

Цифровая обработка СИГНАЛОВ

Digital Signal Processing

A Practical Approach
Second edition

Emmanuel C. Ifeachor
University of Plymouth

Barrie W. Jervis
Sheffield Hallam University



An imprint of Pearson Education

Harlow, England • London • New York • Reading, Massachusetts • San Francisco • Toronto • Don Mills, Ontario • Sydney
Tokyo • Singapore • Hong Kong • Seoul • Taipei • Cape Town • Madrid • Mexico City • Amsterdam • Munich • Paris • Milan

Цифровая обработка СИГНАЛОВ

Практический подход
Второе издание

Эммануил Айфичер
Плимутский университет

Барри Джервис
Университет Шеффилда Халлама



Москва · Санкт-Петербург · Киев
2008

ББК 32.973.26-018.2.75

А36

УДК 681.3.07

Издательский дом “Вильямс”

Зав. редакцией *С.Н. Тригуб*

Перевод с английского *И.Ю. Дорошенко, А.В. Назаренко*

Под редакцией *А.В. Назаренко*

Научный консультант *В.Ю. Чех*

По общим вопросам обращайтесь в Издательский дом “Вильямс” по адресу:
info@williamspublishing.com, <http://www.williamspublishing.com>

Айфичер, Эммануил С., **Джервис**, Барри У.

А36 Цифровая обработка сигналов: практический подход, 2-е издание. : Пер. с англ. — М.: Издательский дом “Вильямс”, 2008. — 992 с. : ил. — Парал. тит. англ.

ISBN 978-5-8459-0710-3 (рус.)

Данная книга предназначена в первую очередь для практиков — инженеров-электронщиков, программистов, людей, занимающихся техникой связи, вычислительными и электронными устройствами. Хотя теоретические основы и изложены в необходимом объеме (не зная основных принципов, работать с цифровыми системами невозможно), основной акцент все же делается на практической разработке современных цифровых устройств, которые можно будет использовать в военной сфере и биомедицине, телекоммуникациях и проигрывателях компакт-дисков, обработке изображений и цифровом телевидении. Для более полного усвоения излагаемого материала в книге подробно разбираются примеры, приводится справочная информация и предлагаются задачи для самостоятельного решения.

Данную книгу можно использовать как учебник при подготовке прикладных специалистов.

ББК 32.973.26-018.2.75

Все названия программных продуктов являются зарегистрированными торговыми марками соответствующих фирм.

Никакая часть настоящего издания ни в каких целях не может быть воспроизведена в какой бы то ни было форме и какими бы то ни было средствами, будь то электронные или механические, включая фотокопирование и запись на магнитный носитель, если на это нет письменного разрешения издательства Pearson Education Limited.

Authorized translation from the English language edition published by Addison Wesley, Copyright © 1993, 2002

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from the Publisher.

Russian language edition published by Williams Publishing House according to the Agreement with R&I Enterprises International, Copyright © 2008

ISBN 978-5-8459-0710-3(рус.)

ISBN 0-201-59619-9 (англ.)

© Издательский дом “Вильямс”, 2008

© Pearson Education Limited, 1993, 2002

Оглавление

Предисловие	17
1. Введение	27
2. Аналоговый интерфейс ввода-вывода для систем ЦОС реального времени	63
3. Дискретные преобразования	135
4. Применение z -преобразования в обработке сигналов	207
5. Корреляция и свертка	281
6. Схема разработки цифровых фильтров	355
7. Разработка фильтров с конечной импульсной характеристикой (КИХ-фильтров)	381
8. Разработка фильтров с бесконечной импульсной характеристикой (БИХ-фильтров)	499
9. Цифровая обработка сигналов при нескольких скоростях	629
10. Адаптивные цифровые фильтры	699
11. Оценка и анализ спектра	737
12. Универсальные и специализированные процессоры ЦОС	783
13. Анализ эффектов конечной разрядности в системах ЦОС с фиксированной запятой	867
14. Приложения и разобранные примеры	935
Предметный указатель	987

Содержание

Предисловие	17
1. Введение	27
1.1. Цифровая обработка сигналов и ее преимущества	27
1.2. Области применения	29
1.3. Ключевые операции ЦОС	31
1.3.1. Свертка	31
1.3.2. Корреляция	33
1.3.3. Цифровая фильтрация	34
1.3.4. Дискретные преобразования	36
1.3.5. Модуляция	38
1.4. Процессоры для цифровой обработки сигнала	38
1.5. Обзор реальных областей применения ЦОС	39
1.6. Применение ЦОС при записи и воспроизведении звука	39
1.6.1. Цифровое микширование	39
1.6.2. Синтез и распознавание речи	41
1.6.3. Цифровая аудиосистема воспроизведения компакт-дисков	44
1.7. Применение ЦОС в телекоммуникации	48
1.7.1. Цифровые сотовые мобильные телефоны	48
1.7.2. Приставка для приема цифрового телевизионного сигнала	53
1.7.3. Адаптивное телефонное эхоподавление	53
1.8. Применение ЦОС в биомедицине	55
1.8.1. Изучение ЭКГ плода	56
1.8.2. Анестезия с регулированием по замкнутому циклу на основе ЦОС	58
1.9. Резюме	61
2. Аналоговый интерфейс ввода-вывода для систем ЦОС реального времени	63
2.1. Характерные системы ЦОС реального времени	64
2.2. Процесс преобразования аналогового сигнала в цифровой	65
2.3. Дискретизация — низкочастотные и полосовые сигналы	66
2.3.1. Дискретизация низкочастотных сигналов	67
2.3.2. Дискретизация полосовых сигналов	82
2.4. Однородное и неоднородное квантование и кодирование	90

2.4.1. Однородное квантование и кодирование (линейная импульсно- коддовая модуляция, ИКМ)	92
2.4.2. Неоднородное квантование и кодирование (нелинейная ИКМ)	94
2.5. Выборка с запасом по частоте при аналого-цифровом преобразовании	96
2.5.1. Введение	96
2.5.2. Выборка с запасом по частоте и фильтры защиты от наложения спектров	97
2.5.3. Выборка с запасом по частоте и разрешающая способность АЦП	100
2.5.4. Применение выборки с запасом по частоте — однобитовый АЦП	104
2.6. Процесс цифроаналогового преобразования: восстановление сигнала	111
2.7. Цифроаналоговый преобразователь	111
2.8. Фильтры защиты от зеркальных частот	113
2.9. Выборка с запасом по частоте при цифроаналоговом преобразовании	113
2.9.1. Цифроаналоговое преобразование с запасом по частоте в проигрывателе компакт-дисков	115
2.10. Ограничения обработки сигналов в реальном времени с аналоговыми входными/выходными сигналами	117
2.11. Примеры применения	118
2.12. Резюме	118
3. Дискретные преобразования	135
3.1. Введение	135
3.1.1. Ряд Фурье	137
3.1.2. Преобразование Фурье	140
3.2. ДПФ и обратное ДПФ	143
3.3. Свойства ДПФ	150
3.4. Вычислительная сложность ДПФ	152
3.5. Алгоритм быстрого преобразования Фурье с децимацией во временной области	153
3.5.1. “Бабочка”	158
3.5.2. Выполнение алгоритма	160
3.5.3. Вычислительные преимущества БПФ	163
3.6. Обратное быстрое преобразование Фурье	163
3.7. Реализация БПФ	165
3.7.1. БПФ с частотной децимацией	165
3.7.2. Сравнение алгоритмов ВД и ЧД	165
3.7.3. Модификации для увеличения скорости	166
3.8. Другие дискретные преобразования	166
3.8.1. Дискретное косинус-преобразование	167
3.8.2. Преобразование Уолша	168
3.8.3. Преобразование Адамара	171
3.8.4. Вейвлетное преобразование	173
3.8.5. Вейвлетный метод кратномасштабного анализа	176

3.8.6. Представление сигнала с помощью сингулярностей: метод вейвлетного преобразования	178
3.9. Применение ДКП: сжатие изображений	183
3.9.1. Дискретное косинус-преобразование	184
3.9.2. Квантование коэффициентов двумерного ДКП	185
3.9.3. Кодирование	185
3.10. Примеры	186
Приложения	195
3.A. Программа на С для прямого вычисления ДПФ	195
3.B. Программа на С для двоичного БПФ с временной децимацией	201
3.B. ДПФ и БПФ в программном пакете MATLAB	204
4. Применение z-преобразования в обработке сигналов	207
4.1. Сигналы и системы дискретного времени	208
4.2. z-преобразование	209
4.3. Обратное z-преобразование	214
4.3.1. Метод степенных рядов	215
4.3.2. Метод разложения на элементарные дроби	217
4.3.3. Метод вычетов	223
4.3.4. Сравнение различных методов обратного z-преобразования	229
4.4. Свойства z-преобразования	229
4.5. Некоторые области применения z-преобразования в обработке сигналов	232
4.5.1. Описание систем дискретного времени с помощью полюсов и нулей	232
4.5.2. Оценка частотной характеристики	235
4.5.3. Геометрическая оценка	235
4.5.4. Непосредственный компьютерный расчет частотной характеристики	239
4.5.5. Вычисление частотной характеристики с помощью БПФ	240
4.5.6. Единицы измерения частоты, которые используются в системах дискретного времени	240
4.5.7. Исследование устойчивости	243
4.5.8. Разностные уравнения	244
4.5.9. Оценка импульсной характеристики	245
4.5.10. Применение в проектировании цифровых фильтров	247
4.5.11. Структуры реализации цифровых фильтров	248
4.6. Резюме	252
Приложения	261
4.A. Рекурсивный алгоритм вычисления обратного z-преобразования	261
4.B. Программа на С для оценки обратного z-преобразования и превращения последовательной структуры в параллельную	263
4.B. Программа на языке С для вычисления частотной характеристики	269
4.Г. Операции z-преобразования с помощью программного пакета MATLAB	270

5. Корреляция и свертка	281
5.1. Введение	281
5.2. Описание корреляции	282
5.2.1. Взаимная корреляция и автокорреляция	288
5.2.2. Применение корреляции	296
5.2.3. Быстрая корреляция	306
5.3. Описание свертки	311
5.3.1. Свойства свертки	319
5.3.2. Круговая свертка	320
5.3.3. Идентификация систем	320
5.3.4. Обращение свертки	322
5.3.5. Слепое обращение свертки	323
5.3.6. Быстрая линейная свертка	325
5.3.7. Вычислительные преимущества быстрой линейной свертки	326
5.3.8. Свертка и корреляция путем сегментации	327
5.3.9. Метод наложения-сложения	329
5.3.10. Метод наложения-записи	334
5.3.11. Вычислительные преимущества быстрой свертки через сегментацию	337
5.3.12. Связь между сверткой и корреляцией	338
5.4. Реализация корреляции и свертки	339
5.5. Примеры применения	339
5.5.1. Корреляция	339
5.5.2. Свертка	344
5.6. Резюме	348
Приложение	354
5.A. Программа на языке C для расчета взаимной корреляции и автокорреляции	354
6. Схема разработки цифровых фильтров	355
6.1. Введение в цифровые фильтры	355
6.2. Типы цифровых фильтров: КИХ- и БИХ-фильтры	357
6.3. Выбор между КИХ- и БИХ-фильтрами	359
6.4. Этапы разработки фильтра	362
6.4.1. Спецификация требований	362
6.4.2. Расчет коэффициентов	364
6.4.3. Представление фильтра подходящей структурой	366
6.4.4. Анализ влияния конечной разрядности	367
6.4.5. Реализация фильтра	371
6.5. Примеры	372
6.6. Резюме	377

7. Разработка фильтров с конечной импульсной характеристикой (КИХ-фильтров)	381
7.1. Вступление	382
7.1.1. Резюме по ключевым особенностям КИХ-фильтров	382
7.1.2. Линейная фазовая характеристика и ее следствия	382
7.1.3. Типы КИХ-фильтров с линейной фазовой характеристикой	385
7.2. Разработка КИХ-фильтров	388
7.3. Спецификации КИХ-фильтра	389
7.4. Методы расчета коэффициентов КИХ-фильтров	390
7.5. Метод взвешивания	390
7.5.1. Некоторые распространенные весовые функции	394
7.5.2. Метод взвешивания: резюме	397
7.5.3. Плюсы и минусы метода взвешивания	404
7.6. Оптимизационные методы	405
7.6.1. Основные концепции	406
7.6.2. Параметры, требуемые для использования оптимальной программы	409
7.6.3. Соотношения для оценки длины фильтра N	409
7.6.4. Резюме по процедуре вычисления коэффициентов фильтра с помощью оптимального метода	410
7.6.5. Примеры	411
7.7. Метод частотной выборки	418
7.7.1. Нерекурсивные фильтры частотной выборки	418
7.7.2. Рекурсивные фильтры частотной выборки	428
7.7.3. Фильтры частотной выборки с простыми коэффициентами	429
7.7.4. Метод частотной выборки: резюме	436
7.8. Сравнение метода взвешивания, оптимального метода и метода частотной выборки	437
7.9. Разработка КИХ-фильтров — специальные вопросы	439
7.9.1. Полуполосные КИХ-фильтры	439
7.9.2. Преобразование частоты	441
7.9.3. Вычислительно эффективные КИХ-фильтры	443
7.10. Структуры реализаций КИХ-фильтров	444
7.10.1. Трансверсальная структура	445
7.10.2. Структура с линейной фазовой характеристикой	446
7.10.3. Другие структуры	447
7.10.4. Выбор структуры	449
7.11. Влияние конечной разрядности на цифровые КИХ-фильтры	450
7.11.1. Ошибки квантования коэффициентов	451
7.11.2. Ошибки округления	457
7.11.3. Ошибки переполнения	458
7.12. Методы реализации КИХ-фильтров	459
7.13. Пример разработки	460
7.14. Резюме	464

7.15. Примеры применения КИХ-фильтров	464
Приложения	481
7.A. Программы на С для разработки КИХ-фильтров	481
7.B. Разработка КИХ-фильтра с помощью MATLAB	485
7.B.1. Метод взвешивания	485
7.B.2. Оптимизационные методы	489
7.B.3. Метод частотной выборки	495
8. Разработка фильтров с бесконечной импульсной характеристикой (БИХ-фильтров)	499
8.1. Вступление: резюме по основным характеристикам БИХ-фильтров	500
8.2. Этапы разработки цифровых БИХ-фильтров	501
8.3. Спецификация производительности	502
8.4. Методы расчета коэффициентов БИХ-фильтров	504
8.5. Расчет коэффициентов фильтра путем размещения нулей и полюсов	504
8.5.1. Основные концепции и пример разработки	504
8.6. Расчет коэффициентов методом инвариантного преобразования импульсной характеристики	508
8.6.1. Основные концепции и пример разработки	508
8.6.2. Метод инвариантного преобразования импульсной характеристики: резюме	511
8.6.3. Метод инвариантного преобразования импульсной характеристики: замечания	511
8.7. Расчет коэффициентов с помощью согласованного z-преобразования	513
8.7.1. Основные концепции и примеры разработки	513
8.7.2. Метод согласованного z-преобразования: резюме	516
8.7.3. Метод согласованного z-преобразования: замечания	516
8.8. Расчет коэффициентов с помощью билинейного z-преобразования	517
8.8.1. Основные концепции и примеры разработки	517
8.8.2. Метод билинейного z-преобразования: резюме	519
8.8.3. Метод билинейного z-преобразования: замечания	523
8.9. Использование для разработки БИХ-фильтров билинейного z-преобразования и классических аналоговых фильтров	527
8.9.1. Характерные особенности классических аналоговых фильтров	527
8.9.2. Методология билинейного z-преобразования с использованием классических аналоговых фильтров	530
8.9.3. Примеры разработки фильтров нижних частот, верхних частот, полосовых и режекторных	535
8.10. Расчет коэффициентов БИХ-фильтра путем отображения полюсов и нулей s-плоскости	544
8.10.1. Основные концепции	544
8.10.2. Примеры	549
8.11. Использование программ разработки БИХ-фильтров	553
8.12. Выбор метода расчета коэффициентов БИХ-фильтров	554
8.12.1. Эффект Найквиста	554

8.13. Структуры реализации цифровых БИХ-фильтров	562
8.13.1. Практические стандартные блоки БИХ-фильтров	562
8.13.2. Каскадная и параллельная реализации БИХ-фильтров больших порядков	565
8.14. Влияние конечной разрядности на БИХ-фильтры	569
8.14.1. Ошибки квантования коэффициентов	570
8.15. Реализация БИХ-фильтров	573
8.15.1. Требования к вычислениям	573
8.16. Подробный пример разработки цифрового БИХ-фильтра	574
8.17. Резюме	579
8.18. Примеры использования БИХ-фильтров в цифровых аудиосистемах и измерительных приборах	580
8.18.1. Цифровые аудиосистемы	580
8.18.2. Цифровое управление	580
8.18.3. Цифровые частотные осцилляторы	582
8.19. Примеры применения БИХ-фильтров в связи	582
8.19.1. Генерация и прием сигналов в цифровых кнопочных телефонах	583
8.19.2. Цифровая телефония: детектирование двухтонального многочастотного сигнала с использованием алгоритма Горцеля	585
8.19.3. Восстановление тактовой частоты в сфере передачи данных	590
Приложения	605
8.А. Программы на С для разработки БИХ-фильтров	605
8.А.1. Реализации на С инвариантного преобразования импульсной характеристики	606
8.В. Разработка БИХ-фильтров с помощью MATLAB	610
8.В.1. Вычисление квадратных корней из комплексных величин с использованием действительной арифметики	627
9. Цифровая обработка сигналов при нескольких скоростях	629
9.1. Введение	629
9.1.1. Некоторые области промышленного применения обработки данных при нескольких скоростях	630
9.2. Концепции обработки при нескольких скоростях	632
9.2.1. Уменьшение частоты дискретизации: децимация с целым шагом	632
9.2.2. Увеличение частоты дискретизации: интерполяция с целым шагом	633
9.2.3. Преобразование частоты дискретизации с нецелым шагом	635
9.2.4. Многокаскадное преобразование частоты дискретизации	639
9.3. Разработка практических конвертеров частоты дискретизации	640
9.3.1. Спецификация фильтра	640
9.3.2. Требования к фильтрам отдельных каскадов	642
9.3.3. Определение числа каскадов и шагов децимации	643
9.3.4. Примеры разработки	644
9.4. Программная реализация дециматоров	652
9.4.1. Программа многокаскадной децимации	653

9.4.2. Тестовый пример использования программы децимации	655
9.5. Программная реализация интерполяторов	655
9.5.1. Программа многокаскадной интерполяции	659
9.5.2. Тестовый пример	661
9.6. Преобразование частоты дискретизации с использованием многофазного фильтра	663
9.6.1. Многофазная реализация интерполяторов	663
9.7. Примеры применения	668
9.7.1. Высококачественное аналого-цифровое преобразование в цифровом аудио	669
9.7.2. Эффективное аналого-цифровое преобразование в высококачественных системах воспроизведения компакт-дисков	669
9.7.3. Сбор высококачественных данных	671
9.7.4. Узкополосная цифровая фильтрация с обработкой при нескольких скоростях	676
9.7.5. Узкополосный спектральный анализ с высоким разрешением	683
9.8. Резюме	683
Приложения	693
9.A. Программы на С для разработки и реализации систем обработки при нескольких скоростях	693
9.B. Цифровая обработка сигналов с помощью MATLAB	693
10. Адаптивные цифровые фильтры	699
10.1. Когда и где стоит использовать адаптивные фильтры	699
10.2. Концепции адаптивной фильтрации	701
10.2.1. Адаптивные фильтры как средство подавления шума	701
10.2.2. Другие конфигурации адаптивного фильтра	702
10.2.3. Основные компоненты адаптивного фильтра	702
10.2.4. Адаптивные алгоритмы	702
10.3. Основы теории фильтров Винера	705
10.4. Стандартный адаптивный алгоритм наименьших квадратов	708
10.4.1. Реализация стандартного алгоритма наименьших квадратов	709
10.4.2. Практические ограничения стандартного алгоритма наименьших квадратов	712
10.4.3. Другие алгоритмы на основе схемы наименьших квадратов	715
10.5. Рекурсивный алгоритм наименьших квадратов	717
10.5.1. Рекурсивный алгоритм наименьших квадратов	718
10.5.2. Ограничения рекурсивного алгоритма наименьших квадратов	719
10.5.3. Алгоритмы факторизации	719
10.6. Сфера приложения 1 — адаптивная фильтрация окулярных артефактов на ЭЭГ человека	720
10.6.1. Физиологическая задача	720
10.6.2. Алгоритм обработки артефактов	721
10.6.3. Реализация в реальном времени	723

10.7. Сфера приложения 2 — адаптивное телефонное эхоподавление	723
10.8. Другие приложения	725
10.8.1. Громкоговорящие телефоны	725
10.8.2. Компенсация многолучевого распространения	725
10.8.3. Адаптивное подавление преднамеренной помехи	727
10.8.4. Обработка сигналов с радаров	727
10.8.5. Отделение сигналов речи от фоновых шума	727
10.8.6. Наблюдение за плодом — подавление ЭКГ матери при родах	728
Приложения	731
10.А. Программа на С для адаптивной фильтрации	731
10.А.1. Адаптивное выделение узкополосных сигналов, спрятанных в шуме	732
10.Б. Программы MATLAB для адаптивной фильтрации	736
11. Оценка и анализ спектра	737
11.1. Введение	737
11.2. Принципы оценки спектра	740
11.3. Традиционные методы	742
11.3.1. Подводные камни	743
11.3.2. Взвешивание	746
11.3.3. Метод и свойства периодограмм	756
11.3.4. Методы модифицированных периодограмм	759
11.3.5. Метод Блэкмена–Тьюки	760
11.3.6. Метод быстрой корреляции	761
11.3.7. Сравнение методов оценки спектральной плотности мощности	761
11.4. Современные параметрические методы оценки	762
11.5. Авторегрессионная оценка спектра	763
11.5.1. Авторегрессионная модель и авторегрессионный фильтр	763
11.5.2. Спектральная плотность мощности авторегрессионного ряда	764
11.5.3. Расчет параметров модели — уравнения Юла–Уокера	765
11.5.4. Решение уравнений Юла–Уокера	767
11.5.5. Порядок модели	769
11.6. Сравнение методов оценки	770
11.7. Примеры применения	770
11.7.1. Использование спектрального анализа с помощью ДПФ для различения болезней мозга	770
11.7.2. Спектральный анализ ЭЭГ с использованием авторегрессионного моделирования	773
11.8. Резюме	774
11.9. Разобранный пример	774
Приложение	781
11.А. Программы MATLAB для оценки и анализа спектра	781

12. Универсальные и специализированные процессоры ЦОС	783
12.1. Вступление	784
12.2. Компьютерные архитектуры обработки сигналов	784
12.2.1. Гарвардская архитектура	786
12.2.2. Конвейерная обработка	788
12.2.3. Аппаратный умножитель-накопитель	792
12.2.4. Специальные команды	795
12.2.5. Средства дублирования	798
12.2.6. Встроенная память/кэш	799
12.2.7. Расширенный параллелизм — SIMD, VLIW и статическая суперскалярная обработка	799
12.3. Универсальные процессоры ЦОС	804
12.3.1. Процессоры ЦОС с фиксированной запятой	805
12.3.2. Процессоры ЦОС с плавающей запятой	815
12.4. Выбор цифрового процессора сигналов	818
12.5. Реализация алгоритмов ЦОС на универсальных процессорах ЦОС	820
12.5.1. Цифровая КИХ-фильтрация	820
12.5.2. Цифровая БИХ-фильтрация	829
12.5.3. Расчет БПФ	836
12.5.4. Обработка при нескольких скоростях	842
12.5.5. Адаптивная фильтрация	845
12.6. Специализированная аппаратура ЦОС	847
12.6.1. Аппаратные цифровые фильтры	848
12.6.2. Аппаратные процессоры БПФ	849
12.7. Резюме	852
Приложение	859
12.A. Программы на языке ассемблера TMS320 для обработки сигналов в реальном времени и программа на языке C для двоичного БПФ с постоянной геометрией	859
13. Анализ эффектов конечной разрядности в системах ЦОС с фиксированной запятой	867
13.1. Вступление	867
13.2. Арифметика ЦОС	868
13.2.1. Арифметика с фиксированной запятой	870
13.2.2. Арифметика с плавающей запятой	874
13.3. Шум квантования АЦП и качество сигнала	877
13.4. Эффекты конечной разрядности в цифровых БИХ-фильтрах	880
13.4.1. Влияние структуры фильтра на эффекты конечной разрядности	881
13.4.2. Ошибки квантования коэффициентов	884
13.4.3. Требования к длинам коэффициентов с точки зрения устойчивости и желаемой частотной характеристики	886
13.4.4. Ошибки переполнения при сложении	890
13.4.5. Принципы масштабирования	892

13.4.6. Масштабирование при каскадной реализации	895
13.4.7. Масштабирование в параллельной реализации	896
13.4.8. Детектирование и предотвращение переполнения на выходе	898
13.4.9. Ошибки округления произведения в цифровых БИХ-фильтрах	898
13.4.10. Влияние ошибок квантования на производительность фильтра	900
13.4.11. Шум округления в каскадных и параллельных реализациях	904
13.4.12. Влияние шума округления произведения в современных системах ЦОС	909
13.4.13. Схемы снижения шума округления	910
13.4.14. Определение практических значений коэффициентов обратной связи по ошибке	916
13.4.15. Граничный цикл вследствие ошибок округления произведений	921
13.4.16. Другие нелинейные явления	922
13.5. Эффекты конечной разрядности в алгоритмах БПФ	923
13.5.1. Ошибки округления в БПФ	923
13.5.2. Ошибки переполнения и масштабирование в БПФ	926
13.5.3. Квантование коэффициентов в БПФ	927
13.6. Резюме	928
Приложения	933
13.А. Программа анализа эффектов конечной разрядности для БИХ-фильтров	933
13.Б. Уравнения масштабных множителей L_2	933
14. Приложения и разобранные примеры	935
14.1. Платы для демонстрации обработки сигналов в реальном времени	935
14.1.1. Основы	935
14.1.2. Целевая плата TMS320C10	936
14.1.3. Демонстрационный модуль для ЦОС реального времени	937
14.1.4. Демонстрационные платы TMS320C54 и DSP56300	938
14.2. Приложения ЦОС	939
14.2.1. Детектирование сердцебиения плода во время родов	939
14.2.2. Адаптивное устранение окулярных артефактов с ЭКГ человека	948
14.2.3. Выравнивание цифровых аудиосигналов	964
14.3. Модельные задачи	967
14.4. Вопросы по ЦОС для компьютерных тестов	975
14.5. Резюме	983
Приложение	986
14.А. Модифицированный алгоритм UD-факторизации	986
Предметный указатель	987

Предисловие

Цель книги

На протяжении последних пяти лет цифровая обработка сигнала (digital signal processing — DSP, ЦОС) оказывает первостепенное и постоянно возрастающее влияние на такие ключевые технологические отрасли, как телекоммуникации, цифровое телевидение и средства информации, биомедицина и цифровая звукозапись. Сегодня ЦОС является ядром множества видов новейших цифровых разработок и различных приложений в информационном обществе (например, цифровая мобильная связь, цифровые видеокамеры, телевидение и системы звукозаписи). В современном мире от инженеров-электронщиков, программистов и связистов даже еще больше, чем во времена первого издания, ожидается и требуется умение работать с ЦОС. Теперь ЦОС — это основной предмет почти во всех курсах, посвященных технике связи, вычислительной и электронной технике.

Второе издание книги стало более современным благодаря тому, что в него вошли дополнительные темы, значение которых постоянно растет, и задачи, основанные на программном пакете MATLAB, а также благодаря наличию справочника-приложения и web-страницы. Эти дополнения появились как реакция на развитие программного обеспечения, расширение доступа к информационным технологиям, достижения в области обработки сигнала и спрос читателей. В высших учебных заведениях все чаще пользуются материалами из сети Internet и такими программными средствами обработки сигналов, как MATLAB. В частности, мы заметили постепенно возрастающий интерес у наших читателей к материалам, основанным на программном пакете MATLAB. Данный язык высокого уровня позволяет осуществлять сложную обработку сигнала и немедленно выводить результат на экран с помощью относительно небольшого числа команд. Можно получать удовольствие от обработки сигнала и решения задач, не отвлекаясь на написание подробных программ. Мы надеемся, что примеры из пакета MATLAB и упражнения, которые вы найдете в этой книге, помогут учащимся приобрести практические навыки и расширят возможности преподавателей.

Целью второго издания, как и первого, было проложить мостик от теории к практике. Поэтому главные особенности книги, — освещение современных тем и предоставление практических примеров и приложений — сохранились. Как и в первом издании,

практические примеры и программы чередуются с теорией, чтобы мотивировать излагаемый материал и поддерживать заинтересованность студентов на высоком уровне для повышения эффективности обучения. Многие главы были значительно переработаны и стали более современными и понятными. Увеличено число задач, предлагаемых в конце каждой главы для проверки, закрепления и дополнения усвоенного материала. При переработке книги мы исходили из собственного опыта и пожеланий читателей первого издания.

Введены такие новые темы: выборка с запасом по частоте (oversampling) и полосовая дискретизация (bandpass sampling) для преобразования аналогового сигнала в цифровой, позволяющие воспользоваться преимуществами ЦОС; вейвлетные преобразования (wavelet transforms), которые применяются при описании и разрешении сигналов в частотной и временной областях; слепое обращение свертки сигнала (blind signal deconvolution) для идентификации входного сигнала по выходу неизвестной системы; параметрическая оценка спектра для повышения разрешения коротких сигналов и уменьшения количества ошибок; архитектура новых процессоров ЦОС и практические советы по снижению шума, возникающего в результате округления, в системах ЦОС с фиксированной запятой; компьютерные вопросы с различными вариантами ответов для улучшения контроля. Многие примеры и упражнения, приведенные в книге, основаны на программном пакете MATLAB.

Эта книга возникла на основе нашего многолетнего опыта преподавания практических специализированных курсов по цифровой обработке сигнала студентам Плимутского университета и университета Шеффилда Халлама (Sheffield Hallam University) и специалистам по применению этих знаний на производстве. Оказалось, что многие из существующих учебников либо слишком элементарны, либо слишком теоретичны для применения в прикладной области. Большинство читателей из собственного опыта знают, насколько глубока пропасть между изучением теоретических основ любого предмета и реальным их воплощением. Поэтому возникло решение написать книгу, которую, как мы надеемся, поймут и оценят студенты, и которая поможет им справиться с практическими заданиями и дипломными проектами по цифровой обработке сигнала. Также мы надеемся, что много полезной информации почерпнут из этой книги и студенты более высокого уровня, и практикующие инженеры, и ученые.

При написании книги источником вдохновения служила также наша собственная исследовательская работа по прикладной цифровой обработке, которая велась на протяжении последних 25 лет, определяя практические вопросы для обсуждения.

Скорее всего, в обозримом будущем интерес к цифровой обработке сигнала будет расти, а ее развитие, как с промышленной, так и с академической точки зрения, будет продолжаться. Существование огромного количества процессоров для цифровой обработки сигналов свидетельствует о большом коммерческом значении ЦОС. Больше всего привлекает то, что цифровая обработка сигналов, позволяющая достичь высокой точности и идеальной воспроизводимости, по своей внутренней гибкости сравнима с аналоговой обработкой сигналов. На производстве многим инженерам все еще не хватает необходимых знаний и опыта работы, чтобы использовать безмерные возможности очень мощных процессоров цифровой обработки сигналов, доступных на сего-

дняшний день. Эта книга даст практические указания и поможет инженерам научиться с помощью таких средств проектировать и разрабатывать практические системы ЦОС.

С теоретической точки зрения ЦОС обычно считают одним из математических предметов электротехнического курса, но, исходя из нашего опыта преподавания, мы свели математические выкладки до того минимума, который посчитали важным, полезным и интересным; кроме этого, мы выделили вопросы, вызывающие определенные проблемы. Как показывает опыт, студенты учатся лучше, если знают о практической востребованности предмета. И хотя теоретические сведения, без сомнения, важны, мы считаем, что выпускники должны также владеть практическими знаниями и умениями. Эта книга была написана именно исходя из таких соображений.

Наша книга не является исчерпывающей работой по ЦОС, но в ней рассматривается большинство аспектов предмета, встречающихся в курсах электротехники, электроники и техники связи. Кроме того, здесь описан ряд методик ЦОС, особенно важных для производства, которые за последние несколько лет начали появляться в курсах обучения. К таким методикам относятся адаптивная фильтрация и обработка при нескольких скоростях.

В книге основное внимание уделяется практическим аспектам ЦОС. Не последнюю роль играет то, что во второе издание включены примеры из программного пакета MATLAB и упражнения по анализу, проектированию, исследованию и обработке сигналов при рациональных затратах времени. Выполнив упражнения из MATLAB, читатель приобретет более глубокие познания в области ЦОС. Поскольку примеры программ для ЦОС на языке C из первого издания оказались достаточно популярными, они также приводятся с незначительными изменениями.

Сегодня MATLAB широко используется в качестве общепринятого инструмента в науке и технике и требует меньших навыков программирования, чем язык C. Этот пакет обладает хорошими графическими и демонстрационными возможностями и обеспечивает неплохую среду для разработки ЦОС. Мы считаем, что студентам не помешало бы ознакомиться с программным пакетом MATLAB, а умение работать с ним — это весьма полезный навык. Все m-файлы из MATLAB, которые упоминаются в книге, доступны в электронном виде в сети Internet. Кроме того, m-файлы (как и программы на языке C из первого издания) есть и на компакт-диске справочника-приложения (см. ниже).

Основные преимущества этой книги

- Обеспечивается понимание теоретических основ, реализации и применения методов ЦОС с практической точки зрения.
- Книга написана ясно и просто, математические выкладки сведены до необходимого для восприятия минимума.
- Методы ЦОС проиллюстрированы практически ориентированными реальными примерами с полным решением, разработанными для более глубокого понимания ЦОС.
- Даются практические советы, чтобы читатели могли проектировать и разрабатывать настоящие системы ЦОС. Приводятся завершенные примеры разработки и практиче-

ские детали реализации, включая программы на языке ассемблера для процессоров ЦОС.

- Приведены задачи и примеры из программного пакета MATLAB для приобретения практического опыта.
- Предлагаются многие алгоритмы и функции ЦОС на языке C, в том числе программы для:
 - разработки фильтров с конечной импульсной характеристикой (КИХ-фильтров) и с бесконечной импульсной характеристикой (БИХ-фильтров);
 - анализа эффекта конечной длины слова для разрабатываемых пользователем БИХ-фильтров с фиксированной запятой;
 - перехода от каскадной структуры реализации к параллельной;
 - расчета корреляции;
 - алгоритмов дискретного и быстрого преобразований Фурье;
 - обратного z-преобразования;
 - оценки частотной характеристики;
 - разработки систем обработки при нескольких скоростях.
- Компьютерные m-файлы для MATLAB доступны в электронном виде в сети Internet (программы на языке C можно найти на компакт-диске к книге [Ifeachor, 2001]; подробнее см. раздел “Web-сайт, компакт-диск и справочник-приложение” этого предисловия).
- Степень усвоения материала помогают проверять задачи в конце каждой главы, а также вопросы с вариантами ответов.
- Для иллюстрации важных принципов и укрепления приобретенных знаний используются реальные примеры.

Предполагаемый круг читателей

Эта книга предназначена для студентов, занимающихся наукой, техникой и программированием, а также для инженеров и ученых, которые хотят приобрести практические знания о ЦОС. В частности, студентам последнего курса, изучающим электронику, электронику или технику связи, эта книга поможет овладеть общими курсами обучения и написать курсовую работу, поскольку все больше и больше студенческих работ связано с теми или иными аспектами ЦОС. Книга может пригодиться и аспирантам, занимающимся указанными науками.

Теоретические вопросы будут очень привлекательны для студентов, и мы верим, что наша книга будет ценным источником информации как на протяжении всего курса обучения, так и потом, когда они начнут работать на производстве.

Данную книгу могут брать за основу большие коммерческие организации или государственные учреждения, которые проводят свои собственные краткие внутренние курсы обучения ЦОС. Мы надеемся, что эта книга послужит как хорошим учебником, так и ценным самоучителем для студентов, аспирантов и практикующих инженеров.

Содержание и структура

Первая глава представляет собой краткий обзор цифровой обработки сигналов и ее применения. Во второй главе с практической точки зрения и с использованием реальных примеров рассмотрены многие теоретические вопросы, которые служат основой ЦОС, в частности, дискретизация и квантование сигналов и их применение в современной цифровой обработке сигналов. По сравнению с предыдущим изданием в главу включены такие новые темы, как методика выборки с запасом по частоте при цифро-аналоговом и аналого-цифровом преобразованиях, дискретизация полосовых сигналов, равномерное и неравномерное квантование. В этой главе вводится понятие сигналов и систем дискретного времени, но более подробно они рассматриваются в главе 4.

Дискретные преобразования, в частности дискретное (ДПФ) и быстрое (БПФ) преобразования Фурье, обеспечивают ЦОС необходимыми математическими аппаратами, а также связывают временную и частотную области. Они описаны в главе 3 вместе с несколькими примерами их применения на практике. Если выводить дискретное преобразование Фурье (ДПФ) из преобразования Фурье и экспоненциальных рядов Фурье, то можно получить логическое обоснование ДПФ, для которого не требуется знание дискретных рядов Фурье, изложение теории которых значительно увеличило бы толщину книги (и объем работы для студентов!). В нашем изложении мы ограничились лишь описанием и осуществлением преобразований. В частности, в этой главе не рассматривается вопрос взвешивания с помощью финитной функции (windowing), более подробно он освещен в главе 11, касающейся спектрального анализа. Описан также важный пример применения дискретного косинус-преобразования (ДКП) — стандарта JPEG для сжатия изображения. В различных приложениях все чаще и чаще применяется вейвлетное преобразование, благодаря тому, что его можно применять к нестационарным сигналам и разрешать сигналы как по частоте, так и по времени. Описаны примеры применения анализа с разной разрешающей способностью и выявления сингулярностей для очистки сигнала от шума.

В главе 4 изложены основные сведения, касающиеся сигналов и систем дискретного времени. Рассказывается о важных моментах z -преобразования — незаменимого средства для описания и анализа сигналов и систем дискретного времени. Обращается внимание на многие области применения z -преобразования, например, говорится о его пользе при разработке, анализе и вычислении частотной характеристики сигналов и систем дискретного времени. Как и во всей книге, основные принципы и области применения z -преобразования проиллюстрированы примерами с полным решением.

Корреляция и свертка — теоретические основы и темы, близко связанные с ЦОС, — описаны в главе 5. Авторы считают, что для ЦОС необходимо знать все, что есть в этой главе, но после предварительного ознакомления с ее содержанием можно посоветовать читателю расширять свои знания постепенно, изучая эту главу поэтапно. Для студентов этот процесс можно растянуть на несколько семестров. В настоящем издании сюда же включены дополнительные темы по идентификации систем, обращению свертки и слепому обращению свертки. Особенно интересно слепое обращение свертки, поскольку

оно позволяет с помощью максимизации информации определить неизвестный входной сигнал, измеряемый на выходе системы с неизвестной импульсной характеристикой.

В главах 6–8 с практической точки зрения подробно рассматривается один из самых важных моментов ЦОС — разработка цифровых фильтров. Эта тема достаточно обширная, и тем, кто впервые с ней сталкивается, может показаться чем-то непреодолимым. В главе 6 дается общая схема построения фильтра.

Методики создания КИХ-фильтров (фильтров с конечной импульсной характеристикой) от спецификации до реализации фильтра изложены в главе 7. Здесь же приведено несколько примеров с полным решением, позволяющих закрепить важные принципы. В этом издании в число дополнительных тем входит автоматическая разработка частотных КИХ-фильтров. Чтобы показать, как свести воедино все этапы разработки фильтра, полностью разобран пример разработки фильтра.

В главе 8 дано подробное описание разработки БИХ-фильтра, в основу которого положено простое пошаговое руководство. Эта глава была существенно переделана и дополнена. В частности, в результате обратной связи с читателями для большей ясности были переписаны разделы о вычислении коэффициентов и добавлены новые материалы для освещения важных вопросов по разработке БИХ-фильтров. Кроме того, глава содержит примеры с решениями, призванные помочь читателям в разработке БИХ-фильтров от спецификации до реализации. Даны примеры разработки с использованием как программного обеспечения MATLAB, так и языка C.

Мы уменьшили общий объем материала по разработке БИХ-фильтров, переместив рассмотрение эффектов конечной длины слова в главу 13. Таким образом, в главах 1–8 собран существенный материал для большинства курсов по ЦОС. Передовые идеи ЦОС изложены в последующих главах. Детальное изучение эффектов конечной длины слова сосредоточено в главе 13.

Методы обработки при нескольких скоростях привели к возникновению таких новейших приложений, как одноразрядные аналого-цифровые и цифроаналоговые преобразователи и цифровая фильтрация с запасом по частоте, которые используются в ряде современных цифровых систем, в том числе и в хорошо знакомом всем проигрывателе компакт-дисков. В главе 9 объясняются основные принципы обработки при нескольких скоростях, проиллюстрированные примерами с решениями и разработкой реальных систем. Материал этой главы был дополнен темой о многофазной фильтрации. В теорию введено больше примеров разработки и приложений, чтобы проиллюстрировать принципы работы и практические вопросы по разработке систем с обработкой при нескольких скоростях.

В главе 10 описаны ключевые аспекты адаптивных фильтров, в основе которых лежат метод наименьших квадратов (least-mean-squares — LMS) и рекурсивный метод наименьших квадратов (recursive least-square — RLS) — два алгоритма, которые чаще всего применяются в адаптивной обработке сигнала. Они изучаются с практической точки зрения, единственный теоретический момент внесен в основной текст главы.

В главе 11 рассматривается важная тема оценки и анализа спектров, которые применяются для описания и изучения сигналов в определенной частотной области. С появлением программных пакетов для параметрической оценки спектров будет уместным подробное знакомство с этими методами. Предложенные сигналы можно точно описать

моделями правильного порядка, а параметрическая оценка спектров применяется для уменьшения длины сигнала и оценки спектра с лучшим разрешением по сравнению с непараметрическими методами. Для иллюстрации метода приведен пример применения авторегрессионной оценки спектра ответных сигналов электроэнцефалограммы. Читатели, которых особенно интересует спектральный анализ, могут изучить и главу 11, и главу 3, поскольку в главе 11 используются объяснения и примеры с решениями из главы 3. Кто овладеет информацией, изложенной в этих двух главах, будет хорошо разбираться в анализе сигналов в частотной области.

Последние 15 лет были годами огромного прогресса в области компьютерного оборудования для ЦОС, и это привело к тому, что недорогие цифровые процессоры стали общедоступными. Для успешного применения этих процессоров в ЦОС необходимо понять принципы, на основе которых разрабатывается оборудование и программное обеспечение для ЦОС. В главе 12 обсуждаются ключевые моменты основных процессоров ЦОС общего и специального назначения, влияние алгоритмов ЦОС на структуру оборудования и программного обеспечения для этих процессоров и структурные требования, необходимые для эффективного осуществления функций ЦОС. Материалы этой главы были дополнены в соответствии с новыми данными. В частности, обсуждаются такие новые архитектуры ЦОС, как архитектура с командными словами сверхбольшой длины и суперскалярная архитектура, новые процессоры для ЦОС с фиксированной и плавающей запятой (включая процессоры с фиксированной запятой от Texas Instruments, например, TMS320C54 и TMS320C62, процессоры с фиксированной запятой от Motorola DSP56300 и Analog Devices TigerSHARC IS0001).

В главе 13 представлен подробный анализ эффектов конечной длины слова в современных системах ЦОС с фиксированной запятой. В соответствующих местах для иллюстрации явления ухудшения рабочих характеристик при использовании арифметики с фиксированной запятой разобраны задачи.

Новая глава 14 (хотя в ней и остались некоторые материалы из первого издания) служит обучающим источником для преподавателей и студентов. Эта глава содержит описание недорогих плат ЦОС для реализации алгоритмов ЦОС и ряда реально существующих приложений в форме конкретных примеров. К числу других отличительных признаков относятся компьютерные вопросы с несколькими вариантами ответов, касающиеся основных моментов всех тем, изложенных в предыдущих главах. Эти вопросы незаменимы для повторения и выставления оценок в больших группах. В главе дается полное описание лабораторных работ и предлагаются идеи для дипломных работ.

Как пользоваться книгой

При преподавании студентам полезным будет такой подход: изложить материал глав 1 и 2, изучить такие фундаментальные понятия, как теорема о дискретном представлении, сигналы и системы дискретного времени, описать преимущества и области применения ЦОС. Затем можно ввести понятие дискретного преобразования, начиная с дискретного (ДПФ) и быстрого (БПФ) преобразований Фурье (глава 3) и z-преобразования (глава 4). Для иллюстрации применения ДПФ и БПФ можно частично воспользоваться

главами 5 и 11. Ознакомившись с корреляционной обработкой с помощью главы 5, следует подробно изучить цифровые фильтры.

Как показывает опыт, студенты учатся лучше, выполняя задания, взятые из реальной жизни. Исходя из этого, поощряются существенные задания по, например, разработке фильтров, обратному z -преобразованию или ДПФ и БПФ. Кроме того, необходимы лабораторные работы для демонстрации методов и закрепления знаний по изученным темам. Важно, чтобы студенты не только посещали лекции, а были активными участниками процесса обучения.

Для студентов последних курсов и аспирантов подход может быть таким же, но скорость прохождения курса будет выше, кроме того, в него следует включить больше специализированных тем по обработке при нескольких скоростях и адаптивным фильтрам.

Web-сайт, компакт-диск и справочник-приложение

Дополнительную информацию об этой книге можно найти на web-сайте www.booksites.net/ifeachor.

С авторами, которые очень ждут ваших отзывов и пожеланий, можно связаться, используя ссылку “Contact us” на сайте www.booksites.net/ifeachor.

Электронные копии всех m-файлов MATLAB можно загрузить через сайт-приложение к данной книге на www.booksites.net/ifeachor.

В материалы входит ряд m-файлов MATLAB, которыми читатели могут пользоваться при выполнении заданий, подобно тому, как они могли это делать с некоторыми программами на языке C в первом издании. Предлагаемые m-файлы MATLAB, программы на языке C и коды на ассемблере также доступны на компакт-диске, который поставляется со справочником-приложением. Программы на языке C из первого издания (с небольшими изменениями) доступны как в виде исполняемых программ, так и в виде исходных кодов. Компилятор для языка C необходим только для запуска исходных кодов, а для запуска исполняемых кодов он не нужен. Программы были написаны на стандартном языке ANSI C под Borland Turbo C версии 2.0. Справочник-приложение *A Practical Guide for MATLAB and C Language Implementation of DSP Algorithms*, опубликованный издательством Pearson, вместе с компакт-диском можно приобрести отдельно. В приложениях к основной книге также содержится множество иллюстративных примеров использования m-файлов MATLAB и программ на языке C.

Благодарности

Мы были счастливы получить множество полезных комментариев и предложений от многих наших настоящих и бывших студентов, которые помогли улучшить техническую сторону книги и сделали ее понятнее. Мы благодарны им всем, а особенно Нику Отраму (Nick Outram), Эдди Риддингтону (Eddie Riddington), Робину Кларку (Robin Clark), Стиву Харпису (Steve Harris), Брагиму Хамадишарефу (Brahim Hamadicharef),

Яну Шоли (Ian Scholey), Франку Аманду (Francois Amand), Николя Гейтеру (Nichola Gater), Роберту Рюсу (Robert Ruse) и Эндрю Поли (Andrew Paulley). Многие упражнения, особенно из главы 14, были разработаны для наших курсов ЦОС Ником Отрамом, Эдди Риддингтоном, Робинотом Кларком и Брагимом Хамадишарефом. Джеймса Бриттона (James Britton) мы благодарим за вычисления и графики в примерах глав 3 и 11. Некоторые из наших бывших студентов дополнили наши курсы, работая на производстве. Огромное спасибо всем им, а особенно Робину Кларку и Нику Отраму за их стимулирующий вклад.

Авторы хотели бы выразить благодарность господину Майку Фрейзеру (Mike Fraser), который несколько лет тому назад был техническим сотрудником Плимутского университета, бывшему главному инженеру компании Rank Toshiba (Плимут). Его значительный опыт и ценные комментарии оказались чрезвычайно полезными. Также мы хотели бы поблагодарить его и Пола Смитсона (Paul Smithson) за разработку и создание плат для ЦОС по нашим исходным чертежам и разработку среды, в которой реализовывались и проверялись многие программы. Мы признательны за помощь и комментарии многим другим коллегам, особенно господину Питеру ван Итвелту (Peter Van Eetvelt) за вывод формул в приложениях 8В и 8Г.

Мы в большом долгу перед множеством читателей из научных и технических кругов по всему миру за неоценимую обратную связь, за то, что они потрудились обратить наше внимание на ошибки в первом издании и сообщить нам, что они думают о нашей книге. Мы очень надеемся, что и они, и другие, продолжают поддерживать с нами связь.

Практический характер книги не всегда позволяет вкладываться в намеченные сроки. Написание каждой главы заняло намного больше времени, чем мы планировали. Мы благодарны первому редактору, Тиму Питтсу (Tim Pitts), за терпение и поддержку. Мы в большом долгу перед Анной Фаерти (Anna Faherty), вторым редактором, за то, что она уговорила нас написать второе издание, и Карен Сюзерленд (Karen Sutherland), Джулией Найт (Julie Knight) и Мэри Линц (Mary Lince) за предварительное принятие проекта.

Наконец, авторы выражают особую благодарность своим семьям за терпение, преданность и поддержку на протяжении всего этого проекта, который отнимал очень много времени.

*Эммануил Айфичер (Emmanuel Ifeakor),
Барри Джервис (Barrie Jervis)
март 2001 года*

Благодарности издателя

Мы благодарны следующим владельцам авторских прав за возможность воспроизведения их материалов.

Компания Allen & Heath, Корнуолл любезно предоставила рис. 1.8; рис. 1.12 и рис. 1.15 взяты из журнала *Philips Technical Review* Vol. 40(6), который издает Konintylke Philips Electronics N. V.; рис. 8.37–8.39 взяты из статьи “Add DTMF generation and decoding to DSP-up designs” в журнале *EDN Magazine* Vol. 30, издаваемом Cahners Business

Information [Mock, P., 1985]; рис. 13.2 взят из журнала *Journal of Audio Engineering Society* Vol. 41(9), издаваемого Audio Engineering Society, Inc. ([Wilson, P., 1993]); табл. 7.11 и 7.18 адаптированы из журнала *An Approach to the Approximation Problem for Nonrecursive Digital Filters*, издаваемого The Institute of Electrical and Electronics Engineers Inc. [Rabiner, L. R., Gold, B., McGonegal, C. A., 1970]; рис. 14.12 перепечатан из *Clinical Neurophysiology Ireland* No. 38, D. G. Girtton and A. J. Kamiya “Electroencephalography and Clinical Neurophysiology”, pp. 623–639, 1973, с разрешения Excerpta Medica Inc.

Хотя мы сделали все возможное, чтобы отследить владельцев авторских прав на материалы, использованные в данной книге, в некоторых случаях сделать это было просто невозможно, поэтому мы приносим официальные извинения, если невольно посягнули на чьи-то права.

От издательства

Вы, читатель этой книги, и есть главный ее критик и комментатор. Мы ценим ваше мнение и хотим знать, что было сделано нами правильно, что можно было сделать лучше и что еще вы хотели бы увидеть изданным нами. Нам интересно услышать и любые другие замечания, которые вам хотелось бы высказать в наш адрес. Мы ждем ваших комментариев и надеемся на них. Вы можете прислать нам бумажное или электронное письмо, либо просто посетить наш Web-сервер и оставить свои замечания там. Одним словом, любым удобным для вас способом дайте нам знать, нравится или нет вам эта книга, а также выскажите свое мнение о том, как сделать наши книги более интересными для вас. Посылая письмо или сообщение, не забудьте указать название книги и ее авторов, а также ваш обратный адрес. Мы внимательно ознакомимся с вашим мнением и обязательно учтем его при отборе и подготовке к изданию последующих книг. Наши координаты:

E-mail: info@williamspublishing.com

WWW: <http://www.williamspublishing.com>

Информация для писем из:

России: 115419, Москва, а/я 783

Украины: 03150, Киев, а/я 152

1.1. Цифровая обработка сигналов и ее преимущества	27
1.2. Области применения	29
1.3. Ключевые операции ЦОС	31
1.4. Процессоры для цифровой обработки сигнала	38
1.5. Обзор реальных областей применения ЦОС	39
1.6. Применение ЦОС при записи и воспроизведении звука	39
1.7. Применение ЦОС в телекоммуникации	48
1.8. Применение ЦОС в биомедицине	55
1.9. Резюме	61
Задачи	61
Литература	61
Дополнительная литература	62

Цель этой главы — объяснить преимущества цифровой обработки сигналов, ознакомить читателя с главными операциями ЦОС, на которых в основном и построена цифровая обработка сигналов, и обозначить широкий диапазон областей ее применения. Для иллюстрации освещаемых концепций приводятся примеры из реальной жизни.

1.1. Цифровая обработка сигналов и ее преимущества

Под *сигналом* мы понимаем любую переменную, которая передает или содержит некий вид информации, и которую можно, например, переносить, выводить на экран или выполнять с ней какие-то действия. Особый интерес представляют такие типы сигналов:

- речь, с которой мы имеем дело, например, при разговоре по телефону, прослушивании радио и в повседневной жизни;
- биомедицинские сигналы, такие как электроэнцефалограмма (сигналы мозга);
- звуки и музыка, подобные тем, что воспроизводит проигрыватель компакт-дисков;
- видео- и телеизображения;
- сигналы радаров, которые исследуют заданный диапазон и пеленгуют отдаленные цели.

Цифровая обработка сигналов связана с цифровым представлением сигналов и использованием цифровых процессоров для анализа, изменения или извлечения информа-

ции из сигнала. Большинство сигналов, существующих в природе, являются по своей форме аналоговыми, что часто означает непрерывное изменение во времени, и описывающими изменение физических величин (например, звуковые волны). Сигналы, применяемые в ЦОС, обычно получают из аналоговых сигналов, дискретизованных через равные интервалы времени и преобразованных в цифровой вид.

Обработка цифрового сигнала обычно нужна для устранения интерференции или шума, получения спектра данных или преобразования сигнала в более удобную форму. В настоящее время ЦОС используется во многих областях, где раньше применялись аналоговые методы, кроме того, появились совершенно новые области применения, где было сложно или невозможно пользоваться аналоговыми устройствами. Привлекательность ЦОС обусловлена такими основными преимуществами.

- *Гарантированная точность.* Точность определяется только числом задействованных битов.
- *Совершенная воспроизводимость.* Можно идентично воспроизвести каждый элемент, поскольку отсутствуют отклонения, обусловленные устойчивостью отдельных составляющих. Например, используя методы ЦОС, цифровые записи можно копировать или воспроизводить многократно без ухудшения качества сигнала.
- Отсутствует искажение характеристик из-за температуры или старости.
- Полупроводниковые технологии позволяют повысить надежность, уменьшить размеры, снизить стоимость, понизить энергопотребление и увеличить скорость работы.
- *Большая гибкость.* Системы ЦОС можно запрограммировать и перепрограммировать на выполнение различных функций без изменения оборудования. Это, пожалуй, одна из самых важных особенностей ЦОС.
- *Превосходная производительность.* ЦОС можно использовать для выполнения функций, которые невозможны при аналоговой обработке сигналов. Например, можно получить линейную фазовую характеристику и реализовать сложные алгоритмы адаптивной фильтрации.
- В некоторых случаях информация уже может быть записана в цифровом виде, и обрабатывать ее можно только методами ЦОС.

В ЦОС есть и свои недостатки. Однако благодаря новым технологиям значение этих недостатков постоянно уменьшается.

- *Скорость и затраты.* Проекты ЦОС могут быть дорогими, особенно при большой ширине полосы сигнала. В настоящее время скоростные АЦП/ЦАП (аналого-цифровые/цифроаналоговые преобразователи) либо слишком дороги, либо не обладают достаточным разрешением для большой ширины полосы. На данный момент для обработки сигналов в мегагерцевом диапазоне можно использовать только специализированные интегральные схемы, но они достаточно дороги. Более того, большинство устройств ЦОС еще не обладают достаточной скоростью и могут обрабатывать сигналы только со средней шириной полосы. Сигналы с шириной порядка 100 МГц все еще обрабатываются аналоговыми методами. Тем не менее устройства ЦОС становятся все более скоростными.

- *Время на разработку.* Пока вы не знакомы с методиками ЦОС, и у вас нет необходимых ресурсов (программных пакетов и т.д.), разработка средств ЦОС будет отнимать очень много времени, а в некоторых случаях будет почти невозможна. Острая нехватка специалистов в этой области хорошо известна. Однако ситуация меняется, так как уже многие выпускники вузов разбираются в цифровых методах, а коммерческие организации все чаще начинают использовать в своей продукции преимущества ЦОС.
- *Проблемы конечной разрядности.* В реальных ситуациях экономические соображения предписывают использовать в алгоритмах ЦОС ограниченное число битов. Если для представления переменной задействуется недостаточное число битов, в некоторых системах ЦОС это приводит к существенному снижению качества работы системы.

1.2. Области применения

ЦОС — это одна из наиболее быстро развивающихся отраслей современной электроники, которая применяется в любой области, где информация содержится в цифровом виде или контролируется цифровым процессором. К числу областей применения можно отнести следующие.

- **Обработка изображений**
 - распознавание образцов;
 - машинное зрение;
 - улучшение изображения;
 - факсимиле;
 - спутниковые карты погоды;
 - анимация.
- **Инструментальные средства/контроль**
 - спектральный анализ;
 - контроль положения и скорости;
 - снижение шума;
 - сжатие информации.
- **Речь/аудио**
 - распознавание речи;
 - синтез речи;
 - озвучивание текста;
 - цифровые аудиосистемы;
 - выравнивание.

■ Военные цели

- безопасная связь;
- работа с радарами;
- работа с сонарами;
- управление ракетами.

■ Телекоммуникации

- устранение эха;
- адаптивное выравнивание;
- транскодеры ADPCM;
- расширение спектра;
- видеоконференц-связь;
- передача данных.

■ Биомедицина

- наблюдение за пациентами;
- сканеры;
- карты электроэнцефалограммы мозга;
- анализ электрокардиограмм;
- хранение (улучшение) рентгеновских снимков.

■ Потребительские цели

- цифровые сотовые мобильные телефоны;
- универсальная мобильная система связи;
- цифровое телевидение;
- цифровые камеры;
- телефонная связь, музыка и видео через Internet;
- цифровые автоответчики, факсы и модемы;
- системы голосовой почты;
- интерактивные развлекательные системы;
- активная подвеска в автомобилях.

С одного взгляда на этот отнюдь не полный список можно понять огромное значение ЦОС. Об осознании значения ЦОС свидетельствует постоянное внедрение мощных средств ЦОС производителями полупроводниковой техники. В то же время не хватает инженеров, обладающих достаточными знаниями в данной сфере. Цель этой книги — обеспечить понимание методов ЦОС и их реализации, дать читателям возможность овладеть практическими знаниями по столь важному предмету.

1.3. Ключевые операции ЦОС

Существует несколько алгоритмов ЦОС, еще больше находится в стадии разработки или ждет своего открывателя. Однако для всех этих алгоритмов, включая самые сложные, необходимы одни и те же основные операции. Для начала будет полезно рассмотреть некоторые из них, чтобы оценить простоту реализации ЦОС. Итак, основные операции ЦОС — это свертка, корреляция, фильтрация, преобразования и модуляция. Все эти операции собраны в табл. 1.1, а ниже дано краткое описание каждой из них. Следует заметить, что для всех основных операций ЦОС потребуется выполнение только простых арифметических действий — умножения, сложения, вычитания и операции сдвига. Также отметим сходство между многими операциями.

1.3.1. Свертка

Свертка — это одна из наиболее используемых операций в ЦОС. Например, это основная операция цифровой фильтрации. Для двух данных конечных и причинных последовательностей $x(n)$ и $h(n)$ с длиной N_1 и N_2 соответственно их свертка определяется как

$$\begin{aligned} y(n) &= h(n) \circledast x(n) = \sum_{k=-\infty}^{\infty} h(k)x(n-k) = \\ &= \sum_{k=0}^{\infty} h(k)x(n-k), \quad n = 0, 1, \dots, (M-1), \end{aligned}$$

где символ $\circledast$ используется для обозначения свертки, а $M = N_1 + N_2 - 1$. Как будет видно из последующих глав, производители устройств ЦОС разработали процессоры для обработки сигналов, которые эффективно выполняют операции умножения с накоплением, задействованные в свертке. Пример линейной свертки двух последовательностей, изображенных на рис. 1.1, *а* и *б*, показан на рис. 1.1, *в*. В этом примере $h(n)$, $n = 0, 1, 2, \dots$, можно рассматривать как импульсную характеристику цифровой системы, а $y(n)$ — как отклик системы на входную последовательность $x(n)$. Численные значения свертки, т.е. $y(n)$, получены путем прямой оценки уравнения (1.1). Например, $y(1)$ получается следующим образом:

$$\begin{aligned} y(1) &= h(0)x(1) + h(1)x(0) + h(2)x(-1) + \dots + h(12)x(-11) = \\ &= 0 \times 1 + (-0,02) \times 1 + 0 \times 0 + \dots + 0 \times 0 = \\ &= -0,02. \end{aligned}$$

Значение свертки становится более очевидным, если рассматривать ее в частотных координатах и использовать тот факт, что свертка во временных координатах эквивалентна умножению в частотной области. Более подробное обсуждение свертки, включая ее свойства и графическую интерпретацию, дается в главе 5.

Таблица 1.1. Обзор основных операций ЦОС

1. *Свертка.* Для двух данных последовательностей конечной длины $x(k)$ и $h(k)$ с длиной N_1 и N_2 соответственно линейная свертка равна

$$\begin{aligned} y(n) &= h(n) \circledast x(n) = \sum_{k=-\infty}^{\infty} h(k)x(n-k) = \\ &= \sum_{k=0}^{\infty} h(k)x(n-k), \quad n = 0, 1, \dots, (M-1), \end{aligned} \quad (1.1)$$

где $M = N_1 + N_2 - 1$.

2. *Корреляция.*

- А. Для двух данных последовательностей $x(k)$ и $y(k)$ длины N с нулевыми средними значениями оценка их взаимной корреляции равна

$$\rho_{xy}(n) = \frac{r_{xy}(n)}{[r_{xx}(0)r_{yy}(0)]^{1/2}}, \quad n = 0, \pm 1, \pm 2, \dots, \quad (1.2)$$

где $r_{xy}(n)$ — оценка взаимной ковариантности, определяемая как

$$\begin{aligned} r_{xy}(n) &= \begin{cases} \frac{1}{N} \sum_{k=0}^{N-n-1} x(k)y(k+n) & n = 0, 1, 2, \dots \\ \frac{1}{N} \sum_{k=0}^{N+n-1} x(k-n)y(k) & n = 0, -1, -2, \dots \end{cases} \\ r_{xx}(0) &= \frac{1}{N} \sum_{k=0}^{N-1} [x(k)]^2, \quad r_{yy}(0) = \frac{1}{N} \sum_{k=0}^{N-1} [y(k)]^2. \end{aligned}$$

- Б. Оценка автокорреляции $\rho_{xx}(n)$ последовательности $x(k)$ длины N с нулевым средним значением задается как

$$\rho_{xx}(n) = \frac{r_{xx}(n)}{r_{xx}(0)}, \quad n = 0, \pm 1, \pm 2, \dots, \quad (1.3)$$

где $r_{xx}(n)$ — оценка автоковариантности, определяемая как

$$r_{xx}(n) = \frac{1}{N} \sum_{k=0}^{N-n-1} x(k)x(k+n), \quad n = 0, 1, 2, \dots$$

3. *Фильтрация.* Уравнение для фильтрации с конечной импульсной характеристикой (КИХ) имеет вид:

$$y(n) = \sum_{k=0}^{N-1} h(k)x(n-k), \quad (1.4)$$

где $x(k)$ и $y(k)$ соответственно — вход и выход фильтра, а $h(k)$, $k = 0, 1, \dots, N-1$ — коэффициенты фильтра.

4. *Дискретное преобразование Фурье.*

$$X(n) = \sum_{k=0}^{N-1} x(k)W^{kn}, \quad \text{где } W = \exp(-2\pi i/N). \quad (1.5)$$

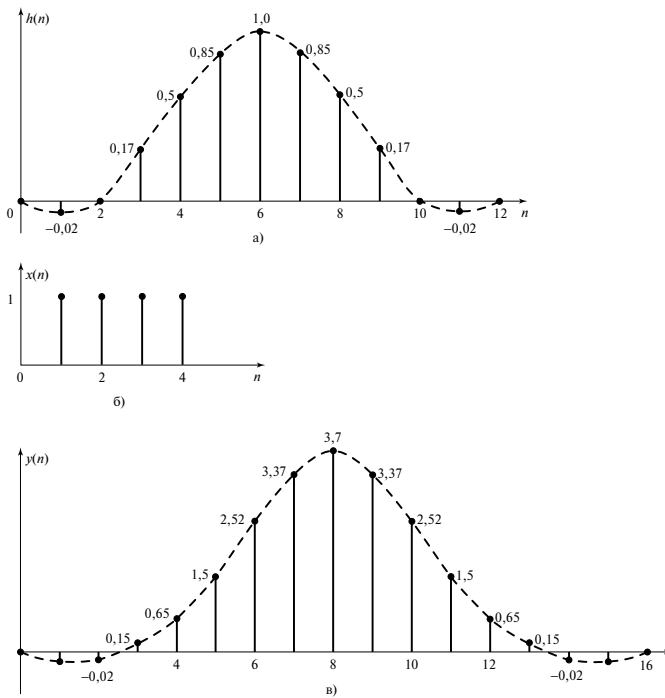


Рис. 1.1. Пример свертки двух последовательностей, $y(n)$ — это свертка $h(n)$ и $x(n)$. Если рассматривать $h(n)$ как импульсную характеристику системы, то $y(n)$ — это выход системы при входе $x(n)$. Значения $y(n)$ были получены непосредственно из уравнения (1.1)

1.3.2. Корреляция

Существует две формы корреляции: автокорреляция и взаимная корреляция.

Взаимно-корреляционная функция (ВКФ) — это показатель сходства или общих свойств двух сигналов. В число областей применения ВКФ входят: взаимный спектральный анализ, детектирование (восстановление) сигналов, скрытых в шуме, например, детектирование ответных сигналов радара, подбор по образцу и измерение задержки. ВКФ определяется по формуле (1.2).

Автокорреляционная функция (АКФ) подразумевает существование только одного сигнала и дает информацию о структуре сигнала или его поведении во времени. Это частный случай ВКФ, и их сферы применения сходны. Автокорреляционная функция, определяемая по формуле (1.3), особенно полезна для выявления скрытой периодичности.

Примеры ВКФ и АКФ некоторых сигналов приведены на рис. 1.2 и 1.3. Стоит заметить, например, что по АКФ искаженного шумом сигнала ясно видно, что за шумом скрывается периодический сигнал (рис. 1.2). На рис. 1.3 показано, как с помощью ВКФ измеряется задержка, которую вводит система, — ее можно узнать, измерив время от начала отсчета до большого максимума.

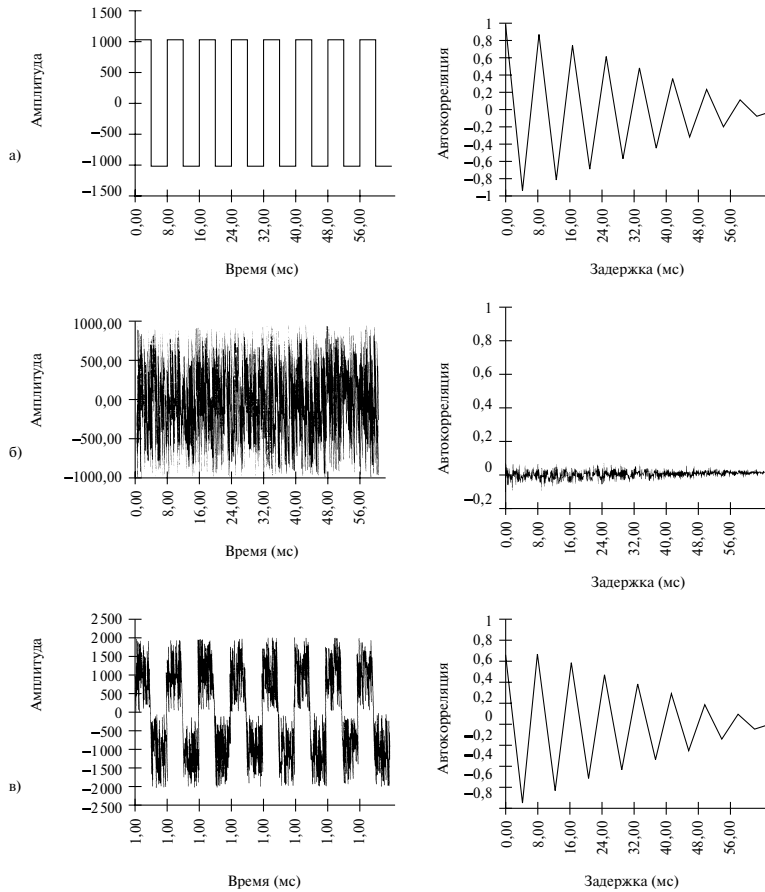


Рис. 1.2. Автокорреляция: а) периодического сигнала, б) шума; в) периодического сигнала с шумом. Обратите внимание на то, что на панели в периодический характер сигнала, скрытого за шумом, все еще заметен; по этой причине автокорреляцию часто используют для выявления скрытой периодичности

1.3.3. Цифровая фильтрация

Как станет очевидным при изучении последующих глав, одной из операций ЦОС, имеющих первостепенное значение, является линейная цифровая фильтрация, которая определяется как

$$y(n) = \sum_{k=0}^{N-1} h(k)x(n-k),$$

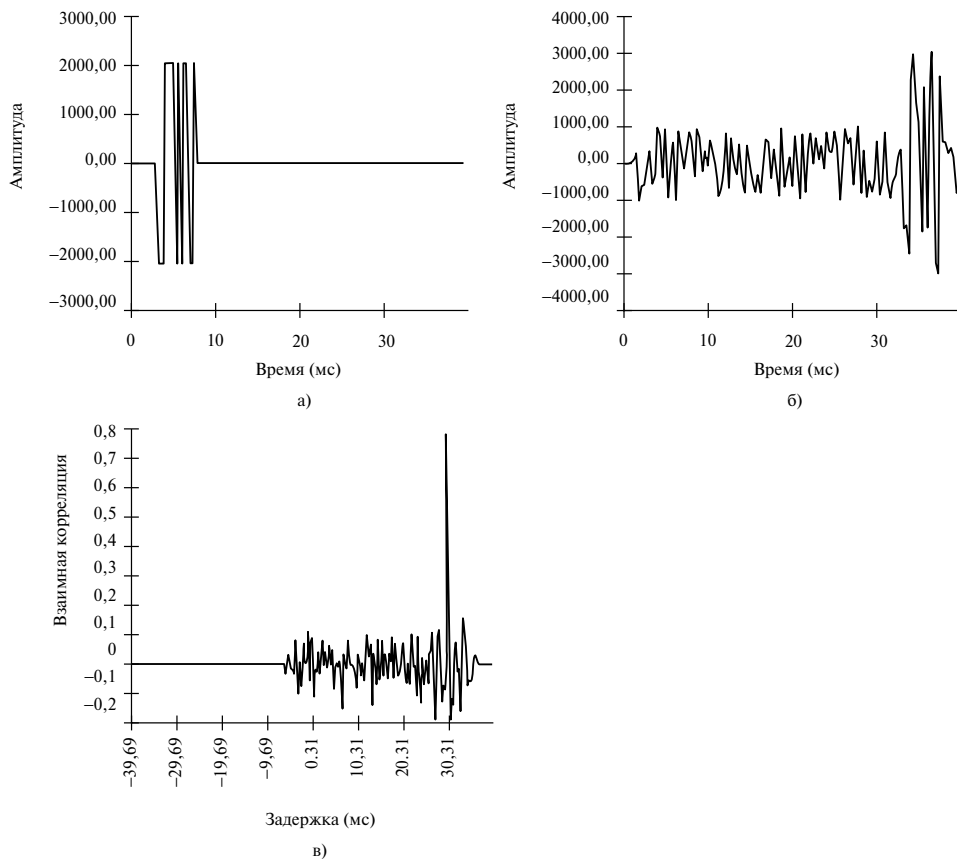


Рис. 1.3. Взаимная корреляция случайного сигнала $x(t)$ с задержанной зашумленной версией того же сигнала $y(t)$. Задержка между двумя сигналами равна времени от начала отсчета до времени появления максимума их взаимной корреляции на панели в

где $h(k)$, $k = 0, 1, \dots, N - 1$ — коэффициенты фильтра, а $x(k)$ и $y(k)$ соответственно — вход и выход фильтра. Для заданного фильтра коэффициенты, определяющие характеристики этого фильтра, однозначны.

Заметим, что фильтрация — это, по сути, свертка сигнала с импульсной характеристикой фильтра во временных координатах, т.е. $h(k)$. На рис. 1.4, а показана блок-схема фильтра, определение которого было дано выше. В таком виде этот фильтр широко известен как *трансверсальный* фильтр. На рисунке через z^{-1} обозначена задержка на один интервал дискретизации.

Как правило, целью фильтрации является устранение или снижение шума в полезном сигнале. Например, на рис. 1.4, б показана низкочастотная фильтрация некоторого биомедицинского сигнала для устранения высокочастотных искажений. Вообще в подобных сферах цифровой фильтр используется в основном для минимизации искажений внутриполосных компонентов сигнала.

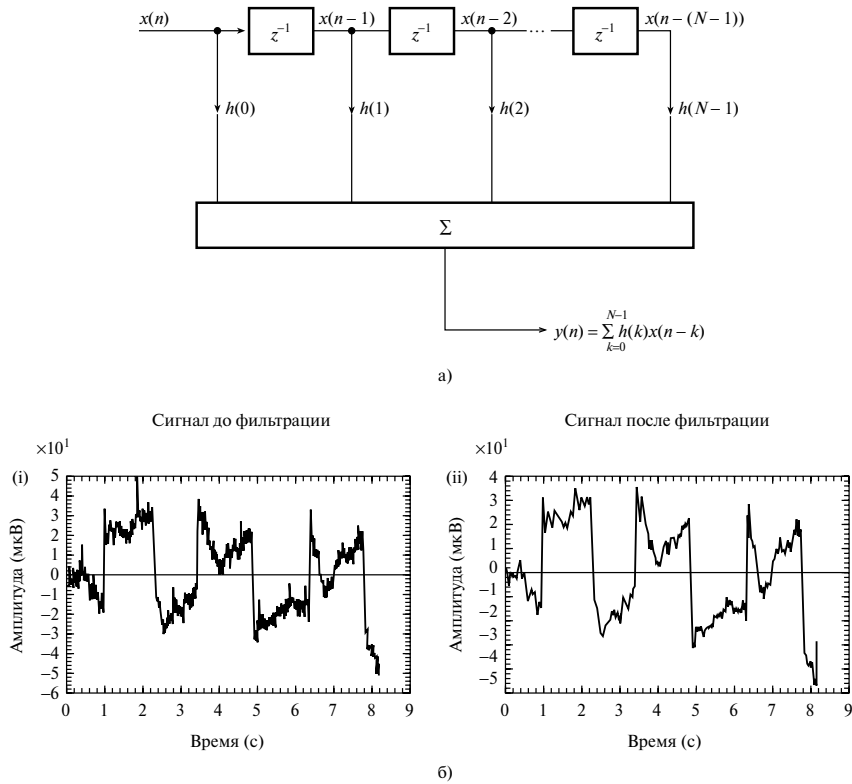


Рис. 1.4. Блок-схема трансверсального фильтра. Здесь $h(k), k = 0, 1, \dots, N-1$ — коэффициенты фильтра, а каждый квадрат с z^{-1} обозначает задержку на один интервал дискретизации (панель а); цифровая низкочастотная фильтрация биомедицинского сигнала с целью устранения шума (панель б)

1.3.4. Дискретные преобразования

Дискретные преобразования позволяют описывать сигналы с дискретным временем в частотных координатах или переходить от описания во временной области к описанию в частотной. Для получения спектра сигнал раскладывается на частотные составляющие с помощью дискретного преобразования. Знание такого спектра неопределимо, например, при определении ширины полосы, что необходимо для передачи сигнала. Переход от временных координат к частотным необходим во многих приложениях ЦОС. Например, он позволяет более эффективно реализовывать такие алгоритмы ЦОС, как цифровая фильтрация, свертка и корреляция.

Существует много дискретных преобразований, из которых самым распространенным является дискретное преобразование Фурье (ДПФ), которое определяется следующим образом:

$$X(n) = \sum_{k=0}^{N-1} x(k)W^{kn}, \text{ где } W = \exp(-2\pi i/N).$$

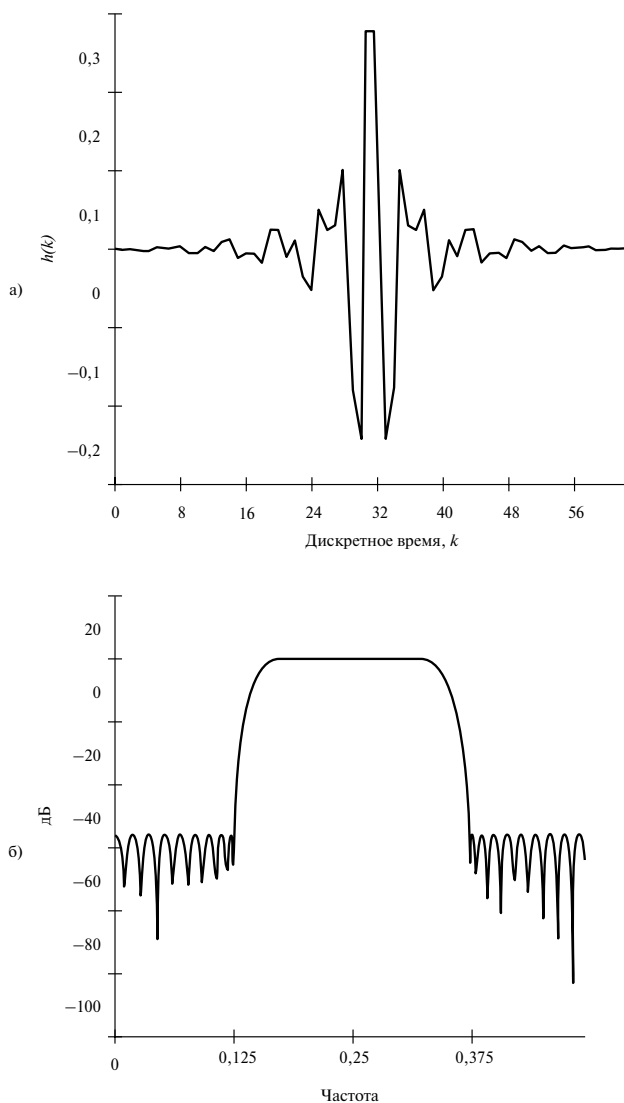


Рис. 1.5. Описание цифрового фильтра во временных и частотных координатах: а) импульсная характеристика; б) спектр фильтра. Спектр фильтра был получен с помощью дискретного преобразования $h(n)$, что иллюстрирует одно из возможных применений ДПФ

Пример использования ДПФ приведен на рис. 1.5. Здесь импульсная характеристика фильтра $h(n)$, $n = 0, 1, \dots, N - 1$, с помощью ДПФ преобразуется в частотную. Более подробно дискретное преобразование Фурье и его применение описано в главах 3, 4 и 11.

1.3.5. Модуляция

Цифровые сигналы редко передаются на большие расстояния или хранятся в большом объеме в необработанном виде. Обычно сигналы модулируются таким образом, чтобы их частотные характеристики совпадали с характеристиками средств передачи и/или хранения, для минимизации искажения сигнала, эффективного использования доступной ширины полосы и придания сигналам некоторых желаемых свойств. Наиболее интенсивно модуляция сигнала используется в связи и цифровых аудиосистемах.

Процесс модуляции часто приводит к изменению свойств высокочастотного сигнала, известного как *несущая частота*, в соответствии с сигналом, который нужно передать или сохранить, называемым *модулирующим сигналом*. Три самые распространенные схемы цифровой модуляции для передачи цифровой информации по широкополосному каналу (например, радиорелейной линии) — это амплитудная (amplitude shift keying — ASK), фазовая (phase shift keying — PSK) и частотная манипуляции (frequency shift keying — FSK). Если цифровые данные передаются по полностью цифровой сети, обычно пользуются схемой, известной как *импульсно-кодовая модуляция* (pulse code modulation — PCM) (см., например, [2]). Для цифровых аудиосистем было разработано несколько других схем модуляции, подробнее о них можно прочитать в работе [21].

1.4. Процессоры для цифровой обработки сигнала

Системы ЦОС характеризуются выполнением операций в реальном времени, причем акцент делается на высокой пропускной способности, а использование алгоритмов требует интенсивных арифметических операций, в особенности, умножения и сложения или умножения-накопления. Это приводит к большому потоку информации через процессор.

Структура стандартных микропроцессоров не удовлетворяет характеристикам ЦОС, и это послужило толчком для развития нового вида процессоров, структура и набор команд которых предназначены специально для операций ЦОС. Новые процессоры, или чипы ЦОС, имеют следующие особенности.

- Встроенные умножители, позволяющие быстро выполнять операции умножения. Новые чипы ЦОС содержат одноктактовые команды умножения-накопления, а в некоторых есть несколько умножителей, работающих параллельно.
- Отдельные шины/области памяти для программ и данных (хорошо известная гарвардская архитектура, которая позволяет перекрываться во времени процессам извлечения команд из памяти и их выполнения).
- Команды, содержащие циклы, которые используются для ветвления и образования циклов. Например, следующие команды для процессора TMS320C25 (Texas Instruments) значительно уменьшают количество циклов и размер программы цифрового фильтра.

RPTK N ; повторить следующую команду N раз

MACD ; занести данные в память, умножить и накопить с задержкой

- Очень большая скорость. Например, в процессоре TMS320C25 тактовая частота равна 40 МГц, а время одного такта — 100 нс.
- Использование конвейерной обработки, что сокращает время, необходимое для выполнения команд, и увеличивает скорость.

Новые чипы ЦОС быстрее и многофункциональнее. Сегодня некоторые из них обладают способностью выполнять арифметические действия с плавающей запятой и содержат такие черты стандартных микропроцессоров, как линии последовательной передачи, расширенная область памяти, таймеры и многоуровневые системы прерываний. Более подробно о чипах ЦОС и их применении рассказывается в главах 12–14.

1.5. Обзор реальных областей применения ЦОС

Методы ЦОС лежат в основе многих новых и только появляющихся продуктов и приложений, связанных с цифровой информацией, которые поддерживают существование информационного общества. Таким продуктам и приложениям часто приходится собирать, обрабатывать, анализировать, передавать, выводить на экран и/или сохранять реально существующую информацию, иногда в реальном времени. Возможности методов ЦОС обрабатывать реальную информацию в цифровом виде позволяют создавать доступные, технически прогрессивные продукты и приложения высокого качества для широкого потребительского рынка (например, цифровые сотовые мобильные телефоны, цифровое телевидение и видеоигры). Сильное влияние ЦОС также заметно в таких областях, как медицина и здравоохранение (например, при наблюдении за пациентами в интенсивной терапии, в цифровых рентгеновских аппаратах, передовом кардиологическом оборудовании и системах составления карт мозговой деятельности), цифровые аудиосистемы (например, проигрыватели компакт-дисков, звуковые микшеры и электронная музыка) и системы для персональных компьютеров (например, диски для эффективного хранения информации и коррекции ошибок, модемы, звуковые карты и организация видеоконференций).

Благотворное влияние методов ЦОС на то, как мы живем, работаем и отдыхаем, показано на рис. 1.6. В следующих трех разделах будут более подробно описаны некоторые технически прогрессивные приложения, причем основное внимание будет уделяться аспектам цифровой обработки данных в этих приложениях.

1.6. Применение ЦОС при записи и воспроизведении звука

1.6.1. Цифровое микширование

Цифровое микширование звука — это отличный пример успешного применения ЦОС для улучшения качества записи и воспроизведения звука, а также повышения функциональности системы.

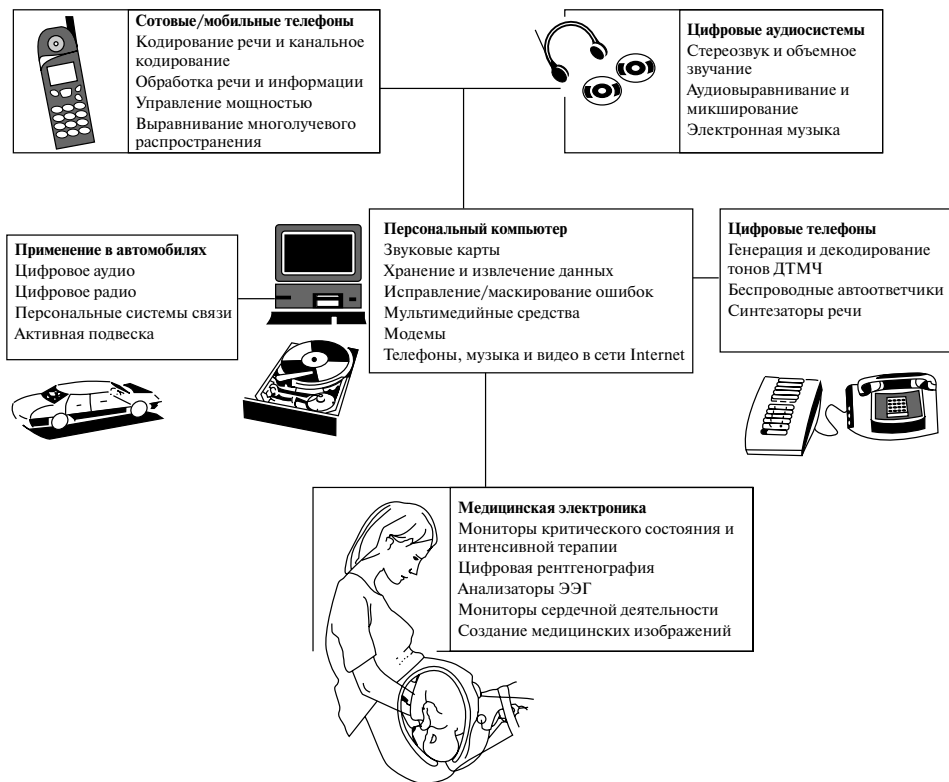


Рис. 1.6. Примеры применения методов ЦОС в современной жизни

Микширование звука применяется в профессиональных и полупрофессиональных аудиоприложениях, например, для студийной звукозаписи, радиовещания, при усилении звука и в системах публичных выступлений. Микшер позволяет регулировать, смешивать и выводить на экран характеристики многоканальных аудиосигналов от различных источников для подстройки их под требования определенных приложений.

Цифровая система микширования состоит из средств аудиовыравнивания, аудиомикширования и обработки после микширования (рис. 1.7). Цифровой аудиоэквалайзер — это набор цифровых фильтров с регулируемыми характеристиками, т.е. с возможностью выполнять действия над различными частями полос входного спектра аудиосистемы для достижения желаемого звука (например, усиливать или обрезать определенные тона) подобно настройке высоких и низких частот. Затем выровненные звуковые сигналы смешиваются с помощью матрицы смешивания (позволяет смешивать любой произвольный звуковой вход с любым выходом). После смешивания обработка сигнала может продолжаться путем реверберации и выравнивания.

Система микширования обладает такими интерактивными средствами контроля для управления параметрами микшера, такими как значения операторов микшера (регуляторов уровня сигнала) и параметры контроля эквалайзера (частота, добротность и коэф-



Рис. 1.7. Упрощенная блок-схема цифровой стереосистемы микширования на 32 входа и 16 выходов

фициент усиления фильтра), в реальном времени. Одна из сложных проблем цифрового микширования звука — это достижение пользовательского контроля над параметрами микшера при относительно высокой скорости обработки данных без заметных искажений [6]. Каждый раз, когда пользователь передвигает рычаги управления, параметры микшера должны изменяться, чтобы соответствовать новым требованиям. Такая регулировка может привести к заметным искажениям, что недопустимо в профессиональных системах смешивания звука. Как говорится в работе [6] и в главе 12, тщательное следование алгоритму смешивания является главным условием достижения профессионального уровня звучания.

На рис. 1.8 показан обычный микшер. Его характерные особенности:

- 8 моновходов (8 микрофонов или 8 линий);
- 2 пары стереоканалов (левый и правый) сигнала;
- главная кодовая шина для централизованного контроля, например, выбора области памяти, усиления по мощности и обработки сигнала.

В цифровом микшере, изображенном на рис. 1.8, используется современный процессор ЦОС, позволяющий реализовывать новейшие алгоритмы, необходимые при смешивании цифровых сигналов (например, выравнивание, стробирование шума, динамический контроль) и выполнении операций после смешивания.

1.6.2. Синтез и распознавание речи

1.6.2.1. Синтез речи

В прошлом синтезированная речь воспринималась как звучание механического голоса. Однако прогресс в области полупроводниковых технологий и ЦОС сделали экономически возможным достижение такого качества речи, что ее нельзя отличить от человеческой.

Примером успешного коммерческого продукта с речевым выходом является система Speak and Spell [11], с которой, вероятно, знакомы многие читатели. Это электронное пособие для обучения детей, в котором используется метод линейного кодирования с пред-



Рис. 1.8. Средства управления на обычном 8-канальном микшере (напечатано с разрешения компании Allen & Heath, Корнуолл, Великобритания)

сказанием (linear predictive coding — LPC), где настоящая человеческая речь, которую нужно воспроизвести, моделируется как отклик переменного во времени цифрового фильтра на периодический или случайный сигнал возбуждения (рис. 1.9). Периодическое возбуждение используется для генерации голосовых звуков (например, гласных) и представляет поток воздуха через вибрирующие голосовые связки. Случайное возбуждение используется для неголосовых звуков (таких, как С, Ш) и представляет шум, создаваемый при проталкивании воздуха через преграды в речевом канале. Человеческая речь содержит очень много лишней информации. Кодер оставляет только важную информацию, необходимую для сохранения таких характеристик речи, как интонация, акцент и диалект, позволяя держать целые минуты звука высокого качества в памяти среднего размера.

В пособии Speak and Spell используется микросхема синтезатора речи TMS5100, который объединяет все элементы модели LPC (цифровой фильтр и источники возбуждения), а также декодер и 8-битовый цифроаналоговый преобразователь (ЦАП). Микросхема синтезатора действует в тесной связи с 4-битовым микропроцессором и двумя 128-килобитовыми ПЗУ, которые вместе вмещают словарь из приблизительно 300 слов и фраз (рис. 1.10). Информация о речи хранится в ПЗУ в форме кадров (содержащих 25 мс речи), каждый кадр характеризуется набором из 10 или 12 параметров LPC. Параметры кадра сообщаются синтезатору каждые 25 мс и используются для обновления коэффициентов цифрового фильтра и выбора источника возбуждения и его уровня энергии. Выход цифрового фильтра преобразуется в аналоговый и подается на громкоговоритель для создания требуемого звука с определенным тоном, амплитудой

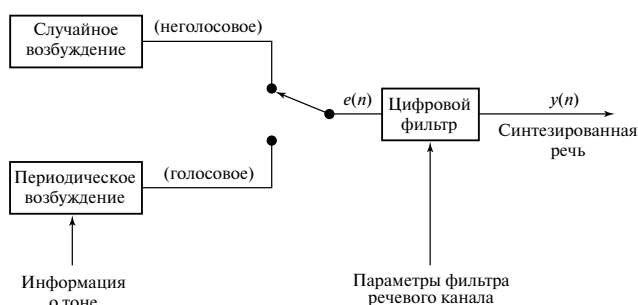


Рис. 1.9. Линейное кодирование речи с предсказанием

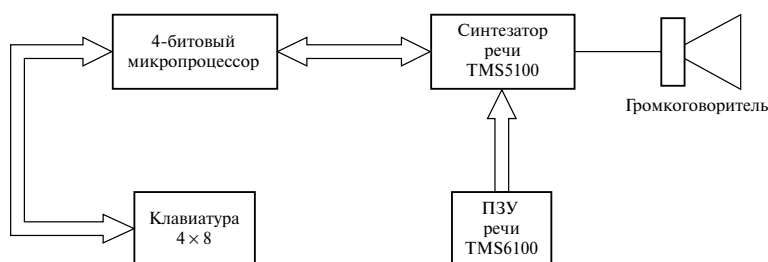


Рис. 1.10. Структура пособия Speak and Spell

и гармоническим содержанием. Для сглаживания переходов в речевом спектре каждые 3 мс синтезатор обновляет параметры LPC, проводя интерполяцию между параметрами предыдущего и следующего кадра.

В первом режиме операции ребенка просят сказать слово по буквам. Ребенок вводит слово по одной букве с помощью клавиатуры. Если написание верно, то при нажатии клавиши <Enter> программа отвечает “Правильно” или “Верно”. Если слово написано неверно, программа говорит “Неправильно, попробуй еще раз”. Если же и следующая попытка неверна, она указывает: “Это неправильно” и добавляет “Правильно это слово пишется так: ...”.

1.6.2.2. Распознавание речи

Под распознаванием голоса подразумевается, что информация вводится в компьютер с помощью человеческого голоса, а компьютер слушает и распознает человеческую речь. Распознавание голоса все еще активно изучается, так как поставленные задачи намного сложнее, чем те, которые возникают при синтезе речи. Поэтому успешные коммерческие системы распознавания речи немногочисленны и встречаются редко. Самыми удачными из них оказались настроенные на диктора системы распознавания изолированных слов. Такие системы работают в одном из двух режимов. В режиме обучения пользователь обучает систему распознавать его или ее голос, произнося каждое слово, подлежащее распознаванию, в микрофон. Система оцифровывает и создает эталон

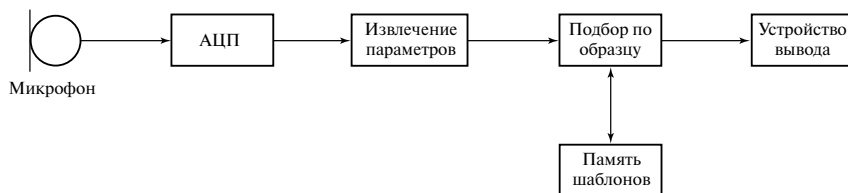


Рис. 1.11. Блок-схема системы распознавания речи

каждого слова, сохраняя его в памяти. В режиме распознавания каждое произнесенное слово снова оцифровывается, и его эталон сравнивается с эталонами из памяти. Если есть соответствие, значит, слово распознано, и система сообщает об этом пользователю или выполняет какое-то действие. Работа таких систем зависит от того, делает ли говорящий достаточно длинные паузы перед каждым словом, существует ли некий шумовой фон и от того, насколько четко и ясно произносятся слова. Двумя самыми важными операциями ЦОС при распознавании являются извлечение параметров, когда из сказанного слова получаются отдельные образцы и создаются эталоны, и подбор по образцу, когда эталоны сравниваются с записями, которые хранятся в памяти (см. рис. 1.11).

Для большинства людей разговор — это самая естественная форма общения, ведь говорить намного быстрее, чем писать или печатать. Поэтому сейчас в офисах есть системы, которые позволяют управлять программами не нажатием клавиш, а произнесением команд. Разрабатываются системы, которые позволяют создавать и отправлять с помощью голоса обычные офисные документы, такие как письма и служебные записки. Системы распознавания слов встречаются в товарах широкого потребления, таких как системы набора телефонного номера с речевым управлением, и в управляемых голосом предметах домашнего обихода для инвалидов с ограниченной способностью передвижения. Это делает таких людей более независимыми, позволяет самостоятельно выполнять некоторые простые действия, например, включать или выключать свет, радио или телевизор.

Конечно, существует еще множество возможных применений распознавания речи. Впрочем, дальнейшее развитие в этой области зависит, главным образом, от технологий искусственного интеллекта, поскольку машины должны не только распознавать, но и понимать человеческую речь.

1.6.3. Цифровая аудиосистема воспроизведения компакт-дисков

Многим читателям знакомы неприятные звуки, доносящиеся из проигрывателя, если пластинка повреждена либо на ней есть царапины, грязь или отпечатки пальцев. Компакт-диск — это современная аудиосистема, у которой уже нет недостатков долгоиграющих пластинок. Основные характеристики пластинок и компакт-дисков сравниваются в табл. 1.2 (согласно [4]).

На компакт-дисках информация записывается в цифровой форме в виде спиральной дорожки, которая состоит из последовательности впадин (рис. 1.12) [5]. Каждый бит, записанный на компакт-диске, занимает площадь всего лишь 1 мкм^2 , т.е. 10^6 бит на квадратный миллиметр, что обеспечивает очень высокую плотность хранения информации.

Таблица 1.2. Сравнение характеристик долгоиграющей пластинки и компакт-диска

<i>Характеристика</i>	<i>Долгоиграющая пластинка</i>	<i>Компакт-диск</i>
Частотная характеристика	От 30 Гц до 20 кГц (± 3 дБ)	От 20 Гц до 20 кГц (от 0,5 до -1 дБ)
Динамический диапазон	70 дБ (при 1 кГц)	> 90 дБ
Соотношение сигнал-шум	60 дБ	> 90 дБ
Гармоническое искажение	1–2 %	0,004 %
Разделение стереоканалов	25–30 дБ	> 90 дБ
Коэффициент детонации	0,03 %	Не регистрируется
Действие пыли, царапин и отпечатков пальцев	Вызывает шум	Приводит к исправимым или маскируемым ошибкам
Долговечность носителей	Высокочастотная характеристика при проигрывании ухудшается	Почти вечные
Время жизни иглы	500–600 ч	Почти бесконечно
Длительность звучания	40–45 мин. (обе стороны)	50–75 мин. (или больше)

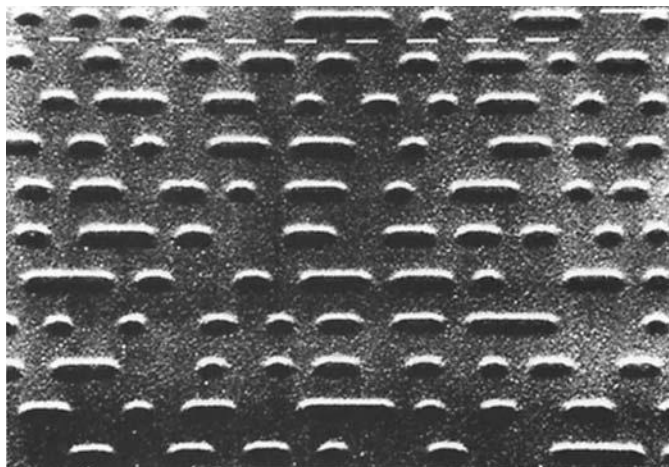


Рис. 1.12. Выжженные лазером впадины на компакт-диске. Ширина каждой впадины — 0,5 мкм, длина — 0,8–3,5 мкм, а глубина — 0,11 мкм. Расстояние между дорожками составляет 1,6 мкм. (Перепечатано с разрешения журнала *Philips Technical Review*, 40(6), 1982)

Упрощенная блок-схема обработки звукового сигнала на компакт-диске во время записи изображена на рис. 1.13. Аналоговый звуковой сигнал в каждом стереоканале дискретизируется с частотой 44,1 кГц и оцифровывается. Каждая выборка представляется 16-битовым кодом, множество всех кодов образует динамический диапазон 90 дБ. Таким образом, в каждый момент дискретизации получается 32 бит, по 16 бит с левого и правого звуковых каналов. Цифровые выборки кодируются с помощью двухуровневого кода Рида–Соломона, что дает возможность обнаруживать и исправлять или скрывать ошибки при воспроизведении звукового сигнала. Для контроля и отображения информации для слушателей добавляются дополнительные биты. Затем полученные потоки битов модулируются для преобразования в форму, более удобную для хранения на диске. С помощью EFM-модуляции (преобразования 8-разрядного кода в 14-разрядный) каждый байт потока данных преобразуется в 14-разрядный код. Полученный поток битов в канале после дальнейшей обработки используется для управления лазерным

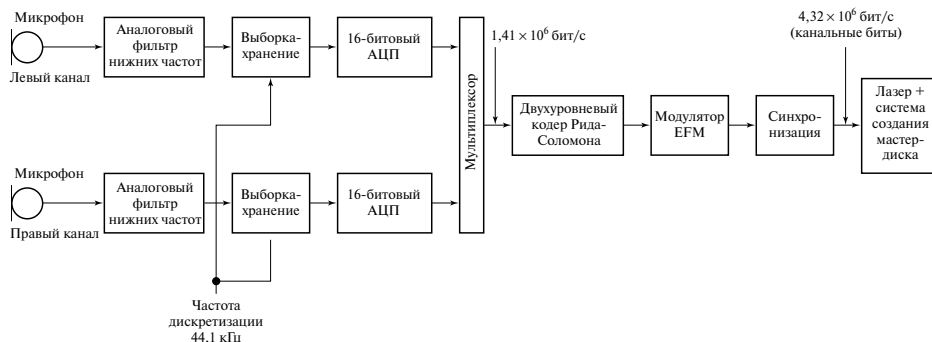


Рис. 1.13. Упрощенная блок-схема обработки аудиосигнала и записи его на компакт-диск

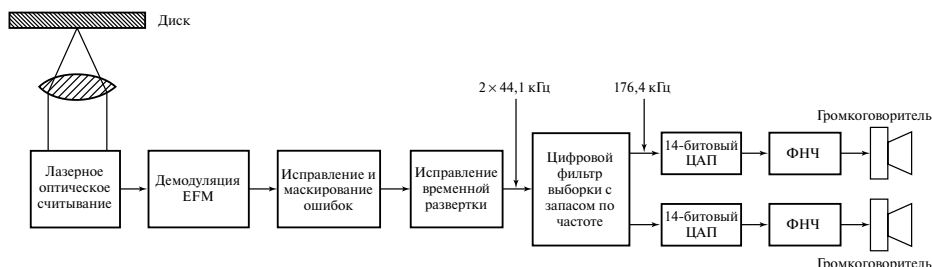


Рис. 1.14. Воспроизведение звукового сигнала с компакт-диска

пучком, который записывает цифровую информацию на светочувствительный слой вращающегося стеклянного диска. С помощью процесса проявки создается узор из впадин на мастер-диске, с которого впоследствии делаются пользовательские компакт-диски.

Во время воспроизведения записи на проигрывателе компакт-дисков дорожки на диске оптически сканируются с постоянной скоростью 1,2 м/с, в то время как диск вращается со скоростью от 8 об./с до приблизительно 3,5 об./с, и таким образом считывается записанная информация (рис. 1.14). Прежде всего, цифровой сигнал с диска демодулируется, детектируется, и по возможности в нем исправляются ошибки. Ошибки могут возникнуть вследствие производственных дефектов, повреждений, наличия отпечатков пальцев или пыли на поверхности диска. Если ошибки неисправимы, они либо маскируются посредством замены неправильного элемента выборки новым, полученным путем интерполяции между соседними элементами, либо, если ошибка обнаружена более чем в одном элементе, обнуляются (приглушаются).

В результате полученные данные представляют собой ряды из 16-битовых слов, каждое из которых является элементом выборки звукового сигнала. Эти элементы могут непосредственно направляться в 16-битовый цифроаналоговый преобразователь, а затем в аналоговый фильтр нижних частот. Однако для этого понадобились бы аналоговые фильтры с весьма строгими спецификациями. В частности, уровень частот, лежащих за 20 кГц, должен быть хотя бы на 50 дБ меньше максимального, а фильтр должен

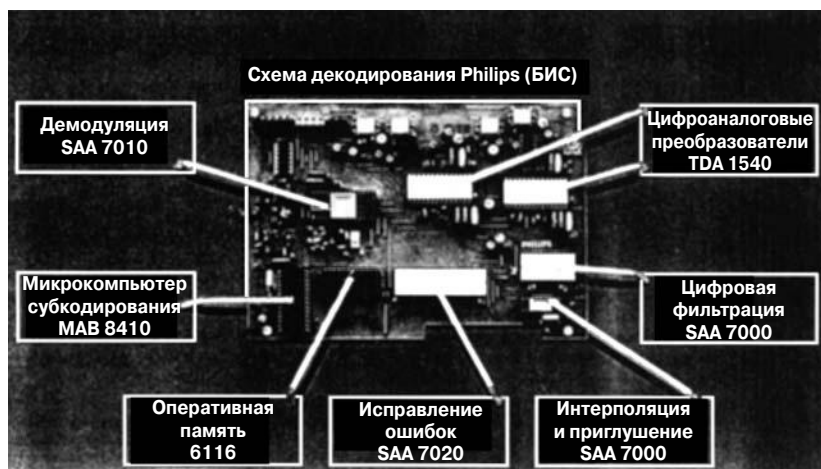


Рис. 1.15. Печатная плата декодирующей схемы проигрывателя компакт-дисков фирмы Philips. (Перепечатано с разрешения журнала *Philips Technical Review*, 40(6), 1982)

иметь линейную фазовую характеристику во избежание ухудшения звукового сигнала. Для этого цифровые сигналы предварительно пропускаются через цифровой фильтр, работающий с частотой, в четыре раза большей, чем частота дискретизации 44,1 кГц. Эффект повышения частоты дискретизации помогает сглаживать выход цифроаналогового преобразователя, упрощает требования к аналоговым фильтрам. Он также позволяет достичь 16-битового отношения сигнал-шум с помощью 14-битового цифроаналогового преобразователя. Использование цифрового фильтра также способствует получению линейной фазовой характеристики, снижает вероятность взаимной модуляции и дает фильтр с характеристикой, которая изменяется с тактовой частотой, делая его нечувствительным к скорости вращения диска. На рис. 1.15 изображена печатная плата декодирующей схемы первого поколения проигрывателей компакт-дисков фирмы Philips. Хорошо различимы основные микросхемы.

Кроме работы с компакт-дисками, цифровая обработка сигналов также играет большую роль и в других областях цифровой аудиотехники. Она широко используется как в товарах массового потребления, так и для профессиональной работы, например, на студиях звукозаписи, при передаче телепрограмм телерадиокомпаниями, при создании фильмов и в музыкальной индустрии. Ниже перечислены специальные области применения ЦОС в цифровой звукозаписи и звуковоспроизведении.

- Использование передовых технологий ЦОС для кодирования, выявления, исправления или маскировки ошибок при воспроизведении, для того, чтобы ограничения носителя информации (магнитного или оптического) не влияли на уровень качества, которого можно достичь при записи и воспроизведении. Таким образом, одна и та же запись будет звучать одинаково на носителях различного качества, с одинаковой частотой появления ошибок.

- Расширение области слышимости и обогащение звука. Например, простые схемы цифровых фильтров применяются для создания эха, естественной реверберации и эффектов хорового звучания.
- Синтез звуков, напоминающих звучание музыкальных инструментов, и даже таких, которые не может издать ни один инструмент.
- На телевидении, в мультипликации и кинофильмах при создании и использовании различных звуковых эффектов, таких как выстрелы, шаги, аплодисменты, звуки машин, удары кулаком, чтобы усилить чувство реальности или сделать сцену более правдоподобной.
- Улучшение качества архивных записей или стенограмм.

1.7. Применение ЦОС в телекоммуникации

1.7.1. Цифровые сотовые мобильные телефоны

1.7.1.1. Введение

Сегодня мобильная связь — это одна из самых быстрорастущих отраслей в мире, мобильный телефон стал незаменимым средством постоянной связи в информационном обществе. По оценкам специалистов, в течение нескольких лет количество людей, пользующихся мобильными телефонами, превысит число тех, кто пользуется стационарными телефонными линиями. В некоторых странах, например, в Финляндии, это уже произошло. ЦОС — это одна из ключевых технологий, делающих возможной революцию в области мобильной связи. ЦОС интенсивно используется для обработки сигналов и информации на базовых радиостанциях и в самих мобильных телефонах (например, для кодирования речи, многоканального выравнивания, измерения силы сигнала, голосовых сообщений, защиты от ошибок, модуляции и демодуляции). Широко доступны ЦОС-микросхемы, предназначенные для беспроводной связи, благодаря чему в области мобильной связи появилась возможность выпускать доступные высококачественные товары широкого потребления.

В современных системах мобильной связи используется принцип цифровой сотовой радиосвязи, хотя в первом поколении мобильных телефонов, с которых началось развитие сотовой радиосвязи, использовались аналоговые методы обработки и передачи голосовых сигналов. К самым удачным системам аналоговой мобильной связи можно отнести системы Advanced Mobile Phone System (AMPS), которая использовалась в Северной Америке, Nordic Mobile Telephone (NMT), разработанную совместно Данией, Финляндией, Норвегией и Швецией, и Total Access Communication Systems (TACS), использовавшуюся в Великобритании. Первые системы были несовместимы друг с другом, поскольку разрабатывались, в основном, для отдельно взятой страны, и не могли в полной мере удовлетворить растущую потребность в мобильной связи. Современные цифровые сотовые сети предлагают более широкие возможности, расширена область покрытия, возросли качество, безопасность и надежность. Проиллюстрируем основные принципы мобильной связи на примере системы GSM (Global System for Mobile communication).

Таблица 1.3. Основные характеристики системы GSM 900

<i>Параметры сотовой системы мобильной связи</i>	<i>Показатели системы GSM 900</i>
Частота	
мобильный передатчик/приемник базовой станции	890–915 МГц
мобильный приемник/передатчик базовой станции	935–960 МГц
Дуплексное разделение по частоте	45 МГц
Частотный интервал канала	200 кГц
Количество каналов	124
Скорость передачи речи	13 Кбит/с (половинная скорость 5,6 Кбит/с)
Общая скорость передачи информации	270 Кбит/с
Способ передачи сигнала	Узкополосный CDMA
Характерный размер ячейки	300 м–35 км

Система GSM была первой полностью цифровой системой радиотелефонной связи, и сейчас считается фактически мировым стандартом цифровой мобильной связи. GSM была разработана в 1992 году, а к концу 1998-го у нее было более 130 миллионов пользователей в более чем 100 странах мира. По прогнозам, к 2005 году количество абонентов мобильной связи буде превышать 1 миллиард. Значительный процент из них будут составлять пользователи GSM. Некоторые характеристики системы GSM 900, которой пользуются в Европе, приведены в табл. 1.3.

1.7.1.2. Архитектура сотовой телефонной сети

Мобильная сотовая телефонная сеть — это двусторонняя телефонная система, которая позволяет отправлять и принимать информацию (речь, данные и сообщения) с помощью мобильного телефона через радиоканалы. В сотовых радиосистемах область покрытия делится на небольшие единицы, называемые *ячейками* (или *сотами*), каждая из которых обслуживается своей базовой радиостанцией. Доступная полоса частот делится на радиочастоты, или *каналы*, с каждой ячейкой связан набор радиочастот. Радиочастоты в пределах области покрытия используются многократно, чтобы эффективно использовать доступную полосу частот, как показано на рис. 1.16 [17, 18]. Для минимизации внутриканальной интерференции между базовыми станциями, работающими на одних радиочастотах, ячейки с одинаковым набором частот располагаются как можно дальше друг от друга.

На практике размер ячейки зависит от интенсивности “движения”. В городах и густонаселенных территориях ячейки небольшие, они удовлетворяют требованиям интенсивного “движения” (часто используются микросоты с размерами, не превышающими 300 м). В сельской местности с низким уровнем “движения” могут использоваться ячейки размером до 35 км. Для таких больших ячеек необходима большая мощность передачи сигнала, при этом могут возникать дыры в области покрытия.

Связь между мобильным телефоном и сетью осуществляется через базовую радиостанцию. Сотовая сеть объединяет базовые радиостанции в единую радиотелефонную систему по всей области покрытия. Каждая базовая станция имеет определенное количество голосовых каналов и связана с мобильным коммутационным центром (mobile switching center — MSC); см. рис. 1.17 и работы [12, 17]. Некоторые коммутационные центры непосредственно связаны с коммутируемой телефонной системой (PSTN)

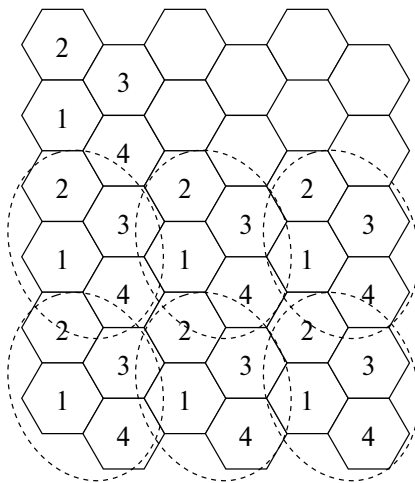


Рис. 1.16. Сотовый принцип, иллюстрирующий схему повторяющихся ячеек, возникающую из-за многократного использования частоты

и таким образом связывают сотовую сеть с PSTN. Базовая радиостанция состоит из контроллеров базовых станций (для управления радиоканалами), каждому из которых соответствует набор базовых станций-передатчиков. MSC содержит информацию о расположении мобильных телефонов и обеспечивает их мобильность.

Двумя ключевыми особенностями мобильной сотовой радиосети, которые позволяют пользователям звонить и принимать звонки на большой территории, являются ее способность определять местонахождение мобильного телефона и передавать линии радиосвязи с телефоном от одной базовой станции к другой. В сети сохраняется информация о местоположении каждого мобильного телефона, зарегистрированного в ней, так что входящие и исходящие звонки могут направляться к мобильному телефону или от него, где бы он не находился в пределах области покрытия. Это крайне важно, поскольку местонахождение пользователя мобильного телефона не фиксированно. При каждом включении мобильного телефона он регистрируется в сети, и это позволяет обновлять информацию о его местонахождении. У каждого мобильного телефона есть, по меньшей мере, два регистра местоположения в сети — домашний (HLR) и регистр местоположения посетителей (VLR). HLR размещается в собственной сети пользователя и содержит информацию о его профиле, в частности перечень услуг, которыми может пользоваться владелец данного телефона. В VLR записывается информация о мобильном телефоне, когда он используется за пределами собственной сети. При регистрации мобильного телефона в сети через ближайшую базовую станцию в сеть передается характерная только для этого телефона информация. Затем она используется для получения необходимых сведений о HLR данного телефона, чтобы он был опознан и получил доступ в сеть.

Как указывалось выше, радиосвязь между мобильным телефоном и сетью осуществляется через базовую станцию. Все базовые станции регулярно передают закодированную контрольную информацию. Мобильный телефон, не задействованный в разговоре,

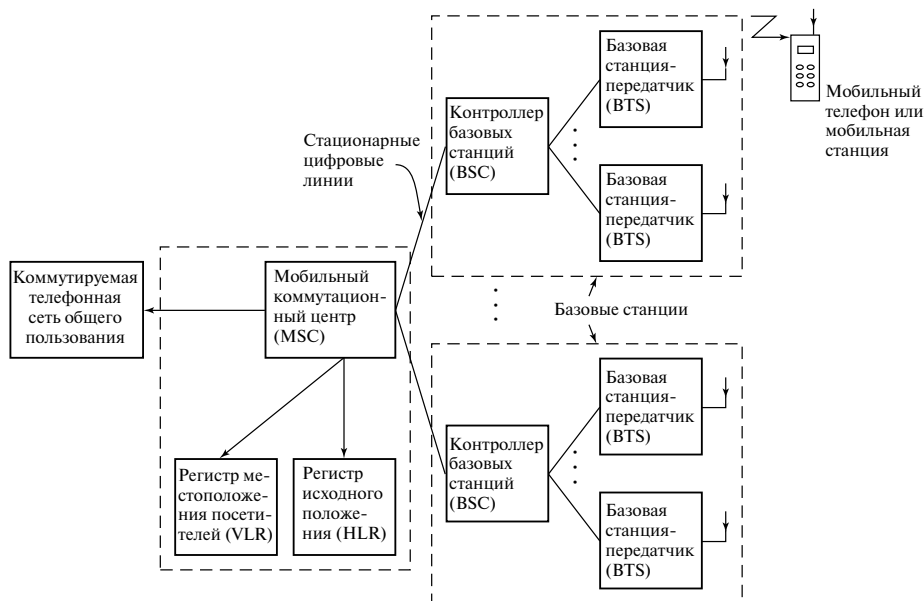


Рис. 1.17. Упрощенная блок-схема мобильной сотовой телефонной системы

ловит контрольный канал ближайшей базовой станции. Перемещаясь по области покрытия, мобильный телефон при необходимости ловит новую базовую станцию, и информация о его местонахождении обновляется. Осуществляется постоянный контроль за силой сигнала от мобильного телефона, по которому разговаривают. Если сила сигнала оказывается ниже определенного порогового значения, то сеть автоматически переключает радиосвязь на другую базовую станцию, которая обеспечит лучший прием, если такая есть. Возможность перебрасывать радиосвязь от одной базовой станции к другой при плохом сигнале — это как раз то, что позволяет пользователям мобильных телефонов свободно перемещаться в пределах области покрытия, не прекращая делать и принимать звонки. Этот процесс передачи занимает несколько секунд, хотя перерыв в разговоре на 200–300 мс уже может быть замечен. Как знает большинство пользователей мобильных телефонов, бывает, что в некоторых отдаленных областях поблизости нет такой базовой станции, которая может обеспечить лучшую связь. Если сила сигнала слишком мала, то голосовой канал обрывается, и пользователь не может делать или принимать звонки. Информация о силе сигнала отображается на экране мобильного телефона в том или ином виде (например, в виде нескольких полосок).

1.7.1.3. Элементы обработки сигнала

В современных сотовых системах радиотелефонной связи, таких как GSM, используются цифровые методы, и поэтому ЦОС — это естественный выбор для обработки и передачи информации. В мобильной связи ЦОС используется для кодирования речи, выравнивания сигналов после многолучевого распространения, измерения силы и ка-

чества сигнала, передачи голосовых сообщений, кодирования с исправлением ошибок, модуляции и демодуляции [17].

В системе GSM кодек речи основан на системе возбуждения регулярными импульсами с линейным кодированием с предсказанием (Regular Pulse Excitation, Linear Predictive Coding — RPE-LPC). В отличие от системы PSTN, в которой речь кодируется со скоростью 64 или 32 Кбит/с (адаптивная дифференциальная импульсно-кодовая модуляция), в мобильной радиотелефонной связи кодирование речи происходит с относительно невысокой скоростью 13 Кбит/с с целью эффективного использования спектра радиочастот. Алгоритм кодирования речи для системы GSM реализован в самых распространенных процессорах ЦОС (таких как Motorola ЦОС56000, Texas Instrument TMS320C50). Скорость обработки данных кодека — 13 Кбит/с, причем данный кодек заменяет 13-разрядные линейные цифроаналоговый и аналого-цифровой преобразователи.

Если мобильный телефон оказывается в неподходящей среде, часто возникают проблемы, связанные с многолучевым распространением. На частотах, которые используются для сотовой радиотелефонной связи, передаваемые сигналы часто могут отражаться от высотных зданий и т.п. Отраженный сигнал задерживается и поступает в принимающее устройство позже, чем неотраженный, поскольку преодолевает большее расстояние. Это приводит к флуктуациям амплитуды и фазы суммарного сигнала в приемнике, зависящим от природы многолучевого распространения и перемещения мобильного телефона. Эффект многолучевого распространения можно снизить, воспользовавшись цифровым выравниванием в приемнике. Для этого через регулярные промежутки передается известная последовательность сигналов длиной 26 бит. По окончании приема эквалайзер, используя эту опорную последовательность, определяет коэффициенты цифрового фильтра, оценивает характеристики полученного радиосигнала и таким образом устраняет эффект многолучевого распространения из полученных данных. Если известна передаточная функция, приемник может определить наиболее вероятную последовательность передаваемых битов и демодулировать сигнал. Алгоритм выравнивания GSM также реализован во многих процессорах ЦОС.

Кроме кодирования речи и выравнивания эффектов многолучевого распространения, методы ЦОС применяются также для цифровой модуляции, в частности, для обработки при нескольких скоростях. Также ЦОС дает возможность измерять силу полученного сигнала, чтобы переключать связь, и регулировать через базовую станцию уровень выходящей мощности мобильного телефона. В сети GSM мобильный телефон следит за сигналами от окружающих базовых станций. В системах сотовой радиосвязи всегда существует проблема внутриканальной интерференции между ячейками с одинаковым набором частот, которая зависит от передаваемой мощности. В системе GSM передаваемую мощность можно контролировать автоматически, снижая таким образом уровень внутриканальной интерференции и увеличивая время работы телефона без подзарядки. Решение об изменении передаваемой мощности принимается на основе уровня и качества принимаемого сигнала. Для того чтобы можно было оценить величину этих параметров, при анализе полученного сигнала применяются методы ЦОС.

При мобильной радиосвязи возникает много ошибок, например, обусловленных случайной интерференцией или затуханием. В системе GSM для уменьшения влияния этих ошибок используется метод сверточного кодирования.

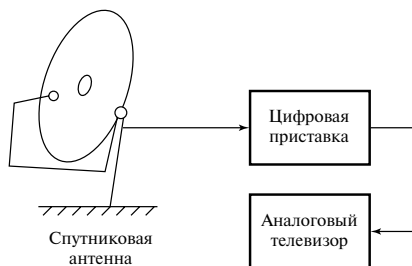


Рис. 1.18. Использование цифровой приставки для приема сигнала цифрового телевидения

1.7.2. Приставка для приема цифрового телевизионного сигнала

Цифровое телевидение возникло не так давно и много обещает потребителям — интерактивность, большой выбор, лучшее качество изображения и звука. Интерактивность дает абонентам возможность играть в игры, иметь доступ в сеть Internet, делать покупки, мгновенно выполнять повторное воспроизведение и т.д. Телевидение снова заявляет о себе как о неотъемлемой части информационного общества.

В цифровом телевидении цифровая информация (видео, аудио, текст) может передаваться на домашние телевизоры через спутник (с помощью спутника и существующих спутниковых антенн), по кабелю (кабельное телевидение) и по наземным линиям связи (с помощью существующих телевизионных антенн и передатчиков). Почти все телевизоры в наших домах могут принимать только аналоговый сигнал, поэтому для приема цифрового телевидения им нужен цифровой декодер (цифровая приставка) (рис. 1.18). Цифровая приставка преобразует цифровой сигнал в подходящий для приема аналоговым телевизором вид. Новейшие телевизоры оснащены встроенным декодером.

В цифровом телевидении ЦОС играет ключевую роль в обработке сигнала, кодировании и модуляции/демодуляции видео- и аудиосигналов от точки захвата сигнала до момента его появления на экране телевизора [3]. Без ЦОС замечательное качество изображения и звука, которое мы сейчас воспринимаем как должное, было бы невозможным. Например, ЦОС лежит в основе алгоритмов кодирования MPEG, которые используются для сжатия видео- и аудиоинформации перед их передачей (для эффективного использования ширины полосы). В цифровой приставке для декодирования информации используется декодер MPEG. Главный элемент алгоритма MPEG — это дискретное косинус-преобразование, подробно описанное в главе 3.

1.7.3. Адаптивное телефонное эхоподавление

Эффект эха в системах связи возникает преимущественно тогда, когда на пути сигнала встречается неподходящее сопротивление. На рис. 1.19 показана упрощенная схема телефонной цепи для больших расстояний. Гибридная схема телефонного аппарата преобразовывает двухпроводную линию, идущую от абонента, в четырехпроводную, и предоставляет отдельный тракт для передачи сигнала в каждом направлении. Это

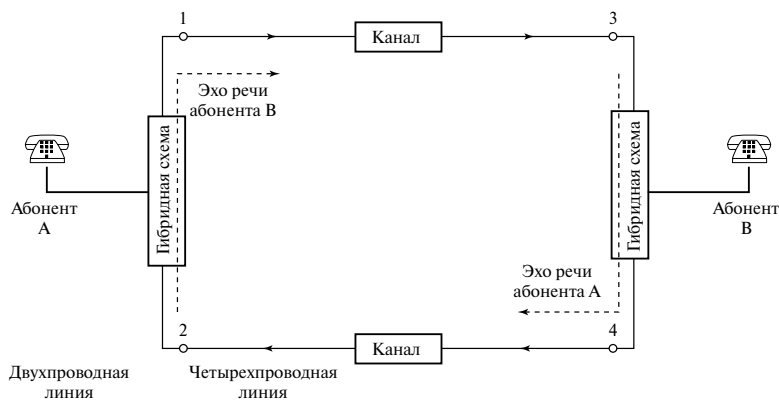


Рис. 1.19. Упрощенная схема телефонной цепи для больших расстояний

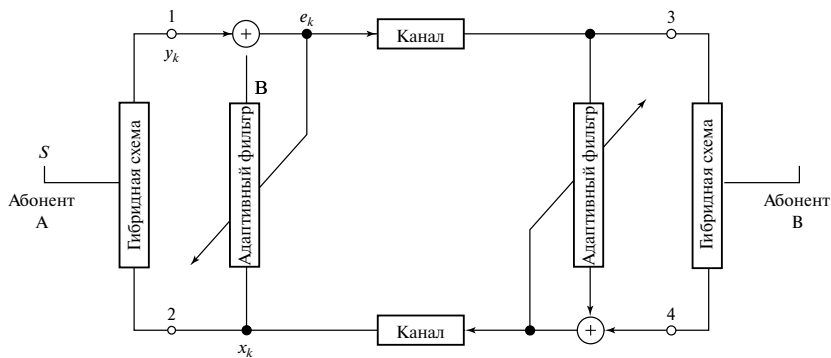


Рис. 1.20. Эхоподавление в междугородной телефонной связи

делается в основном из соображений экономии, например, чтобы сделать возможной множественную или одновременную передачу нескольких звонков.

В идеале, речевой сигнал от абонента А распространяется по верхнему тракту передачи к гибридной схеме справа, а оттуда к абоненту В, тогда как сигнал от пользователя В распространяется по нижней линии передачи к пользователю А. Гибридная схема на каждом конце обеспечивает подачу речевого сигнала от удаленного пользователя только на ее двухпроводный выход (на внешний выход сигнал не поступает). Однако из-за несоответствий в сопротивлении гибридная сеть пропускает некоторые входные сигналы на выход, которые слышны говорящему как эхо. Если телефонный звонок делается с большого расстояния (например, через геостационарный спутник), то эхо может задерживаться на 540 мс и вызывать помехи, раздражающие пользователей. Эти помехи будут возрастать с увеличением расстояния. Для решения этой проблемы в сети устанавливаются пары эхоподавителей, как показано на рис. 1.20 [9].

На каждом конце системы связи (рис. 1.20) входной сигнал x_k подается на гибридную схему и на адаптивный фильтр (АФ). Компенсация осуществляется с помощью оценки эха и вычитания его из возвращенного сигнала y_k . Оценка эха осуществляется по формуле

$$\hat{y}_k = \sum_{i=0}^{N-1} w_{k+1}(i) x_{k-i},$$

где x_k — это входной сигнал от дальнего абонента, а $w_k(i)$, $i = 0, 1, \dots, N-1$ — оценка импульсной характеристики тракта передачи эха в дискретный момент времени k .

1.8. Применение ЦОС в биомедицине

Биомедицина — это важная и очень продуктивная область как для применения традиционной цифровой обработки сигнала, так и для разработки новых мощных алгоритмов ЦОС. Часто медицинские данные не отличаются хорошими характеристиками, и это бросает вызов профессионалам в области ЦОС, которые должны предложить новый способ работы с данными. В большинстве случаев медицинские данные попадают в диапазон аудиочастот. Поэтому методы ЦОС, появившиеся благодаря биомедицинским проблемам, нашли применение и в других областях, таких как системы звукозаписи, звуковоспроизведения, телекоммуникации, и наоборот.

К ряду многочисленных приложений ЦОС в биомедицине относятся усиление сигнала и/или извлечение из него характеристик, представляющих медицинский интерес. Необходимость в усилении сигнала возникает из-за проблемы артефактов (ложных изображений) или загрязнения сигналов, что очень часто встречается в биомедицине. Артефакты могут возникать как из-за внешних источников (например, электросети или другого медицинского оборудования), так и из-за внутренних (движения головы и тела, мышечная и сердечная активность, движения глаз). Артефакты снижают медицинскую информативность сигналов и усложняют как ручной, так и автоматический его анализ, а в некоторых случаях даже делают его невозможным из-за сходства артефактов с сигналом, представляющим медицинский интерес [15].

Задачи усиления сигнала зачастую сопровождаются двумя неразрывно связанными проблемами: низкий уровень сигнала по сравнению с шумом и перекрывание спектров сигнала и шума. Следовательно, нужно приложить много усилий, чтобы минимизировать искажение сигнала, представляющего медицинский интерес (см. [20, 23]). В следующих двух разделах будут описаны два новых биомедицинских приложения ЦОС, которые подразумевают усиление сигнала и/или извлечение необходимых характеристик. Основные интересующие нас здесь биомедицинские сигналы имеют физиологическую природу, в частности это сигналы электрокардиограмм (электрической активности сердца) и электроэнцефалограмм (электрической активности мозга).

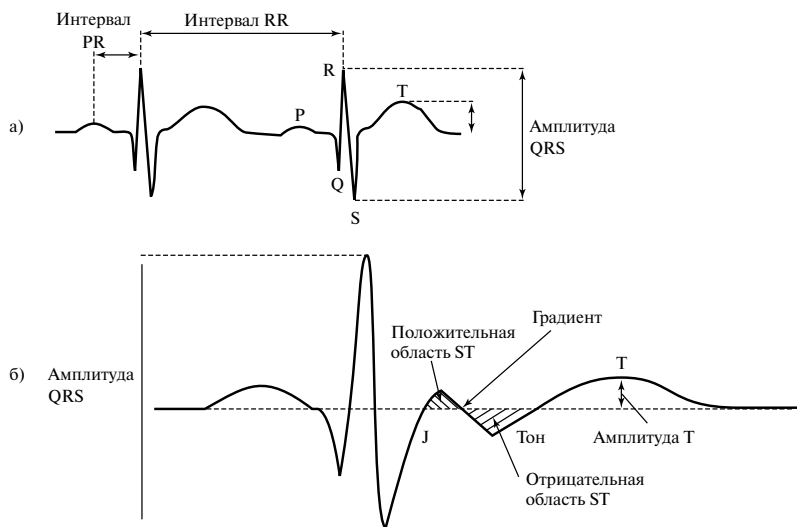


Рис. 1.21. ЭКГ плода, на которой показаны основные элементы, представляющие медицинский интерес

1.8.1. Изучение ЭКГ плода

Электрокардиограмма (ЭКГ) плода показывает электрическую активность сердца ребенка, которая измеряется на поверхности тела [20]. Частота сердцебиения плода (Fetal Heart Rate — FHR) определяется по расстоянию R-R на кардиограмме (рис. 1.21). Для оценки состояния плода во время родов обычно применяют визуальный анализ непрерывного изображения FHR вместе с сокращениями матки (маточной активности), которое называют *кардиотокограммой* (КТГ). Сложности, возникающие при интерпретации КТГ во время родов, могут привести к ненужному медицинскому вмешательству (например, кесаревому сечению или наложению щипцов), повреждению плода или неоказанию необходимой помощи [16].

Правильное применение комбинированного анализа ЭКГ плода и КТГ может значительно снизить процент ненужных медицинских вмешательств без каких-либо неблагоприятных последствий для младенцев [22]. В частности, для этого был создан коммерческий аппарат для ЭКГ плода, ST Analyser (STAN, Neoventa AB, Швеция).

Упрощенная схема снятия ЭКГ плода изображена на рис. 1.22. ЭКГ снимается с черепного электрода, чтобы получить хорошее отношение сигнал-шум, полоса ограничивается диапазоном 0,05–100 Гц, дискретизация проводится с точностью до 12 бит с частотой 500 раз в секунду. Данные ЭКГ плода обрабатываются для снижения шума и извлечения необходимых характеристик, а затем анализируются, чтобы можно было измерить и вывести на экран изменения формы сигнала, связанные с КТГ.

Важная характеристика ЭКГ плода — это форма сегмента ST. К числу существенных изменений формы сигнала, связанных со сжатием или растяжением, относятся постоянное увеличение амплитуды T сигнала, отрицательные значения волны T и по-

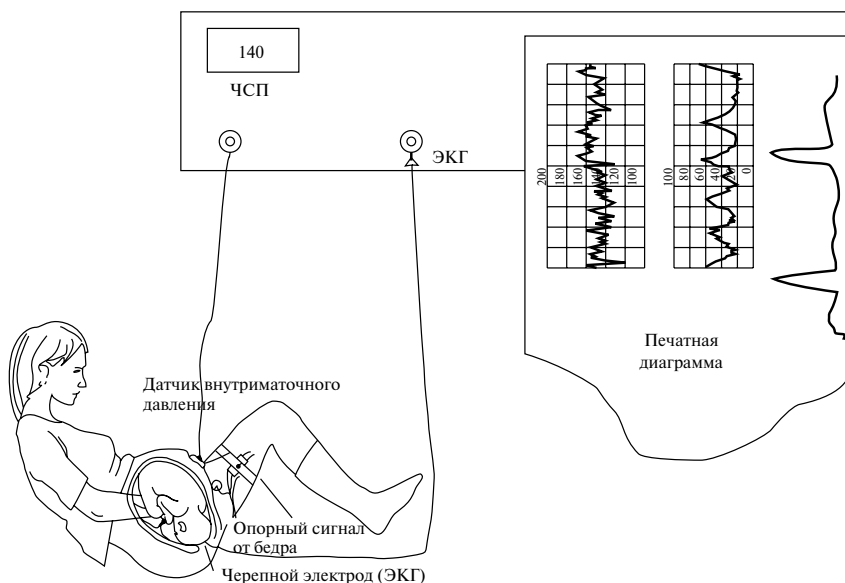


Рис. 1.22. Снятие ЭКГ плода во время родов

давление элементов ST. Изменения формы сигнала можно измерить как отношение амплитуды T к амплитуде QRS, которое называют *отношением T/QRS* (см. рис. 1.21). К числу остальных характеристик можно отнести область ST, изменения отрезков R-R, продолжительность P-волны и ширину комплекса QRS.

Черепная ЭКГ плода восприимчива к низкочастотному шуму и другим артефактам, дающим ложные изменения формы сигнала, например, шум от электросети, смещение базовой линии, мышечная активность или случайные шумы. Артефакты препятствуют извлечению необходимых характеристик и могут привести к неточностям анализа характеристик ЭКГ и формы сигнала.

Для снижения шума и извлечения основных характеристик из ЭКГ применяется множество методов обработки сигнала (включая подгонку кривой и цифровую фильтрацию при нескольких скоростях). Общая схема обработки ЭКГ плода показана на рис. 1.23. Первая главная задача — точно определить R-волну. Затем из необработанной ЭКГ удаляют смещение базовой линии, шум, вызванный мышечной активностью, и частоты питающей линии, чтобы получить форму сигнала, подходящую для надежного анализа. Пример оценки и устранения смещения базовой линии показан на рис. 1.24.

Очевидно, что исследование ЭКГ плода невозможно без ЦОС. Однако, хотя проблемы обработки сигнала, которые ранее препятствовали анализу ЭКГ плода, решены, исследование ЭКГ плода все еще ограничено в некоторых вопросах. В настоящее время анализ и интерпретация изменений в ЭКГ вместе с КТГ выполняются с помощью визуального наблюдения. Врачей нужно еще научить правильно снимать ЭКГ плода, но даже после соответствующего обучения существует вероятность того, что без постоянного

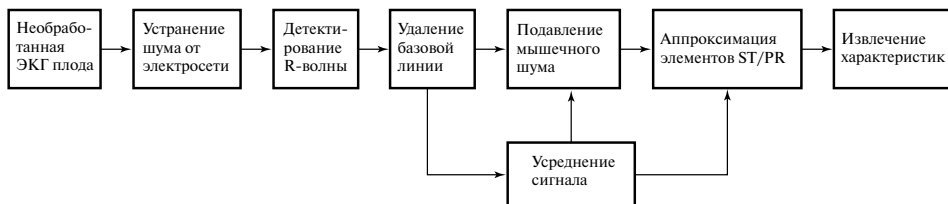


Рис. 1.23. Схема обработки сигнала ЭКГ плода

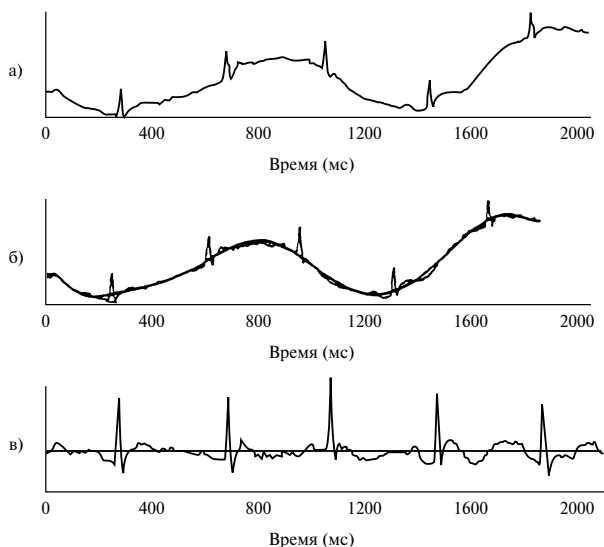


Рис. 1.24. Аппроксимация (панель а) и устранение (панель б) смещения базовой линии из необработанной ЭКГ плода

наблюдения специалиста могут быть пропущены важные особенности ЭКГ. Сейчас разрабатывается “сиделка” — разумная система контроля для непрерывного анализа и интерпретации изменений в ЭКГ и КТГ, которая будет помогать врачам (см. [15, 16]).

1.8.2. Анестезия с регулированием по замкнутому циклу на основе ЦОС

Огромное количество примеров применения ЦОС в медицине можно встретить в отделениях интенсивной терапии всех крупных больниц. Постоянно создаются новые современные методы работы с анестезией. Во время операции пациенты обычно находятся под наркозом, например, после введения анестезирующих препаратов внутривенно, так что они не чувствуют боли, и хирург может оперировать в спокойной обстановке. Задача анестезиолога — ввести правильную дозу препарата, чтобы как можно быстрее вызвать настолько глубокое состояние анестезии, насколько это нужно, и поддерживать это состояние без изменений на протяжении необходимого времени. Если ввести слишком большую дозу, это может привести к осложнениям или другим побочным эффектам,

а при недостаточной дозе пациент может очнуться во время операции, что также может иметь долговременные физиологические последствия [13]. В большинстве случаев глубину бессознательного состояния пациента оценивает опытный анестезиолог, наблюдая за клиническими признаками. При этом анестезиолог соответствующим образом изменяет дозу препарата, чтобы контролировать состояние анестезии. Автоматический ввод препаратов, основанный на методике регулирования по замкнутому циклу, дает большие преимущества загруженным работой анестезиологам и обеспечивает лучший уход за пациентом при меньших затратах. Эта методика снижает вероятность превышения дозы и дает анестезиологу возможность заметить и должным образом отреагировать на нарушения, которые могли бы остаться незамеченными или показаться слишком незначительными, чтобы послужить причиной ручного ввода препарата.

Однако для ввода наркотических препаратов с регулированием по замкнутому циклу необходимы надежные средства контроля глубины анестетического состояния, которые могли бы определять необходимые для поддержания этого состояния изменения в дозе. В современных анестетических системах с регулированием по замкнутому циклу для измерения глубины анестетического состояния используются биологические сигналы, также они применяются для организации обратной связи, по ним определяют поправки, которые нужно сделать при вводе препаратов. В частности, для обработки электроэнцефалограмм (ЭЭГ) применяются различные методы обработки сигнала, помогающие извлечь такие характеристики, как слуховой отклик (Auditory Evoke Response — AER) и биспектральный коэффициент, и по ним оценить глубину анестетического состояния [13]. ЭЭГ — это электрическая активность мозга, измеренная с помощью электродов, прикрепленных к голове, а слуховой отклик — это электрическая реакция мозга на внешний звуковой раздражитель. Сигналы AER незаменимы при определении момента перехода от бессознательного состояния в сознательное, но их сложно получить, поскольку они скрыты в сигналах ЭЭГ, которые в несколько раз превышают AER. Для их выявления часто пользуются усреднением сигналов отклика на эффективный слуховой раздражитель. Итак, сигналы AER необходимо обработать, извлечь их из фона ЭЭГ, а затем определить по ним характеристики, представляющие медицинский интерес (например, пики, времена задержки и форму). Биспектральный коэффициент определяется путем спектрального анализа ЭЭГ высшего порядка [19]. Он представляет собой количественную меру комплексных изменений и взаимосвязей между частотными компонентами ЭЭГ при различных уровнях сознания. Известно, что он линейно зависит от концентрации в крови пропофола, самого распространенного анестетического препарата.

Упрощенная блок-схема анестетической системы с регулированием по замкнутому циклу, основанной на ЭЭГ, изображена на рис. 1.25 (согласно [7, 8]). Главный элемент системы — это анализатор ЭЭГ, подсоединенный к пациенту через бифронтальные электроды для получения необработанных сигналов ЭЭГ. Для снижения шума, извлечения характеристик, анализа изменений этих характеристик и вычисления соответствующего показателя ЭЭГ в анализаторе используются различные методы обработки сигнала. К ним относятся импульсные преобразования, усреднение сигнала, биспектральный анализ и нейронные сети.

Рассчитанный показатель ЭЭГ, такой как биспектральный коэффициент, говорит о мере воздействия введенного препарата на пациента. Кроме того, рассчитанный показа-

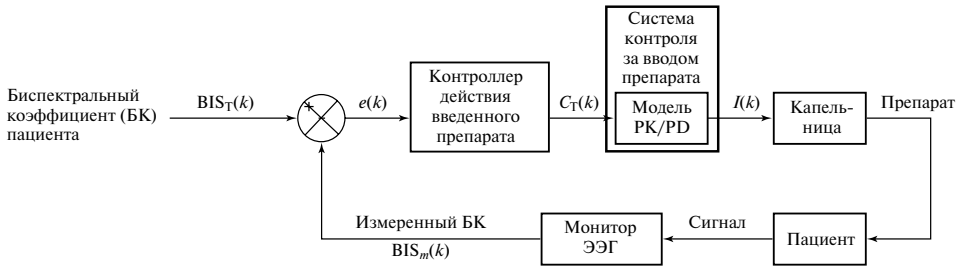


Рис. 1.25. Основанная на ЦОС анестетическая система с регулированием по замкнутому циклу

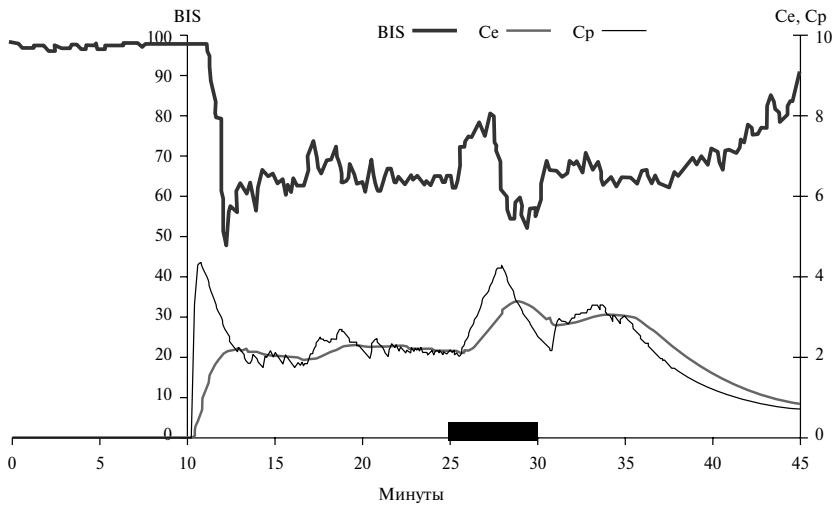


Рис. 1.26. Пример изменения биспектрального коэффициента пациента во время анестезии с регулированием по замкнутому циклу

тель ЭЭГ играет роль сигнала обратной связи, который сравнивается с показателем ЭЭГ пациента, чтобы определить изменения, которые нужно внести в концентрацию крови пациента $C_T(k)$. Для вычисления биспектрального коэффициента и других показателей (например, A-1000, Aspect Medical System) создан коммерческий анализатор ЭЭГ. С помощью фармакинетической/фармадинамической (PK/PD) модели определяют скорость введения препарата пациенту $I(k)$. Эта скорость зависит от концентрации крови пациента и других его характеристик, таких как возраст, вес и пол. Внутривенные препараты обычно вводят пациенту с помощью соответствующей анестетической капельницы под контролем специальной программы (например, STANPUMP®, S. L. Shafer, Stanford University). Пример изменения биспектрального коэффициента во время анестезии приведен на рис. 1.26.

1.9. Резюме

В этой главе рассказывается о значении цифровой обработки сигнала, обсуждаются области ее применения, и дается определение основных операций ЦОС.

Отдельные рассмотренные примеры применения показывают, что ЦОС уже оказывает значительное влияние на электронику профессионального и потребительского уровня.

Задачи

- 1.1. Назовите и обоснуйте два главных преимущества и два главных недостатка ЦОС по сравнению с системами аналоговой обработки сигнала.
- 1.2. Опишите с помощью блок-схемы процесс воспроизведения аудиосигнала в проигрывателе компакт-дисков. Назовите и обоснуйте четыре преимущества применения методов ЦОС для этой цели.
- 1.3. Что такое современная система GSM? Объясните, что она представляла собой в прошлом.
- 1.4. Опишите с помощью блок-схемы работу сотового радиотелефона.
- 1.5. Коротко расскажите о роли ЦОС в цифровой сотовой радиотелефонной связи. Почему ЦОС является естественным выбором для обработки голосовой информации в цифровых радиотелефонных системах? Какие проблемы возникают при использовании ЦОС в мобильных радиотелефонах?
- 1.6. Объясните, что представляет собой радиочейка. Изобразите схему трехкратного повторения.

Литература

1. Advanced Mobile Phone Service. *Bell System Technical Journal*, **58**(1), January 1979.
2. Bellamy J. C. (1982) *Digital Telephony*. New York: Wiley.
3. Benoit H. (1997) *Digital Television: MPEG-1, MPEG-2 and Principles of the DVB System*. London: Arnold.
4. Bloom P. J. (1985) High-quality digital audio in the entertainment industry: an overview of achievements and challenges. *IEEE ASSP Magazine*, October, 2–25.
5. Carasso M. G., Peek J. B. H. and Sinjou J. P. (1982) The compact disc digital audio system. *Philips Technical Rev.*, **40**(6), 151–156.
6. Clark R. J., Ifeachor E. C., Rogers G. M. and Van Eetvelt P. W. J. (2000) Techniques for generating digital equaliser coefficients. *Journal of Audio Engineering Society*, **48**(4), 281–298.
7. Dong C., Kehoe J., Henry J., Ifeachor E. C., Reeve C. D. and Sneyd J. R. (1998) Closed loop computer controlled sedation with propofol. *British Journal of Anaesthesia*, **81**, 631P.
8. Dong C., Reeve C. D., Sneyd J. R. and Ifeachor E. C. (1999) Closed-loop control of intravenous drug infusion. *IEE Proc. Sci. Meas. Technol.* (submitted).
9. Duttweiler D. L. (1978) A twelve-channel digital echo canceler. *IEEE Trans. Communications*, **26**, 647–653.
10. ETSI/GSM Recommendations: ETSI, BP 152, F-06561 Valbonne CEDEX, France.
11. Frantz G. A. and Wiggins R. H. (1982) Design case history: Speak and Spell learns to talk. *IEEE Spectrum*, February, 45–49.

12. Horrocks R. J. and Scarr R. W. A. (1993) *Future Trends in Telecommunications*. New York: Wiley.
13. Huang J. W., Lu Y., Nayak A. and Roy R. J. (1999) Depth of anaesthesia estimation and control. *IEEE Trans. Biomed. Eng.*, **46**(1), 71–81.
14. Ifeakor E. C., Hellyar M. T., Mapps D. J. and Allen E. M. (1990) Knowledge-based enhancement of EEG signals. *Proceedings of IEEE Radar and Signal Processing*, **37**, 302–310.
15. Ifeakor E. C. and Outram N.J. (1995) A fuzzy expert system to assist in the management of labour. *Proc. International ICSC Symposium on Fuzzy Logic*, ICSC, C97–102, Zürich, Switzerland.
16. Keith R. D. F., Beckley S., Garibaldi J. M., Westgate J., Ifeakor E.C. and Greene K.R. (1995) A multi-centre comparison study of 17 experts and an intelligent computer system for managing labour using the cardiotocogram. *Brit. J. Obstet. Gynaecol.*, **102**, 688–700.
17. Macario R. C. V. (ed.) (1991) *Personal and Mobile Radio Systems* (Chapters 4, 9, 13 and 14). Peter Peregrinus Ltd for the Institution of Electrical Engineers.
18. Macario R. C. V. (ed.) (1996) *Modern Personal Radio Systems* (Chapters 3, 8, 11 and 12). The Institution of Electrical Engineers, London.
19. Nikias C. L. and Raghuveer M. R. (1987) Bispectrum estimation: a digital signal processing framework. *Proc. IEEE*, **75**, July, 869–891.
20. Outram N. J., Ifeakor E. C., Van Eetvelt P. W. J. and Curnow J. S. H. (1995) Techniques for optimal enhancement and feature extraction of fetal electrocardiogram. *IEE Proc. Sci. Meas. Technol.*, **142**(6), November, 482–489.
21. Watkinson J. (1994) *The Art of Digital Audio*. Second edition. Oxford: Butterworth-Heinemann.
22. Westgate J., Harris M., Curnow J. and Greene K. R. (1993) Plymouth randomised trial of the cardiotocogram only versus ST waveform plus cardiotocogram for intrapartum monitoring in 2400 cases. *Am. J. Obstet. Gynecol.*, **169**, 1151–1160.
23. Wu J., Ifeakor E. C., Allen E. M., Wimalaratna S. K. and Hudson N. R. (1997) Intelligent artefact identification in electroencephalography signal processing. *IEEE Proceedings Science, Measurement and Technology*, **144**(5), 193–201.

Дополнительная литература

- Mitra S. K. (1998) *Digital Signal Processing — A Computer-Based Approach*. New York: McGraw-Hill.
- Mulgrew B., Grant P. and Thompson J. (1999) *Digital Signal Processing — Concepts and Applications*. Basingstoke: Macmillan.
- Oppenheim A. V. and Schaffer R. W. (1975) *Digital Signal Processing*. Englewood Cliffs NJ: Prentice-Hall.
- Oppenheim A. V. and Schaffer R. W. (1989) *Discrete-Time Signal Processing*. Englewood Cliffs NJ: Prentice-Hall.
- Orfanidis S. J. (1996) *Introduction to Signal Processing*. Englewood Cliffs NJ: Prentice-Hall.
- Papamichalis P. (1987) *Practical Approaches to Speech Coding*. Englewood Cliffs NJ: Prentice-Hall.
- Rabiner L. R. and Gold B. (1975) *Theory and Applications of Digital Signal Processing*. Englewood Cliffs NJ: Prentice-Hall.

Аналоговый интерфейс ввода-вывода для систем ЦОС реального времени

2

2.1. Характерные системы ЦОС реального времени	64
2.2. Процесс преобразования аналогового сигнала в цифровой	65
2.3. Дискретизация — низкочастотные и полосовые сигналы	66
2.4. Однородное и неоднородное квантование и кодирование	91
2.5. Выборка с запасом по частоте при аналого-цифровом преобразовании	96
2.6. Процесс цифроаналогового преобразования: восстановление сигнала	110
2.7. Цифроаналоговый преобразователь	111
2.8. Фильтры защиты от зеркальных частот	113
2.9. Выборка с запасом по частоте при цифроаналоговом преобразовании	113
2.10. Ограничения обработки сигналов в реальном времени с аналоговыми входными/выходными сигналами	117
2.11. Примеры применения	118
2.12. Резюме	119
Литература	133
Дополнительная литература	133

Во многих реальных приложениях сигналы изначально имеют аналоговую форму, а ЦОС оперирует цифровыми данными. Следовательно, для успешного взаимодействия систем ЦОС с реальным миром необходим аналоговый интерфейс ввода-вывода, позволяющий осуществлять переход от аналогового формата к цифровому. В системах ЦОС, работающих в реальном времени, аналоговый интерфейс ввода-вывода является слабым звеном, поскольку он вносит неустранимую погрешность и ограничивает скорость работы (см. далее). Главным условием успешной разработки системы ЦОС реального времени с аналоговым входом и/или выходом, а также систем, предназначенных для других областей ЦОС (например, для обработки при нескольких скоростях), является глубокое понимание вопросов, касающихся проектирования аналогового интерфейса ввода-вывода. Многие вопросы, связанные с проектированием подобных систем, рассмотрены в этой главе. В частности, прочитав эту главу, читатель будет

- 1) знать теоретические основы проектирования аналогового интерфейса ввода-вывода для систем ЦОС реального времени (например, теоремы о низкочастотной и полосовой дискретизации и то, как их применять для решения практических задач, а также природу ошибок, возникающих в аналоговом интерфейсе ввода-вывода);
- 2) уметь устанавливать, оценивать и определять основные параметры аналогового интерфейса ввода-вывода (например, частоту дискретизации и уровень искажения от наложения спектров);
- 3) понимать основные принципы метода выборки с запасом по частоте в аналоговом интерфейсе ввода-вывода (например, выборка с запасом по частоте и ограничение шума, проектирование и анализ простых преобразователей для выборки с запасом по частоте).

Для иллюстрации этих понятий мы воспользовались примерами из областей записи и воспроизведения звука, телекоммуникаций и биомедицины.

2.1. Характерные системы ЦОС реального времени

Блок-схема типичной системы ЦОС, работающей в реальном времени, изображена на рис. 2.1. Аналоговый входной фильтр используется для ограничения полосы частот входного аналогового сигнала перед его оцифровкой, чтобы уменьшить наложение (см. далее). Аналого-цифровой преобразователь (АЦП) трансформирует аналоговый входной сигнал в цифровую форму. Если у сигнала широкая полоса частот или если применяется низкоскоростной АЦП, то перед оцифровкой сигнала нужно воспользоваться схемой выборки-хранения, хотя в новейшие АЦП такие схемы уже встроены. После цифровой обработки в процессоре цифроаналоговый преобразователь (ЦАП) вновь преобразует обработанный сигнал в аналоговую форму. Выходной фильтр сглаживает выход ЦАП и устраняет нежелательные высокочастотные компоненты.

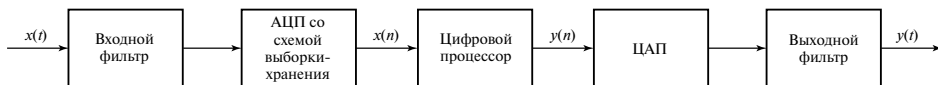


Рис. 2.1. Блок-схема упрощенной обобщенной системы цифровой обработки сигналов реального времени. В некоторых приложениях входной фильтр и АЦП или ЦАП и выходной фильтр не нужны

Сердце системы, изображенной на рис. 2.1, — это цифровой процессор, которым может быть обычный универсальный процессор, такой как Motorola MC68000, чип процессора цифровой обработки сигнала, например, Texas Instruments TMS320C50 или Motorola DSP56000, или какое-либо другое устройство. Цифровой процессор может выполнять один из нескольких алгоритмов ЦОС, например цифровую фильтрацию, и отображать вход $x(n)$ в выход $y(n)$.

При обработке сигнала цифровым процессором подразумевается, что перед обработкой входной сигнал нужно преобразовать в цифровую форму. В некоторых приложениях

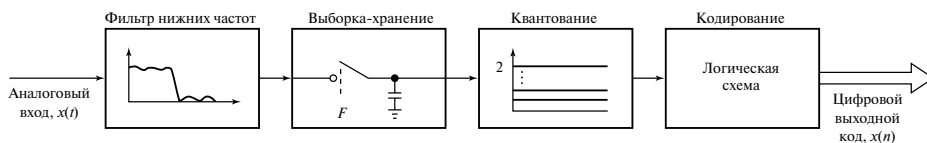


Рис. 2.2. Графическое представление процесса преобразования аналогового сигнала в цифровой

ях, работающих в реальном времени, данные уже представлены в цифровом виде или их не нужно преобразовывать в аналоговый сигнал. Например, после обработки сигнал может храниться в памяти компьютера с целью использования в будущем или изображаться графически на экране. В других приложениях может возникнуть необходимость создания цифровых сигналов. Примерами таких приложений служат синтез речи, цифровой синтез частот и генерация псевдослучайных бинарных последовательностей. В данной книге в большинстве случаев считается, что сигнал имеет цифровой вид или был оцифрован соответствующим образом, который описан в следующем разделе.

2.2. Процесс преобразования аналогового сигнала в цифровой

Как говорилось выше, прежде чем выполнять любой алгоритм ЦОС, сигнал нужно представить в цифровом виде. Большинство сигналов в природе существуют в аналоговом виде, поэтому для них необходим процесс аналого-цифрового преобразования, который состоит из таких этапов.

- Вначале сигнал (с ограниченной полосой) дискретизируется, т.е. аналоговый сигнал преобразуется в дискретный по времени сигнал с непрерывной амплитудой.
- Амплитуда каждого дискретного элемента сигнала квантуется в один из 2^B уровней, где B — число битов, которым дискретная выборка представлена в АЦП.
- Дискретные уровни амплитуды представляются или кодируются в виде различных бинарных слов, каждое из которых имеет длину B бит.

Описанный процесс изображен на рис. 2.2. На этом рисунке можно выделить три различных типа сигнала.

- *Аналоговый входной сигнал* непрерывен как по времени, так и по амплитуде.
- *Дискретный сигнал* непрерывен по амплитуде, но определяется только в дискретных точках во времени.
- *Цифровой сигнал* $x(n)$ ($n = 0, 1, \dots$) существует только в дискретных точках во времени и в каждой временной точке может иметь одно из 2^B значений (дискретный во времени сигнал с дискретной амплитудой). Именно этот тип сигнала и будет интересовать нас в данной книге.

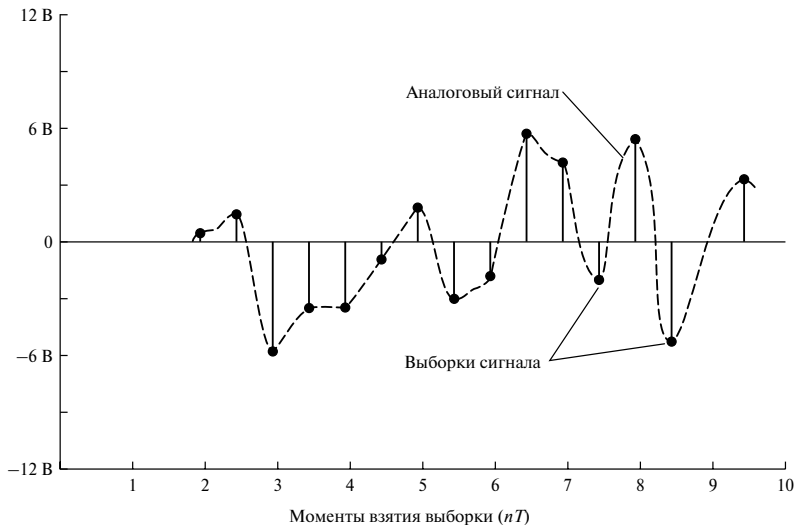


Рис. 2.3. Пример дискретного сигнала (идеальная дискретизация). Дискретные значения сигнала равны значениям исходного аналогового сигнала в эти моменты времени

Заметим, что и дискретный во времени (т.е. дискретизованный сигнал), и цифровой сигнал можно передать в виде последовательности чисел $x(nT)$, или просто $x(n)$ ($n = 0, 1, 2, \dots$). Рассмотрим процесс оцифровки сигнала подробнее, шаг за шагом.

2.3. Дискретизация — низкочастотные и полосовые сигналы

Дискретизация — это определение значений непрерывного сигнала (например, аналогового) в дискретные моменты времени. Это основное понятие обработки сигнала в реальном времени. Пример аналогового сигнала, подвергшегося дискретизации, показан на рис. 2.3. Отметим, что в данном идеальном случае после дискретизации аналоговый сигнал представлен только в дискретные моменты времени, причем значения сигнала в эти моменты равны соответствующим значениям исходного аналогового сигнала.

Ниже будет дано качественное описание теоремы о дискретном представлении, определяющей частоту, с которой должна осуществляться дискретизация аналогового сигнала, чтобы вся важная информация, содержащаяся в сигнале, собиралась и сохранялась при дискретизации. На практике чаще всего встречаются два вида дискретизации — дискретизация низкочастотных сигналов и полосовых сигналов. Как будет показано ниже, дискретизацию полосовых сигналов можно рассматривать как частный случай более общей низкочастотной дискретизации.

2.3.1. Дискретизация низкочастотных сигналов

2.3.1.1. Теорема о дискретном представлении¹

Если $f_{\max}$ — самый высокочастотный компонент сигнала, то, чтобы элементы выборки полностью описывали сигнал, дискретизация сигнала должна осуществляться с частотой не ниже $2f_{\max}$:

$$F_s \geq 2f_{\max}, \quad (2.1)$$

где F_s — частота дискретизации. Следовательно, если максимальная частота аналогового сигнала составляет 4 кГц, то для того, чтобы собрать или сохранить всю информацию, содержащуюся в сигнале, его дискретизация должна осуществляться с частотой 8 кГц или больше. Дискретизация с частотой, меньшей той, которую дает теорема о дискретном представлении, приведет к появлению перегибов или *наложению* зеркальных частот в интересующей нас частотной области. Следовательно, если захочется преобразовать дискретную информацию обратно в аналоговую, исходный сигнал будет уже невозможно восстановить. Важно помнить о том, что часто значительная доля энергии сигнала может попадать за пределы интересующей нас частотной области, и/или сигнал может содержать шум, ширина полосы которого всегда будет большой. Например, в телефонной связи самая высокая из представляющих интерес частот составляет приблизительно 3,4 кГц, но частоты речевого сигнала могут превышать 10 кГц. Поэтому, если не удалить лишний сигнал или шум за пределы полосы интересующих нас частот, теорема о дискретном представлении выполняться не будет. На практике это достигается путем предварительного пропускания сигнала через аналоговый фильтр защиты от наложения спектров.

2.3.1.2. Наложение и спектры дискретных сигналов

Предположим, мы выполнили дискретизацию сигнала в определенной временной области с интервалом T (в секундах) (т.е. частота дискретизации равна $1/T$ (в герцах)). Видно (рис. 2.4), что в исходном сигнале есть еще одна частотная составляющая с таким же набором дискретных значений. Следовательно, этот частотный компонент можно ошибочно принять за компонент с более низкой частотой. Это и есть наложение. С точки зрения анализа следствий или поиска решения задачи наложения исследовать наложение лучше в частотных координатах.

На рис. 2.5 показан процесс дискретизации, который можно рассматривать как умножение аналогового сигнала $x(t)$ на выборочную функцию $p(t)$. Функция $p(t)$ состоит из импульсов единичной амплитуды с шириной dt (бесконечно малой величиной) и периодом T . Спектр сигнала $x(t)$, функция $p(t)$ и их произведение показаны на рис. 2.5. Заметим, что $X'(f)$ — это свертка $X(f)$ и $P(f)$, следовательно, умножение во временных координатах эквивалентно свертке в частотных координатах.

Для дискретного сигнала (рис. 2.5, z) следует отметить такие моменты.

¹Известная также как теорема Котельникова, теорема Найквиста, теорема отсчетов. — Прим. ред.

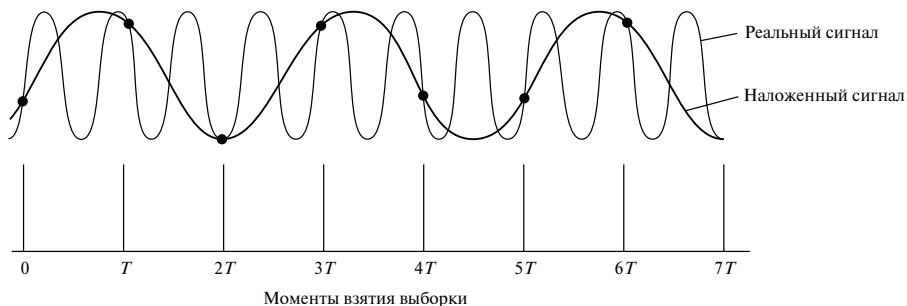


Рис. 2.4. Пример наложения во временных координатах. Обратите внимание на то, что оба сигнала имеют одинаковые значения в одних и тех же точках, хотя их частоты различны

- Спектр идентичен исходному аналоговому спектру, только повторяется в точках, кратных частоте дискретизации F_s . Компоненты более высокого порядка с центрами в точках, кратных F_s , называются *зеркальными частотами*.
- Если частота дискретизации F_s недостаточно высока, то зеркальные частоты с центром в F_s будут, например, накладываться на частоты основной полосы (рис. 2.6). В этом случае полезную информацию, содержащуюся в сигнале, невозможно отличить от его образа в области наложения.
- Перекрывание (или наложение) происходит в районе точки F_N , равной половине частоты дискретизации. Эту точку часто называют *максимальной частотой сигнала*, *частотой Найквиста*, *частотой Котельникова* и т.п.

На практике наложение существует всегда, из-за шума и наличия энергии сигнала за пределами полосы частот, которая представляет интерес. Поэтому задача разработчика — определить уровень допустимого наложения, создать подходящий фильтр защиты от наложения спектров и выбрать подходящую для этого частоту дискретизации.

2.3.1.3. Фильтр защиты от наложения спектров

Для уменьшения эффектов наложения обычно используют фильтры защиты от наложения спектров с резким срезом, которые ограничивают полосу частот сигнала и/или увеличивают частоту дискретизации, чтобы отодвинуть спектр сигнала и зеркальный спектр дальше друг от друга. В идеале фильтр защиты от наложения спектров должен устранять все частотные компоненты с частотой, превышающей частоту наложения, т.е. его частотная характеристика должна быть похожей на ту, которая изображена на рис. 2.7, *а*. Более реальная характеристика показана на рис. 2.7, *б*, где f_c и f_s — это соответственно частота среза и частота полосы подавления. Из рис. 2.7, *б* и *в* видно, что практическая характеристика искажает амплитуду сигнала, поскольку ее полоса пропускания не плоская. Кроме того, компоненты сигнала, превышающие f_s , будут подавляться на $A_{\min}$, а амплитуды компонентов, которые лежат между f_c и f_s (ширина полосы перехода), будут спадать монотонно.

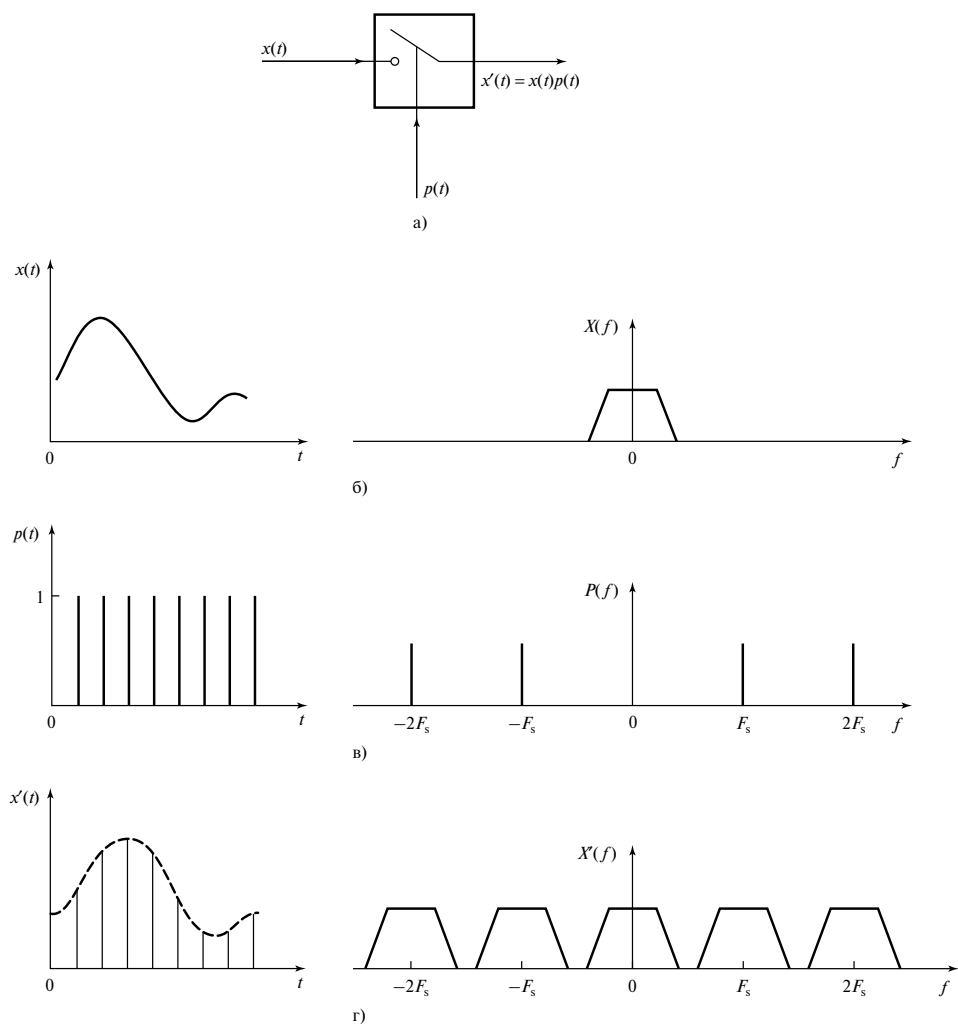


Рис. 2.5. Описание процесса дискретизации во временной и частотной областях. Сравните спектры сигнала до (б) и после (г) дискретизации. Обратите внимание на изменения в дискретном сигнале и, в частности, на то, что спектр дискретного сигнала повторяется в точках, кратных частоте дискретизации F_s

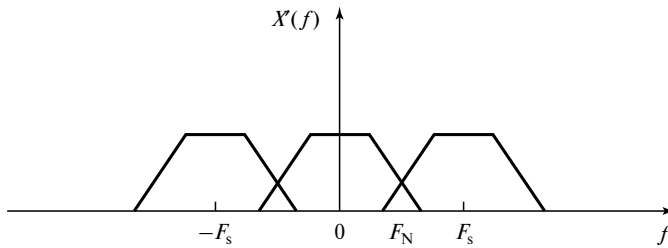


Рис. 2.6. Спектр сигнала, прошедшего процесс дискретизации, на котором показано наложение. Сигналы в области наложения не поддаются восстановлению. F_N равно половине частоты дискретизации и обычно называется *частотой Найквиста*. Чтобы восстановить все компоненты сигнала, нужно провести выборку с частотой, превышающей удвоенную частоту самого высокочастотного компонента (или равной ей)

Таблица 2.1. Оценка минимального затухания в полосе подавления фильтра нижних частот ($A_{\min}$) для различных значений разрешающей способности АЦП (B)

B	$A_{\min}$ (дБ)
8	50
10	62
12	74
16	98

Фильтр защиты от наложения спектров должен обеспечивать достаточное подавление характеристики на частотах, превышающих частоту Найквиста. Из-за неидеальности характеристик тех фильтров, которые используются на практике, в качестве эффективной частоты Найквиста берется f_s (частота среза). При спецификации фильтра защиты от наложения спектров полезно также учитывать требования к разрешению АЦП. Итак, фильтр защиты от наложения спектров следует разрабатывать так, чтобы частоты, превышающие частоту Найквиста, подавлялись до уровня, неразличимого для АЦП, например, до уровня, меньшего, чем шум квантования (см. далее). Так, для системы, в которой используется B -битовый линейный АЦП, минимальное затухание в полосе подавления фильтра, как правило, будет равно

$$A_{\min} = 20 \lg(\sqrt{1,5} \times 2^B), \quad (2.2)$$

где B — это количество битов АЦП (подробнее см. пример 2.3). Значения $A_{\min}$ для различных значений B даны в табл. 2.1.

Применение для предварительной обработки данных системы ЦОС аналогового фильтра накладывает еще одно ограничение, так называемое *условие фазового искажения*. На рис. 2.7, *г* изображена фазовая характеристика фильтра защиты от наложения спектров, амплитудная характеристика которого дана на рис. 2.7, *в*. Здесь показано, что фазовая характеристика зависит от частоты нелинейно, так что компоненты искомого сигнала будут иметь смещенную фазу или задерживаться на величину, не пропорцио-

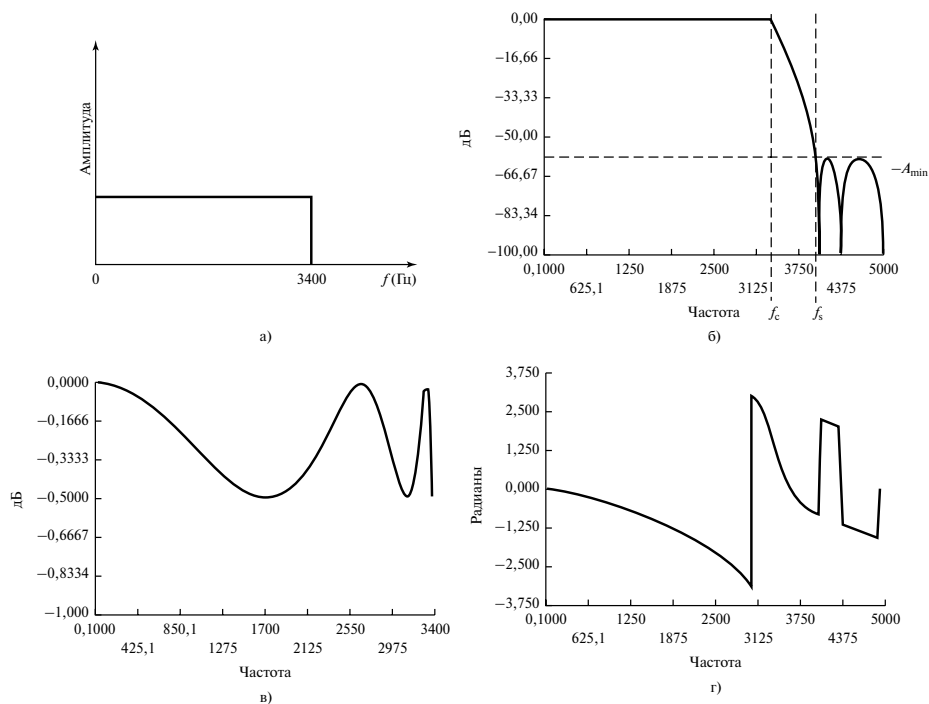


Рис. 2.7. Идеальная и реальные частотные характеристики фильтров защиты от наложения спектров, показывающие ошибки, которые дают реальные фильтры: а) идеальная характеристика; б) реальная амплитудная характеристика; в) реальная полосовая амплитудная характеристика; г) реальная фазовая характеристика. Сравним, к примеру, плоскую полосовую характеристику на панели а и реальную полосовую характеристику на панели в: неравномерность реальной характеристики приводит к амплитудному искажению внутриполосных компонентов сигнала

нальную их частотам. Величина искажения зависит от характеристик фильтра, включая то, насколько круто спадает его амплитудно-частотная характеристика. В большинстве случаев, чем круче спад (т.е. чем уже полоса пропускания), тем большее фазовое искажение вносит фильтр, и тем труднее согласовать амплитуду и групповую задержку каналов в многоканальной системе. Однако применение фильтров с крутым срезом позволяет использовать низкую частоту дискретизации и менее скоростные, т.е. более дешевые АЦП.

При обработке сигналов в реальном времени наблюдается тенденция к использованию высокой частоты дискретизации, т.е. выборке с запасом по частоте, даже если это осуществляется за счет применения скоростных и дорогих АЦП. Этому есть несколько причин. Во-первых, так можно использовать простые фильтры защиты от наложения спектров, минимизирующие фазовое искажение, и, что важно для многоканальных систем, снижающих стоимость. Во-вторых, выборка с запасом по частоте в сочетании с дополнительной цифровой обработкой сигнала приводит к улучшению отношения сигнал-шум (см. главу 9). Чтобы систему ЦОС с аналоговым устройством предвари-

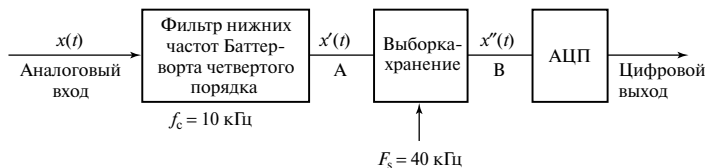


Рис. 2.8. Устройство предварительной обработки данных системы ЦОС реального времени

тельной обработки данных можно было использовать в различных приложениях, частота среза фильтра должна быть переменной. В употребление входят программируемые аналоговые фильтры, например MF10, но их производительность не совсем удовлетворительна, а для многоканальных систем они могут оказаться слишком дорогими. Дискретизация аналогового сигнала с запасом по частоте позволяет применять метод цифрового преобразования частоты дискретизации (см. главу 9), чтобы проще было соответствовать требованиям переменной частоты среза.

2.3.1.4. Примеры, иллюстрирующие выбор частоты дискретизации и контроль за наложением спектров

К числу факторов, которые влияют на выбор частоты дискретизации, относятся:

- частотное содержание входного сигнала;
- требования к фильтрации для защиты от наложения спектров;
- допустимый уровень искажения от наложения спектров;
- разрешающая способность АЦП;
- требования к памяти.

Влияние первых двух факторов на частоту дискретизации уже обсуждалось. Для определения допустимого уровня искажения от наложения спектров можно использовать один из множества разработанных методов; в большинстве случаев — это взаимосвязь между частотой дискретизации, уровнем искажения от наложения и параметрами фильтра. Например, можно определить допустимый уровень искажения для заданных характеристик фильтра защиты от наложения спектров, а затем определить частоту дискретизации, необходимую для достижения этого уровня. Можно также при заданной частоте дискретизации рассчитать минимальное затухание в полосе подавления, которое даст фильтр защиты от наложения спектров, необходимое для обеспечения определенного уровня искажения. На практике следует также учитывать разрешающую способность АЦП, так как она определяет минимальный уровень шумов в системе.

Следующие примеры иллюстрируют некоторые моменты, касающиеся выбора частоты дискретизации и контроля за наложением спектров.

Пример 2.1

Иллюстрация дискретизации и взаимосвязи между уровнем искажения от наложения и частотой дискретизации. На рис. 2.8 изображено устройство предварительной обработки данных системы ЦОС реального времени. Предположим, что входной сигнал является широкополосным.

- А. Изобразите спектр сигнала до дискретизации (точка А) и после нее (точка В) в пределах области $\pm F_s/2$.
- Б. Найдите уровень сигнала и искажения от наложения при частоте 10 кГц и частоте Найквиста (т.е. 20 кГц).
- В. Определите минимальную частоту дискретизации ($F_s(\min)$), которая дает отношение сигнала к уровню искажения от наложения 10:1 при частоте 10 кГц. Укажите все дополнительные предположения.

Решение

- А. Спектр сигнала до и после дискретизации изображен на рис. 2.9. Заметим, что форма каждого спектрального компонента имеет ту же форму, что и характеристика фильтра Баттерворта, т.е.

$$|H(f)| = \frac{1}{\sqrt{1 + \left(\frac{f}{f_c}\right)^8}}.$$

- Б. Спектр сигнала на выходе фильтра равен произведению спектра сигнала и характеристики фильтра, т.е. $X(f)|H(f)|$. Для входного широкополосного сигнала спектр $X(f)$, по сути, плоский. Если предположить, что максимальное значение $X(f)$ и $H(f)$ равно единице (т.е. они нормированы), уровень сигнала до дискретизации (на выходе фильтра) и после нее (после схемы выборки-хранения) определяется характеристикой аналогового фильтра.

Итак, при частоте 10 кГц уровень нормированного сигнала (из приведенного выше уравнения) равен просто 0,707 (т.е. $1/\sqrt{2}$). Уровень искажения от наложения (согласно рис. 2.9, б) задается как

$$\text{уровень наложения, } X_a = \frac{1}{\sqrt{1 + \left(\frac{30}{10}\right)^8}} = 0,012.$$

Частота Найквиста равна 20 кГц (т.е. половине частоты дискретизации). Это точка пересечения характеристик на рис. 2.9, б, поэтому уровни сигнала и искажения от наложения одинаковы. И уровень сигнала, и уровень наложения при 20 кГц (исходя из выражения для характеристики Баттерворта, где $f = 20$ кГц, а $f_c = 10$ кГц) равны 0,062.

- В. При 10 кГц уровень сигнала составляет 0,707. Отношение уровня сигнала к уровню наложения 10:1 предполагает уровень наложения в 0,0707. Зеркальный компонент, который вызывает наложение, определяется уравнением Баттерворта.

Следовательно, из указанного уравнения находим:

$$\frac{1}{\sqrt{1 + \left(\frac{f}{10}\right)^8}} = 0,0707,$$

т.е. $f = 19,39$ кГц.

Это соответствует частоте наложения при 10 кГц, т.е. f_a на приведенном выше рис. 2.9, б. Следовательно, частота дискретизации $F_s = f_a + 10 = 29,39$ кГц.

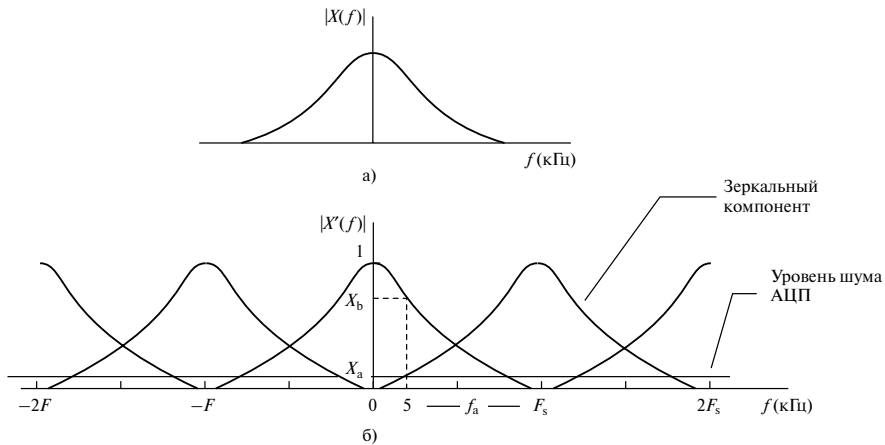


Рис. 2.9. Спектр дискретного сигнала, на котором показано искажение, вызванное наложением и шумом квантования АЦП

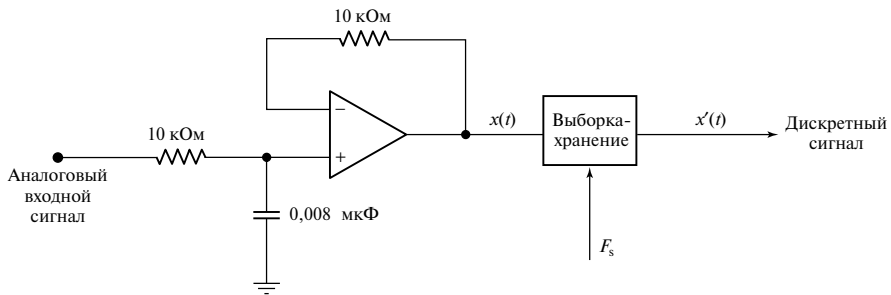


Рис. 2.10. Устройство предварительной обработки в простой системе сбора данных. Простой активный фильтр ограничивает полосу частот сигнала перед дискретизацией с частотой F_s

Пример 2.2

На рис. 2.10 изображено устройство предварительной обработки в простой системе сбора данных. Найдите минимальную частоту дискретизации F_s , при которой искажение от наложения будет составлять меньше 2 % от уровня сигнала в полосе пропускания.

Решение

Амплитудная характеристика активного фильтра задается следующим образом:

$$|H(f)| = \frac{1}{[1 + (f/f_c)^2]^{1/2}}, \quad \text{где } f_c = 1/2\pi RC = 2 \text{ кГц.}$$

Спектр входного сигнала с ограниченной полосой частот и спектр дискретного сигнала показаны на рис. 2.11, где предполагалось, что на вход подается аналоговый широкополосный сигнал.

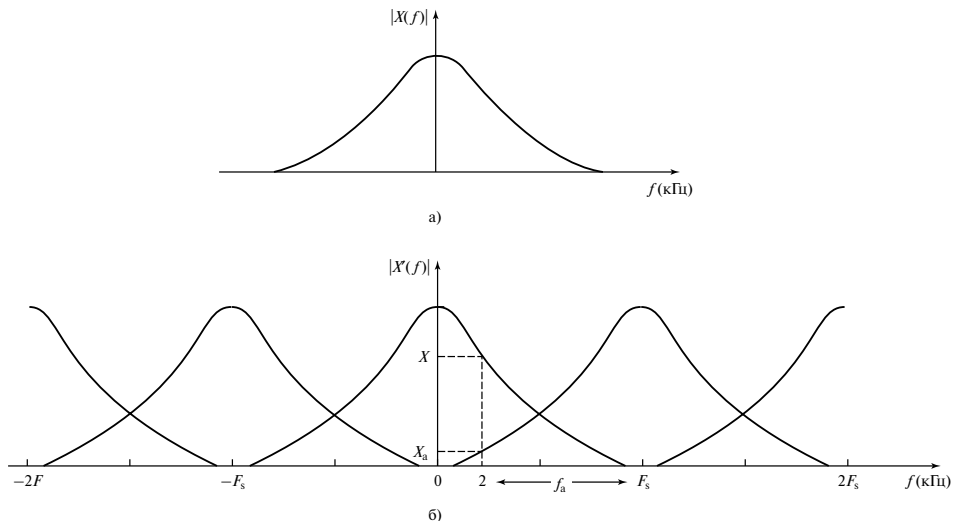


Рис. 2.11. Спектр сигнала а) на выходе аналогового фильтра и б) после дискретизации (пример 2.2)

На рисунке видно, что спектр дискретного сигнала повторяется на частотах, кратных частоте дискретизации. Перекрытие зеркальных частот с интересующей нас полосой частот (от 0 до 2 кГц) — это наложение.

При частоте 2 кГц уровень сигнала $X_b = 0,7071$, так что искомый уровень наложения $< 0,7071 \times 2/100 = 0,01414$.

Следовательно,

$$0,01414 < \frac{1}{[1 + (f_a/2)^2]^{1/2}},$$

где f_a — частота наложения. Решив это уравнение относительно f_a , получим: $f_a < 141,4$ кГц. Следовательно,

$$F_s(\min) > f_c + f_a = 2 \text{ кГц} + 141,4 \text{ кГц} = 143,4 \text{ кГц}.$$

Если учесть спецификации и появление зеркальных частот в точках $2F_s$, $3F_s$ и т.д. (что ранее не учитывалось), то $F_s(\min) > 143,4$ кГц. Положим $F_s(\min) = 150$ кГц.

Пример 2.3

Иллюстрация взаимосвязи между разрешающей способностью АЦП и параметрами фильтра. На рис. 2.12 изображена система ЦОС реального времени. Предположив, что интересующая нас полоса частот занимает положение от 0 до 4 кГц, и что используется 12-битовый биполярный АЦП, оцените:

- 1) минимальное затухание в полосе подавления $A_{\min}$ для фильтра защиты от наложения спектров;

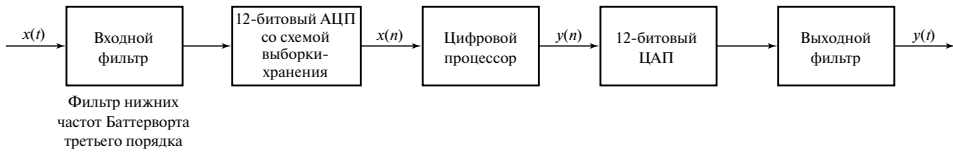


Рис. 2.12. Система цифровой обработки сигналов реального времени

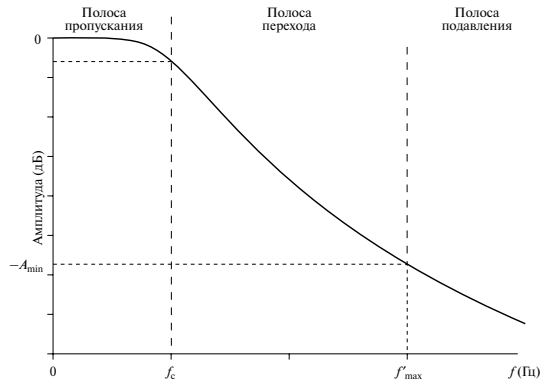


Рис. 2.13. Типичная амплитудно-частотная характеристика реального фильтра защиты от наложения спектров

- 2) минимальную частоту дискретизации F_s ;
- 3) уровень искажения от наложения спектров относительно уровня сигнала в полосе пропускания для оцененных $A_{\min}$ и F_s .

Изобразите спектр сигнала на выходе аналогового фильтра, предположив, что на вход подается широкополосный сигнал, а также спектр сигнала после дискретизации.

Решение

Согласно условиям теоремы о дискретном представлении фильтр защиты от наложения спектров должен ограничивать полосу частот спектра входного сигнала таким образом, чтобы во избежание наложения удалить частотные компоненты, превышающие частоту Найквиста.

На практике, поскольку идеального фильтра не существует, фильтр защиты от наложения обычно подавляет частотные компоненты, превышающие частоту Найквиста, до уровня, меньшего чем среднеквадратический уровень шума квантования АЦП, так что эти компоненты становятся неразличимы для АЦП.

Типичная амплитудно-частотная характеристика фильтра защиты от наложения спектров изображена на рис. 2.13, где показаны полосы пропускания, подавления и перехода. Фильтр защиты от наложения спектров разработан для подавления уровня частотных компонентов в полосе подавления, т.е. частот $\geq f'_{\max}$, до уровня, не превышающего среднеквадратический уровень шума квантования АЦП.

Итак, эффективная частота Найквиста равна $f'_{\max}$, а эффективную частоту дискретизации можно найти так:

$$F_s \geq 2f'_{\max}. \quad (2.3)$$

Теперь величина шага квантования задается как

$$q = \frac{V_{fs}}{2^B - 1} \approx \frac{V_{fs}}{2^B},$$

где B — количество битов АЦП, а V_{fs} — полный диапазон входного сигнала.

Следовательно,

$$V_{fs} \approx q \times 2^B.$$

Среднеквадратический уровень шума квантования задается как

$$\sqrt{\frac{q^2}{12}} = \frac{q}{2\sqrt{3}}.$$

Если предположить для простоты, что входной сигнал имеет вид синусоиды с максимальной амплитудой A (такой сигнал точно вмещается в диапазон частот АЦП), то максимальный уровень сигнала полосы пропускания будет равен

$$V_{fs} = 2A = q \times 2^B.$$

Следовательно,

$$A = \frac{q \times 2^B}{2}.$$

Отношение максимального уровня сигнала в полосе пропускания к уровню сигнала в полосе подавления показывает меру максимального затухания сигнала в полосе подавления фильтра:

$$\begin{aligned} \frac{\text{максимальный уровень сигнала в полосе пропускания}}{\text{уровень сигнала в полосе подавления}} &= \frac{q \times 2^B / 2\sqrt{2}}{q / 2\sqrt{3}} = \\ &= \sqrt{1,5} \times 2^B. \end{aligned}$$

1. Итак, в системе ЦОС минимальное затухание $A_{\min}$ в полосе подавления (при синусоидальном входе) задается как

$$A_{\min} = 20 \lg(\sqrt{1,5} \times 2^B) \text{ дБ} = 74 \text{ дБ}.$$

2. Спектр сигнала до и после дискретизации (без зеркальных частот более высоких порядков) показан на рис. 2.14.

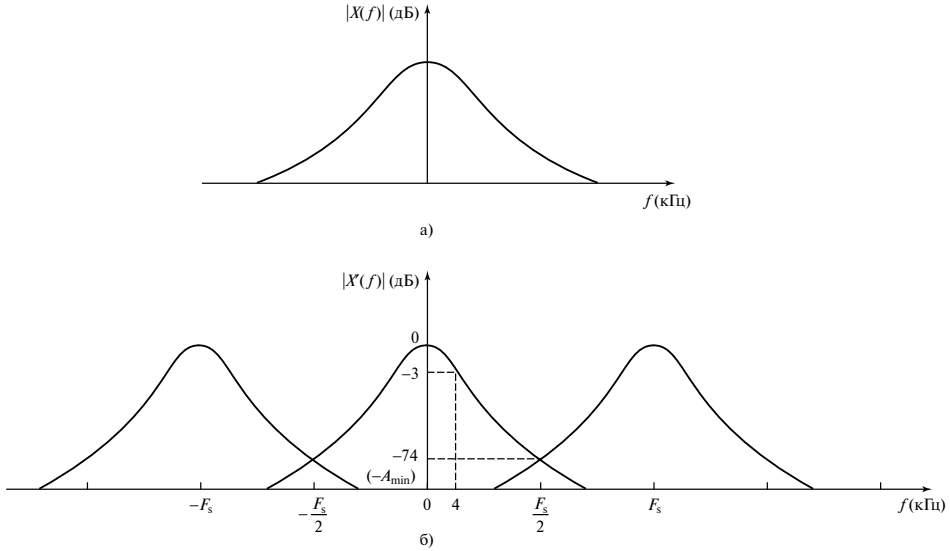


Рис. 2.14. Спектр сигнала: а) на выходе аналогового фильтра; б) после дискретизации (пример 2.3)

Из

$$A_{\min} = 74 = 20 \lg \left[1 + \left(\frac{f'_{\max}}{f_c} \right)^6 \right]^{\frac{1}{2}}$$

получаем

$$\left(\frac{f'_{\max}}{f_c} \right)^6 = (5011,87)^2 - 1,$$

откуда $f'_{\max} = 68,45$ кГц.

Из уравнения (2.3) получаем: $F_s = 2f'_{\max} = 136,9$ кГц.

3. При частоте 4 кГц уровень наложения составляет

$$\frac{1}{\left[1 + \left(\frac{136,9-4}{4} \right)^6 \right]^{\frac{1}{2}}} = 2,73 \times 10^{-5}.$$

Уровень наложения спектров относительно уровня сигнала при частоте 4 кГц равен

$$\frac{2,73 \times 10^{-5}}{0,7071}.$$

Если нужно подавить граничную частоту полосы пропускания настолько, чтобы она точно была ниже уровня шума квантования, частоту дискретизации можно снизить до $68,4 + 4 = 72,4$ кГц.

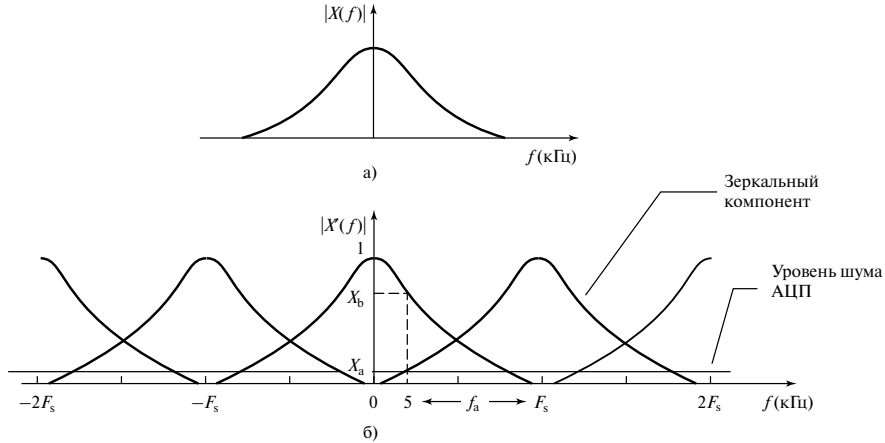


Рис. 2.15. Спектр дискретного сигнала, на котором показано искажение, вызванное наложением и квантованием АЦП (пример 2.4)

Пример 2.4

Использование минимального уровня шума АЦП как эталона. Полоса частот аналогового сигнала с однородной спектральной плотностью мощности ограничена фильтром защиты от наложения спектров со следующей амплитудной характеристикой:

$$|H(f)| = \frac{1}{\left[1 + \left(\frac{f}{f_c}\right)^8\right]^{\frac{1}{2}}},$$

где $f_c = 5$ кГц. Сигнал оцифрован с помощью линейного 12-разрядного биполярного АЦП. Найдите:

- 1) минимальную частоту дискретизации, при которой максимальное искажение от наложения не будет превышать уровень шума квантования;
- 2) максимальный уровень сигнала в полосе пропускания (в дБ) относительно минимального уровня шума квантования АЦП.

Решение

1. Частоту дискретизации следует выбирать так, чтобы фильтр защиты от наложения спектров ослаблял искажение от наложения, попадающее в полосу пропускания, до уровня, меньшего, чем минимальный среднеквадратический шум АЦП, чтобы оно было неразличимо для АЦП (рис. 2.15).

Предположим, что входной сигнал имеет вид синусоиды с максимальной амплитудой A (такой сигнал точно вмещается в диапазон АЦП), тогда

среднеквадратическое значение входного сигнала: $\frac{A}{\sqrt{2}},$

шаг квантования: $q \approx \frac{2A}{2^B},$

среднеквадратический шум квантования: $\frac{q}{2\sqrt{3}} = \frac{A}{\sqrt{3 \times 2^B}}.$

При частоте 5 кГц максимальное искажение вследствие наложения равно

$$\frac{A}{\sqrt{2}} \times \frac{1}{\left[1 + \left(\frac{f_a}{5}\right)^6\right]^{\frac{1}{2}}} = \frac{A}{\sqrt{3 \times 2^B}}. \quad (2.4)$$

При $B = 12$ бит это уравнение можно решить относительно частоты наложения f_a и, следовательно, найти частоту дискретизации F_s :

$$f_a = 85,59 \text{ кГц},$$

$$F_s = f_a + 5 = 90,59 \text{ кГц}.$$

2. Максимальный уровень сигнала относительно минимального уровня шума АЦП:

$$\frac{\text{максимальное среднеквадратическое значение уровня сигнала}}{\text{максимальное искажение в результате квантования}} = \frac{A/\sqrt{2}}{A/(\sqrt{3} \times 2^B)} = \sqrt{1,5} \times 2^B.$$

$$\text{Сигнал-минимальный уровень шума АЦП} = 20 \lg(\sqrt{1,5} \times 2^B). \quad (2.5)$$

Заметим также, что отношение сигнал-минимальный уровень шума АЦП в данном случае можно найти из следующего выражения (см. уравнение (2.4)):

$$\text{сигнал} - \text{минимальный уровень шума} = 20 \lg \left[1 + \left(\frac{f_a}{5} \right)^6 \right]^{\frac{1}{2}} \text{ дБ}.$$

2.3.1.5. Другие практические вопросы, связанные с дискретизацией: точность и ограничение ширины полосы

На практике мгновенная выборка, показанная на рис. 2.5, *з*, невозможна; вместо этого функция отсчетов имеет конечную ширину. Это приводит к возникновению проблемы, называемой *апертурным эффектом*, возникающей, поскольку сигнал измеряется за конечный отрезок времени, а не мгновенно. Ненулевое апертурное время ограничивает точность и максимальную частоту сигнала, который можно оцифровать, поскольку сам сигнал может измениться во время дискретизации. Величину апертурного эффекта можно найти, предположив, что входное напряжение на протяжении апертурного отрезка времени может изменяться, скажем, не более чем на $\frac{1}{2}$ МЗБ (младшего значащего бита). Следовательно, для синусоидального входного сигнала максимальная частота, которая может быть оцифрована с точностью до $\frac{1}{2}$ МЗБ, в системе с B -битовым АЦП задается как

$$f_{\max} = \frac{1}{\pi 2^{B+1} \tau}, \quad (2.6)$$

где τ — это апертурное время (доказательство см. в примере 2.5).

Пример 2.5

В системе ЦОС реального времени используется 12-битовый АЦП со временем преобразования 35 мкс и без схемы выборки-хранения. Какова наибольшая частота, поддающаяся оцифровке в пределах точности $\frac{1}{2}$ МЗБ, если предположить, что система бинарна с однородным квантованием? Результат прокомментируйте.

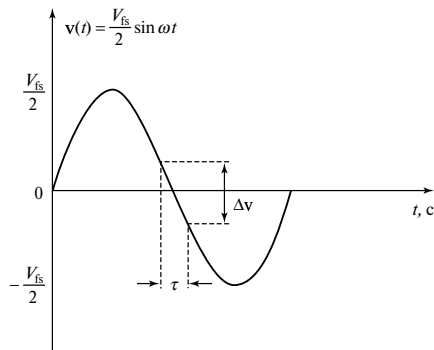


Рис. 2.16. Синусоидальный сигнал (пример 2.5)

Решение

Рассмотрим синусоидальный сигнал с максимальной амплитудой, равной половине полного диапазона АЦП, $V_{fs}/2$ (рис. 2.16). На рисунке τ — это апертурное время, а $\Delta\nu$ — изменение $\nu(t)$ за время τ . Самое большое изменение происходит в точке $t = 0$, и, чтобы АЦП выдавал сигнал с желаемой точностью, это нужно учесть. В этой точке

$$\left. \frac{d\nu(t)}{dt} \right|_{t=0} = (V_{fs}/2)\omega \cos \omega t = \pi f V_{fs} (B \text{ c}^{-1}) = \frac{\Delta\nu}{\tau}.$$

Для точности $\frac{1}{2}$ МЗБ $\Delta\nu = \alpha/2$, где $\alpha = (V_{fs}/2^B)$. Следовательно, $\Delta\nu/\tau = \pi f V_{fs}$. Подставив значения V_{fs} и $\Delta\nu$ и упростив выражение, получим

$$f_{\max} = \frac{1}{\pi 2^{B+1} \tau}.$$

Для рассматриваемой системы ЦОС $B = 12$ бит и $\tau = 35$ мкс. Следовательно, $f_{\max} = 1,11$ Гц.

Ясно, что от АЦП, который может преобразовывать только максимальную частоту 1,11 Гц, мало пользы. На практике перед АЦП обычно стоит устройство выборки-хранения, которое замораживает элемент сигнала во время преобразования и дает возможность точно оцифровывать сигналы в килогерцевом диапазоне. Например, если перед АЦП, рассмотренным выше, расположить схему выборки-хранения с апертурным временем 25 нс и временем захвата сигнала 2 мкс, то максимальная частота, которую можно преобразовать, будет равна

$$2f_{\max} \leq F_s = 1/(35 + 2 + 0,025) \times 10^{-6} \text{ кГц}, \text{ т.е. } f_{\max} = 13,5 \text{ кГц}.$$

Следовательно, сигнал с максимальной частотой 13,5 кГц будет дискретизоваться с частотой 27 кГц, или с интервалом $(35 + 2 + 0,025) \text{ мкс} = 37,025 \text{ мкс}$.

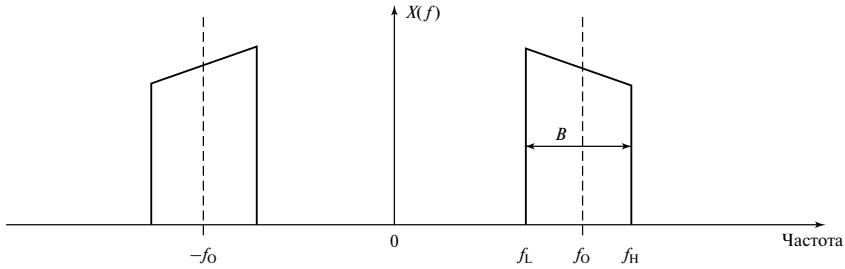


Рис. 2.17. Полосовой сигнал

2.3.2. Дискретизация полосовых сигналов

2.3.2.1. Введение и основные понятия

В некоторых приложениях, например, в системах связи, полезный сигнал занимает только узкую часть доступной полосы частот (рис. 2.17).

В таких случаях ширина полосы сигнала B зачастую очень мала по сравнению с верхней и нижней граничными частотами полосы (f_L и f_H), поэтому использовать теорему о низкочастотной дискретизации неэкономно. Чтобы побороть эту трудность, можно использовать теорему о полосовой дискретизации (уравнение 2.7):

$$\frac{2f_H}{n} \leq F_s \leq \frac{2f_L}{n-1}, \quad (2.7)$$

где

$$n = \frac{f_H}{B} \quad (n - \text{целое, округленное до ближайшего большего целого числа}).$$

Теорема о полосовой дискретизации позволяет выполнять дискретизацию узкополосных высокочастотных сигналов со значительно сниженной частотой и при этом избегать наложения [4, 6]. Существует два общих метода полосовой дискретизации без наложения с недостаточной выборкой. Один из них — это так называемая *дискретизация целочисленной полосы*, а в другом используется квадратурная модуляция. Второй метод в данной книге не обсуждается.

2.3.2.2. Методы дискретизации с недостаточной выборкой для целочисленных полос

Если для данного полосового сигнала граничные частоты полосы (f_L и f_H) — целые числа, кратные ширине полосы сигнала, сигнал можно оцифровать без наложения с теоретической минимальной частотой $2B$:

$$F_s(\min) = 2B \quad (2.8, a)$$

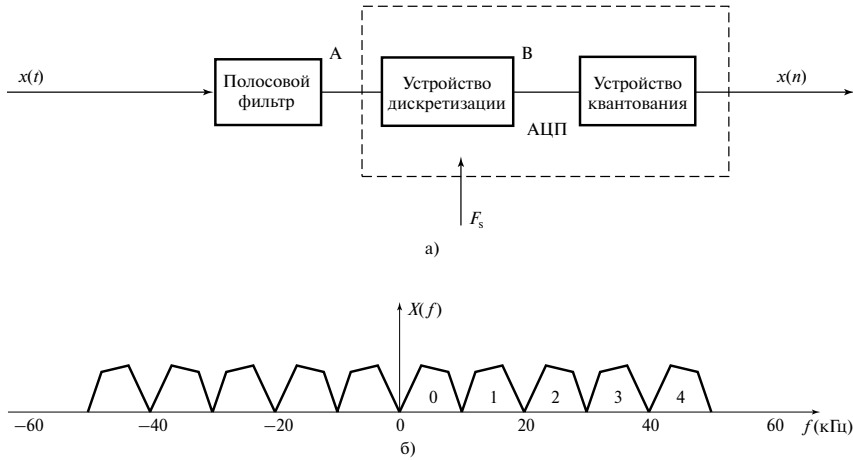


Рис. 2.18. Устройство для предварительной обработки данных системы (панель а); спектр принятого сигнала (панель б) (пример 2.6)

Уравнение (2.8, а) справедливо, если отношение низкочастотного края полосы к ширине полосы сигнала и/или высокочастотного края к ширине полосы сигнала — целые числа:

$$n = \frac{f_H}{B} \text{ или } n = \frac{f_L}{B}. \quad (2.8, б)$$

Если условия уравнения (2.8, б) соблюдены, то полосу сигнала называют *целочисленно расположенной*. Если полоса сигнала не целочисленная, граничные частоты полосы можно сместить таким образом, чтобы эффективная полоса стала целочисленной.

Пример 2.6

Иллюстрация принципов полосовой дискретизации с недостаточной выборкой. На рис. 2.18, а показано устройство предварительной обработки данных приемника многоканальной системы связи. Спектр принимаемого сигнала показан на рис. 2.18, б с указанием номеров каналов. Для выделения сигнала в нужном канале перед оцифровкой сигнала с наименьшей возможной частотой служит полосовой фильтр.

Предположим, что полосовой фильтр — идеальный и имеет следующие характеристики:

$$H(f) = 140 \text{ кГц} \leq f \leq 50 \text{ кГц}$$

0 — в других случаях

2.6.1. А. Найдите минимальную теоретическую частоту дискретизации.

Б. Изобразите спектр сигнала до дискретизации (точка А) и после дискретизации (точка В).

2.6.2. Повторите пункты А и Б для полосового фильтра, который пропускает сигнал канала 3.

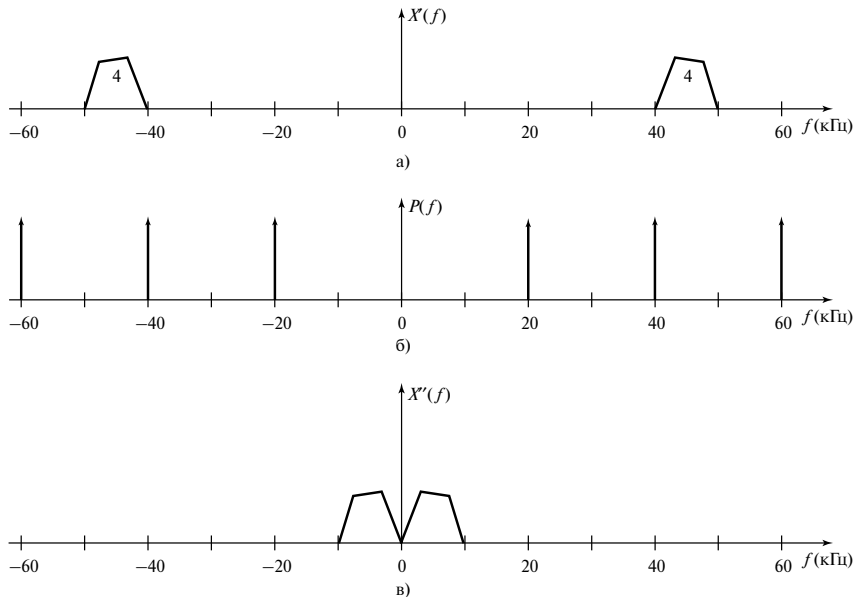


Рис. 2.19. Выход полосового фильтра (панель *а*); функция дискретизации (панель *б*); выход устройства дискретизации (панель *в*) (пример 2.6)

Решение

2.6.1. А. Минимальная теоретическая частота дискретизации равна 2×10 кГц, т.е. 20 кГц.

Б. Спектр в точке А (выход полосового фильтра) — это просто спектр сигнала для канала 4 (рис. 2.19, *а*).

Спектр в точке В (т.е. после дискретизации) можно найти, выполнив свертку спектра сигнала на выходе полосового фильтра (рис. 2.19, *а*) со спектром функции дискретизации (рис. 2.19, *б*). Результат данного процесса изображен на рис. 2.19, *в*.

2.6.2. А. Частота дискретизации остается 20 кГц.

Б. Производя те же действия, что и в п. 2.6.1, получаем изображенные на рис. 2.20, *а* и *в* соответственно спектры в точках А и В.

Пример 2.7

Иллюстрация требований метода полосовой дискретизации с недостаточной выборкой без наложения. На рис. 2.21 изображен спектр узкополосного сигнала. Найдите и изобразите спектр дискретного сигнала в диапазоне $\pm F_s/2$ при следующих условиях:

- 1) $\frac{f_H}{B} = 4$,
- 2) $\frac{f_H}{B} = 5$,
- 3) $\frac{f_H}{B} = 6, 5$.

Считайте в каждом случае, что ширина полосы сигнала $B = 4$ кГц, а сигнал дискретизируется с частотой $2B$.

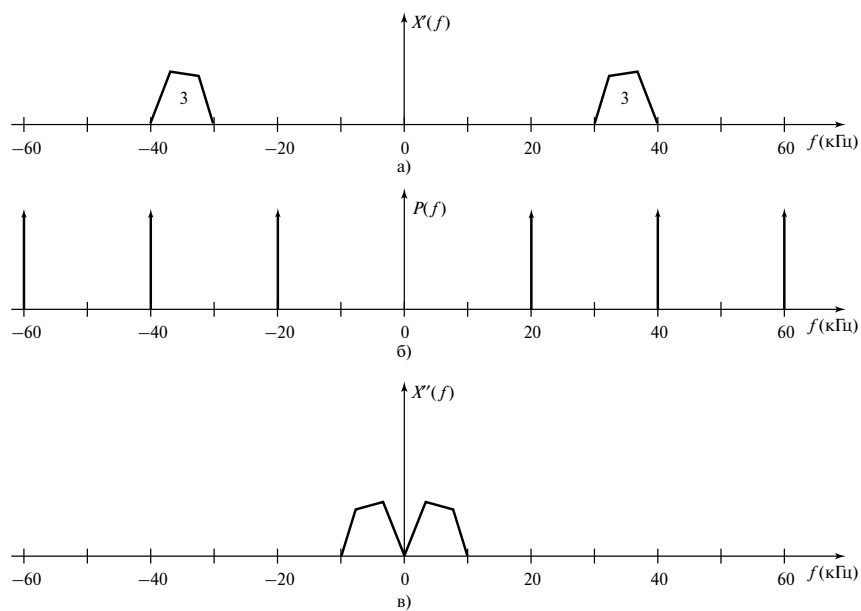


Рис. 2.20. Выход полосового фильтра (панель *а*); функция дискретизации (панель *б*); выход устройства дискретизации для канала 3 (панель *в*) (пример 2.6)

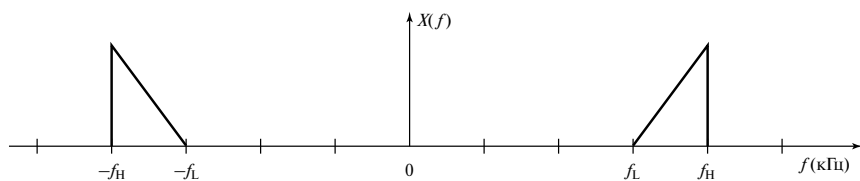


Рис. 2.21. Спектр узкополосного сигнала (пример 2.7)

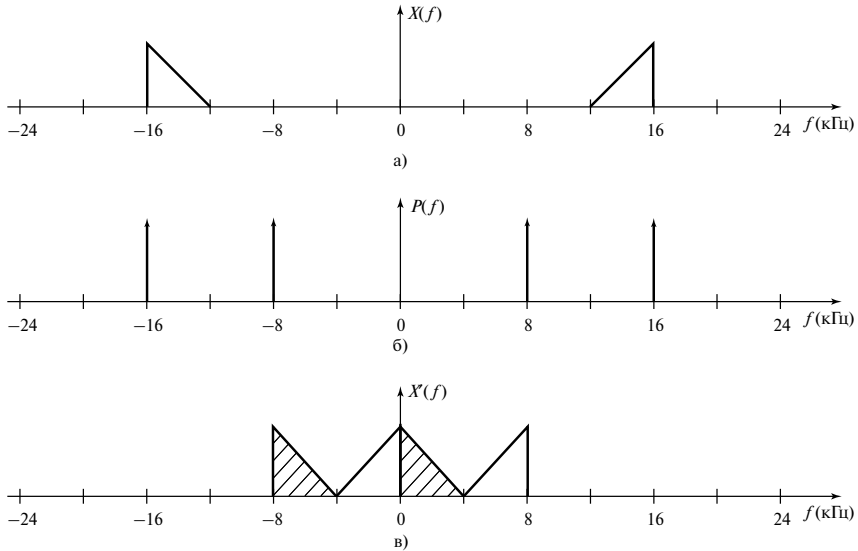


Рис. 2.22. Спектр дискретного сигнала для четных целых n ($n = f_H/B = 4$) (пример 2.7, вариант 1)

Решение

1. Соответствующий спектр сигнала показан на рис. 2.22, а. Дискретизуя сигнал с частотой $2B$, получаем частоту дискретизации 8 кГц. Спектр дискретного сигнала можно найти с помощью графической свертки спектра сигнала (рис. 2.22, а) и выборочной функции (рис. 2.22, б). Чтобы выполнить свертку, зафиксируем спектр сигнала и сместим спектр выборочной функции. Для выполнения графической свертки перед началом смещения перевернем сигнал, чтобы его положение изменилось относительно вертикальной оси. Впрочем, поскольку функция отсчетов симметрична относительно оси частот, этот шаг необязателен, так как в конце концов мы получим сигнал точно такой же формы.

Видно, что точка, соответствующая частоте -16 кГц на рис. 2.22, б, точно совпадает с отрицательной частью спектра сигнала. Следовательно, если сместить спектр выборочной функции вправо, то произойдет свертка точки, соответствующей частоте -16 кГц, с частью спектра сигнала с отрицательной частотой. Это даст спектр в диапазоне от 0 до 4 кГц на рис. 2.22, в. Затем сворачивается точка спектра, соответствующая частоте 8 кГц на рис. 2.22, б, с частью полосы сигнала между 12 и 16 кГц, что дает спектр между 4 и 8 кГц на рис. 2.22, в.

Зеркальное изображение спектра дискретного сигнала можно получить, смещая спектр выборочной функции влево. Части спектра дискретного сигнала, полученные с помощью отрицательных частотных компонентов, заштрихованы.

Комментарий. Часть спектра дискретного сигнала между 0 и 4 кГц перевернута по сравнению с исходным спектром между 12 и 16 кГц. Кроме того, в полосе сигнала нет наложения, поэтому он может быть восстановлен с помощью соответствующего обратного спектрального алгоритма.

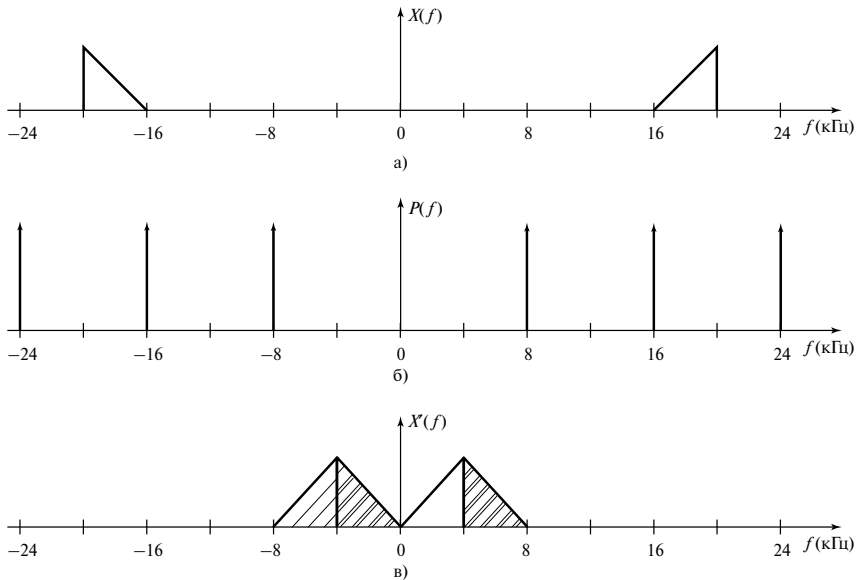


Рис. 2.23. Спектр дискретного сигнала для нечетных целых n ($n = f_H/B = 5$) (пример 2.7, вариант 2)

2. Частота дискретизации также равна 8 кГц, а спектры сигнала и выборочной функции показаны на рис. 2.23, а и б.

Как и прежде, зафиксировав спектр сигнала и сместив спектр выборочной функции сначала вправо, а затем влево, получаем спектр дискретного сигнала, изображенный на рис. 2.23, в.

Части спектра дискретного сигнала, полученные с помощью отрицательных частотных компонентов, на рисунке заштрихованы.

Комментарий. Часть спектра дискретного сигнала между 0 и 4 кГц не перевернута по сравнению с исходным спектром между 12 и 16 кГц, а в полосе сигнала нет наложения, так что его можно восстановить.

3. Как и в предыдущем случае, частота дискретизации равна 8 кГц, а спектры сигнала и функции дискретизации показаны на рис. 2.24, а и б.

Как и прежде, зафиксируем спектр сигнала и сместим спектр выборочной функции сначала вправо, а затем влево. В результате получим спектр дискретного сигнала, изображенный на рис. 2.24, в.

Видно, что точка, соответствующая частоте –24 кГц, на рис. 2.24, б находится в середине отрицательной части спектра сигнала, а точка, соответствующая частоте 24 кГц, — в середине части спектра 22–26 кГц. Следовательно, если сместить спектр выборочной функции вправо, выполнится свертка точки, соответствующей частоте –24 кГц, с частью спектра, имеющей отрицательные частоты, а точки, соответствующей частоте 24 кГц, — с положительной частью спектра сигнала. В результате получим спектр в диапазоне 0–2 кГц на рис. 2.24, в. Часть спектра дискретного сигнала, полученная с помощью отрицательных частотных компонентов, показана пунктиром, а полученная с помощью положительных частотных компонентов — сплошной линией.

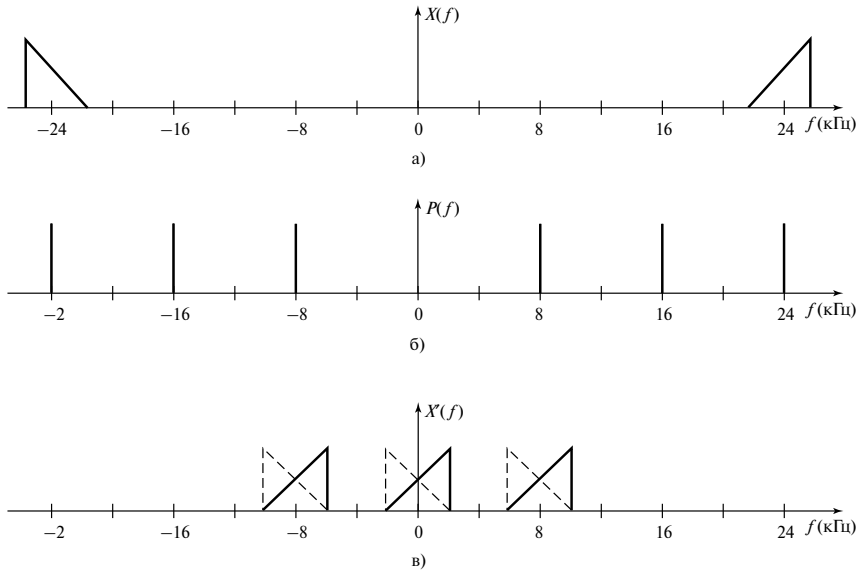


Рис. 2.24. Спектр дискретного сигнала ($n = f_H/B = 6,5$) (пример 2.7, вариант 3)

После дальнейшего смещения на 4 кГц выполняется свертка точки спектра, соответствующей 16 кГц на рис. 2.24, б, с положительной полосой сигнала, что дает спектр с центром в 8 кГц на рис. 2.24, в, показанный сплошной линией. Пунктирная линия получена с помощью точки спектра, соответствующей –16 кГц. Зеркальное изображение спектра дискретного сигнала получается в результате смещения выборочной функции влево.

Комментарий. *Перекрытие спектров, полученных с помощью положительных и отрицательных спектральных точек, свидетельствует о наложении, следовательно, в этом случае при дискретизации с частотой 8 кГц компоненты сигнала восстановить нельзя.*

2.3.2.3. Расширение полосы частот для полосовой дискретизации с недостаточной выборкой без наложения

Как было показано для целочисленной дискретизации полосового сигнала, при условии, что одна из граничных частот полосы — целое число, кратное ширине полосы, узкую полосу высокочастотного сигнала можно дискретизовать со значительно уменьшенной частотой ($2B$) и при этом избежать искажения вследствие наложения.

Следовательно, важным параметром целочисленной дискретизации полосовых сигналов является отношение верхней границы полосы к ширине этой полосы B (или, что эквивалентно, отношение нижнего края полосы (f_L) к ширине полосы):

$$n = \frac{f_H}{B} \quad (2.9, a)$$

или

$$n = \frac{f_L}{B}. \quad (2.9, б)$$

В любом из этих случаев дискретизация будет осуществляться с частотой $2B$ без наложения, т.е.:

$$F_s = 2B. \quad (2.10)$$

Если это отношение равно четному целому числу, спектр дискретного сигнала в области полосы частот исходного сигнала будет перевернут.

Если значение n в уравнении (2.9, а) или (2.9, б) — нецелое число, то наблюдается наложение. Наложения можно избежать, сместив граничные частоты или центральную частоту таким образом, чтобы n стало целым. Например, можно перенести нижнюю граничную частоту из f_L в f_1 , так что

$$f_1 \leq f_L \quad (2.11, а)$$

$$f_H = n(f_H - f_1) = nB'. \quad (2.11, б)$$

Из уравнения (2.11, б) можно записать

$$f_1 = \left(\frac{n-1}{n} \right) f_H, \quad (2.12)$$

а из уравнений (2.11, а) и (2.12) —

$$\left(\frac{n-1}{n} \right) f_H \leq f_L$$

и получить выражение для n :

$$n \leq \frac{f_H}{f_H - f_L} = \frac{f_H}{B}. \quad (2.13)$$

Следовательно, чтобы достичь желаемой связи между граничными частотами полосы и ее шириной, можно сместить нижнюю граничную частоту. В этом случае используется уравнение (2.12), где n — ближайшее меньшее целое число, найденное из уравнения (2.13).

Можно показать, что желаемой цели можно достичь также, сместив верхнюю граничную частоту:

$$f_2 = \left(\frac{n}{n-1} \right) f_L, \quad (2.14)$$

где n определяется из уравнения (2.13). Доказательство уравнения (2.14) оставляется читателям в качестве упражнения.

Пример 2.8

Иллюстрация расширения полосы частот во избежание наложения. Найдите минимальную частоту дискретизации, позволяющую избежать наложения в примере 2.7 (вариант 3) путем соответствующего смещения нижней граничной частоты полосы.

Изобразите спектр измененного сигнала до и после дискретизации с новой частотой.

Решение

Отношение верхней граничной частоты к ширине полосы равно

$$\frac{f_H}{B} = \frac{26}{4} = 6,5.$$

Если n равно 6 (ближайшему меньшему целому числу), то нижнюю граничную частоту можно снизить до нового значения:

$$f'_L = \left(\frac{n-1}{n} \right) f_H = 21,66 \text{ кГц}.$$

Теперь новая ширина полосы B' и частота дискретизации F'_s приобретают значения

$$B' = f_H - f'_L = 4,34 \text{ кГц},$$

$$F'_s = 2B' = 8,68 \text{ кГц}.$$

В качестве упражнения читателям предлагается самостоятельно изобразить спектр сигнала до и после дискретизации и таким образом показать, что наложение отсутствует.

В некоторых приложениях можно достичь условий целочисленной дискретизации без наложения, изменяя не ширину полосы, а центральную частоту. Это возможно в цифровой радиосвязи, где разработчик выбирает локальную промежуточную частоту по своему усмотрению.

2.4. Однородное и неоднородное квантование и кодирование

После дискретизации амплитуда аналоговых выборок может в зависимости от приложения подвергаться однородному и неоднородному квантованию и кодированию. В биомедицине и системах записи и воспроизведения звука обычно используется однородное квантование и кодирование, тогда как в системах связи интенсивно применяется неоднородное квантование и кодирование, поскольку здесь требуется сжимать речевые сигналы (см. далее).

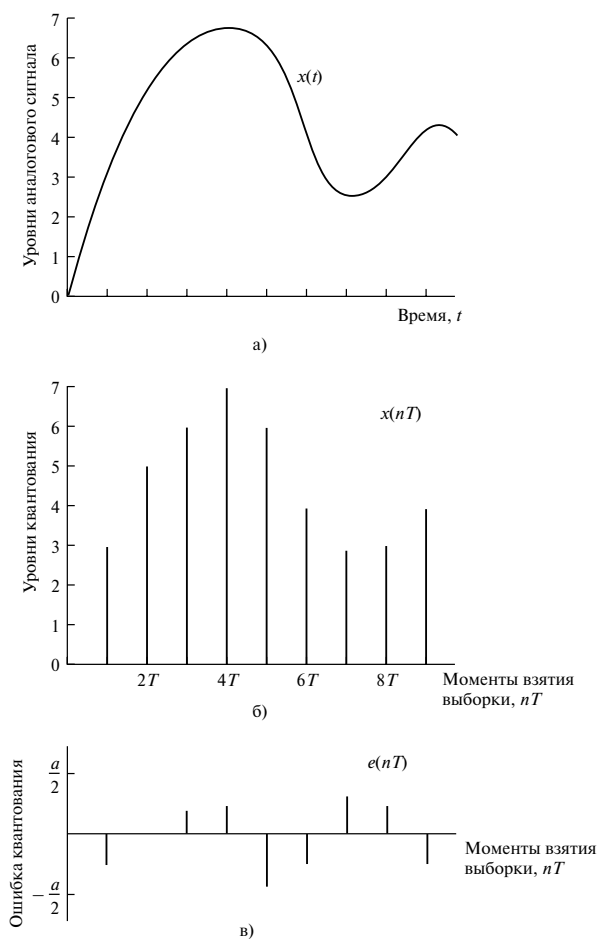


Рис. 2.25. Квантование выборок аналогового сигнала. Для нахождения ошибок квантования на панели *в* из квантованных выборок (3-битовое устройство квантования; панель *б*) сигнала вычитаются выборки сигнала (панель *а*)

2.4.1. Однородное квантование и кодирование (линейная импульсно-кодовая модуляция, ИКМ)

При однородном квантовании и кодировании каждой аналоговой выборке присваивается одно из 2^B значений (см. рис. 2.25), где B — количество битов АЦП. Этот процесс, называемый *квантованием*, вносит неустранимую погрешность. Уровень этой погрешности является функцией числа битов АЦП, которое приблизительно равно половине МЗБ (предполагая округление). Например, у 12-битового АЦП с диапазоном входных напряжений ± 10 В МЗБ будет равен $20/2^{12}$ В, т.е. 4,9 мВ, а ошибка квантования — 2,45 мВ.

Для АЦП с B двоичными цифрами количество уровней квантования равно 2^B , а расстояние между уровнями, т.е. размер шага квантования q , задается как

$$q = V_{fs}/(2^B - 1) \approx V_{fs}/(2^B), \quad (2.15)$$

где V_{fs} — полный диапазон АЦП со входом в виде биполярного сигнала. Максимальная ошибка квантования, когда значения округляются до ближайшего большего или меньшего числа, равна $\pm q/2$. Для синусоидального входа с амплитудой A (такой, что удвоенная амплитуда сигнала как раз заполняет весь входной диапазон АЦП), размер шага квантования становится равным

$$q = 2A/2^B. \quad (2.16)$$

Ошибка квантования для каждой выборки e обычно полагается случайной и однородно распределенной на отрезке $\pm q/2$ с нулевым средним значением. В этом случае мощность шума квантования, или дисперсия, задается как

$$\begin{aligned} \sigma_e^2 &= \int_{-q/2}^{q/2} e^2 P(e) de = \\ &= \frac{1}{q} \int_{-q/2}^{q/2} e^2 de = \frac{q^2}{12}. \end{aligned} \quad (2.17)$$

Для синусоидального входа средняя мощность сигнала равна $A^2/2$. Отношение сигнал-шум квантования (SNR(q)) равно (в децибелах)

$$\text{SNR}(q) = 10 \lg \left(\frac{A^2/2}{q^2/12} \right) = 10 \lg \left(\frac{3 \times 2^{2B}}{2} \right) = 6,02B + 1,76 \text{ дБ}. \quad (2.18)$$

Это теоретический максимум. На практике, когда используются реальные входные сигналы, достижимое SNR(q) меньше, чем это значение. Однако с увеличением количества разрядов B SNR(q) возрастает. Количество используемых битов ограничивают практические факторы, такие как скорость, внутреннее отношение сигнал-шум (signal-to-noise ratio — SNR) аналогового сигнала и затраты. Например, совсем не обязательно пользоваться преобразователем, дающим точность, превышающую SNR аналогового сигнала,

который подвергается преобразованию, поскольку это даст только более точное описание шума. Во многих приложениях ЦОС подходящей разрешающей способностью АЦП является 12–16 бит.

Затем цифровые выборки $x(n)$, которые во многих случаях представлены в двоичной форме, перекодируются в форму, удобную для дальнейших манипуляций. Это кодирование заключается в присвоении квантованным элементам дискретных кодов. Самые распространенные формы кодирования в ЦОС — это представление в формате с фиксированной запятой (дополнение до двух), плавающей запятой и блочное представление в формате с плавающей запятой. Заметим, что можно одновременно выполнять все три операции — дискретизации, квантования и кодирования. Это бывает в тех случаях, когда используется АЦП без схемы выборки-хранения.

Пример 2.9

Объясните значение динамического диапазона и апертурного времени в контексте процесса аналого-цифрового преобразования.

Если в примере 2.2 динамический диапазон АЦП больше, чем 70 дБ, а выборки нужно оцифровать с точностью $\frac{1}{2}$ МЗБ, найдите

- 1) минимальное разрешение АЦП в битах и
- 2) максимальное допустимое апертурное время, предположив, что максимальная важная частота, которую нужно оцифровать, равна 20 кГц.

Решение

Динамический диапазон — это отношение максимального уровня сигнала к минимальному, с которыми может работать система аналого-цифрового преобразователя. Динамический диапазон чаще всего выражается в децибелах через количество битов преобразователя:

$$D = 20 \lg 2^B. \quad (2.19)$$

В некоторых приложениях динамический диапазон определяется через мощность сигнала. Например, в цифровых системах записи и воспроизведения звука он может определяться как отношение максимальной мощности сигнала к минимальной, которую можно отличить от мощности шума. Если используется только АЦП, апертурное время — это, по сути, время преобразования АЦП, т.е. период времени, на протяжении которого аналоговый вход должен оставаться постоянным, чтобы можно было осуществить точное преобразование. Относительно схемы выборки-хранения это время, необходимое для записи после того, как отдана команда о сохранении.

1. Из выражения для D получим

$$70 = 20 \lg 2^B,$$

откуда $B = 11,62$. Пусть $B = 12$ бит (ближайшее целое число).

2. Максимально допустимое апертурное время τ задается как

$$\tau = 1/2^{B+1} \pi f_{\max} = 1/(2^{13} \times \pi \times 20 \times 10^3) \text{ с} = 1,94 \text{ нс}.$$

Данное небольшое апертурное время требуется для использования схемы выборки-хранения на входе АЦП.

2.4.2. Неоднородное квантование и кодирование (нелинейная ИКМ)

Процесс линейного аналого-цифрового преобразования, о котором шла речь выше, иногда называют *линейной импульсно-кодовой модуляцией* (ИКМ). В преобразователях, реализующих ИКМ, уровень шума квантования непосредственно связан с числом битов АЦП. Подобные преобразователи хорошо подходят для приложений, в которых амплитуда сигнала по каким-либо причинам распределена неоднородно, или в которых длина слова АЦП не имеет первостепенного значения. В приложениях, где амплитуда сигнала распределена неоднородно (например, телефонная связь), для точной передачи информации может понадобиться большое количество битов, а это не всегда эффективно.

Некоторые сигналы, например, речь, состоят из гармоник как большой, так и малой амплитуды, хотя малые амплитуды более вероятны. Следовательно, для речи однородное квантование не подходит. Неоднородное квантование может дать больше уровней квантования для сигнала с невысоким уровнем, чем однородное квантование с таким же количеством битов, а в телефонной связи это означает, что система более успешно может подстроиться под абонента и с громким, и с тихим голосом.

Стандартное неоднородное квантование, которое используется в телефонной связи (как в сетях общего пользования, так и в частных телефонных сетях), определяется по известному распределению амплитуды речи. Речевой сигнал дискретизируется с частотой 8 кГц, каждый элемент выборки квантуется и кодируется 8 бит, составляющими систему со скоростью передачи 64 бит/с. Амплитуды элементов речевого сигнала перед передачей сжимаются логарифмически в 8 бит. После приема уплотненные сигналы расширяются.

Процесс сжатия речевого сигнала и его расширения называют *компандированием* (от английского COMpressing — “сжатие” и exPANDING — “расширение”). Этот процесс изображен на рис. 2.26. На практике компандирование осуществляется с помощью кодеков или комби-кодеков (комбинации кодека ИКМ и фильтров защиты от наложения спектров и подавления зеркальных частот), которые подбираются к каждому речевому сигналу в цифровых системах телефонной связи.

В современной цифровой телефонии с кодеками необходим обратный процесс компандирования в цепи, чтобы данные могли обрабатываться методами ЦОС. Уплотненные ИКМ-данные преобразуются в линейные ИКМ-данные, а после цифровой обработки сигнала эти данные преобразуются обратно в неоднородную ИКМ. Операции обращения компандирования могут выполняться микросхемами ЦОС (например, чипами серии Texas Instrument TMS320 или Motorola 56000) как поиск в таблице или путем реализации алгоритмов сжатия и расширения в реальном времени.

2.4.2.1. Способы компандирования: ИКМ с μ - и A-характеристикой

Для неоднородного квантования в телефонии используют два международных стандарта (см. [3]): стандарт с μ -характеристикой (μ -законом) (США и Япония) и стандарт с A-характеристикой (A-законом) (Европа). Оба стандарта сжимают речь в 8-битовые выборки, которые эквивалентны приблизительно 14 бит линейных АЦП.

Характеристики компандирования и уравнения для μ -характеристики показаны на рис. 2.26. Для описания каждого элемента выборки используются разные по знаку

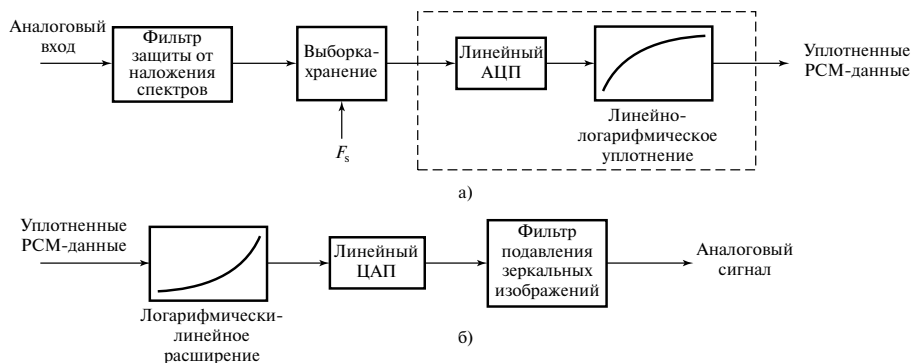


Рис. 2.26. Неоднородные аналого-цифровые и цифроаналоговые преобразования

восьмиразрядные слова. Характеристики компрессирования приблизительно описываются набором из восьми прямолинейных отрезков (см. рис. 2.27). Видно, что последовательно возрастающие элементы входного сигнала сжимаются, формируя на выходе однородный масштаб. Величина шага входного сигнала при переходе от одного отрезка к другому постоянно удваивается. Это облегчает преобразование линейной ИКМ в неоднородную. ИКМ с μ -характеристикой состоит из 8-битовых слов. На рисунке старший значащий разряд (СЗР) — это разряд, обозначающий знак, следующие 3 разряда представляют собой номер отрезка, а последние 4 — положение на нем. На практике до передачи биты инвертируются, чтобы увеличить плотность единиц, так как это облегчает восстановление синхронизации и исправление ошибок. Данный процесс необходим, поскольку речь — это сигнал с небольшой энергией. Как правило, отношение сигнала к шуму квантования при неоднородном квантовании сравнимо с этим же отношением для 14-битового линейного АЦП.

Характеристики A -типа показаны на рис. 2.28. По своим свойствам они похожи на μ -характеристики. Однако A -характеристики дают меньшую точность для небольших сигналов, но обладают превосходной характеристикой динамического диапазона.

2.4.2.2. Адаптивная дифференциальная импульсно-кодовая модуляция (АДИКМ)

При неоднородной ИКМ элементы речевого сигнала квантуются в 8 бит для каждого элемента информации. Известно, что при таком подходе не используется преимущество избыточности речевого сигнала. При АДИКМ [1,2] значение каждого элемента зависит от значения выборки. Это снижает количество битов, которые используются для представления каждого элемента, с 8 до 4 ($8 \text{ кГц} \times 4 \text{ бит} = 32 \text{ Кбит/с}$). Коды АДИКМ передают разницу между предсказанным и действительным значением элемента выборки. На практике для повышения вместимости речевых каналов в системы ИКМ могут встраиваться транскодеры АДИКМ. Кодер АДИКМ принимает значения ИКМ как входные данные, а декодер АДИКМ дает на выходе значения ИКМ.

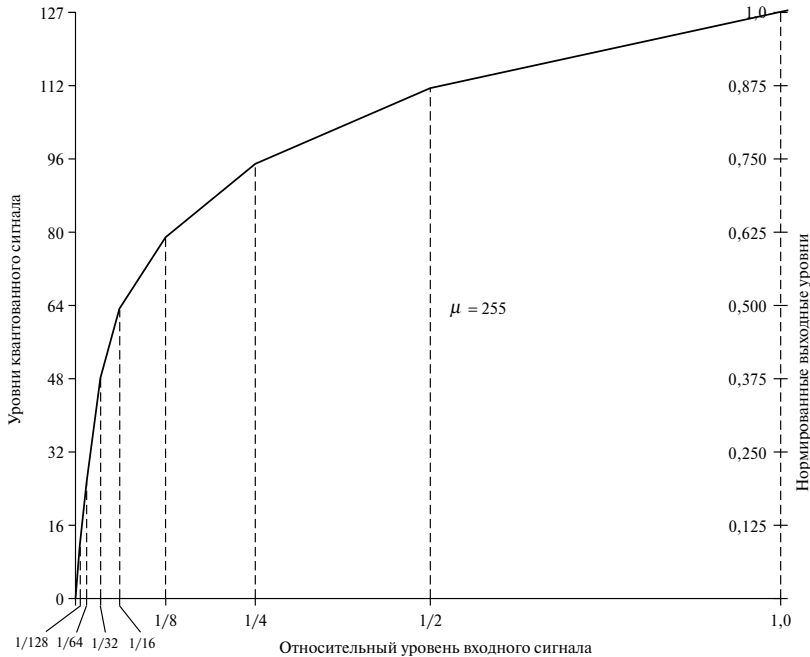


Рис. 2.27. Характеристики компандирования для μ -характеристики с $\mu = 255$ ([ITU, 1998]). Характеристики подчиняются уравнению $F(x) = \text{sgn}(x) \frac{\ln(1+\mu|x|)}{\ln(1+\mu)}$, где $\mu = 255$, x — нормированный входной сигнал, sgn — его знак, а $F(x)$ — сжатый выходной сигнал

Из-за разнообразия существующих стандартов кодирования речи их можно использовать в качестве своеобразных контрольных задач для тестирования многих служб и приложений в сфере коммуникаций. В большинстве случаев акцент делается на снижении скорости передачи информации. Например, скорость передачи речи в системе мобильных телефонов GSM равна 13,2 Кбит/с. Стандарт АДИКМ МККТТ G.721 работает со скоростью 32 Кбит/с.

2.5. Выборка с запасом по частоте при аналого-цифровом преобразовании

2.5.1. Введение

На практике выборка с запасом по частоте означает дискретизацию входного сигнала с частотой намного большей, чем частота Найквиста. Отношение действительной частоты дискретизации к частоте Найквиста называется *коэффициентом перевыборки* (сигнал предполагается низкочастотным):

$$\text{коэффициент перевыборки} = F_s / 2f_{\max}. \quad (2.20)$$

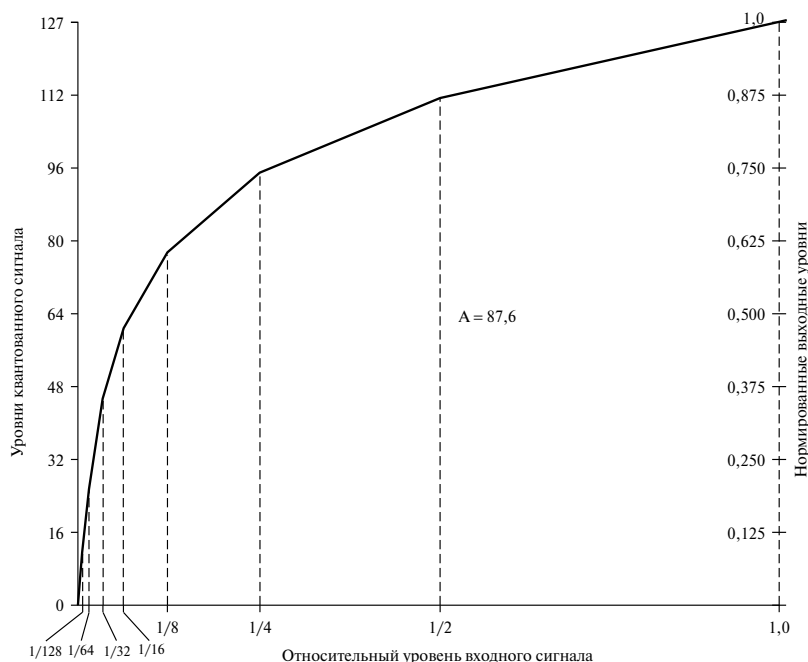


Рис. 2.28. Характеристики компрессирования для А-характеристики ($A = 255$) [ITU, 1998]. Характеристики подчиняются уравнению $F(x) = \text{sgn}(x) \frac{A|x|}{1+\ln(A)}$, $0 \leq |x| < 1/A$; $F(x) = \text{sgn}(x) \frac{1+\ln(A|x|)}{1+\ln(A)}$, $1/A \leq |x| < 1$, где $A = 87,6$, x — нормированный входной сигнал, sgn — его знак, а $F(x)$ — сжатый выходной сигнал

Во многих областях современной ЦОС существует тенденция всегда использовать выборку с запасом по частоте, чтобы сыграть на практических результатах теоремы о дискретном представлении. Для аналого-цифрового интерфейса главными преимуществами выборки с запасом по частоте являются: 1) упрощение фильтра защиты от наложения спектров; 2) возможность использовать фильтры защиты от наложения спектров с переменной частотой среза (каждая частота среза требует своей частоты дискретизации) и 3) снижение минимального уровня шума АЦП путем распределения шума квантования по более широкой полосе частот. Это позволяет использовать АЦП с меньшим количеством битов, чтобы достичь такого же отношения сигнал-шум, как и при более высоком разрешении АЦП.

2.5.2. Выборка с запасом по частоте и фильтры защиты от наложения спектров

В высокоточных цифровых системах из-за необходимости поддерживать низкий уровень искажения от наложения часто приходится пользоваться относительно сложными аналоговыми фильтрами защиты от наложения спектров. В многоканальных системах каждому аналоговому сигналу должен соответствовать отдельный фильтр защиты от

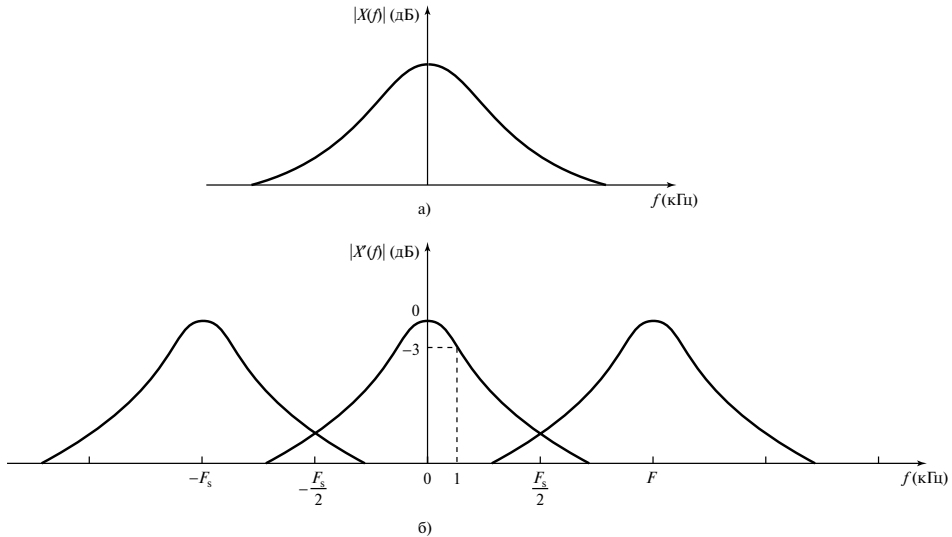


Рис. 2.29. Спектр входного сигнала: а) до дискретизации; б) после дискретизации

наложения, поскольку такие фильтры сложно уплотнить. При большом количестве каналов это зачастую очень дорого (например, в биомедицинских приложениях бывает нужно 64 канала). Более того, может оказаться трудно согласовать по фазе узкоспециализированные аналоговые фильтры (для сохранения взаимосвязи между компонентами сигнала во всех каналах).

Многие из этих проблем позволяют преодолеть методы выборки с запасом по частоте. Как следует из теории спектра дискретного сигнала, чем выше частота дискретизации, тем дальше друг от друга находятся зеркальные компоненты и полоса частот исходного сигнала (см., например, рис. 2.29). В главе 9 показано, как методы выборки с запасом по частоте сочетаются с цифровой обработкой сигналов при нескольких скоростях и удовлетворяют требованиям высокоточных систем.

Пример 2.10

1. К универсальной многоканальной (до 64 каналов) системе, собирающей нейрофизиологические данные, существует одно требование. Каждый аналоговый канал должен быть индивидуально задан пользователем так, чтобы граничная частота его полосы пропускания была в пределах от 0,5 кГц до 500 Гц, а выбираемая частота дискретизации — от 1 Гц до 5 кГц. Максимально допустимая неравномерность в полосе пропускания равна 0,5 дБ, а зеркальные компоненты должны быть по крайней мере на 40 дБ ниже компонента сигнала. Объясните подход, использованный для удовлетворения вышеизложенного требования. Ответ должен включать следующие пункты:
 - а) рассуждения о вопросах, специфичных для данного приложения;
 - б) указания, как в этом приложении для наиболее эффективного и экономного удовлетворения указанного требования (используйте расчет цены/числа компонентов) можно использовать методы дискретизации с запасом по частоте.

2. Пусть для всех каналов системы из п. 1 используются идентичные фильтры защиты от наложения спектров, каждый из которых имеет следующую характеристику типа Баттерворта:

$$A(f) = \frac{1}{\sqrt{1 + \left(\frac{f}{f_c}\right)^6}},$$

где $f_c = 3$ дБ — частота среза фильтра.

3. Изобразите спектры данных до и после дискретизации и определите:

- частоту среза f_c ;
- подходящую общую частоту дискретизации F_s .

Ответ прокомментируйте.

Решение

1. АЦП и ЦАП с высоким разрешением ограничивают максимально достижимую частоту дискретизации — данное ограничение является основным для ряда приложений реального времени. Для его преодоления может потребоваться использовать несколько устройств АЦП/ЦАП и/или методов ЦОС с обработкой при нескольких скоростях.

Изменение характеристики на выходе по закону $\sin x/x$, постепенно снижающее высокочастотные компоненты сигнала, можно компенсировать, используя после обработки цифровой фильтр с характеристикой $x/\sin x$. (В своем ответе вы можете использовать и другие технические решения.)

- а. Чтобы сохранить информацию сигнала, представляющую клинический интерес, и амплитудные, и фазовые искажения должны быть как можно ниже. Кроме того, должна сохраняться временная связь между характеристиками всех каналов. Желательно использовать идентичные фильтры защиты от наложения спектров с достаточно хорошими амплитудно-фазовыми характеристиками.
- б. Чтобы уменьшить число/цену компонентов и размер устройства управления системой, всем 64 каналам должны соответствовать идентичные простые фильтры защиты от наложения спектров. Затем должна выполняться дискретизация каналов с запасом по частоте с общей фиксированной скоростью. Высокую общую скорость дискретизации можно снизить до желаемого значения с помощью методов обработки при нескольких скоростях. Во избежание чрезмерно высокой скорости общей дискретизации следует использовать фильтр Баттерворта как минимум второго порядка.
- Рассмотрев спецификации и спектр данных до и после дискретизации, находим следующее.
- а. Чтобы не нарушать спецификации, ошибка по амплитуде в пределах от 0 до 500 кГц должна удовлетворять следующему критерию:

$$20 \log \left[1 + \left(\frac{500}{f_c} \right)^6 \right]^{\frac{1}{2}} \leq 0,5 \text{ дБ},$$

учитывая, что используется фильтр Баттерворта третьего порядка с частотой среза f_c .

Решив это уравнение относительно f_c , находим, что

$$f_c = \frac{500^6}{0,122} \approx 710 \text{ Гц.}$$

Сделав поправку на дополнительные искажения на последующих этапах (и для удобства), положим $f_c = 1000$ Гц (это эквивалентно максимальной неравномерности 0,26 дБ).

- 3, б. После ограничения полосы частот каждого канала спектр дискретного сигнала будет иметь вид, показанный на рис. 2.29.

Теперь F_s выбирается так, чтобы уровень искажения вследствие наложения при частоте 500 Гц по крайней мере был больше 40 дБ, т.е.

$$20 \log \left[1 + \left(\frac{F_s - 500}{1000} \right)^6 \right]^{\frac{1}{2}} \geq 40 \text{ дБ.}$$

Решив это уравнение относительно F_s , получим $F_s = 5141,5$ Гц.

К выбору общей частоты дискретизации F_s следует отнестись внимательно и учесть эффективное снижение скорости дискретизации. Возможное решение — 8192 Гц, что допускает снижение общей скорости в n раз (n — простое целое число).

2.5.3. Выборка с запасом по частоте и разрешающая способность АЦП

При выборке с запасом по частоте энергия шума квантования входного сигнала распределяется по более широкому диапазону частот, таким образом снижается уровень шума в важной полосе и увеличивается разрешающая способность АЦП. Это используется в системах записи и воспроизведения звука для получения так называемого однобитового АЦП или АЦП с выборкой с запасом по частоте. Приведем для начала основные понятия, связанные с квантованием и его ошибками.

2.5.3.1. Квантование и ошибки квантования

В традиционном процессе аналого-цифрового преобразования каждая выборка сигнала квантуется в один из 2^B уровней и описывается B бинарными разрядами, где B — число битов АЦП. Квантование вносит ошибку, которая является функцией числа битов АЦП.

Мощность шума квантования (для однородно распределенной ошибки с нулевым средним значением) задается как

$$\sigma_e^2 = \frac{q^2}{12},$$

где q — величина шага квантования. Теоретическое максимальное отношение сигнала к шуму квантования (SNR(q)) для линейного АЦП задается как

$$\text{SNR}(q) = 6,02B + 4,77 - 20 \log(A/\sigma_x) \text{ дБ}, \quad (2.21)$$

где $\pm A$ — входной диапазон АЦП, а σ_x — среднеквадратическое значение входного сигнала. Для синусоидального входа с максимальной амплитудой A , которая как раз заполняет диапазон АЦП, $\sigma_x = \frac{A}{\sqrt{2}}$, $20 \log \left(\frac{A}{\sigma_x} \right) = 3,01$ дБ, таким образом, уравнение (2.21) сводится к знакомому виду:

$$\text{SNR}(q) = 6,02B + 1,7 \text{ дБ.} \quad (2.22)$$

У биполярного линейного 16-битового АЦП, например с входным диапазоном ± 5 В, величина шага квантования $q = \frac{10\text{В}}{2^{16}-1} = 0,152$ мВ, максимальная ошибка квантования — $\frac{q}{2} = 76$ мкВ, а $\text{SNR}(q) = 98$ дБ.

Задание

Синусоидальный сигнал с удвоенной амплитудой 10 В оцифровывается с помощью 12-битового АЦП. Предположив, что квантование линейно, определите

- 1) шаг квантования;
- 2) мощность шума квантования;
- 3) теоретическое максимальное отношение сигнала к шуму квантования.

2.5.3.2. Выборка с запасом по частоте и мощность шума квантования

Собственная мощность шума квантования, который вносится при преобразовании аналогового сигнала в цифровой, задается как

$$\sigma_e^2 = \frac{q^2}{12} = \frac{2^{-2(B-1)}}{12} \quad (\text{нормированная}), \quad (2.23)$$

где B — длина слова АЦП (включая бит знака).

Для достаточно больших или случайных входных аналоговых сигналов энергия шума квантования равномерно распределяется по всему доступному спектру, т.е. от 0 до $F_s/2$, где F_s — частота дискретизации. В этом случае спектральная плотность мощности шума квантования $P_e(f)$ задается как (см. рис. 2.30)

$$P_e(f) = \frac{\sigma_e^2}{F_s}. \quad (2.24)$$

Следовательно, эффективное разрешение АЦП можно увеличить, дискретизируя входные данные с высокой частотой, чтобы энергия шума квантования распределялась по более широкой полосе частот, снижая таким образом уровень шума в важной полосе. Именно это и понимается под дискретизацией с запасом по частоте.

Согласно уравнению (2.20) при преобразовании с частотой Найквиста $f_{\max} = F_s/2$, а общая мощность шума в полосе задается площадью, указанной на рис. 2.30, а, т.е. σ_e^2 . При преобразовании с выборкой с запасом по частоте (рис. 2.30, а) некоторая часть мощности шума квантования попадает за пределы нужной полосы (поскольку $f_{\max} < F_s/2$), и шум в полосе меньше, чем при частоте Найквиста.

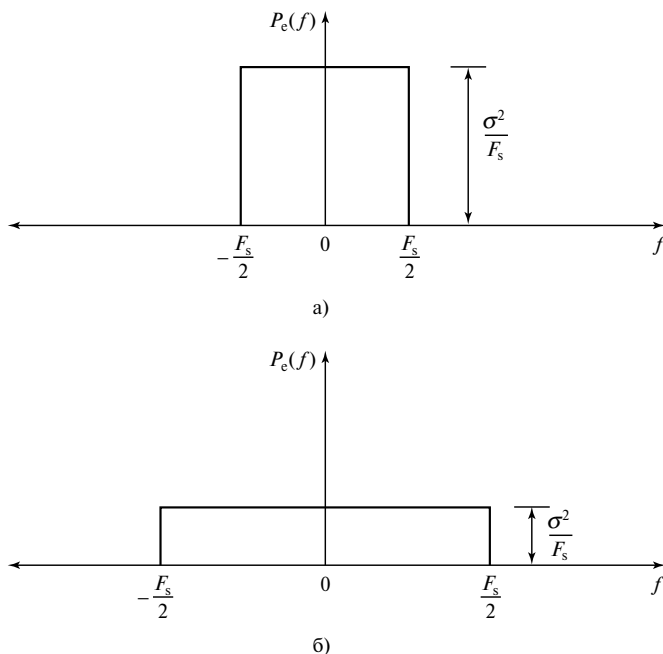


Рис. 2.30. Спектральная плотность мощности шума квантования: а) для преобразования с частотой Найквиста; б) для выборки с запасом по частоте. (Общая мощность шума одинакова в обоих случаях, но при выборке с запасом по частоте мощность шума распределяется по намного большему частотному диапазону, что приводит к снижению уровня мощности шума в полосе)

Мощность шума в полосе при выборке с запасом по частоте задается как

$$P_e = \int_{-f_{\max}}^{f_{\max}} P_e(f) df = \frac{2f_{\max}}{F_s} \sigma_e^2. \quad (2.25)$$

Следовательно, если сигнал с ограниченной полосой частот дискретизовать с запасом по частоте, энергия шума квантования в полосе сигнала уменьшается на коэффициент пере выборки. На практике для простоты вычислений выбирается коэффициент, равный двум в целой степени.

Пример 2.11

1. Система записи и воспроизведения звука работает с сигналами, основная полоса частот которых лежит в диапазоне 0–20 кГц. Найдите коэффициент перевыборки и минимальную частоту дискретизации, которая необходима для достижения такого же качества при использовании 12-битового преобразователя, которое можно было бы получить с помощью 16-битового АЦП.
2. В цифровой системе звукозаписи для оцифровки аналогового входного сигнала с частотными компонентами в диапазоне 0–4 кГц используется метод выборки с запасом по частоте и 8-битовый биполярный преобразователь, работающий с частотой Найквиста. Оцените эффективное разрешение преобразователя (в битах), если частота дискретизации равна 40 МГц. Прокомментируйте практические проблемы, связанные с этим методом.

Решение

1. При частоте Найквиста (т.е. $F_s = 2f_{\max}$) нормированная мощность шума квантования в полосе для 12- и 16-битового преобразователей соответственно равна

$$\sigma_1^2 = 2^{-\frac{2(B_1-1)}{12}} \quad (\text{где } B_1 = 12),$$

$$\sigma_2^2 = 2^{-\frac{2(B_2-1)}{12}} \quad (\text{где } B_2 = 16).$$

Для достижения 16-битового качества с 12-битовым АЦП вход 12-разрядного преобразователя нужно дискретизовать с запасом по частоте, чтобы снизить мощность шума квантования в полосе на величину, равную коэффициенту перевыборки

$$(\sigma'_1)^2 = \frac{2f_{\max}}{F_s} \sigma_1^2.$$

Приравнивая новый шум квантования в полосе к шуму 16-битового АЦП, получим

$$\frac{2f_{\max}}{F_s} \sigma_1^2 = \sigma_2^2.$$

Таким образом,

$$\frac{2f_{\max}}{F_s} = \frac{\sigma_2^2}{\sigma_1^2} = \frac{2^{-2(B_2-1)}}{2^{-2(B_1-1)}} = 2^{-2(B_2-B_1)} = \frac{1}{256}.$$

Следовательно, коэффициент перевыборки задается как

$$F_s/(2f_{\max}) = 256, \text{ т.е. } F_s = 10,24 \text{ МГц.}$$

2. Шум квантования в полосе снижается в число раз, равное коэффициенту перевыборки, т.е. на

$$\frac{40000}{2 \times 4} = 5000.$$

Учитывая, что $\frac{\sigma_1^2}{\sigma_2^2} = \frac{2^{-2(B_1-1)}}{2^{-2(B_2-1)}} = 2^{2(B_2-B_1)} = 5000$ и $B_1 = 8$ бит, находим, что разрешение АЦП B_2 приблизительно равно 14 бит.

Как следует из предыдущих примеров, метод выборки с запасом по частоте сам по себе невыгоден с точки зрения достижения желаемого разрешения при использовании АЦП с небольшим разрешением, поскольку повышение разрешающей способности часто требует очень высокой частоты дискретизации, которую данный метод может не поддерживать.

На практике выборка с запасом по частоте сочетается с ограничением шума, что позволяет сместить шум квантования в область более высоких частот за пределы полосы частот сигнала, где его можно устранить с помощью фильтра. В следующем разделе описан принцип действия АЦП для дискретизации с запасом по частоте.

2.5.4. Применение выборки с запасом по частоте — одноканальный АЦП

Требования таких высококачественных систем ЦОС, как цифровые системы записи и воспроизведения звука, — высокое качество, большая частота и очень скоростные (и дешевые!) АЦП, — сложно удовлетворить с помощью обычного последовательного приближения или преобразователей с двойными огибающими из-за ошибок, присущих аналоговым частям таких преобразователей (например, корректирующих схем АЦП, фильтров защиты от наложения спектров и контуров выборки-хранения).

Одноканальные АЦП или, точнее, АЦП для выборки с запасом по частоте не нуждаются в усилителях выборки-хранения, в них применяются простые фильтры защиты от наложения спектров, или вообще не используются никаких фильтров, таким образом, они не имеют большей части вышеперечисленных недостатков.

Существует два подхода, делающих возможным существование одноканальных АЦП.

- Выборка с запасом по частоте, которая распределяет энергию шума квантования по намного большему частотному интервалу, снижая таким образом уровень шума в представляющей интерес полосе частот.
- Формирование спектра шума, сдвигающее большую часть шума в область более высоких частот, за пределы важной полосы сигнала, где шум можно устранить с помощью цифрового фильтра.

Принцип действия АЦП выборки с запасом по частоте показан на рис. 2.31. Аналоговый входной сигнал дискретизируется с запасом по частоте (например, в 64 раза), чтобы распределить мощность шума квантования по широкой полосе частот. Впоследствии дискретные данные обрабатываются для формирования спектра шума, чтобы сместить мощность шума в область более высоких частот, удалить ее из полосы частот сигнала. Затем частота дискретизации путем децимации уменьшается до частоты Найквиста. Этот процесс также служит для преобразования данных из одноканального в многоканальный поток информации. Более подробно децимация описывается в главе 9.

Одним из самых эффективных способов формирования спектра шума является сигма-дельта-модуляция. На рис. 2.32 изображен сигма-дельта-модулятор первого порядка. Он состоит из интегратора, одноканального устройства квантования (т.е. одноканального АЦП,

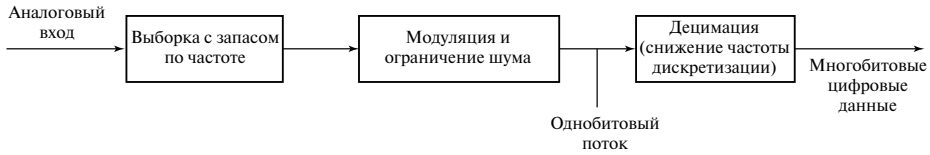


Рис. 2.31. Принципы (однобитового) аналого-цифрового преобразования с запасом по частоте



Рис. 2.32. Сигма-дельта-модулятор первого порядка

выполняющего роль компаратора) и однобитового ЦАП для обеспечения обратной связи. Аналоговый входной сигнал $x(t)$ дискретизируется с очень высокой частотой, а затем квантуется в однобитовый поток, который содержит очень большой шум квантования. Соответствующим подбором характеристик интегратора спектр шума ограничивают так, чтобы большая часть энергии шума смещалась за пределы полосы сигнала.

Чтобы выяснить, как модулятор формирует спектр шума квантования, рассмотрим модель сигма-дельта-модулятора первого порядка на z -плоскости (рис. 2.33), где предполагается, что элементы выборки шумового сигнала не коррелируют.

Согласно рисунку z -преобразование выхода задается как

$$\begin{aligned} Y(z) &= E(z) + [X(z) - Y(z)z^{-1}] \left(\frac{1}{1 - z^{-1}} \right) = \\ &= X(z) + E(z)(1 - z^{-1}) = X(z) + E(z)H_n(z), \end{aligned} \quad (2.26)$$

где

$X(z)$ = z -образ входного сигнала,

$Y(z)$ = z -образ однобитового выхода,

$E(z)$ = z -образ шума квантования,

$H_n(z) = (1 - z^{-1})$ — передаточная функция шума.

Уравнение (2.26) ясно показывает, что образ выхода идентичен образу входа плюс шум квантования, модифицированный передаточной функцией шума. Передаточная функция шума $(1 - z^{-1})$ описывает, по сути, фильтр верхних частот с нулем на нулевой частоте. Действие этого фильтра заключается в смещении энергии шума квантования в более высокочастотную область спектра (см. рис. 2.34).

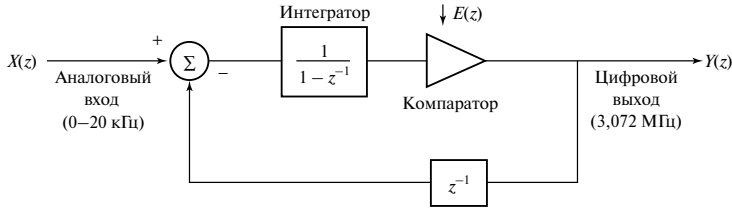
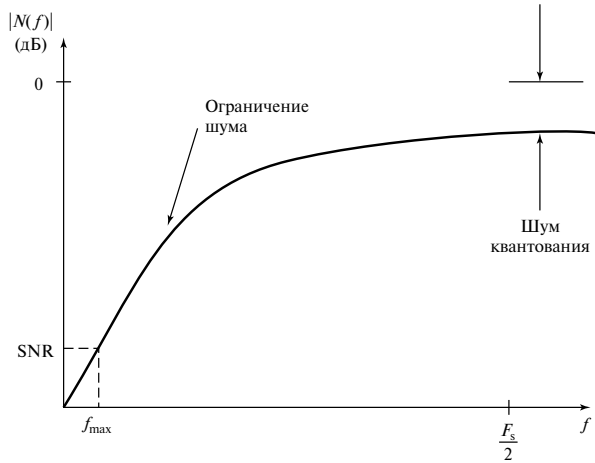
Рис. 2.33. Модель сигма-дельта-модулятора первого порядка на z -плоскости

Рис. 2.34. Влияние схемы формирования спектра шума на шум квантования

Для системы, входная полоса частот которой ограничена величиной $f_{\max}$, мощность шума в полосе после ограничения шума задается как

$$\sigma_n^2 = \int_{-f_{\max}/F_s}^{f_{\max}/F_s} |H_n(f)|^2 P_d df. \quad (2.27)$$

Ясно, что действие сигма-дельта-модулятора не зависит от коэффициента перевыборки и способности сигма-дельта-модулятора ограничивать спектр шума. Для сигма-дельта-модулятора первого порядка удвоение частоты дискретизации увеличивает отношение сигнал-шум на 9 дБ, из которых 6 относится к ограничению шума, а оставшиеся 3 — к выборке с запасом по частоте. Еще большего уменьшения шума квантования можно достичь, увеличив порядок передаточной функции шума (т.е. интегратора). Можно показать, что для сигма-дельта-модулятора N -го порядка выходное преобразование задается как

$$Y(z) = X(z) + E(z)(1 - z^{-1})^N. \quad (2.28)$$

Это дает фильтр шума с характеристикой сглаживания $6N$ дБ/октаву. К сожалению, при $N > 3$ устойчивость модулятора гарантировать нельзя из-за большого фазового

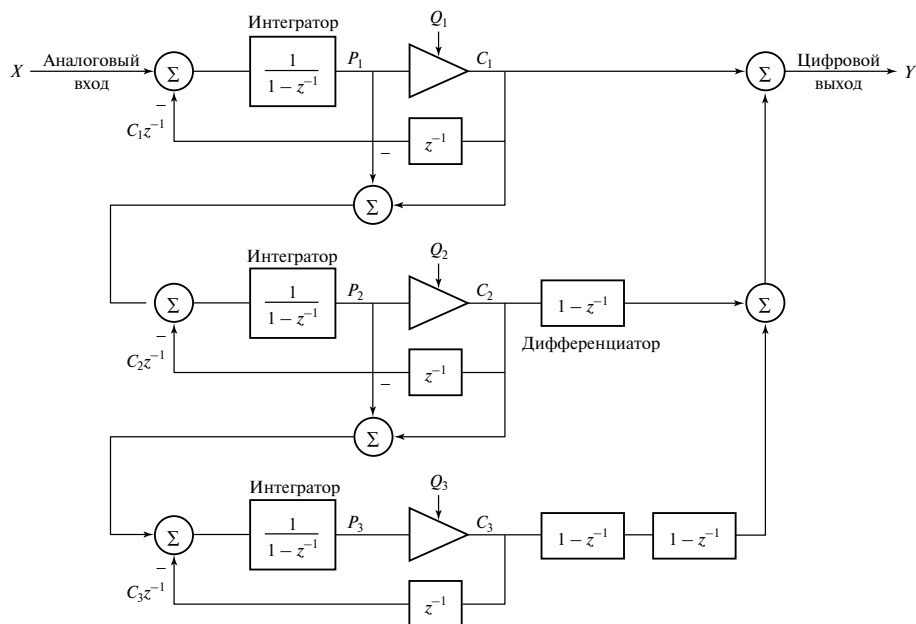


Рис. 2.35. Модель сигма-дельта-модулятора MASH третьего порядка на z -плоскости

сдвига. Для сигма-дельта-модуляторов, порядок которых больше двух, для борьбы с нестойчивостью используют специальные конфигурации. Одна из таких конфигураций известна под названием MASH. На рис. 2.35 показано устройство MASH для сигма-дельта-модулятора третьего порядка.

Выход сигма-дельта-модулятора MASH третьего порядка задается как

$$Y(z) = X(z) + E(z)(1 - z^{-1})^3. \quad (2.29)$$

Заметим, что на выход влияет только шум квантования $E_3(z)$ последнего этапа, а шум первых двух этапов подавляется.

Независимо от порядка сигма-дельта-модулятора полоса частот его выхода содержит очень мало шума квантования, но за пределами полосы этот шум очень велик. Для устранения шума за пределами полосы используется цифровой фильтр нижних частот. Из-за высокой частоты дискретизации непосредственно использовать цифровой фильтр непрактично. Вместо этого фильтрация осуществляется с помощью децимации, которая служит также для снижения частоты до желаемого значения. После фильтрации полученный сигнал приобретает вид B -битовых квантованных данных. Фильтрация служит для усреднения высокого шума квантования. Как правило, коэффициенты КИХ-фильтра для децимации описываются 16–24 бит.

Упрощенная блок-схема быстрого процесса однобитового аналого-цифрового преобразования показана на рис. 2.36. Сперва аналоговый звуковой сигнал преобразуется в однобитовый поток с помощью сигма-дельта-модуляции на частоте 3,071 МГц. Затем

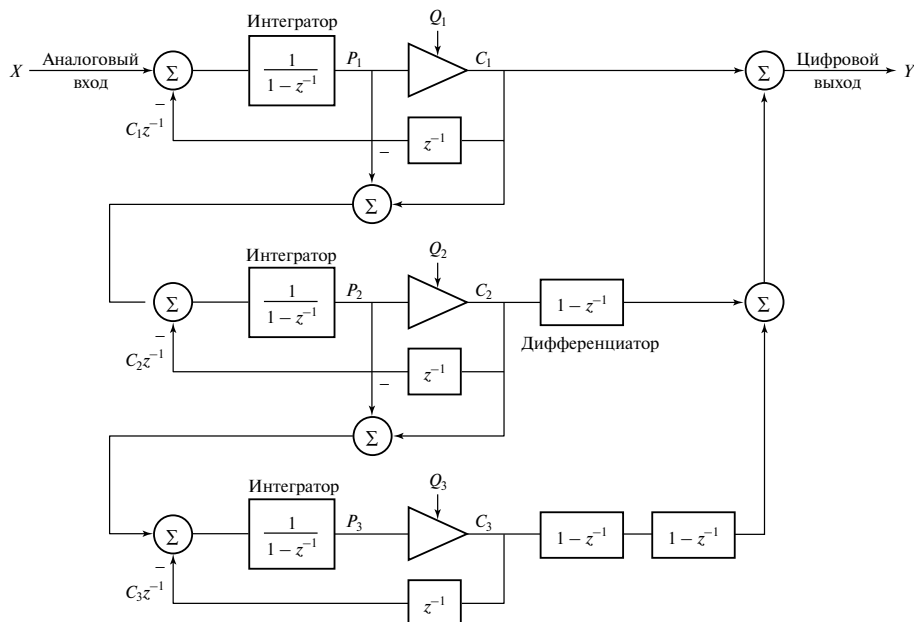


Рис. 2.36. Упрощенный однобитовый АЦП

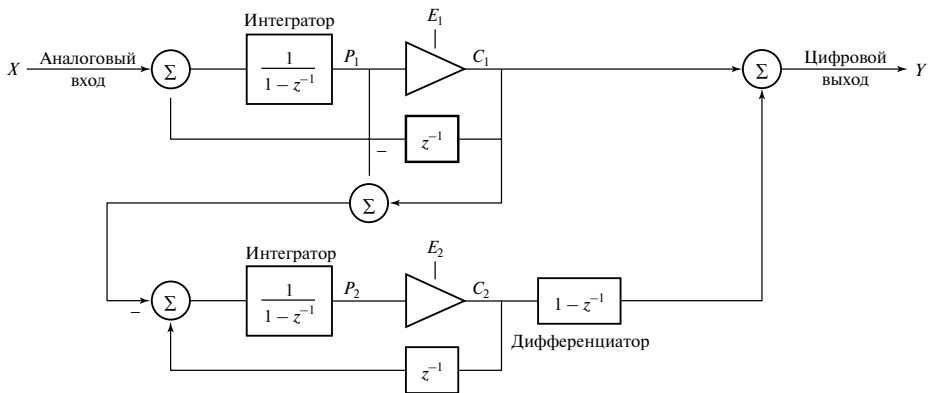
однобитовый поток дискретизуется с недостаточной частотой до 48 кГц с помощью многокаскадного дециматора (см. главу 9) и преобразуется в 16-битовые слова линейной ИКМ.

Для заданного типа сигнала эффективная длина слова АЦП определяется по отношению сигнал-шум, которое достигается после выборки с запасом по частоте, ограничения шума и децимации. Например, если нужно получить 16-битовый АЦП, то отношение сигнал-шум должно составлять как минимум 96 дБ. В настоящее время широко доступно множество коммерческих АЦП для выборки с запасом по частоте.

Пример 2.12

В системе цифровой обработки сигнала с аналоговым звуковым входом в интервале частот 0–20 кГц для преобразования сигнала в цифровой поток битов с частотой 3,072 МГц используется метод выборки с запасом по частоте и сигма-дельта-модулятор второго порядка. Модель сигма-дельта-модулятора на z -плоскости изображена на рис. 2.37.

1. Объясните, каким образом цифровой поток битов можно преобразовать в цифровой многобитовый поток с частотой 48 кГц.
2. Определите общее улучшение отношения сигнал-шум, возможное благодаря дискретизации с запасом по частоте и формированию спектра шума, а затем оцените (в битах) эффективную разрешающую способность устройства для оцифровки.

Рис. 2.37. Модель сигма-дельта-модулятора второго порядка на z -плоскости (пример 2.12)

Решение

1. Однобитовый поток преобразуется в многобитовые слова с помощью децимации (процесса понижения частоты дискретизации). Полоса частот выхода сигма-дельта-модулятора содержит очень мало шума квантования, но за пределами полосы этот шум очень велик. Для устранения шума за пределами полосы используется цифровой фильтр нижних частот. Из-за высокой частоты дискретизации непосредственно использовать цифровой фильтр непрактично. Вместо этого фильтрация осуществляется с помощью децимации, которая служит также для снижения частоты до желаемого значения. Как правило, используется двухкаскадный дециматор (с коэффициентами децимации 16 и 4). После фильтрации полученный сигнал приобретает вид B -битовых квантованных данных. Фильтрация служит для усреднения высокого шума квантования. Как правило, коэффициенты КИХ фильтра децимации описываются 16–24 бит.
2. Эффективное разрешение можно оценить с помощью следующего упрощенного анализа.

Мощность шума снижается путем дискретизации с запасом по частоте и ограничения шума. Снижение мощности шума, обусловленное выборкой с запасом по частоте, задается коэффициентом перевыборки.

Коэффициент перевыборки равен

$$\frac{F_s}{2f_{\max}} = \frac{3,072 \times 10^6}{2 \times 24 \times 10^3} = 64.$$

Это эквивалентно снижению мощности шума квантования на 18 дБ.

Из модели сигма-дельта-модулятора на z -плоскости находим, что передаточная функция шума квантования задается как

$$N(z) = (1 - z^{-1})^2.$$

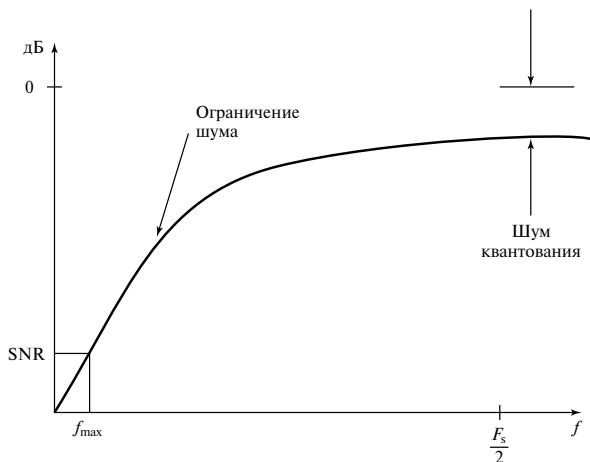


Рис. 2.38. Влияние схемы формирования спектра шума на шум квантования

Эта функция, по сути, описывает фильтр верхних частот с двойным нулем на нулевой частоте, который ослабляет компоненты шума в низкочастотной области (см. рис. 2.38). Характеристика этого фильтра задается следующим образом:

$$|N(z)|_{z=e^{i\omega T}}^2 = |(1 - e^{-i\omega T})^2|^2.$$

При $f = 24$ кГц (край полосы) и $F_s = 3,072$ МГц, $\omega T = 2,8125^\circ$ и

$$|N(e^{i\omega T})|^2 = 2,412 \times 10^{-3}.$$

Это позволяет снизить отношение сигнал-шум квантования на 52,35 дБ. Эффективная длина слова АЦП определяется, главным образом, по отношению сигнал-шум, которое достигается путем дискретизации с запасом по частоте и ограничения шума. Общее снижение отношения сигнал-шум квантования составляет 70,41 дБ. Это соответствует эффективному разрешению АЦП в 11,4 бит (поскольку отношение сигнал-шум квантования равно $6,02B + 1,77$ дБ).

2.6. Процесс цифроаналогового преобразования: восстановление сигнала

Процесс цифроаналогового преобразования позволяет трансформировать цифровой сигнал в аналоговую форму после его цифровой обработки, передачи или хранения в цифровой форме. Такое преобразование необходимо, например, для генерации звукового сигнала, который передается в динамик (как в системе воспроизведения компакт-дисков), или озвучивания аварийного сигнала. Самое распространенное из используемых устройств показано на рис. 2.39; видно, что оно состоит из двух главных элементов: ЦАП (цифроаналогового преобразователя) и фильтра нижних частот, который иногда называют *фильтром восстановления*, *сглаживания* или *фильтром подавления зеркальных частот*, *интегрирующим фильтром*, но чаще всего, конечно, ФНЧ.

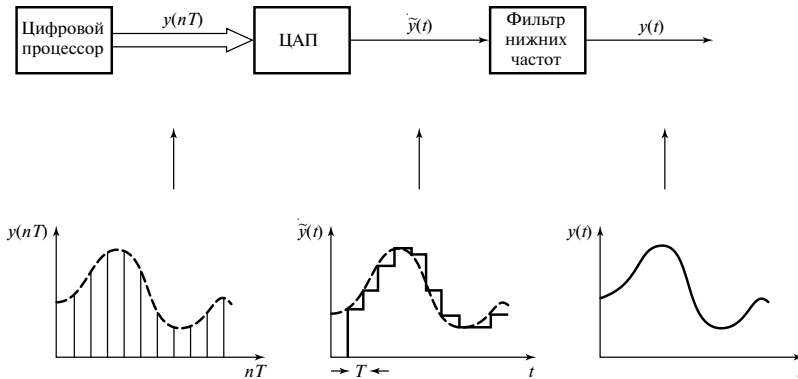


Рис. 2.39. Процесс цифроаналогового преобразования, который используется для восстановления аналогового сигнала после цифровой обработки. Заметим, что вход ЦАП — это серия импульсов, тогда как выход ЦАП имеет форму лесенки, поскольку каждый импульс не изменяется на протяжении времени T (s)

2.7. Цифроаналоговый преобразователь

Стандартный ЦАП принимает параллельные цифровые данные и создает аналоговый выходной сигнал, являющийся функцией входного цифрового кода. Для буферизации входа ЦАП используется регистр, чтобы выход ЦАП оставался без изменений, пока ЦАП принимает следующую порцию цифрового входа. Регистр может быть внешним по отношению к ЦАП или являться частью чипа ЦАП, как на рис. 2.39. В некоторых приложениях может потребоваться дополнительный контур, который будет предотвращать появление на выходе ЦАП кратковременных пиков, вызванных ложными цифровыми кодами.

ЦАП, показанный на рисунке, называют *ЦАП с задержкой нулевого порядка*. При сравнении его выхода $\tilde{y}(t)$ и входа $y(nT)$ становится очевидно, что для каждого цифрового кода, вводимого в ЦАП, его выход задерживается на время T . В результате получается ступенчатая выходная характеристика ЦАП. В частотных координатах задерживающее действие ЦАП вносит некоторое искажение, известное как *искажение по закону $\sin x/x$* , или *апертурное искажение*, где $x = \omega T/2$.

На рис. 2.40 показаны входной и выходной сигналы ЦАП с задержкой нулевого порядка в частотных и временных координатах, позволяющее сделать следующие выводы.

- Входной и выходной сигналы ЦАП имеют широкую полосу частот. Каждый из них состоит из спектра оцифрованного сигнала плюс бесконечное число зеркальных изображений исходного спектра, центрированных на частотах, кратных частоте дискретизации.
- Амплитуда спектра выходного сигнала умножается на функцию $\sin x/x$, которая действует как фильтр нижних частот, значительно ослабляя зеркальные частоты.

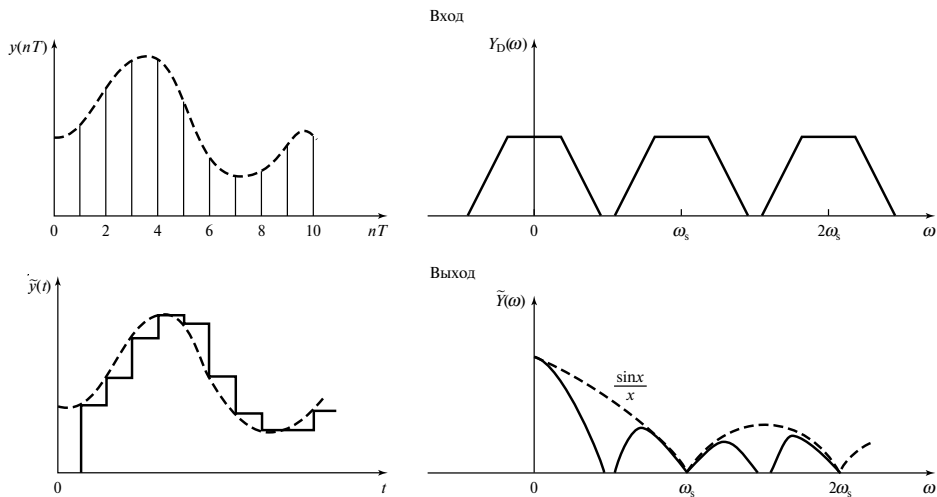


Рис. 2.40. Входной и выходной сигналы ЦАП в частотных и временных координатах. Заметим, что изменение характеристики по закону $\sin x/x$ можно представить гипотетическим фильтром нижних частот

Искажение по закону $\sin x/x$ обусловлено задерживающим действием ЦАП, и при восстановлении сигнала оно является причиной амплитудного искажения. Среднюю ошибку, обусловленную этим действием, при заданной частоте можно выразить в процентном отношении:

$$(1 - \sin x/x) \times 100\%. \quad (2.30)$$

Для задержки нулевого порядка функция $\sin x/x$ спадает приблизительно до 4 дБ на половине частоты дискретизации ($F_s/2$), что дает среднюю ошибку приблизительно в 36,4%. Апертурную ошибку можно исключить с помощью выравнивания. На практике для этого сигнал до его преобразования в аналоговую форму пропускается через цифровой фильтр, амплитудно-частотная характеристика которого имеет равна $\sin x/x$.

В некоторых приложениях для вставки новых выборок между действительными элементами выборки, которые задает ЦАП (данный процесс также называется *интерполяцией*), можно использовать цифровой процессор. Это помогает сглаживать аналоговый сигнал и дает значительно лучший результат, чем простая задержка нулевого порядка. Еще один способ, который пользуется все большей популярностью в тех случаях, когда нужны высококачественные звуковые сигналы, — цифроаналоговое преобразование с намного большей частотой, чем это предусмотрено теоремой о дискретном представлении (см., например, [5]). Для этого используются методы обработки при нескольких скоростях. В результате получается лучшее отношение сигнал-шум и упрощается фильтр защиты от зеркальных частот. Более подробно данный подход описывается в разделе 2.9 и главе 9.

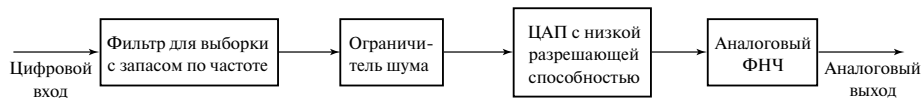


Рис. 2.41. Блок-схема основных принципов цифроаналогового преобразования с выборкой с запасом по частоте

2.8. Фильтры защиты от зеркальных частот

Выход ЦАП наряду с нужными частотными компонентами содержит и нежелательные высокочастотные компоненты, или зеркальные изображения, центрированные на частотах, кратных частоте дискретизации (т.е. частоте обновления). В зависимости от приложения высокочастотные компоненты могут вызывать нежелательные побочные эффекты. Например, хотя в проигрывателе компакт-дисков зеркальные частоты и не слышны, они могут перегружать усилитель и вызывать взаимную модуляцию с частотными компонентами, попадающими в основную полосу частот. Результат — недопустимое снижение качества звучания.

Роль выходного фильтра (т.е. фильтра подавления зеркальных частот) заключается в сглаживании шагов выхода ЦАП и устранении таким образом нежелательных высокочастотных компонентов. Требования к спаду характеристики выходного фильтра зависят от влияния, которое аналоговый сигнал окажет на последующие аналоговые этапы обработки. В общих чертах требования к фильтру подавления зеркальных частот похожи на требования к фильтру защиты от наложения спектров.

2.9. Выборка с запасом по частоте при цифроаналоговом преобразовании

Мотивация использования выборки с запасом по частоте в ЦАП подобна мотивации ее использования в АЦП. При ЦАП для выборки с запасом по частоте частота дискретизации данных, которые нужно преобразовать в аналоговые, возрастает во много раз (например, в 64 раза), что позволяет получать аналоговые выборки с намного меньшими промежутками между ними. Таким образом, для сглаживания или устранения шума вне полосы сигнала нужен всего лишь относительно простой фильтр подавления зеркальных частот. Дискретизация с очень высокой частотой создает сигнал с шириной полосы частот от 0 до половины частоты дискретизации. По этой более широкой полосе равномерно распределяется мощность шума квантования, в результате чего можно достичь высокого разрешения цифроаналогового преобразования при низкой разрешающей способности ЦАП.

Как и для АЦП, выборка с запасом по частоте сама по себе не позволяет достичь желаемого разрешения ЦАП, поэтому необходимо еще ограничить шум. Следовательно, реальные ЦАП для выборки с запасом по частоте, как правило, состоят из четырех основных частей: цифрового фильтра для выборки с запасом по частоте, схемы формирования спектра шума (например, сигма-дельта-модулятора), ЦАП с маленьким раз-

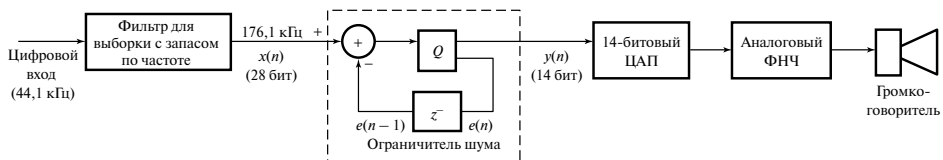


Рис. 2.42. Упрощенная блок-схема воспроизведения аудиосигнала в проигрывателе компакт-дисков с четырехкратной перевыборкой и формированием спектра шума

решением (например, однобитового ЦАП) и простого аналогового фильтра подавления зеркальных частот (см. рис. 2.41).

Фильтр для выборки с запасом по частоте используется для повышения частоты дискретизации и ослабления зеркальных компонентов. Уменьшение длины слова (например, с 16 бит до 1) приводит к появлению большого шума квантования, но этот шум распределяется по намного большей полосе частот, поскольку значительно увеличилась частота дискретизации. Кроме того, схема формирования спектра шума смещает большую часть шума квантования за пределы полосы сигнала в область более высоких частот. Шум за пределами полосы сглаживается с помощью простого фильтра подавления зеркальных частот.

2.9.1. Цифроаналоговое преобразование с запасом по частоте в проигрывателе компакт-дисков

Проиллюстрируем принципы цифроаналогового преобразования с запасом по частоте на примере его реализации в некоторых проигрывателях компакт-дисков (см. рис. 2.42). После декодирования и исправления ошибок цифровые сигналы, которые считываются с компакт-диска, имеют вид 16-битовых слов, передающих звуковую информацию с частотой 44,1 кГц. Если бы цифровые коды непосредственно преобразовывались в аналоговые, то появлялись бы полосы с зеркальными частотами, центрированные в точках, кратных частоте дискретизации 44,1 кГц (см. рис. 2.43, а). И хотя зеркальные частоты не были бы слышны (их частота превышает 20 кГц), они могли бы привести к перегрузке, если бы попали на усилитель и динамик проигрывателя, или вызвать интермодуляционное искажение сигнала. Следовательно, частотные компоненты, превышающие основную частоту, должны подавляться, как минимум, на 50 дБ. Аналоговые фильтры, которые могут обеспечить такой уровень подавления, должны будут соответствовать очень строгой спецификации, а для того, чтобы убедиться, что фильтры двух разных каналов подходят друг другу, нужна будет кропотливая подгонка.

Во избежание таких проблем в проигрывателях компакт-дисков используются фильтры выборки с запасом по частоте. Для реализации соответствующего метода можно умножить частоту дискретизации данных перед цифроаналоговым преобразованием на четыре, получив 176,4 кГц ($4 \times 44,1$ кГц). Частотный спектр сигнала с четырехкратной перевыборкой, отфильтрованного с помощью цифрового фильтра, показан на рис. 2.43, б. Видно, что зеркальные компоненты, превышающие 20 кГц, после выборки с запасом по частоте существенно уменьшаются, в результате чего их легче отфильтро-

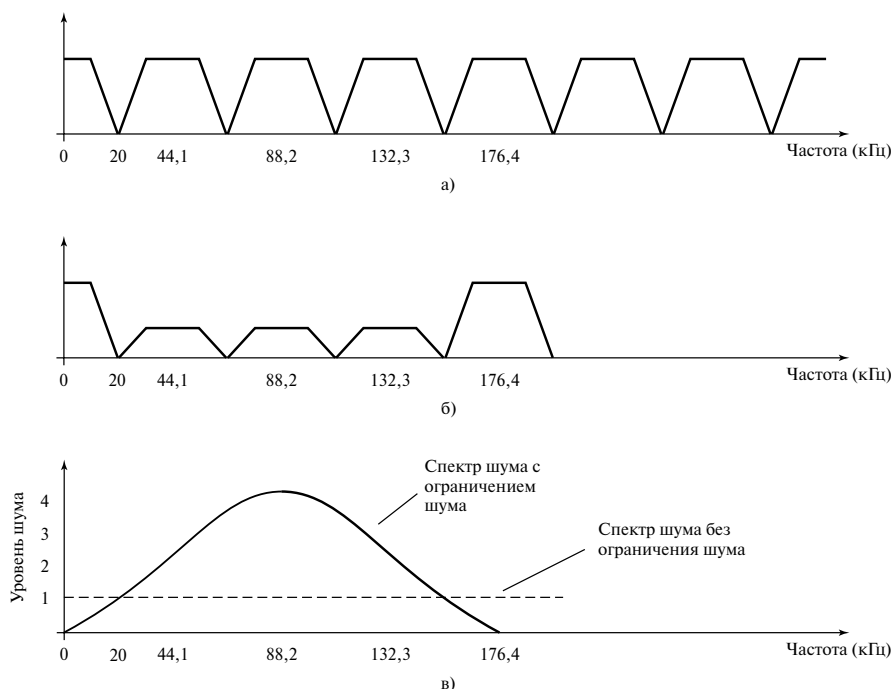


Рис. 2.43. Спектр выборок звукового сигнала с частотой 44,1 кГц (панель а); спектр сигнала с четырехкратной переборкой, на котором показаны значительно уменьшенные зеркальные компоненты (панель б); действие схемы ограничения шума на спектр шума (панель в). Обратите внимание на то, что в полосе звуковых частот (0–20 кГц) уровень шума значительно меньше, чем в области высоких частот, и сравним со спектром шумов, если бы не было ограничения шума (показанного пунктирной линией)

вать. Во временных координатах это выглядит как создание сигнала с намного меньшим шагом. В частотных же координатах зеркальные компоненты сдвигаются в область более высоких частот.

Выход фильтра выборки с запасом по частоте (28-битовые слова) подается на схему формирования спектра шума, а затем квантуется в 14-битовые слова путем округления (см. рис. 2.42). (Длина слова коэффициентов фильтра выборки с запасом по частоте составляет 12 бит, а его вход состоит из 16-битовых слов. После фильтрации выход состоит из 28-битовых слов.) Ошибка квантования возвращается и объединяется с выходом фильтра выборки с запасом по частоте. Для смещения шума квантования в сторону более высоких частот используется схема формирования спектра шума.

Объединенное действие выборки с запасом по частоте, фильтрации и ограничения шума существенно уменьшает зеркальные компоненты и уровень квантования в полосе сигнала. Это позволяет использовать 14-битовый ЦАП и при этом достигать такого отношения сигнал-шум, которое соответствует 16-битовому ЦАП. Можно показать, что фильтр с четырехкратной выборкой с запасом по частоте и ограничитель шума улучшают отношение сигнал-шум на 6 и 7 дБ соответственно.

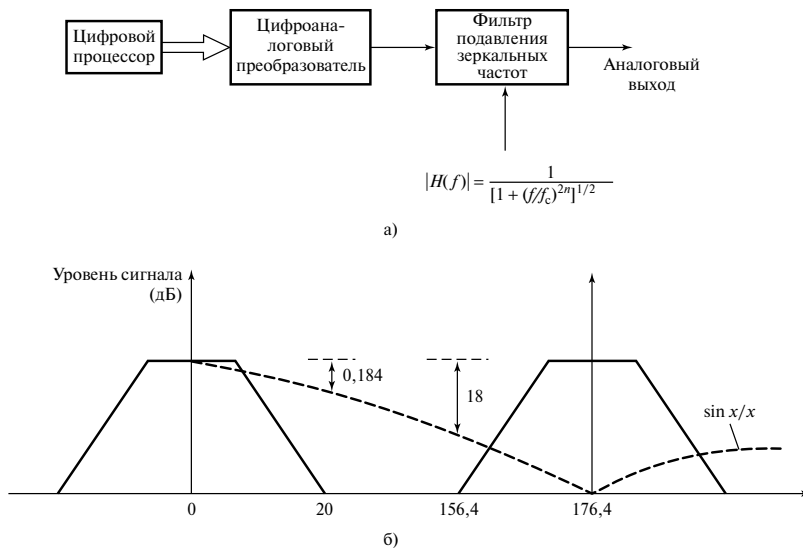


Рис. 2.44. Устройство предварительной обработки данных системы ЦОС реального времени (панель а). Спектр на выходе ЦАП (панель б)

Эффект задержки в ЦАП приводит к искажению по закону $\sin x/x$ в спектре, что способствует дальнейшему уменьшению зеркальных компонентов. Затем для восстановления звукового сигнала используется простой фильтр подавления зеркальных частот.

Пример 2.13

На рис. 2.44, а изображено устройство, которое применяется для восстановления аналогового сигнала после его цифровой обработки в определенной цифровой системе записи и воспроизведения звука реального времени. Основная полоса частот аналогового сигнала расположена между 0 и 20 кГц, а цифроаналоговый преобразователь обновляется с частотой 176,4 кГц. Зеркальные частоты следует подавить как минимум на 50 дБ, а важные компоненты сигнала должны измениться не больше, чем на 0,5 дБ. Найдите минимальный порядок и частоту среза фильтра защиты от зеркальных частот, предполагая, что он имеет баттервортовскую характеристику. Укажите все сделанные предположения.

Решение

Если предположить задержку нулевого порядка, то спектр на выходе ЦАП представляет собой произведение спектра сигнала и характеристики $\sin x/x$ (см. рис. 2.44, б). Ослабление сигнала, обусловленное спектром $\sin x/x$, на двух критических частотах 20 и 156,4 кГц (ближайшие к основной полосе частот зеркальные частоты) равно:

$$\text{при } 20 \text{ кГц} : \frac{\sin x}{x} = 0,9789 (\text{при } x = \omega T/2) \rightarrow -0,184 \text{ дБ},$$

$$\text{при } 156,4 \text{ кГц} : \frac{\sin x}{x} = 0,125 \rightarrow -18 \text{ дБ}.$$

Следовательно, неравномерность в полосе пропускания выходного фильтра не должна превышать $0,5 - 0,184 = 0,316$ дБ. В полосе подавления необходимо дополнительное ослабление не меньше, чем на $50 - 18 = 32$ дБ. Следовательно,

$$20 \log [1 + (20/f_c)^{2n}]^{1/2} \leq 0,316 \text{ дБ},$$

$$20 \log [1 + (156,4/f_c)^{2n}]^{1/2} \geq 32 \text{ дБ}.$$

Решая эту систему уравнений относительно n , получаем $n = 2,4 \simeq 3$ (целое), а $f_c = 30,76$ кГц.

2.10. Ограничения обработки сигналов в реальном времени с аналоговыми входными/выходными сигналами

Основные ограничения и ошибки, возникающие при процессах аналого-цифрового и цифроаналогового преобразования в системах ЦОС реального времени, обсуждались выше. Ниже кратко изложены основные ограничения и предлагаются возможные решения.

- Представление данных с помощью конечного числа битов вносит исходную ошибку (ошибку квантования), которая увеличивается при дальнейшей обработке сигнала. Существует два способа устранения этой ошибки: увеличить разрешение АЦП и дискретизовать сигнал с запасом по частоте с последующей цифровой обработкой для улучшения отношения сигнал-шум (подробнее см. главу 9).
- АЦП и ЦАП с большой разрешающей способностью обычно низкоскоростные (за исключением очень дорогих преобразователей). Как правило, для преобразования аналогового элемента АЦП требуется несколько микросекунд, а ЦАП тратит на решение значительную часть микросекунды. Такие задержки ограничивают максимальную возможную частоту дискретизации. Фактически при современных технологиях АЦП и ЦАП являются главным критическим параметром большинства приложений ЦОС реального времени.
- АЦП и ЦАП подвержены ряду других ошибок, включая температурные эффекты и нелинейности. Таким образом, у хорошей системы ЦОС реального времени с аналоговым входом должны быть качественные элементы аналогового ввода-вывода.
- Выход устройства выборки-хранения — это широкая полоса частот (“благодаря” зеркальным частотам), которая увеличивает шум на входе АЦП.
- Из-за наличия энергии сигнала за пределами представляющей интерес полосы частот всегда существует искажение от наложения. Чтобы снизить его до приемлемого уровня, перед дискретизацией необходимо ограничить полосу частот сигнала и по возможности провести выборку с запасом по частоте.

- При использовании ЦАП нулевого порядка возникает эффект $\sin x/x$, который постепенно уменьшает высокочастотные компоненты сигнала. Этот эффект можно компенсировать, применив цифровой фильтр с характеристикой $x/\sin x$.
- Фильтры защиты от наложения спектров также вносят ошибки. Как правило, это амплитудные и фазовые искажения. Амплитудные характеристики этих фильтров в представляющей интерес полосе не плоские. Аналоговые фильтры с достаточно хорошей амплитудной характеристикой неизменно обладают плохой фазовой характеристикой, что означает, что гармонические отношения между компонентами сигнала искажаются. В многоканальных системах проблема усложняется тем, что искажения от генераторов аналоговых сигналов для разных каналов различаются, и может возникнуть необходимость в их компенсации.
- К ошибкам дискретизации с задержкой относятся время детектирования, апертурная неопределенность, спад за время преобразования и прохождение сигнала в режиме задержки.
- В современных системах ЦОС, особенно в цифровых системах записи и воспроизведения звука, таких как проигрыватели компакт-дисков, наблюдается тенденция к использованию однобитовых АЦП и ЦАП. Эти более новые устройства используют преимущества обработки при нескольких скоростях (подробнее см. главу 9).

2.11. Примеры применения

Применение аналогового интерфейса ввода-вывода широко распространено в большинстве систем ЦОС реального времени. В новейших приложениях используются теорема о дискретном представлении и факторы, связанные с дискретизацией, а именно: частота дискретизации, повторение спектра дискретных данных в точках, кратных частоте дискретизации, конечная ширина полосы реального сигнала (например, настоящие компоненты сигнала ограничены, скажем, частотами 0 и $F_s/2$), действие дискретизации на шум квантования и т.д. Это привело к развитию недорогих аналого-цифровых и цифроаналоговых преобразователей с высокой разрешающей способностью (см. разделы 2.5 и 2.9). В основе этих приложений лежат методы обработки при нескольких скоростях, описанные в главе 9, и поэтому мы отложим их обсуждение до этой главы.

Для улучшения характеристик приемника в системах связи используются методы полосовой дискретизации. Вопросы, связанные с этими приложениями, требуют хорошего понимания методов обработки при нескольких скоростях (эту тему мы также отложим до главы 9).

2.12. Резюме

В общих чертах, система ЦОС реального времени — это система, которая состоит из блока аналого-цифрового преобразования, цифрового процессора и блока цифроаналогового преобразования. Время, необходимое для преобразования в таких системах аналогового сигнала в цифровой, ограничивает максимальную ширину полосы сигнала, с которым может работать система, а устройства, которые используются в процессах преобразования, могут приводить к значительным ошибкам или ухудшению сигнала.

Большую часть этих ошибок можно минимизировать, тщательно подобрав устройства (АЦП, ЦАП и т.д.) и параметры системы (частоты дискретизации и т.п.). Например, наложение можно уменьшить с помощью дискретизации с достаточно высокой частотой и подходящих фильтров, ограничивающих полосу частот сигнала.

Задачи

Дискретизация и контроль за наложением

- 2.1. Что вы понимаете под следующими выражениями?
- Частота Найквиста (Котельникова).
 - Скорость Найквиста.
 - Частота дискретизации.
 - Скорость дискретизации.
- 2.2. На рис. 2.45 изображен спектр сигнала. Найдите минимальную частоту дискретизации, позволяющую избежать наложения. Предположите, что сигнал дискретизируется с частотой 16 кГц, и изобразите спектр дискретного сигнала в диапазоне ± 16 кГц. Укажите на своем рисунке частоты, представляющие практический интерес, в том числе и частоту наложения.

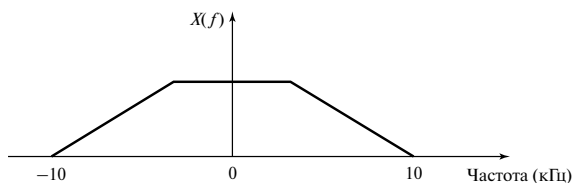


Рис. 2.45

- 2.3. Объясните, почему одной только теоремы о дискретном представлении недостаточно для определения настоящей частоты дискретизации, которая используется в практической системе ЦОС.
- 2.4. Объясните роль фильтров защиты от наложения спектров и фильтров подавления зеркальных частот в системе ЦОС реального времени. Почему в системах ЦОС требования к этим двум фильтрам часто бывают одинаковыми?
- 2.5. К аналоговому входному блоку определенной работающей в реальном времени системы ЦОС выдвигаются такие требования:
- | | |
|---|----------------|
| полоса частот, представляющая практический интерес, | 0–4 кГц, |
| максимально допустимая неравномерность в полосе пропускания | $\leq 0,5$ дБ, |
| затухание в полосе подавления | ≥ 50 дБ. |

Найдите минимальный порядок фильтра защиты от наложения спектров с характеристикой типа Баттерворта и подходящую частоту дискретизации для удовлетворения этим требованиям.

- 2.6. Аналоговый вход системы ЦОС реального времени оцифровывается с помощью 16-битового АЦП в биполярном режиме. Удвоенная амплитуда входного сигнала лежит в диапазоне ± 10 В, полоса частот сигнала — от 0 до 10 кГц. Оцените
- а) минимальное затухание $A_{\min}$ в полосе подавления фильтра защиты от наложения спектров;
 - б) минимальную частоту дискретизации F_s , поддерживающую искажение от наложения в полосе пропускания как раз ниже уровня шума квантования (предположите, что для защиты от наложения спектров используется фильтр Баттерворта шестого порядка).

- 2.7. Полоса частот аналогового сигнала с однородной спектральной плотностью мощности ограничена фильтром с такой амплитудной характеристикой

$$|H(f)| = \frac{1}{[1 + (f/f_c)^6]^{1/2}},$$

где $f_c = 3,4$ кГц. Сигнал оцифровывается с помощью линейного 8-битового АЦП. Найдите такую минимальную частоту дискретизации, чтобы максимальное искажение от наложения не превышало уровень ошибки квантования в полосе пропускания.

- 2.8. Полоса частот аналогового входного сигнала системы ЦОС реального времени перед оцифровкой была ограничена до 30 Гц с помощью аналогового фильтра Баттерворта третьего порядка. При условии, что обусловленное дискретизацией искажение от наложения составляет меньше 1% от уровня сигнала в полосе пропускания, найдите минимальную частоту дискретизации F_s , необходимую для этой системы.

Какова будет средняя ошибка, обусловленная апертурными эффектами при частоте 30 Гц, если сигналу после оцифровки и обработки был возвращен аналоговый вид? Предположите, что входной сигнал оцифрован с помощью идеального устройства дискретизации и АЦП, но восстановлен с помощью ЦАП с задержкой нулевого порядка. Общей частотой дискретизации на входе и на выходе можно считать частоту 256 Гц.

- 2.9. На рис. 2.46 изображено устройство предварительной обработки данных системы ЦОС. Предположим, что на вход подается широкополосный сигнал.

1. Изобразите спектр сигнала до (точка А) и после дискретизации (точка В) в диапазоне $\pm F_s/2$.
2. Найдите уровень сигнала и искажения от наложения при 15 кГц и частоте Найквиста (т.е. 30 кГц).
3. Найдите минимальную частоту дискретизации $F_s(\min)$, которая при частоте 15 кГц даст отношение сигнала к уровню искажения от наложения 10:1. Укажите все сделанные предположения.

- 2.10. На рис. 2.47 изображена система ЦОС реального времени. Предположив, что полоса частот, представляющая практический интерес, находится в диапазоне 0–100 Гц, и что используется 16-битовый *биполярный* АЦП, оцените:

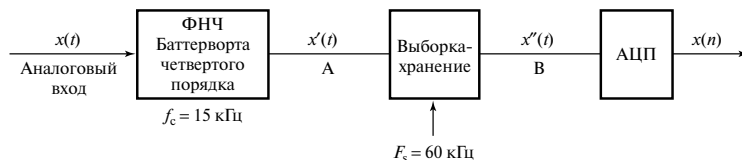


Рис. 2.46

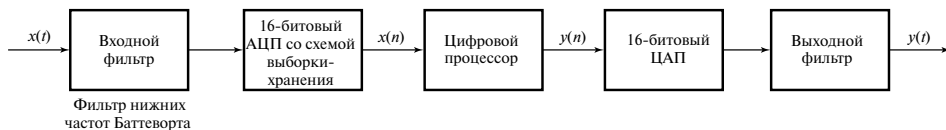


Рис. 2.47. Система цифровой обработки сигналов реального времени

- минимальное затухание $A_{\min}$ в полосе подавления фильтра защиты от наложения спектров;
- минимальную частоту дискретизации F_s ;
- уровень искажения от наложения относительно уровня сигнала в полосе пропускания для оцененных $A_{\min}$ и F_s .

Изобразите спектр сигнала на выходе аналогового фильтра (снабдив рисунок необходимыми пометками), предположив, что на вход подается широкополосный сигнал, и спектр сигнала после дискретизации.

- Коротко расскажите о трех основных факторах, определяющих уровень искажения от наложения в реальных системах ЦОС. Укажите значения каждого из них для контроля за наложением спектров.
- Полоса частот аналогового сигнала с однородной спектральной плотностью мощности ограничена с помощью фильтра защиты от наложения спектров со следующей амплитудно-частотной характеристикой:

$$\frac{1}{\sqrt{1 + \left(\frac{f}{f_c}\right)^8}},$$

где $f_c = 40$ Гц. Сигнал оцифрован с помощью линейного 12-битового биполярного АЦП. Найдите:

- минимальную частоту дискретизации, при которой максимальное искажение от наложения в полосе пропускания не превышает уровень шума квантования;
 - максимальный уровень сигнала в полосе пропускания (в дБ) относительно минимального уровня шума квантования АЦП.
- Укажите все сделанные предположения.
- Запишите уравнение, выражающее теорему о полосовой дискретизации. Объясните, почему эта теорема представляет интерес с точки зрения цифровых систем связи.

4. Из уравнения, записанного в п. 3, выведите выражение для теоретической минимальной частоты дискретизации полосового сигнала. Предположите, что отношение верхнего края полосы частот к ширине полосы сигнала равно целому числу. Прокомментируйте, почему теоретическую минимальную частоту дискретизации нельзя использовать на практике.

Полосовая дискретизация с недостаточной выборкой

- 2.12. На рис. 2.48, *а* изображено устройство предварительной обработки данных приемника многоканальной системы связи. Спектр принимаемого сигнала изображен на рис. 2.48, *б* с указанием номеров каналов. Перед оцифровкой сигнала на минимальной возможной частоте для изоляции сигнала в нужном канале применяется полосовой фильтр.

Предположим, что используется идеальный полосовой фильтр со следующей характеристикой:

$$H(f) = 1, \text{ если } 10 \text{ кГц} \leq f \leq 20 \text{ кГц},$$

0 — в иных случаях.

- 1, *а*. Найдите минимальную теоретическую частоту дискретизации.
- 1, *б*. Изобразите спектр сигнала до дискретизации (точка А) и после дискретизации (точка В).
2. Повторите пункты *а* и *б* для полосового фильтра, пропускающего канал 2.

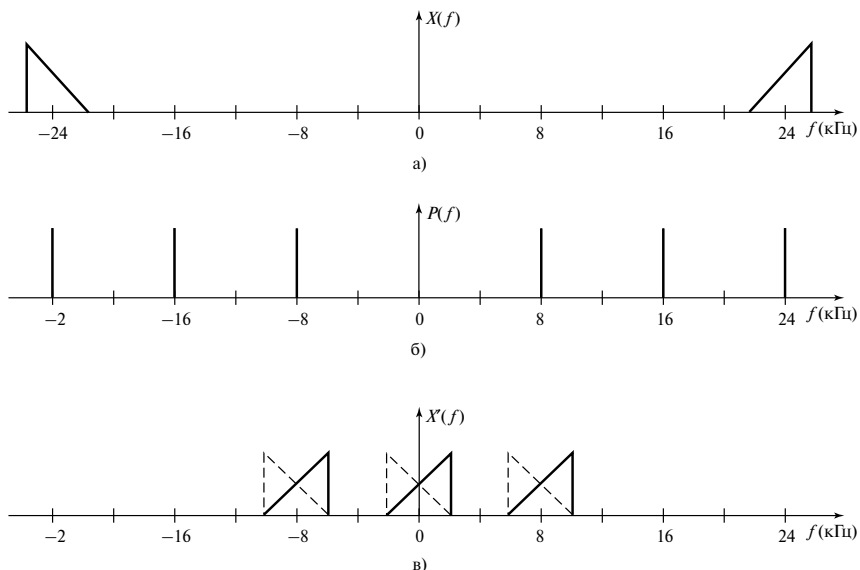


Рис. 2.48. Устройство предварительной обработки данных системы (панель *а*); спектр принимаемого сигнала (панель *б*)

2.13. На рис. 2.49 изображен спектр узкополосного сигнала. Получите и изобразите спектр дискретного сигнала в диапазоне $\pm F_s/2$ для каждого из следующих трех случаев:

- 1) $\frac{f_H}{B} = 3$,
- 2) $\frac{f_H}{B} = 4$,
- 3) $\frac{f_H}{B} = 4,5$.

Пусть ширина полосы сигнала $B = 5$ кГц, и в каждом случае сигнал дискретизируется с частотой $2B$.

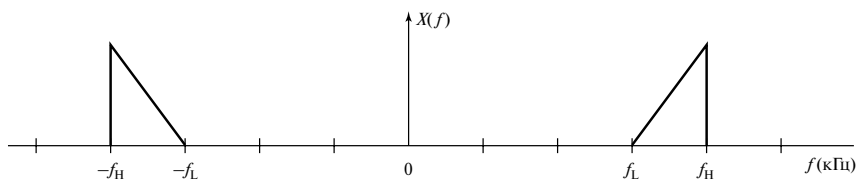


Рис. 2.49

- 2.14. 1. Найдите минимальную теоретическую частоту дискретизации F_s , позволяющую избежать наложения, для полосового сигнала с частотными компонентами в диапазоне $20 \text{ МГц} < f < 30 \text{ МГц}$. Обоснуйте свой ответ и объясните, почему минимальной теоретической частотой дискретизации нельзя воспользоваться на практике.
2. Пусть полосовой сигнал из п. 1 имеет спектр, изображенный на рис. 2.50. Найдите граничные частоты всех частотных полос сигнала (включая зеркальные компоненты) после дискретизации в интервале $\pm 2F_s$. Изобразите спектр дискретного сигнала в этом интервале, отметив на рисунке необходимые элементы.
3. Рассчитайте допустимый диапазон частот дискретизации, позволяющих избежать наложения спектров, если аналоговый полосовой сигнал усиливается защитной полосой в 5 кГц на каждом краю полосы.

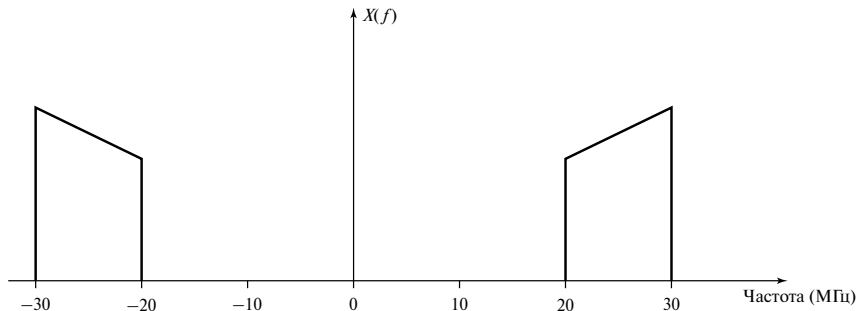


Рис. 2.50

- 2.15. 1. Кратко объясните принцип полосовой дискретизации с недостаточной выборкой. Прокомментируйте преимущества применения этого метода на практике.
2. Цифровой радиоприемник на втором этапе работает на промежуточной частоте 50 кГц.
- А. Найдите минимальную частоту дискретизации F_s , позволяющую избежать наложения спектров в этой системе, если ширина полосы промежуточной частоты сигнала равна 6 кГц.
- Б. Изобразите спектр дискретного сигнала в интервале $\pm F_s$. Объясните, как вы получили спектр дискретного сигнала и прокомментируйте его вид.

Пусть используется метод целочисленной дискретизации и спектр сигнала на втором этапе промежуточной частоты выглядит так, как показано на рис. 2.51.

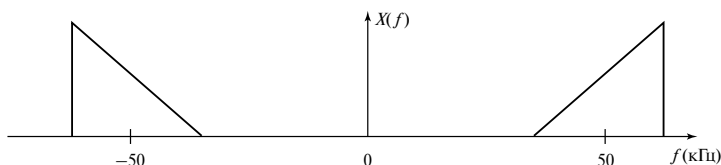


Рис. 2.51

- 2.16. 1. На рис. 2.52, а изображен спектр сигнала на втором этапе обработки в цифровом приемнике (преобразование на промежуточную частоту), где промежуточная частота равна 2,976 МГц. С помощью соответствующих схем покажите, что сигнал промежуточной частоты можно дискретизовать без наложения с частотой 128 кГц.
2. С помощью соответствующих схем покажите, что если промежуточная частота сигнала будет равна 3 МГц, то появится наложение, если сигнал на промежуточной частоте дискретизован с частотой 128 МГц.

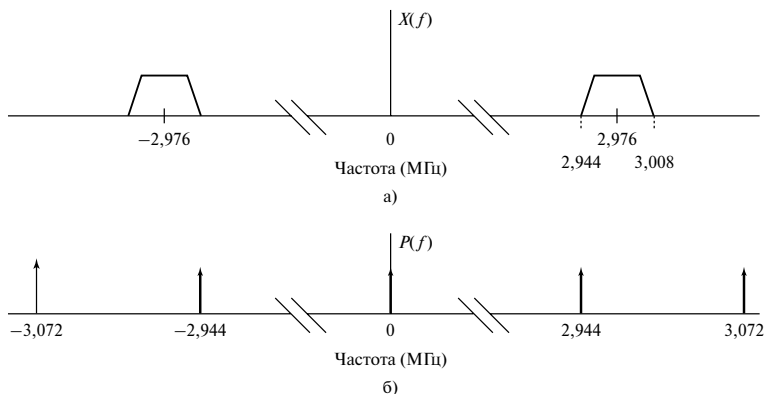


Рис. 2.52. Спектр сигнала на втором промежуточном этапе (панель а); спектр выборочной функции (панель б)

- 2.17. 1. Запишите уравнение, выражающее теорему о полосовой дискретизации. Объясните, почему теорема о полосовой дискретизации представляет интерес для цифровой связи.
2. Из записанного уравнения выведите выражение для теоретической минимальной частоты дискретизации полосового сигнала. Предположите, что отношение верхнего края полосы частот к ширине полосы сигнала равно целому числу. Объясните, почему теоретическая минимальная частота дискретизации может оказаться непригодной на практике.

Шум квантования при аналого-цифровом преобразовании

- 2.18. В системе ЦОС реального времени используется 16-битовый АЦП в биполярном режиме с входным диапазоном ± 5 В. Какова максимальная ошибка квантования? Рассчитайте теоретическое максимальное отношение сигнал-шум квантования (в децибелах) для этой системы.
- 2.19. Синусоидальный сигнал с удвоенной амплитудой 5 В оцифровывается с помощью 16-битового АЦП. Предположив, что квантование линейно, найдите:
- а) величину шага квантования;
 - б) среднеквадратическое отношение сигнал-шум квантования.

Укажите все сделанные предположения.

- 2.20. Аналоговый вход системы ЦОС оцифровывается с частотой 100 кГц при однородном квантовании. Предполагая, что на вход подается синусоидальный сигнал с удвоенной амплитудой ± 5 В, найдите минимальное число битов АЦП, позволяющее достигать отношения сигнал-шум квантования не менее 90 дБ. Перечислите все сделанные предположения.
- 2.21. Покажите, что отношение сигнал-шум квантования линейного АЦП задается как

$$SNR(q) = 6,02B + 4,77 - 20 \log(A/\sigma_x)(\text{дБ}),$$

где B — количество битов АЦП, $\pm A$ — входной диапазон АЦП, а σ_x — среднеквадратическое значение входного сигнала. Найдите отношение сигнал-шум квантования, если разрешение АЦП равно 16 бит, а вход имеет вид:

- а) синусоидального сигнала;
- б) сигнала со среднеквадратическим значением $A/4$.

Укажите все сделанные предположения.

- 2.22. Среднеквадратическое значение аналогового входного сигнала B -битового АЦП равно σ_x (В). Входной диапазон АЦП подходит для значений в пределах $\pm 3\sigma_x$ (В). Найдите выражение для отношения сигнал-шум квантования (в децибелах) преобразователя. Укажите все сделанные предположения.

Выборка с запасом по частоте при аналого-цифровом преобразовании — наложение спектров и контроль за уровнем шума квантования

- 2.23. 1. С помощью подходящих схем объясните принцип метода выборки с запасом по частоте и его использование для увеличения эффективной разрешающей способности аналого-цифрового преобразователя, работающего на частоте Найквиста.
2. В цифровой системе записи и воспроизведения звука для оцифровки аналогового входного сигнала с частотными компонентами в диапазоне 0–4 кГц используется метод выборки с запасом по частоте и 8-битовый биполярный преобразователь, работающий на частоте Найквиста. Оцените эффективную разрешающую способность (в битах) преобразователя, если частота дискретизации равна 40 МГц. Покажите, как вы получили этот ответ. Прокомментируйте практические проблемы, возникающие в связи с этим методом.
- 2.24. 1. Универсальная многоканальная (до 64 каналов) система сбора данных должна собирать медицинские данные. Каждый аналоговый канал должен задаваться пользователем отдельно, чтобы граничная частота его полосы пропускания была между 0,5 и 200 Гц, а выбираемая частота дискретизации — в диапазоне от 1 Гц до 2 кГц. Максимально допустимая неравномерность в полосе пропускания равна 0,5 дБ, а зеркальные компоненты должны быть как минимум на 40 дБ ниже, чем компонент сигнала. Объясните стратегию, выбранную для удовлетворения вышеизложенным требованиям. Ответ должен включать следующие пункты:
- рассуждения о вопросах, специфичных для данного приложения;
 - указания, как использовать в этом приложении методы дискретизации с запасом по частоте, чтобы наиболее эффективно и экономно удовлетворить данные условия (применяйте расчет цены/числа компонентов).
2. Пусть для всех каналов системы из п.1 используются идентичные фильтры защиты от наложения спектров, каждый из которых имеет следующую характеристику типа Баттерворта:

$$A(f) = \frac{1}{\sqrt{1 + \left(\frac{f}{f_c}\right)^6}},$$

где $f_c = 3$ дБ — частота среза фильтра.

3. Изобразите спектры данных до и после дискретизации и определите:
- частоту среза f_c ;
 - подходящую общую частоту дискретизации F_s .
- Ответ прокомментируйте.
- 2.25. Система записи и воспроизведения звука работает с сигналами, основная полоса частот которых занимает диапазон 0–20 кГц. Найдите коэффициент перевыборки и минимальную частоту дискретизации, которая необходима для достижения с 8-битовым преобразователем того качества, которого можно было бы достичь с 16-битовым преобразователем.

- 2.26. 1. Напишите краткое резюме по каждому из перечисленных ниже методов для однобитового АЦП:
- а) выборка с запасом по частоте;
 - б) ограничение спектра шума.
2. Почему в высококачественных системах ЦОС предпочтительнее однобитовые АЦП, а не обычные АЦП с достаточной аппроксимацией?
3. Для оцифровки аналогового сигнала в системе цифровой обработки сигналов с аналоговым звуковым входом в диапазоне 0–20 кГц применяется метод дискретизации с запасом по частоте и сигма-дельта-модулятор первого порядка, изображенный на рис. 2.53. Предположив, что частота дискретизации равна 3,072 МГц, найдите характеристику фильтра ограничения шума при 20 кГц. Оцените эффективную разрешающую способность (в битах) устройства оцифровки.



Рис. 2.53. Сигма-дельта-модулятор первого порядка

- 2.27. В системе цифровой обработки сигналов с аналоговым звуковым входом в диапазоне 0–20 кГц для преобразования аналогового сигнала в цифровой поток битов с частотой 6,144 МГц применяется метод выборки с запасом по частоте и сигма-дельта-модулятор второго порядка. На рис. 2.54 изображена модель сигма-дельта-модулятора на z -плоскости.

Определите общее улучшение отношения сигнал-шум квантования, которое становится возможным благодаря дискретизации с запасом по частоте и ограничению шума, и оцените эффективную разрешающую способность (в битах) устройства оцифровки.

- 2.28. В системе цифровой обработки сигналов с аналоговым звуковым входом в диапазоне 0–20 кГц для преобразования аналогового сигнала в цифровой поток битов с частотой 6,144 МГц применяется метод выборки с запасом по частоте и сигма-дельта-модулятор первого порядка. На рис. 2.55 изображена модель сигма-дельта-модулятора на z -плоскости.

1. Объясните, как цифровой поток битов можно преобразовать в цифровой многобитовый поток с частотой 92 кГц.
2. Найдите общее улучшение отношения сигнал-шум квантования, которое становится возможным благодаря дискретизации с запасом по частоте и ограничению шума, и оцените эффективную разрешающую способность (в битах) устройства оцифровки.

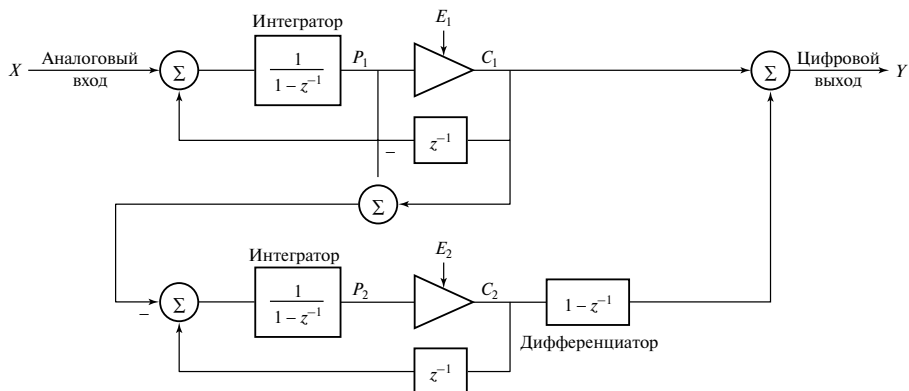


Рис. 2.54. Результат преобразования $Y(z)$ на выходе сигма-дельта-модулятора второго порядка задается как $Y(z) = X(z) + E_2(z)(1 - z^{-1})^2$

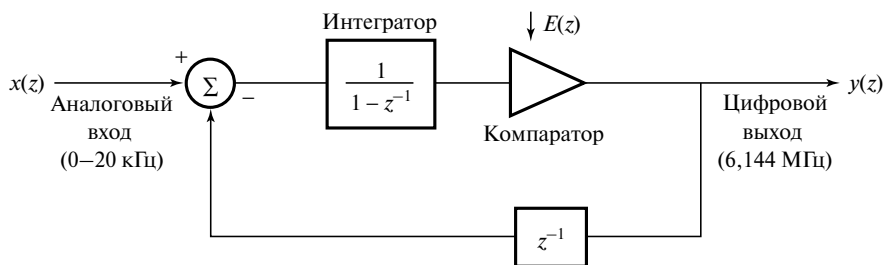


Рис. 2.55. Модель сигма-дельта-модулятора на z -плоскости

Цифроаналоговые преобразования и эффект $\sin x/x$

- 2.29. На рис. 2.56, а показана блок-схема системы ЦОС реального времени со аналоговым выходом, а на рис. 2.56, б изображена основная полоса частот спектра сигнала, который подается на ЦАП. Изобразите спектр сигнала на выходе ЦАП в интервале от 0 до $2F_s$, где F_s — частота дискретизации. Найдите амплитуды компонентов сигнала, учитывая, что частота дискретизации равна 15 кГц.
- 2.30. В системе ЦОС реального времени используется 16-битовый процессор, 12-битовый АЦП с временем преобразования 15 мкс и 12-битовый ЦАП со временем установки сигнала 500 нс. Необходимая операция ЦОС — это сверточное суммирование, которое задается как

$$y(n) = \sum_{k=0}^{N-1} h(k)x(n-k),$$

где переменные имеют обычные значения, а вычисления должны выполняться между элементами выборки. Оцените пропускную способность системы в реальном времени, указав все сделанные предположения.

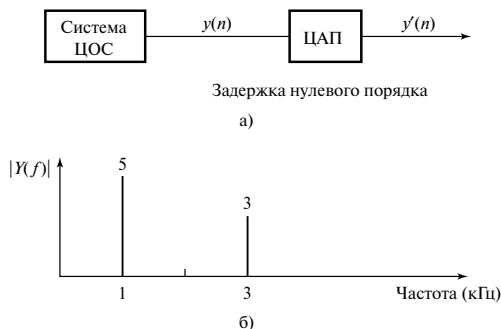


Рис. 2.56. Система ЦОС реального времени с аналоговым выходом (панель а); спектр сигнала, который подается на ЦАП (панель б)

- 2.31. Выход цифроаналогового преобразователя задается как реакция на цифровую последовательность:

$$y(t) = \sum_n y(n)h(t - nT),$$

где $h(t)$ — это импульсная характеристика ЦАП, а $1/T$ — скорость подачи данных на ЦАП. Пусть ЦАП имеет задержку нулевого порядка, а $h(t)$ — квадрат импульса длительностью $T(s)$.

Изобразите выход ЦАП как реакцию на входную последовательность $y(n)$, показанную на рис. 2.57. Покажите, что действие спектрального ограничения ЦАП на спектр сигнала можно компенсировать с помощью цифрового фильтра, спектр которого имеет вид

$$|H(\omega)| = \frac{\omega T}{2 \sin(\omega T/2)}.$$

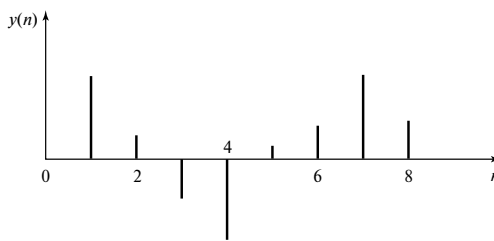


Рис. 2.57

- 2.32. Критически оцените основные ограничения и ошибки, возникающие в процессе аналогового/цифрового преобразования в системах цифровой обработки сигналов реального времени, и подумайте о том, как можно уменьшить каждую из этих ошибок и снизить влияние ограничивающих факторов.

Выборка с запасом по частоте в цифроаналоговом преобразовании: контроль за зеркальными частотами и шумом квантования

- 2.33. На рис. 2.58 изображено устройство, которое используется для восстановления аналогового сигнала после того, как он был обработан определенной цифровой системой записи и воспроизведения звука, работающей в реальном времени. Основная полоса частот аналогового сигнала простирается от 0 до 24 кГц, а частота дискретизации равна 192 кГц.

Зеркальные частоты следует приглушать как минимум на 50 дБ, чтобы при этом компоненты звукового сигнала не подавлялись больше, чем на 0,5 дБ. Воспользовавшись соответствующими изображениями спектра сигнала, найдите минимальные значения порядка и частоты среза фильтра подавления зеркальных частот, предположив, что он имеет характеристику типа Баттерворта. Перечислите все сделанные предположения.

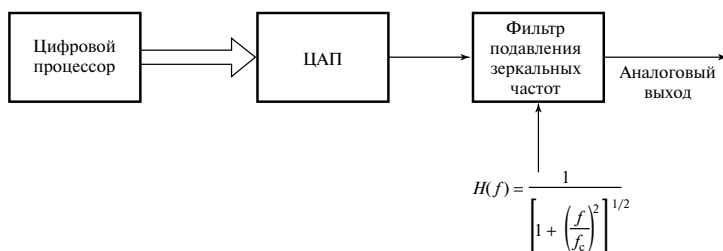


Рис. 2.58. Выходной каскад системы ЦОС реального времени

- 2.34. На рис. 2.59 изображено устройство, которое используется для восстановления аналогового сигнала после обработки цифровой системой записи и воспроизведения звука, работающей в реальном времени. Основная полоса частот аналогового сигнала простирается от 0 до 20 кГц, а частота дискретизации равна 176,4 кГц.

Ограничитель шума описывается следующими уравнениями:

$$y'(n) = x(n) - e(n-1),$$

$$e(n) = y(n) - y'(n).$$

1. Выведите выражение для передаточной функции, которая действует на шум квантования, а затем изобразите спектр шума квантования после ограничения шума.
2. Найдите улучшение отношения сигнал-шум квантования, возможность которого дает выборка с запасом по частоте и ограничение шума, а затем оцените эффективную разрешающую способность ЦАП (в битах). Предположите, что осуществляется четырехкратная выборка с запасом по частоте.

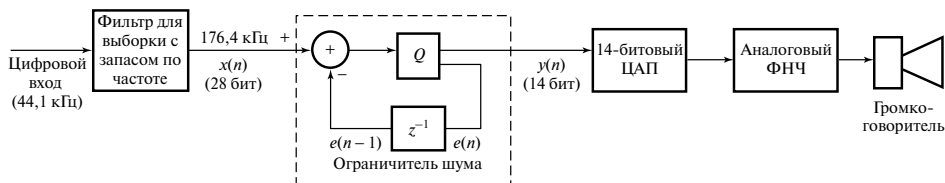


Рис. 2.59. Упрощенная блок-схема воспроизведения звукового сигнала проигрывателем компакт-дисков с четырехкратной выборкой с запасом по частоте и ограничением формы шума

- 2.35. Системе ЦОС предшествует устройство дискретизации с задержкой с апертурным временем 10 нс и временем детектирования 1 μ с, за которым идет 8-битовый АЦП. Найдите максимальное время преобразования АЦП, которое поддерживает частоту дискретизации 100 кГц.

Задачи для решения с помощью MATLAB

- 2.36. На рис. 2.59 изображено устройство, которое используется для восстановления аналогового сигнала после обработки цифровой системой записи и воспроизведения звука реального времени. Основная полоса частот аналогового сигнала расположена от 0 до 20 кГц, а частота дискретизации равна 176,4 кГц.
1. С помощью MATLAB рассчитайте и постройте график спектра удобного фильтра четырехкратной выборки с запасом по частоте с КИХ. Укажите коэффициенты этого фильтра.
 2. Представьте коэффициенты фильтра в виде 12-битовых чисел с фиксированной запятой.
 3. Сгенерируйте звуковой сигнал (16-битовый при частоте 44,1 кГц).
 4. Создайте в MATLAB модель цифроаналогового процесса, пользуясь арифметикой с фиксированной запятой, и постройте график сигнала на выходе каждого блока в частотных и временных координатах.
- 2.37. На рис. 2.60, а изображен спектр сигнала в системе связи с шириной полосы B и несущей частотой f_c . Аналоговый сигнал проходит через фильтр защиты от наложения спектров и дискретизируется с частотой F_s . Искомый спектр дискретного сигнала изображен на рис. 2.60, б.
1. Предоставьте спецификацию подходящего фильтра защиты от наложения спектров и обоснуйте свое решение.
 2. Назовите подходящие частоты дискретизации, которые позволят восстановить спектр на рис. 2.60, б без наложения. Изобразите спектр, полученный в каждом случае.
 3. Чтобы частота дискретизации была как можно ниже, выберите наименьшую частоту дискретизации из п. 2 и проверьте спектр повторно дискретизованного сигнала с помощью пакета MATLAB. Приведите в ответе код MATLAB.
- Подсказка.** Дискретизация — это умножение во временных координатах, что математически равнозначно свертке в частотных координатах. Возьмите

частотный спектр выборочной функции и выполните его свертку со спектром на рис. 2.60, а.

4. Предоставьте спецификацию цифрового фильтра для восстановления сигнала и укажите, как добавление защитных полос упрощает этот фильтр.

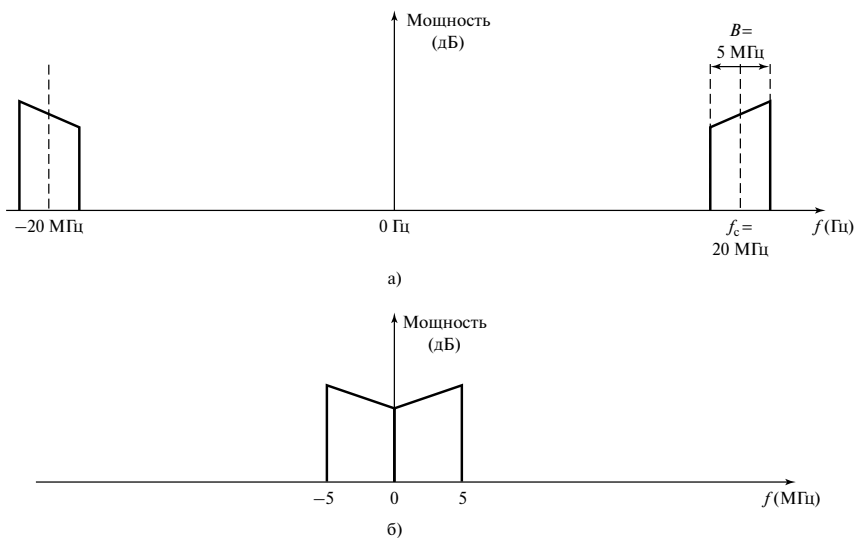


Рис. 2.60

- 2.38. 1. Спроектируйте (на уровне блок-схемы) простой цифровой АМ-приемник, в котором используется метод полосовой дискретизации с недостаточной частотой и квадратурное смешивание для демодуляции принятого сигнала. Пусть промежуточная частота имеет ширину полосы 6 кГц. Ответ должен включать следующие составляющие:
- спецификацию (с обоснованием) подходящей центральной промежуточной частоты в диапазоне 40–60 кГц, оптимальной частоты дискретизации во избежание наложения спектров, соответствующей частоты квадратурного осциллятора и подходящих цифровых фильтров;
 - изображение спектра сигнала на промежуточной частоте до и после дискретизации;
 - точное описание того, как работает цифровой приемник;
 - указание всех сделанных предположений.
2. Постройте и проверьте (в пакете MATLAB) упрощенную модель цифрового АМ-приемника.

Литература

1. CCITT (1989) Possible applications for 16 kbits/sec voice coding. Appendix 3 — Annex 1 to Question 21/XV, 13–22, March.
2. CCITT Recommendation G.726 (1990) 40, 32, 24 and 16kbit/s Adaptive Differential Pulse Code Modulation (ADPCM), ITU Geneva, Switzerland.
3. CCITT Recommendation G.711 (1998) Pulse Code Modulation (PCM) of Voice Frequencies, ITU Geneva, Switzerland.
4. Del Re E. (1978) Bandpass signal filtering and reconstruction through minimum-sampling-rate digital processing. *Alta Frequenza*, **47**(9), September, 395E/675–398E/678.
5. Goedhart D., Van de Plassche R. J. and Stikvoort E. F. (1982) Digital-to-analog conversion in playing compact disc. *Philips Technical Rev.*, **40**(6), 174–179.
6. Vaughan R. G., Scott N. L. and White D. R. (1991) The theory of bandpass sampling. *IEEE Transactions on Signal Processing*, **39**(9), September, 1973–1984.

Дополнительная литература

- Aziz P. M., Sorensen H. V. and Spiegel J. V. D. (1996) An overview of sigma-delta converters. *IEEE Signal Processing Magazine*, January, 61–84.
- Bellamy J. (1982) *Digital Telephony*. New York: John Wiley & Sons.
- Berkhout P. J. and Eggermont L. D. J. (1985) Digital audio systems. *IEEE ASSP Magazine*, October, 45–67.
- Betts J. A. (1978) *Signal Processing, Modulation and Noise*. Unibooks, Hodder and Stoughton.
- Blessier B. A. (1978) Digitization of audio: a comprehensive examination of theory, implementation, and current practice. *J. Audio Eng. Soc.*, **26**(10), 739–771.
- Blessier B., Locanthi B. and Stockham Jr, T. G. (eds) (1982) *Digital Audio*. New York: Audio Engineering Society.
- Candy J. C., Wooley B. A. and Benjamin O. J. (1981) A voice band codec with digital filtering. *IEEE Trans. Communications*, **COM-29**(6), June, 815–830.
- Garret P. H. (1981) *Analog I/O Design*. Reston VA: Reston Publishing Co. Inc.
- ITTCC (1986) Study Group XVIII P Report R26C, Recommendation G7221. 32 kbit/s Adaptive Differential Pulse-Code Modulation (ADPCM).
- Jayant N. S. and Noll P. (1984) *Digital Coding of Waveforms*. Englewood Cliffs NJ: Prentice-Hall.
- Macario R. C. V. (1991) *Signal Coding B: Speech Coding*. C. Xydeas, 82–99.
- Mueller H. R., Schindler H. R. and Vettiger P. (1978) Signal-to-noise analysis of a PCM voice system based on analogue/digital filtering. *IEEE Trans. Communications*, **COM-26**(5), May, 653–659.
- Natvig J. E. (1988) Speech coding in the pan-European digital mobile radio systems. *Speech Communication Magazine*, January.
- Oliver B. M., Pierce J. R. and Shannon C. E. (1948) The philosophy of PCM. *Proc IRE*, November, 1324–1331.
- Oppenheim A. and Schaffer R. W. (1975) *Digital Signal Processing*. Englewood Cliffs NJ: Prentice-Hall.
- Papamichalis P. (1987) *Practical Approaches to Speech Coding*. Englewood Cliffs NJ: Prentice-Hall.
- Rabiner L. R. and Gold B. (1975) *Theory and Applications of Digital Signal Processing*. Englewood Cliffs NJ: Prentice-Hall.
- Sheingold D. H. (ed.) (1986) *Analog-Digital Conversion Handbook*. Englewood Cliffs NJ: Prentice-Hall.
- Steer Jr, R. W. (1989) Antialiasing filters reduce errors in A/D converters. *EDN*, March, 171–186.
- Tiefenthaler C. (1987) Oversampling to increase signal to noise ratio of ADCs. *Electronic Product Design*, March, 59–62.
- Van Doren A. H. (1982) *Data Acquisition Systems*. Reston VA: Reston Publishing Co. Inc.

Дискретные преобразования

3

3.1. Введение	137
3.2. ДПФ и обратное ДПФ	143
3.3. Свойства ДПФ	150
3.4. Вычислительная сложность ДПФ	152
3.5. Алгоритм быстрого преобразования Фурье с децимацией во временной области	153
3.6. Обратное быстрое преобразование Фурье	163
3.7. Реализация БПФ	165
3.8. Другие дискретные преобразования	166
3.9. Применение ДКП: сжатие изображений	183
3.10. Примеры	186
Литература	194
Приложения	195

В данной главе рассказывается о пользе дискретных преобразований и выводится широко распространенное быстрое преобразование Фурье. Кратко описаны дискретное косинус-преобразование, а также преобразование Уолша и Адамара. Рассказано о вейвлетном преобразовании, интерес к которому постоянно растет, и объясняется, как его можно использовать для очистки сигнала от шумов на основе анализа с переменной разрешающей способностью и детектирования сингулярностей.

3.1. Введение

В этой главе рассказывается о преобразовании дискретных данных из временной области в частотную. Зависимость напряжения от времени принято выражать через зависимость амплитуды (или фазы) от частоты, и наоборот. Эти два описания (представления) в различных областях дают дополняющую друг друга информацию об одних и тех же данных. Следовательно, в некоторых приложениях исследование графика зависимости амплитуды сигнала от частоты на предмет изменения напряжения при определенной частоте может дать больше полезной информации, чем наблюдение за формой сигнала напряжения, например, для определения первых признаков износа деталей машины. Для этого к выходным данным применяется быстрое преобразование Фурье (БПФ). Еще один пример — это использование анализатора на основе дискретного преобразования Фурье и осциллографа для проверки выхода модулятора системы связи, где пробные сигналы должны порождать амплитудные составляющие на определенных известных

частотах. Внимание, которое в этих двух примерах спектрального анализа уделяется избранному и четко ограниченному набору частот, говорит о том, что выгоднее применить преобразование, при котором маловажной информацией можно пренебречь, облегчив таким образом последующую интерпретацию данных. Дискретные преобразования, в частности, дискретное косинус-преобразование, используются также для сжатия речевых и видеосигналов, что позволяет передавать их с меньшей шириной полосы. Кроме того, они применяются при обработке изображений (распознавание образов) для сокращения набора признаков. Преобразования полезны и как математическое средство ускорения вычислений в других областях обработки сигналов, таких как нахождение корреляции (используется в гидролокации для поиска объекта на большом расстоянии), или при свертке для определения взаимосвязи между системой и ее входными или выходными данными. При таких расчетах важен переход как из частотной области во временную, так и в обратном направлении. Вообще, эта тема сугубо математическая, но сегодня можно с уверенностью сказать, что в большинстве приложений дискретные преобразования стали стандартной процедурой, так что инженерам-разработчикам едва ли понадобятся глубокие математические познания или знание специальных теорем. Исключением является спектральный анализ сигналов. Здесь каждую задачу нужно рассматривать в своем контексте, при этом важно хорошо разбираться в предмете, чтобы избежать многочисленных подводных камней, связанных с необходимостью сбора достаточного количества правильных дискретных выборок, а также наложения спектров, “эффекта частотного колла” и просачивания спектральных составляющих. Все это будет подробно обсуждаться в главе 11.

Из всех возможных преобразований самыми известными и, пожалуй, самыми важными являются дискретное преобразование Фурье (ДПФ) и алгоритм его быстрого вычисления — быстрое преобразование Фурье (БПФ). Этому есть несколько причин: они позволяют адекватно описывать в частотных координатах все, кроме самых кратковременных (< 1 с) сигналов; усеченные по частоте Фурье-компоненты описывают данные более правдоподобно, чем любые другие степенные ряды. Отдельные компоненты представляют собой синусоиды и не искажаются при передаче через линейные системы, что позволяет использовать их как хорошие пробные сигналы. И, наконец, БПФ можно посчитать очень быстро. Еще одна причина — это то, что анализ Фурье существует с момента опубликования работы Фурье в 1822 году, и с тех пор стал широко известен, заслужил уважение и развился вместе с рядом областей своего применения.

В последнее время значительное внимание уделяется вейвлетному преобразованию, поскольку оно позволяет описывать через амплитуду сигнала стохастические сигналы с переменным во времени частотным спектром. Эта тема также сугубо математическая, но ниже излагаются основные ее концепции, а для иллюстрации разобраны два примера извлечения сигнала из спектра шумов.

Студентов технических специальностей, изучающих электротехнику и электронику, с первых курсов учат анализировать электрическое поведение контуров с помощью преобразования Лапласа. Это объясняется тем, что преобразование Фурье неприменимо ни к пошаговому входу, ни к ненулевым начальным условиям.

Когда студенты переходят к изучению частотных характеристик и устойчивости таких дискретных систем, как фильтры с конечной импульсной характеристикой, воз-

никает необходимость использования z -преобразования (дискретного преобразования Лапласа). Таким образом, главная область применения преобразования Фурье — это быстрые вычисления при обработке сигналов, где применяется БПФ, и спектральный анализ. Однако эти три преобразования взаимосвязаны. Преобразование Лапласа считается более общим, поскольку из него можно вывести два остальных. Так, переменная Лапласа равна $s = \sigma + i\omega$, тогда как переменная преобразования Фурье равна $s = i\omega$, а переменная z -преобразования задается как $z = e^{sT}$, где T — это промежуток времени между двумя выборками. Наконец, переменные преобразования Фурье и z -преобразования связаны соотношением $z = e^{i\omega T}$ (см. главу 4).

3.1.1. Ряд Фурье

Любой периодический сигнал $f(t)$ можно представить в виде суммы бесконечного числа синусоидальных и косинусоидальных членов и одного постоянного члена. Это представление называется *рядом Фурье* и задается следующим образом:

$$f(t) = a_0 + \sum_{n=1}^{\infty} a_n \cos(n\omega T) + \sum_{n=1}^{\infty} b_n \sin(n\omega T), \quad (3.1)$$

где t — независимая переменная, которая часто обозначает время, но может обозначать, например, расстояние или любую другую величину; $f(t)$ часто обозначает функцию зависимости напряжения от времени, но может обозначать и любой сигнал, $\omega = 2\pi/T_p$ называют *циклической частотой* первой (или основной) гармоники, связанной с основной частотой f соотношением $\omega = 2\pi f$, T_p — период повторения сигнала, через

$$a_0 = \frac{1}{T_p} \int_{-T_p/2}^{T_p/2} f(t) dt$$

обозначается постоянная, равная усредненному по времени сигналу $f(t)$ за один период, которая может представлять, например, уровень постоянного напряжения,

$$a_n = \frac{2}{T_p} \int_{-T_p/2}^{T_p/2} f(t) \cos(n\omega t) dt$$

и

$$b_n = \frac{2}{T_p} \int_{-T_p/2}^{T_p/2} f(t) \sin(n\omega t) dt.$$

Частоты $n\omega$ называют n -ми *гармониками* частоты ω . Следовательно, бесконечный ряд (3.1) содержит зависящие от частоты синусоидальные и косинусоидальные члены с различными амплитудами a_n и b_n на положительных частотах гармоник $n\omega$. Этот ряд можно записать компактнее с помощью экспоненциального представления, кроме

того, в таком виде намного упрощается выполнение математических операций. Итак, в экспоненциальной форме ряд Фурье выглядит так:

$$f(t) = \sum_{n=-\infty}^{\infty} d_n e^{in\omega t}, \quad (3.2)$$

где

$$d_n = \frac{1}{T_p} \int_{-T_p/2}^{T_p/2} f(t) e^{-in\omega t} dt \quad (3.3)$$

являются комплексными числами, а $|d_n|$ — величины, измеряемые в вольтах.

При суммировании учитываются и отрицательные значения n , так что половину ряда составляют отрицательные частоты $-n\omega$. Они не имеют физического значения и являются чисто математическим понятием¹, но вследствие этого модули $|d_n|$ комплексных амплитуд d_n численно уменьшены в два раза. Это означает равное распределение амплитуды по соответствующим отрицательной и положительной частотам. Следовательно, правильное значение амплитуды на частоте $n\omega$ можно найти, удвоив рассчитанную величину. Комплексная и тригонометрическая формы связаны следующими соотношениями:

$$|d_n| = (a_n^2 + b_n^2)^{1/2} \quad (3.4)$$

и

$$\phi_n = -\arctg(b_n/a_n), \quad (3.5)$$

где ϕ_n — сдвиг фазы компонента n -й гармоники, который также задается как арктангенс отношения мнимой и действительной частей d_n . Следовательно, каждая гармоника сигнала характеризуется своим фазовым сдвигом и амплитудой.

Пример 3.1

В качестве примера рассмотрим периодический однополюсный импульсный сигнал, показанный на рис. 3.1, *a*. Намеренно сместим начало отсчета относительно центра, а край импульса выберем так, чтобы проиллюстрировать фазовые особенности ряда Фурье. Подстановка соответствующих значений в уравнение (3.3) дает:

¹Физический смысл отрицательной частоты — вращение в направлении, противоположном тому, которое принято за положительное. — *Прим. ред.*

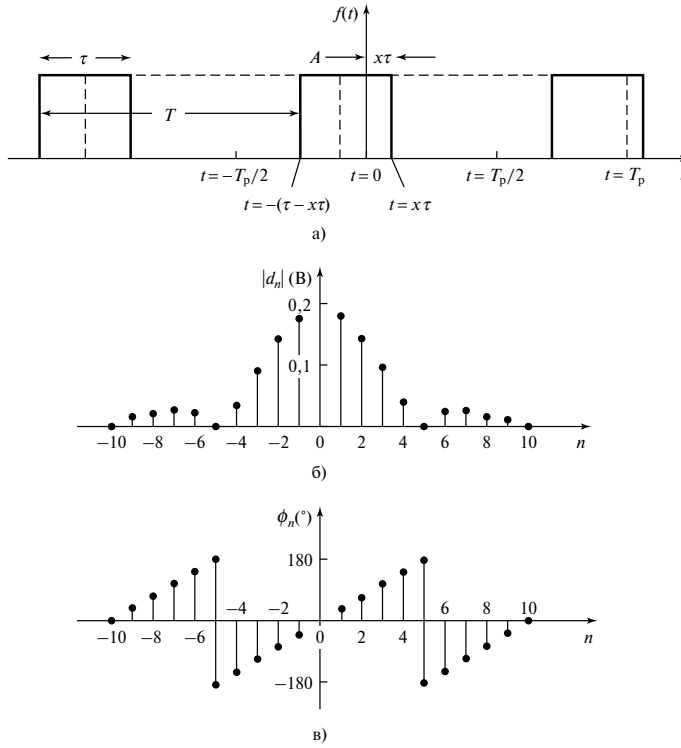


Рис. 3.1. Сигнал $f(t)$ (панель а); амплитудный спектр $|d_n|$ (панель б); фазовый спектр ϕ_n (панель в)

$$\begin{aligned}
 d_n &= \frac{1}{T_p} \int_{-(\tau-x\tau)}^{x\tau} A e^{-in\omega t} dt = \\
 &= \frac{A}{T_p} \left[\frac{e^{-in\omega t}}{-in\omega} \right]_{-(\tau-x\tau)}^{x\tau} = \\
 &= \frac{A}{n\omega T_p} \frac{e^{-in\omega x\tau} - e^{-in\omega(\tau-x\tau)}}{-i} = \\
 &= \frac{A}{n\omega T_p} e^{-in\omega x\tau} \left[\frac{e^{in\omega\tau} - 1}{i} \right] = \\
 &= \frac{2A}{n\omega T_p} e^{-in\omega x\tau} \left[\frac{e^{in\omega\tau/2} - e^{-in\omega\tau/2}}{2i} \right] e^{in\omega\tau/2} =
 \end{aligned} \tag{3.6}$$

$$\begin{aligned}
&= \frac{2A}{n\omega T_p} e^{in\omega(\tau/2-x\tau)} \sin\left(\frac{n\omega\tau}{2}\right) = \\
&= \frac{2A}{n\omega T_p} \frac{n\omega\tau}{2} e^{in\omega(\tau/2-x\tau)} \frac{\sin(n\omega\tau/2)}{n\omega\tau/2} = \\
&= \frac{A\tau}{T_p} \operatorname{sinc}\left(\frac{n\omega\tau}{2}\right) e^{in\omega(0,5-x)\tau},
\end{aligned} \tag{3.7}$$

где

$$\operatorname{sinc}\left(\frac{n\omega\tau}{2}\right) = \frac{\sin(n\omega\tau/2)}{n\omega\tau/2}$$

называется *функцией отсчетов* аргумента $n\omega\tau/2$. Модуль d_n равен

$$|d_n| = \frac{A\tau}{T_p} \left| \operatorname{sinc}\left(\frac{n\omega\tau}{2}\right) \right|$$

и его график изображен на рис. 3.1, б. Величина $n\omega(0,5-x)\tau$ представляет собой фазу ϕ_n (в радианах), связанную с n -й гармоникой. Чтобы построить график зависимости этой фазы от порядка гармоники n , рассмотрим частный случай. Пусть $x = 0$, т.е. поместим начало отсчета в запаздывающий край импульса, и пусть $\tau = T_p/5$, причем

$$\phi_n = \frac{n\omega\tau}{2} = n \frac{2\pi}{T_p} \frac{\tau}{2} = n \frac{2\pi}{T_p} \frac{T_p}{5} \frac{1}{2} = \frac{n}{5}\pi.$$

График ϕ_n показан на рис. 3.1, в, где по договоренности $-180^\circ \leq \phi_n \leq 180^\circ$. От выбора начала отсчета времени зависит вид фазового спектра (зависимости ϕ_n от n). Обычно анализ упрощается, если поместить начало отсчета времени в точку, обладающую некоторой симметрией, например, в центр импульса периодической серии импульсов. В выбранном случае амплитудный спектр (рис. 3.1, б) выглядит, как четная функция ($|d_n| = |d_{-n}|$), тогда как фазовый спектр (рис. 3.1, в) выглядит, как нечетная функция ($\phi_{-n} = -\phi_n$). Фазы ϕ_n, ϕ_{-n} дают относительный сдвиг фаз гармоник относительно друг друга. В момент времени t абсолютный сдвиг фаз равен (из уравнения (3.2)) $\{n\omega(0,5-x)\tau + n\omega t\}$.

3.1.2. Преобразование Фурье

Если сигнал непериодический, то метод разложения в ряд Фурье слегка модифицируется. Для яркой иллюстрации подобного подхода используем одиночный прямоугольный импульс, который можно получить из периодического сигнала на рис. 3.1, а, увеличив период T_p до бесконечности. При увеличении T_p расстояние между гармониками $1/T_p = \omega/2\pi$ уменьшается до $d\omega/2\pi$, в конце концов превращаясь в нуль. Это соответствует переходу от дискретной частотной переменной $n\omega$ к непрерывной

переменной ω , а фазовый и амплитудный спектры становятся непрерывными. Следовательно, $d_n \rightarrow d(\omega)$ при $T_p \rightarrow \infty$. С этими изменениями уравнение (3.3) приобретает вид

$$d(\omega) = \frac{d\omega}{2\pi} \int_{-\infty}^{\infty} f(t)e^{-i\omega t} dt. \quad (3.8)$$

Для удобства нормируем эту формулу, разделив ее на $d\omega/2\pi$, и получим

$$\frac{d(\omega)}{d\omega/2\pi} = F(i\omega) = \int_{-\infty}^{\infty} f(t)e^{-i\omega t} dt. \quad (3.9)$$

Здесь комплексное $F(i\omega)$ называется *интегралом Фурье* или просто *Фурье-образом*. Если положить

$$F(i\omega) = \operatorname{Re}(i\omega) + i\operatorname{Im}(i\omega) = |F(i\omega)|e^{i\phi(\omega)}, \quad (3.10)$$

то

$$|F(i\omega)| = [\operatorname{Re}^2(i\omega) + \operatorname{Im}^2(i\omega)]^{1/2}, \quad (3.11)$$

а измеряться данная величина будет уже в В/Гц, а не в вольтах. Следовательно, $|F(i\omega)|$ — это плотность амплитуды, которую еще называют *спектральной плотностью амплитуды* или *амплитудным спектром*. Соответствующий сдвиг фаз $\phi(\omega)$ равен

$$\phi(\omega) = \arctg[\operatorname{Im}(i\omega)/\operatorname{Re}(i\omega)]. \quad (3.12)$$

Величина $|F(i\omega)|^2$ измеряется в $\text{В}^2\text{Гц}^{-2}$. Поскольку нормированная электрическая мощность, т.е. мощность, рассеянная сопротивлением 1 Ом, измеряется в В^2 , что эквивалентно Дж/с или Дж·Гц (Дж обозначает джоуль, единицу энергии), то $\text{В}^2\text{Гц}^{-2}$ равно $\text{Дж}\cdot\text{Гц} \times \text{Гц}^{-2} = \text{Дж}\cdot\text{Гц}^{-1}$. Следовательно, $|F(i\omega)|^2$ измеряется в единицах энергии на Гц^{-1} , т.е. $|F(i\omega)|^2$ — это спектральная плотность энергии. Площадь под кривой $|F(i\omega)|$ на графике зависимости от f между частотами $f_0 - df$ и $f_0 + df$ выражает среднее напряжение на частоте f_0 , а площадь под соответствующей кривой $|F(i\omega)|^2$ на графике зависимости от f выражает среднюю энергию на частоте f_0 . Кроме того, при спектральном анализе довольно часто строятся графики зависимости спектральной плотности энергии от частоты.

Пример 3.2

Вернемся к рассмотрению одиночного импульса и вычислим его амплитудный спектр, пользуясь уравнением (3.9) и рис. 3.1, а. Искомое выражение приобретает вид

$$F(i\omega) = \int_{-(\tau-x\tau)}^{x\tau} A e^{-i\omega t} dt \quad (3.13)$$

и отличается от уравнения (3.6) только постоянной $1/T_p$. Отсюда следует, что

$$F(i\omega) = A\tau e^{i\omega(1/2-x)\tau} \text{sinc}(\omega\tau/2), \quad (3.14)$$

что в T_p раз больше, чем d_n . Это соответствует тому факту, что $|F(i\omega)|$ измеряется в вольтах, умноженных на единицу времени или в В/Гц. Кстати, результат (3.14) можно получить проще, чем было получено уравнение (3.7), если учесть определенные свойства преобразования Фурье. Итак, импульс с шириной τ единичной высоты с центром в $t = 0$, обозначенный $\text{rect}(t/\tau)$, имеет Фурье-образ $\tau \text{sinc}(\omega\tau/2)$. Поскольку $Af(t)$ преобразуется в $AF[f(t)]$, где через F обозначено преобразование Фурье, то импульс высотой A преобразуется в $A\tau \text{sinc}(\omega\tau/2)$. В контексте рис. 3.1, a импульс смещается влево на $\tau/2 - x\tau$ и действительно является прямоугольным импульсом, $\text{rect}\{[t + (\tau/2 - x\tau)]/\tau\}$. Кроме того, из свойства задержки преобразования Фурье следует, что для импульса, смещенного вправо на t_0 , $f(t - t_0) = se^{-i\omega t_0} F[f(t)]$. Применяя это свойство к $A\tau \text{sinc}(\omega\tau/2)$, получаем искомый вид Фурье-образа:

$$\begin{aligned} F(i\omega) &= e^{+i\omega(\tau/2-x\tau)} A\tau \text{sinc}\left(\frac{\omega\tau}{2}\right) = \\ &= A\tau e^{i\omega(1/2-x)\tau} \text{sinc}\left(\frac{\omega\tau}{2}\right), \end{aligned}$$

т.е. тот же результат, что приведен в формуле (3.14).

Если начало отсчета времени находится в центре импульса, т.е. $x = \frac{1}{2}$, то Фурье-образ этого импульса задается как

$$F(i\omega) = \frac{A\tau \sin(\omega\tau/2)}{\omega\tau/2} = A\tau \text{sinc}(\omega\tau/2) \quad (3.15)$$

и является действительным. Функция $|F(i\omega)|$ непрерывна и ее график для значений $A = 1$ В, $T_p = 10$ с и $\tau = 2$ с изображен на рис. 3.2, a . Этот амплитудный спектр, пропорционально ограниченный функцией отсчетов, всегда порождается прямоугольными импульсами, а также любыми сигналами конечной длительности τ . Последние можно рассматривать как бесконечный сигнал, умноженный на $\text{rect}[(t \pm t_0)/\tau]$, т.е. на единичный импульс. Экспериментально полученные сигналы также подпадают под эту категорию, так как имеют конечную длительность. Используемая функция отсчетов проходит через нуль при каждом $\sin(\omega\tau/2) = 0$, т.е. когда $\omega\tau/2 = m\pi$ ($m \neq 0, m$ — целое). Следовательно, нули амплитуды появляются в точках $f = 1/\tau, 2/\tau, 3/\tau, \dots$. Если $\omega \rightarrow 0$, $\sin(\omega\tau/2) \rightarrow \omega\tau/2$ и $\text{sinc}(\omega\tau/2) = \sin(\omega\tau/2)/(\omega\tau/2) \rightarrow 1$, так что $F(i\omega) = A\tau$ при $\omega = 0$, т.е. при $f = 0$. График спектральной плотности энергии импульса с амплитудой 2 В изображен на рис. 3.2, b , а на рис. 3.2, a для сравнения изображен амплитудный спектр.

Отметим также, что из частотной области во временную можно перейти с помощью обратного преобразования Фурье. В этом случае

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(i\omega) e^{i\omega t} d\omega = \int_{-\infty}^{\infty} F(i\omega) e^{i\omega t} df. \quad (3.16)$$

3.2. ДПФ и обратное ДПФ

На практике Фурье-компоненты сигнала получаются как результат цифровых вычислений, а не вследствие аналоговой обработки. Поскольку аналоговый сигнал состоит из бесконечного числа соприкасающихся точек, описать все их значения практически невозможно. Следовательно, для использования в цифровой системе аналоговые значения следует дискретизовать через равные промежутки времени, а затем выборки преобразовать в цифровой бинарный вид. Это осуществляется с помощью контура выборки-хранения, за которым следует аналого-цифровой преобразователь. Получающееся число выборок в секунду достаточно велико для адекватного описания сигнала. Теоретически необходимая скорость дискретизации называется *частотой Найквиста* и равна $2f_{\max}$, где $f_{\max}$ — частота самого высокочастотного синусоидального компонента сигнала с существенной амплитудой. В этой главе будем считать, что рассматриваются цифровые значения, которые поддаются преобразованию, а такие аспекты, как обработка методом взвешивания, необходимая при спектральном анализе, будут обсуждаться в главе 11. Итак, данные, которые нужно преобразовать, — уже дискретные и, возможно, непериодические. В таком случае преобразованием Фурье воспользоваться нельзя, поскольку оно предназначено для непрерывных данных. Однако существует аналоговое преобразование, которое можно применять к дискретным данным, — дискретное преобразование Фурье (ДПФ).

Предположим, что сигнал дискретизован через равные промежутки времени T , и в результате получена дискретная последовательность $\{x(nT)\} = x(0), x(T), \dots, x[(N-1)T]$ из N выборок, где n — номер выборки, n пробегает значения от $n = 0$ до $n = N-1$. Значения $x(nT)$ будут действительными только тогда, когда они представляют собой значения такого временного ряда, как спектр напряжения. В подобном случае ДПФ последовательности $x(nT)$ можно определить как последовательность комплексных значений $\{X(k\Omega)\} = X(0), X(\Omega), \dots, X[(N-1)\Omega]$ в частотной области, где Ω — частота первой гармоники, которая задается как $\Omega = 2\pi/(N-1)T \simeq 2\pi/NT$ для $N \gg 1$. Следовательно, $X(k\Omega)$ в общем случае имеет действительные и мнимые компоненты, так что для k -й гармоники

$$X(k) = \operatorname{Re}(k) + i\operatorname{Im}(k) \quad (3.17)$$

и

$$|X(k)| = [\operatorname{Re}^2(k) + \operatorname{Im}^2(k)]^{1/2}, \quad (3.18)$$

а сдвиг фазы $X(k)$

$$\phi(k) = \arctg[\operatorname{Im}(k)/\operatorname{Re}(k)], \quad (3.19)$$

где под $X(k)$ понимают $X(k\Omega)$. Следовательно, приведенные уравнения аналогичны уравнениям для преобразования Фурье: сравните уравнения (3.17)–(3.19) с уравнениями (3.10)–(3.12).

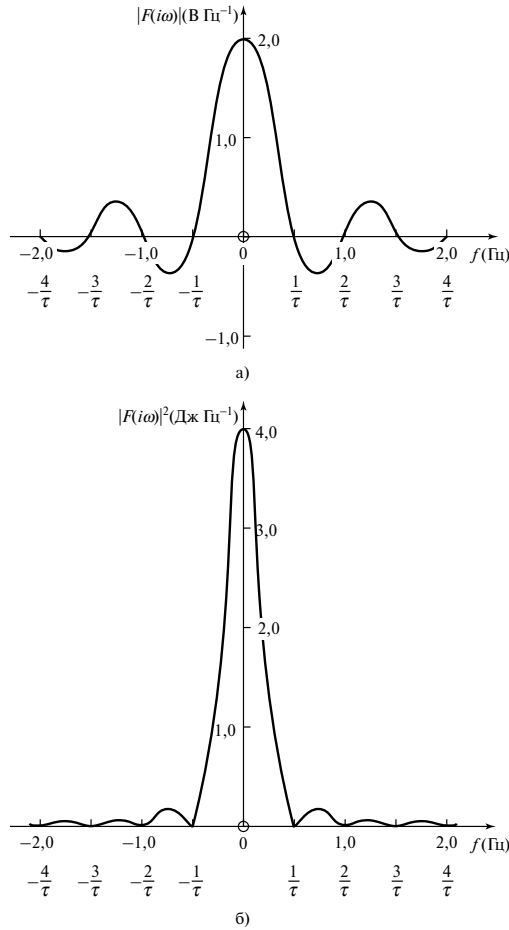


Рис. 3.2. Импульс 2 В: а) амплитудный спектр; б) энергетический спектр

Заметим, что N действительных значений (во временной области) преобразуется в N комплексных значений ДПФ (в частотной области). Значения ДПФ $X(k)$ задаются как

$$X(k) = F_D[x(nT)] = \sum_{n=0}^{N-1} x(nT)e^{-ik\Omega nT}, k = 0, 1, \dots, N-1, \quad (3.20)$$

где через F_D обозначено дискретное преобразование Фурье. В этом уравнении k — номер гармоники компонента преобразования. При замене $x(nT) = f(t)$, $k\Omega = \omega$ и $nT = t$ видно, что это уравнение переходит в преобразование Фурье (уравнение (3.9)) при $f(t) = 0$ для $T < 0$ и $t > (N-1)T$, так что можно ожидать, что эти два преобразования будут иметь общие свойства. В то же время эти преобразования не равнозначны.

Так, сделав указанные замены в уравнении (3.9) и положив $dt = T$, а также заменив интеграл суммой, получим частоты гармоник kf_s , где $f_s = 1/(N-1)T = 2\pi/\Omega$,

$$\sum_{n=0}^{N-1} x(nT)e^{-ik\Omega nT}T = F(i\omega) \quad (3.21)$$

для $0 \leq t \leq (N-1)T$. Затем, сравнив уравнения (3.20) и (3.21), получим

$$F(i\omega) = TX(k), \quad (3.22)$$

откуда видно, что компоненты преобразования Фурье связаны с компонентами ДПФ через интервал дискретизации, и их можно найти, умножив компоненты ДПФ на интервал дискретизации.

Заметим также, что в практических приложениях $N \gg 1$, и часто используется приближение $\Omega = 2\pi/NT$. Поэтому далее во всех примерах в этой главе (даже при $N = 4$) мы также будем принимать это приближение.

Пример 3.3

Здесь будет уместно проиллюстрировать использование уравнения (3.20), рассмотрев простой случай. Найдем ДПФ последовательности $\{1, 0, 0, 1\}$. Стоит заметить, что если непрерывность реальных данных нарушается, то для задания значения в месте разрыва при расчетах берется среднее значение с обеих сторон от разрыва. Это касается первого и последнего значения набора данных, а также всех остальных случаев нарушения непрерывности. Однако, чтобы избежать искажения спектров, вызванного нарушением непрерывности в начале и в конце последовательности данных, при нахождении спектра сигнала необходимо реализовать процедуру вырезания (взвешивания). Подробно эта тема обсуждается в главе 11, а здесь будем просто считать, что последовательность $\{1, 0, 0, 1\}$ уже прошла предварительную обработку. Предположим, что приведенные данные описывают четыре последовательных значения напряжения $x(0) = 1$, $x(T) = 0$, $x(2T) = 0$, $x(3T) = 1$, записанные с временным интервалом T . Таким образом, $N = 4$. Далее нужно найти комплексные значения $X(k)$ для $k = 0$, $k = 1$, $k = 2$ и $k = 3$ (поскольку $N - 1 = 3$). Для $k = 0$ уравнение (3.20) приобретает вид

$$\begin{aligned} X(0) &= \sum_{n=0}^3 x(nT)e^{-i0} = \\ &= \sum_{n=0}^3 x(nT) = x(0) + x(T) + x(2T) + x(3T) = \\ &= 1 + 0 + 0 + 1 = 2, \end{aligned}$$

так что $X(0) = 2$ полностью действительное с модулем 2 и фазой $\phi(0) = 0$. Для $k = 1$ уравнение (3.20) приобретает вид

$$X(1) = \sum_{n=0}^3 x(nT)e^{-i\Omega nT}.$$

Здесь T неизвестно, но оно сокращается, если вспомнить, что $\Omega = 2\pi/NT$. Получим

$$\begin{aligned} X(1) &= \sum_{n=0}^3 x(nT) e^{-i\Omega n 2\pi/N\Omega} = \sum_{n=0}^3 x(nT) e^{-i2\pi n/N} = \\ &= 1 + 0 + 0 + 1e^{-i2\pi 3/4} = 1 + e^{-i3\pi/2} = \\ &= 1 + \cos\left(\frac{3\pi}{2}\right) - i \sin\left(\frac{3\pi}{2}\right) = 1 + i. \end{aligned}$$

Следовательно, $X(1) = 1 + i$ и является комплексным числом с модулем $\sqrt{2}$ и фазой $\phi(\Omega) = \arctg 1 = 45^\circ$. Для $k = 2$ уравнение (3.20) приобретает вид

$$\begin{aligned} X(2) &= \sum_{n=0}^3 x(nT) e^{-i2\Omega nT} = \\ &= \sum_{n=0}^3 x(nT) e^{-i2n 2\pi/N} = \\ &= \sum_{n=0}^3 x(nT) e^{-i4\pi n/N} = \\ &= 1 + 0 + 0 + 1e^{-i4\pi 3/4} = 1 + 0 + 0 + e^{-i3\pi} = 1 - 1 = 0. \end{aligned}$$

Итак, $X(2) = 0$ с модулем 0 и неопределенной фазой $\phi(2)$. Наконец, для $k = 3$ уравнение (3.20) приобретает вид

$$\begin{aligned} X(3) &= \sum_{n=0}^3 x(nT) e^{-i3n 2\pi/N} = \\ &= 1 + 0 + 0 + e^{-i9\pi/2} = 1 - i. \end{aligned}$$

Итак, $X(3) = 1 - i$ с модулем $\sqrt{2}$ и фазой $\phi(3) = -45^\circ$.

Таким образом, было показано, что у временного ряда $\{1, 0, 0, 1\}$ есть ДПФ-образ, который задается комплексной последовательностью $\{2, 1 + i, 0, 1 - i\}$.

ДПФ принято представлять в виде графиков зависимости $|X(k)|$ от $k\Omega$ и $\phi(k)$ от $k\Omega$. Это можно сделать через гармоники Ω или частоты, если Ω известно. Чтобы найти Ω , необходимо знать значение интервала дискретизации T . Если предположить, что использованная выше последовательность данных была дискретизована с частотой 8 кГц, то $T = 1/(8 \times 10^3) = 125$ мкс. Тогда $\Omega = 2\pi/NT = 2\pi/(4 \times 125 \times 10^{-6}) = 12,57 \times 10^3$ рад/с. Следовательно, $2\Omega = 25,14 \times 10^3$ рад/с, а $3\Omega = 37,71 \times 10^3$ рад/с. На рис. 3.3, а показан график зависимости $x(nT)$ от t , на рис. 3.3, б — график зависимости $|X(k)|$ от $k\Omega$, а на рис. 3.3, в — график зависимости $\phi(k)$ от $k\Omega$. Следует заметить, что график “амплитуды” на рис. 3.3, б симметричен относительно компонента второй гармоники,

т.е. относительно гармоники с номером $N/2$, а на рис. 3.3, в фазовый сдвиг является нечетной функцией с центром в этой гармонике. Отметим, что полученные результаты справедливы и в более общем случае.

Если сравнить k -й компонент ДПФ $X(k)$ с $(k+N)$ -м компонентом $X(k+N)$, можно вывести важное свойство ДПФ. Таким образом,

$$X(k) = \sum_{n=0}^{N-1} x(nT)e^{-ik\Omega nT} = \sum_{n=0}^{N-1} x(nT)e^{-ik2\pi n/N}$$

и

$$\begin{aligned} X(k+N) &= \sum_{n=0}^{N-1} x(nT)e^{-ik2\pi n/N} e^{-iN2\pi n/N} = \\ &= \sum_{n=0}^{N-1} x(nT)e^{-ik2\pi n/N} e^{-i2\pi n} = \\ &= \sum_{n=0}^{N-1} x(nT)e^{-ik2\pi n/N} = X(k), \end{aligned}$$

поскольку n — целое, то $e^{-i2\pi n} = 1$.

Из $X(k+N) = X(k)$ следует, что ДПФ-образ повторяется с периодом N . Это *свойство цикличности* ДПФ, т.е. значения компонентов ДПФ повторяются. Если $k = 0$, то $k+N = N$, а $X(0) = X(N)$. В вышеприведенном примере $X(0) = 2$, следовательно, $X(4)$ также равно 2. Это явление демонстрируется на рис. 3.3, б, где амплитуда четвертой гармоники взята в точке 50,28 кГц. Симметрия распределения амплитуды относительно второй гармоники очевидна. Общий вывод: амплитудный спектр N -точечного ДПФ симметричен относительно гармоники $N/2$ при условии, что график содержит нулевую и $(N+1)$ -ю гармоники. Аналогично функция зависимости фазового сдвига, будучи нечетной, проявляет антисимметричные свойства относительно гармоники $N/2$. Если на протяжении t секунд берутся $2f_{\max}$ элементов выборки сигнала в секунду, то $2f_{\max}t = N$, так что $1/t = 2f_{\max}/N$ — это частота первой гармоники. Следовательно, симметрия гармоники $N/2$ будет повторяться на частоте $(N/2)/(2f_{\max}/N) = f_{\max}$ — максимальной частоте, присутствующей в сигнале. Следовательно, все компоненты сигнала полностью описываются амплитудным спектром, график которого построен до $f_{\max}$ или гармоники $N/2$, и дальше график строить не нужно. В таком контексте $f_{\max}$ называют *частотой перегиба*, поскольку спектр между гармониками $N/2$ и N можно перегнуть по оси симметрии на частоте $f_{\max}$, и при этом он точно наложится на низкочастотную половину спектра. Теперь видно, что N действительных значений данных преобразуются в $N/2$ комплексных значений ДПФ. Последние состоят из $N/2$ действительных и $N/2$ мнимых значений, дающих в сумме N значений, которые и получаются из N исходных значений данных. Наконец, значения компонентов преобразования Фурье $F(i\omega)$ последовательности данных $\{1, 0, 0, 1\}$ из примера 3.3 можно найти, умножив компоненты ДПФ на $T = 125$ мкс. Следовательно, $F(0) = 250$ мкВ/Гц, $F(12,57 \text{ кГц}) = (125 + i125)$ мкВ/Гц, $F(25,14 \text{ кГц}) = 0$ В/Гц, $F(37,71 \text{ кГц}) = (125 - i125)$ мкВ/Гц.

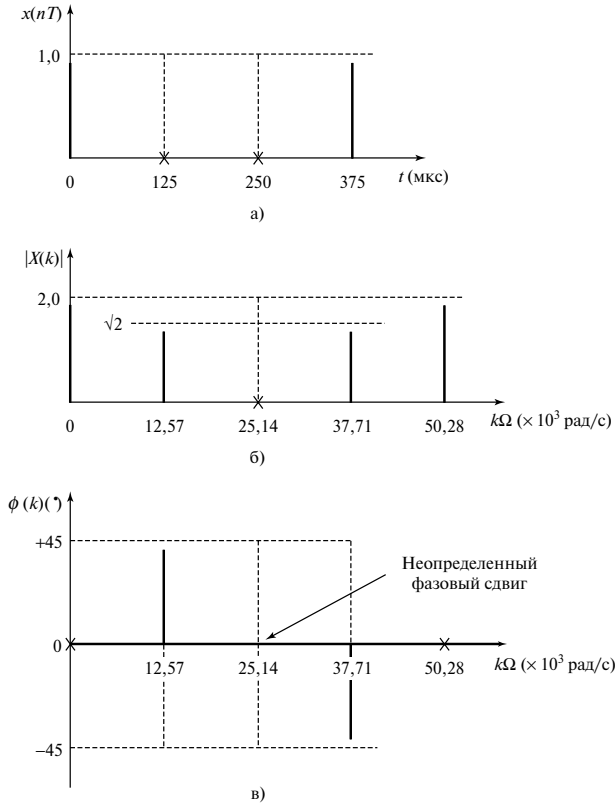


Рис. 3.3. График зависимости $x(nT)$ от t (панель а), график зависимости $|X(k)|$ от k (панель б); график зависимости $\phi(k)$ от k (панель в)

Как объяснялось во введении к данной главе, часто возникает необходимость выполнить дискретное преобразование из частотной области во временную. Это можно сделать с помощью обратного дискретного преобразования Фурье (ОДПФ), которое определяется как

$$x(nT) = F_D^{-1}[X(k)] = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{ik\Omega nT}, \quad n = 0, 1, \dots, N-1, \quad (3.23)$$

где через F_D^{-1} обозначено обратное дискретное преобразование Фурье.

Аналогия обратного дискретного преобразования Фурье с преобразованием, определяемым уравнением (3.16), очевидна. Здесь достаточно легко показать, что обратное преобразование Фурье можно получить из ОДПФ, разделив ОДПФ на T . Справедливость уравнения (3.23) можно продемонстрировать, подставив $x(nT)$ в уравнение (3.20).

Пример 3.4

Для иллюстрации обратного дискретного преобразования Фурье выведем временной ряд $\{1, 0, 0, 1\}$ из его ДПФ-образа $[2, 1 + i, 0, 1 - i]$.

При $n = 0$

$$\begin{aligned} x(nT) &= x(0) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) = \\ &= \frac{1}{4} [X(0) + X(1) + X(2) + X(3)] = \\ &= \frac{1}{4} [2 + (1 + i) + 0 + (1 - i)] = 1, \end{aligned}$$

как и ожидалось. При $n = 1$

$$\begin{aligned} x(nT) &= x(T) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{ik\Omega T} = \\ &= \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{ik2\pi/N} = \\ &= \frac{1}{4} \sum_{k=0}^{N-1} X(k) e^{ik\pi/2} = \\ &= \frac{1}{4} [2 + (1 + i) e^{i\pi/2} + 0 + (1 - i) e^{i3\pi/2}] = \\ &= \frac{1}{4} [2 + (1 + i)i + (1 - i)(-i)] = \frac{1}{4} (2 + i - 1 - i - 1) = 0, \end{aligned}$$

чего и следовало ожидать. При $n = 3$

$$\begin{aligned} x(nT) &= x(3T) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{ik\pi} = \\ &= \frac{1}{4} [2 + (1 + i) e^{i\pi} + (1 - i) e^{i3\pi}] = \\ &= \frac{1}{4} [2 - (1 + i) - (1 - i)] = 0, \end{aligned}$$

снова, как и ожидалось. Наконец, при $n = 3$

$$\begin{aligned} x(nT) &= x(3T) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{ik3\pi/2} = \\ &= \frac{1}{4} [2 + (1 + i) e^{i3\pi/2} + (1 - i) e^{i9\pi/2}] = \\ &= \frac{1}{4} [2 + (1 + i)(-i) + (1 - i)i] = \frac{1}{4} (2 - i + 1 + i + 1) = 1. \end{aligned}$$

Таким образом, получен правильный последний член ряда.

3.3. Свойства ДПФ

ДПФ обладает рядом математических свойств, которыми можно воспользоваться, чтобы упростить задачу или построить удачные приложения. Некоторые из этих свойств перечислены ниже. Для упрощения выражений последовательность данных $x(nT)$ записывается как $x(n)$.

1. *Симметрия.*

$$\operatorname{Re}[X(N - k)] = \operatorname{Re}X(k), \quad (3.24)$$

где Re означает действительную часть. Данная формула выражает симметрию амплитудного спектра, которая обсуждалась выше, а формула

$$\operatorname{Im}[X(N - k)] = -\operatorname{Im}[X(k)] \quad (3.25)$$

(где Im означает мнимую часть) выражает антисимметрию фазового спектра. Данное свойство полезно при определении значений компонентов.

2. *Четные функции.* Если $x(n)$ — четная функция $x_{\text{ч}}(n)$, т.е. $x_{\text{ч}}(n) = x_{\text{ч}}(-n)$, то

$$F_D[x_{\text{ч}}(n)] = X_{\text{ч}}(k) = \sum_{n=0}^{N-1} x_{\text{ч}}(n) \cos(k\Omega nT). \quad (3.26)$$

3. *Нечетные функции.* Если $x(n)$ — нечетная функция $x_{\text{н}}(n)$, т.е. $x_{\text{н}}(n) = -x_{\text{н}}(-n)$, то

$$F_D[x_{\text{н}}(n)] = X_{\text{н}}(k) = \sum_{n=0}^{N-1} x_{\text{н}}(n) \sin(k\Omega nT). \quad (3.27)$$

4. *Теорема Парсеваля.* Нормированная энергия сигнала равна

$$\sum_{n=0}^{N-1} x^2(n) = \frac{1}{N} \sum_{n=0}^{N-1} |X(k)|^2. \quad (3.28)$$

Правая часть уравнения (3.28) — это среднеквадратическая спектральная амплитуда, а левая часть — сумма квадратов амплитуд временного ряда.

5. *Дельта-функция.*

$$F_D[\delta(nT)] = 1. \quad (3.29)$$

6. С помощью ДПФ можно найти линейную взаимную корреляцию двух последовательностей данных или рядов. Линейная взаимная корреляция двух конечных последовательностей $x_1(n)$ и $x_2(n)$, каждая длиной N , определяется как

$$r_{x_1 x_2}(j) = \frac{1}{N} \sum_{n=-\infty}^{\infty} x_1(n) x_2(n + j), \quad -\infty \leq j \leq \infty. \quad (3.30)$$

Круговую корреляцию периодических последовательностей конечной длины $x_{1p}(n)$ и $x_{2p}(n)$ можно определить как

$$r_{cx_1x_2}(j) = \frac{1}{N} \sum_{n=0}^{N-1} x_{1p}(n)x_{2p}(n+j), j = 0, \dots, N-1 \quad (3.31)$$

(поскольку круговая корреляция определяется с помощью ДПФ). Следовательно,

$$r_{cx_1x_2}(j) = F_D^{-1}[X_1^*(k)X_2(k)]. \quad (3.32)$$

Уравнение (3.32) называют *теоремой о корреляции*. Круговую корреляцию, которая задается уравнением (3.32), можно преобразовать в линейную с помощью дополняющих нулей. Теперь для последовательностей $x_1(n)$ с длиной N_1 и $x_2(n)$ с длиной N_2 их линейная корреляция будет иметь длину $N_1 + N_2 - 1$. Для этого последовательность $x_1(n)$ заменяют последовательностью $x_{1a}(n)$, которая состоит из последовательности $x_1(n)$ с добавленными $N_2 - 1$ нулями, а последовательность $x_2(n)$, дополненная $N_1 - 1$ нулями, превращается в последовательность $x_{2a}(n)$. Теперь линейная взаимная корреляция последовательностей $x_1(n)$ и $x_2(n)$ задается как

$$r_{x_1x_2}(j) = F_D^{-1}[X_{1a}^*(k)X_{2a}(k)], \quad (3.33)$$

где

$$X_{1a}(k) = F_D[X_{1a}(n)] \text{ и } X_{2a}(k) = F_D[X_{2a}(n)].$$

Более полно эта тема освещается в главе 5.

7. ДПФ также можно использовать для вычисления круговой и (с помощью дополняющих нулей) линейной свертки. Эти свертки могут быть как во временной, так и в частотной области. В теореме о свертке во временной области говорится, что

$$x_{3p}(n) = x_{1p}(n) \otimes x_{2p}(n) = F_D^{-1}[X_1(k)X_2(k)], \quad (3.34)$$

где $\otimes$ обозначает круговую свертку, а $x_{1p}(n)$, $x_{2p}(n)$, $x_{3p}(n)$ — конечные периодические последовательности одинаковой длины.

Аналогично уравнению (3.31) $x_{3p}(n)$ можно также записать как

$$x_{3p}(n) = \sum_{m=0}^{N-1} x_{1p}(m)x_{2p}(n-m). \quad (3.35)$$

Кроме того,

$$X_3(k) = X_1(k)X_2(k), \quad (3.36)$$

где $X_3(k) = F_D[x_3(n)]$.

Следующее уравнение — это формулировка теоремы о свертке в частотной области:

$$\frac{1}{N}X_1(k) \circledast X_2(k) = F_D[x_1(n)x_2(n)], \quad (3.37)$$

где

$$X_1(k) \circledast X_2(k) = \sum_{m=0}^{N-1} X_1(m)X_2(k-m). \quad (3.38)$$

Уравнение (3.34) дает основания утверждать, что свертка во временной области эквивалентна умножению в частотной, тогда как уравнение (3.37) приводит к наблюдению о том, что свертка в частотной области эквивалентна умножению во временной. Более подробно свертка рассмотрена в главе 5.

3.4. Вычислительная сложность ДПФ

Для вычисления ДПФ необходимо выполнить большое количество операций сложения и умножения. Для 8-точечного ДПФ (согласно уравнению (3.20)) расписанная последовательность $X(k)$ имеет вид:

$$X(k) = \sum_{n=0}^7 x(n)e^{-ik2\pi n/8}, \quad k = 0, \dots, 7, \quad (3.39)$$

а замена $k2\pi/8 = K$ позволяет записать выражение

$$\begin{aligned} X(k) = & x(0)e^{-iK0} + x(1)e^{-iK1} + x(2)e^{-iK2} + x(3)e^{-iK3} + x(4)e^{-iK4} + \\ & + x(5)e^{-iK5} + x(6)e^{-iK6} + x(7)e^{-iK7}, \quad k = 0, \dots, 7. \end{aligned} \quad (3.40)$$

Правая сторона уравнения (3.40) содержит восемь членов. Вычисление каждого члена включает операцию умножения всегда комплексного экспоненциального множителя на другой множитель, который может быть либо действительным, либо комплексным (например, во временных рядах напряжения он действительный). Затем все эти произведения складываются. Таким образом, для вычисления этой величины необходимо выполнить восемь операций комплексного умножения и семь операций комплексного сложения. Для N -точечного ДПФ их число будет составлять соответственно N и $N-1$. Кроме этого, надо вычислить также восемь компонентов гармоник ($k = 0, \dots, 7$). Для N -точечного ДПФ это число равно N . Следовательно, для вычисления восьмиточечного ДПФ необходимо выполнить $8^2 = 64$ операции комплексного умножения и $8 \times 7 = 56$ операций комплексного сложения. Для N -точечного ДПФ эти числа равны соответственно N^2 и $N(N-1)$. Если $N = 1024$, то понадобится приблизительно один миллион операций комплексного умножения и один миллион операций комплексного сложения. Ясно, что это число нужно каким-то образом уменьшить.

Объем необходимых вычислений можно сократить, если вспомнить о значительной избыточности, в частности, таких уравнений, как (3.4). Например, если $k = 1$, а $n = 2$, то $e^{-ik2\pi n/8} = e^{-i\pi/2}$, а если $k = 2$, а $n = 1$, то $e^{-ik2\pi n/8}$ также равно $e^{-i\pi/2}$.

3.5. Алгоритм быстрого преобразования Фурье с децимацией во временной области

В этом разделе будет показано, как внутреннюю избыточность вычислений, свойственную ДПФ, можно использовать для уменьшения числа требуемых операций и, соответственно, увеличения скорости вычислений. Для 1024-точечного ДПФ объем требуемых вычислений можно снизить в 204,8 раз. Соответствующие алгоритмы получили название “быстрого преобразования Фурье” (БПФ). Если этот алгоритм применяется во временной области, его называют БПФ с прореживанием во временной области (временной децимацией — ВД). Первый алгоритм БПФ-ВД появился благодаря Кули (Cooley) и Тьюки (Tukey) [9], в честь которых его часто и называют. Децимация приводит к значительному снижению числа операций, выполняемых с данными во временной области. Следует заметить, что выигрыш на вычислениях растет по закону $N^2 - (N/2) \log_2 N$.

Итак, упростим вначале запись и установим некоторые математические соотношения. Уравнение (3.20) можно переписать как

$$X_1(k) = \sum_{n=0}^{N-1} x_n e^{-2\pi i n k / N}, \quad k = 0, \dots, N-1. \quad (3.41)$$

Кроме того, выражение $e^{-i2\pi/N}$ запишем как W_N , следовательно

$$W_N = e^{-2\pi i / N}. \quad (3.42)$$

Так что уравнение (3.41) приобретает вид

$$X_1(k) = \sum_{n=0}^{N-1} x_n W_N^{kn}, \quad k = 0, \dots, N-1. \quad (3.43)$$

Здесь следует обратить внимание на некоторые соотношения, содержащие W_N . Во-первых,

$$W_N^2 = (e^{-2\pi i / N})^2 = e^{-i2\pi 2 / N} = e^{-2\pi i / (N/2)} = W_{N/2}. \quad (3.44)$$

Во-вторых,

$$\begin{aligned} W_N^{(k+N/2)} &= W_N^k W_N^{N/2} = W_N^k e^{-(2\pi i / N)(N/2)} = W_N^k e^{-i\pi} = \\ &= -W_N^k. \end{aligned} \quad (3.45)$$

Для удобства обобщим полезные результаты, касающиеся W_N , и получим

$$W_N = e^{-2\pi i / N}, \quad (3.46, a)$$

$$W_N^2 = W_{N/2}, \quad (3.46, б)$$

$$W_N^{k+N/2} = -W_N^k. \quad (3.46, в)$$

При использовании вычислительной избыточности, выражаемой уравнениями (3.46), последовательность данных делится на две равные последовательности, одна из которых состоит из элементов с четными порядковыми номерами, а другая — с нечетными. Чтобы длина последовательностей была одинаковой, они должны состоять из четного количества элементов. Если исходная последовательность состоит из нечетного числа элементов, то необходимо добавить дополняющий нуль, чтобы их число стало четным. Это позволит записать ДПФ $X_1(k)$ через два ДПФ, $X_{11}(k)$ и $X_{12}(k)$, которые будут соответственно ДПФ-образами данных с четными и нечетными значениями (табл. 3.1). Следовательно, N -точечное ДПФ превращается в два ДПФ, каждое из которых содержит $N/2$ точек. Затем этот процесс повторяется до тех пор, пока $X_1(k)$ не распадется на $N/2$ ДПФ, каждое из которых состоит из двух точек, представляющих собой исходные данные. Таким образом, на практике происходит переупорядочение исходных данных, и вычисляются $N/2$ двухточечных ДПФ, которые получают исходные данные парами. Результаты этих ДПФ удобно объединять в четверки, всего получается $N/4$ четырехточечных ДПФ, которые вычисляются и соответственно объединяются в $N/8$ восьмиточечных ДПФ, которые также вычисляются, и так далее, до тех пор, пока не получится конечное N -точечное ДПФ $X_1(k)$. На каждом этапе для снижения количества комплексных операций используется общий множитель, представляющий собой W_N в некоторой степени. Правомерность этой процедуры доказывается следующим образом.

Индекс n в уравнении (3.43) пробегает значения от $n = 0$ до $n = N - 1$, что соответствует значениям данных $x_0, x_1, x_2, x_3, \dots, x_{N-1}$. Последовательность с четными номерами — это $x_0, x_2, x_4, \dots, x_{N-2}$, а последовательность с нечетными номерами — $x_1, x_3, \dots, x_{N-1}$. Обе последовательности содержат по $N/2$ точек. Члены четной последовательности можно обозначить x_{2n} , где n пробегает значения от нуля до $N/2 - 1$, тогда как члены нечетной последовательности — x_{2n+1} . Тогда уравнение (3.43) можно переписать как

$$\begin{aligned}
 X_1(k) &= \underbrace{\sum_{n=0}^{N/2-1} x_{2n} W_N^{2nk}}_{\text{четная последовательность}} + \underbrace{\sum_{n=0}^{N/2-1} x_{2n+1} W_N^{(2n+1)k}}_{\text{нечетная последовательность}} = \\
 &= \sum_{n=0}^{N/2-1} x_{2n} W_N^{2nk} + W_N^k \sum_{n=0}^{N/2-1} x_{2n+1} W_N^{2nk}, \quad k = 0, \dots, N-1.
 \end{aligned} \tag{3.47}$$

Из уравнения (3.46, б) $W_N^{2nk} = W_{N/2}^{nk}$, так что уравнение (3.47) приобретает вид

$$X(k) = \sum_{n=0}^{N/2-1} x_{2n} W_{N/2}^{nk} + W_N^k \sum_{n=0}^{N/2-1} x_{2n+1} W_{N/2}^{nk}, \quad k = 0, \dots, N-1. \tag{3.48}$$

Уравнение (3.48) можно переписать в виде:

$$X_1(k) = X_{11}(k) + W_N^k X_{12}(k), \quad k = 0, \dots, N-1. \tag{3.49}$$

Сравнивая уравнение (3.49) с уравнением (3.43), видим, что $X_{11}(k)$ — действительно ДПФ-образ четной последовательности, тогда как $X_{12}(k)$ — образ нечетной последовательности. Следовательно, как утверждалось ранее, ДПФ-образ $X_1(k)$ можно выразить

через два ДПФ-образа: $X_{11}(k)$ и $X_{12}(k)$. При таком подходе коэффициент $W_{N/2}^k$ появляется и в $X_{11}(k)$, и в $X_{12}(k)$, и вычислять его нужно только один раз.

В табл. 3.1 изображен процесс вычисления восьмиточечного ДПФ. В первой строке записаны данные, тогда как во второй строке дается выражение для ДПФ-образа этих данных, записанное через ДПФ-образы четной и нечетной последовательностей $X_{11}(k)$ и $X_{12}(k)$ соответственно. В третьей строке приведены переупорядоченные данные, из которых получают $X_{11}(k)$ и $X_{12}(k)$. В четвертой строке даются ДПФ-образы последовательностей из строки 3, выраженные через ДПФ-образы их четных и нечетных последовательностей $X_{21}(k)$, $X_{22}(k)$, $X_{23}(k)$ и $X_{24}(k)$. Эти последовательности показаны в строке 5, и видно, что они состоят из двухточечных последовательностей, ДПФ-образы которых $X_{21}(k)$, $X_{22}(k)$, $X_{23}(k)$ и $X_{24}(k)$ выражаются через данные из строки 6. Итак, единое восьмиточечное ДПФ раскладывается на четыре двухточечных ДПФ, каждое из которых дает два значения, например, для $X_{21}(k)$ — это $X_{21}(0)$ и $X_{21}(1)$. Описанный процесс включает два разложения, и коэффициент W_N^k на каждом шаге возводится в квадрат. Из строки 6 видно, что

$$X_{21}(k) = x_0 + W_{N/4}^k x_4 \quad k = 0, \dots, N/4 - 1, \quad \text{т.е. } k = 0, 1 \quad (3.50)$$

Следовательно,

$$X_{21}(0) = x_0 + x_4,$$

тогда как

$$\begin{aligned} X_{21}(1) &= x_0 + W_{N/4} x_4 = \\ &= x_0 + W_2 x_4 = x_0 + e^{-2\pi i/2} x_4 = x_0 + e^{-\pi i} x_4 = x_0 - x_4. \end{aligned}$$

Аналогично

$$\begin{aligned} X_{22}(0) &= x_2 + x_6, & X_{22}(1) &= x_2 - x_6, \\ X_{23}(0) &= x_1 + x_5, & X_{23}(1) &= x_1 - x_5, \\ X_{24}(0) &= x_3 + x_7, & X_{24}(1) &= x_3 - x_7, \end{aligned}$$

откуда видно, что значения при $k = 0$ отличаются от значений при $k = 1$ только знаком. Этот момент особо важен, если рассматривается случай $X_{11}(k)$ ($k = 0, 1, 2, 3$). Теперь

$$X_{11}(k) = X_{21}(k) + W_{N/2}^k X_{22}(k), \quad (3.51)$$

так что

$$X_{11}(0) = X_{21}(0) + W_{N/2}^0 X_{22}(0) = X_{21}(0) + X_{22}(0) \quad (3.52)$$

$$\begin{aligned} X_{11}(1) &= X_{21}(1) + W_{N/2}^1 X_{22}(1) = \\ &= X_{21}(1) + e^{-\pi i/2} X_{22}(1) = X_{21}(1) - i X_{22}(1) \end{aligned} \quad (3.53)$$

$$\begin{aligned} X_{11}(2) &= X_{21}(2) + W_{N/2}^2 X_{22}(2) = X_{21}(2) + e^{-(2\pi i/8)2 \times 2} X_{22}(2) = \\ &= X_{21}(2) + e^{-i\pi} X_{22}(2) = X_{21}(2) - X_{22}(2). \end{aligned} \quad (3.54)$$

Таблица 3.1. Структура 8-точечного БПФ

Номер строки	Содержание строки	Диапазон изменения k				Диапазон изменения N					
1	Последовательность данных A_0	A_0	$x_0x_1x_2x_3x_4x_5x_6x_7$				$0, \dots, 7$				
2	Восьмиточечное ДПФ последовательности A_0	$X_1(k) = X_{11}(k) + W_N^k X_{12}(k)$						$0, \dots, N-1$ $(0, \dots, 7)$	$0, \dots, 7$		
3	Переупорядоченная последовательность A_0 : две последовательности A_1 и A_2	A_1	$x_0x_2x_4x_6$	A_2	$x_1x_3x_5x_7$			$0, \dots, 3$			
4	Четырехточечные ДПФ последовательностей A_1 и A_2	$X_{11}(k) = X_{21}(k) + W_{N/2}^k X_{22}(k)$			$X_{12}(k) = X_{23}(k) + W_{N/2}^k X_{24}(k)$			$0, \dots, N/2-1$ $(0, \dots, 3)$	$0, \dots, 3$		
5	Переупорядоченные последовательности A_1 и A_2 : четыре последовательности A_3, A_4, A_5, A_6	A_3	x_0x_4	A_4	x_2x_6	A_5	x_1x_5	A_6	x_3x_7	$0, 1$	
6	Двухточечные ДПФ последовательностей A_3, A_4, A_5, A_6	$X_{21}(k) = x_0 + W_{N/4}^k x_4$		$X_{22}(k) = x_2 + W_{N/4}^k x_6$		$X_{23}(k) = x_1 + W_{N/4}^k x_5$		$X_{24}(k) = x_3 + W_{N/4}^k x_7$		$0, \dots, N/4-1$ $(0, 1)$	$0, 1$

Теперь

$$X_{21}(2) = x_0 + W_{N/4}^2 x_4 = x_0 + W_2^2 x_4 = x_0 + x_4 = X_{21}(0),$$

и

$$X_{22}(2) = x_2 + W_{N/4}^2 x_6 = x_2 + x_6 = X_{22}(0).$$

Следовательно, уравнение (3.54) эквивалентно уравнениям

$$X_{11}(2) = X_{21}(0) - X_{22}(0) \quad (3.55)$$

$$X_{11}(3) = X_{21}(3) - W_{N/2}^3 X_{22}(3) \quad (3.56)$$

Теперь

$$\begin{aligned} X_{21}(3) &= x_0 + W_{N/4}^3 x_4 = \\ &= x_0 + e^{-i(2\pi/2)^3} x_4 = x_0 + e^{-i3\pi} x_4 = x_0 - x_4 = X_{21}(1) \end{aligned}$$

и

$$X_{22}(3) = x_2 - x_6 = X_{22}(1).$$

Следовательно, уравнение (3.56) эквивалентно уравнению

$$\begin{aligned} X_{11}(3) &= X_{21}(1) + e^{-(2\pi i/4)^3} X_{22}(1) = X_{21}(1) + e^{-3\pi i/2} X_{22}(1) = \\ &= X_{21}(1) + iX_{22}(1). \end{aligned} \quad (3.57)$$

Сведя эти результаты вместе, получим

$$X_{11}(0) = X_{21}(0) + X_{22}(0) = X_{21}(0) + W_8^0 X_{22}(0), \quad (3.58, a)$$

$$X_{11}(2) = X_{21}(0) - X_{22}(0) = X_{21}(0) - W_8^0 X_{22}(0), \quad (3.58, б)$$

$$X_{11}(1) = X_{21}(1) - iX_{22}(1) = X_{21}(1) + W_8^2 X_{22}(1), \quad (3.58, в)$$

$$X_{11}(3) = X_{21}(1) + iX_{22}(1) = X_{21}(1) + W_8^2 X_{22}(1). \quad (3.58, г)$$

Внимательно изучая эти уравнения, видим, как ДПФ-образы $X_{11}(k)$ связаны с ДПФ-образами их последовательностей с четными и нечетными номерами, и что $X_{11}(0)$ и $X_{11}(2)$ задаются выражениями с одинаковыми членами, которые отличаются только знаком. То же справедливо и для $X_{11}(1)$ и $X_{11}(3)$. Уравнения, подобные этим, называются *рекомпозиционными*, поскольку, взяв пары данных и формируя $X_{21}(k)$, $X_{22}(k)$, $X_{23}(k)$ и $X_{24}(k)$, можно найти $X_{11}(k)$ и $X_{12}(k)$, а следовательно, и $X_1(k)$. При этом количество необходимых операций комплексного сложения и умножения уменьшается, поскольку, во-первых, рекомпозиционные уравнения записываются через рекуррентный коэффициент W_N в некоторой степени, во-вторых, используются также соотношения типа $X_{21}(2) = X_{21}(0)$ и $X_{21}(3) = X_{21}(1)$ и, в-третьих, различие между парами используемых выражений состоит только в знаке. Этот алгоритм известен под названием алгоритма Кули–Тьюки.

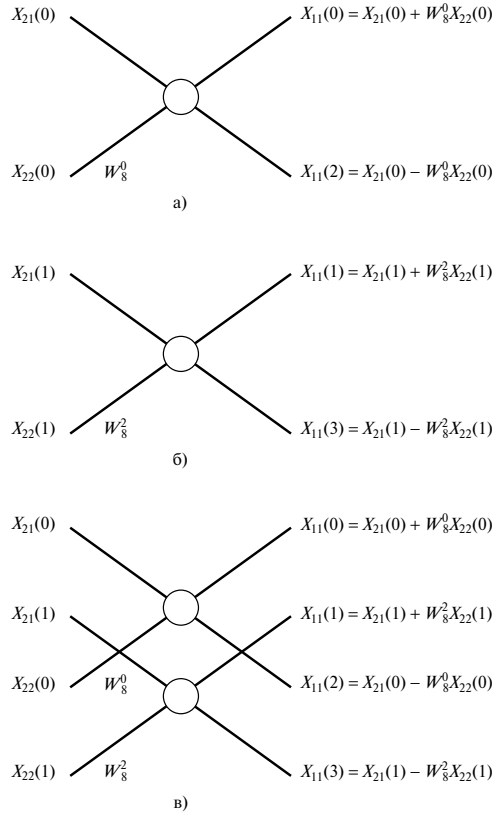


Рис. 3.4. “Бабочки” БПФ

3.5.1. “Бабочка”

Уравнения (3.58) можно изобразить на диаграмме, взяв их попарно и воспользовавшись симметрией знаков. Итак, согласно уравнениям (3.58, а) и (3.58, б) результатом рекомпозиции будут $X_{11}(0)$ и $X_{11}(0)$, получившиеся из входных $X_{21}(0)$ и $X_{22}(0)$. Это показано на рис. 3.4, а. Входные данные находятся с левой стороны от креста, а выходные — с правой. На рис. 3.4, б показано, как с помощью диаграммы можно получить $X_{11}(1)$ и $X_{11}(3)$. При наложении рис. 3.4, а на рис. 3.4, б получается составная диаграмма, на которой выходные ДПФ расположены в порядке возрастания k (см. рис. 3.4, в). Структура, изображенная на рис. 3.4, а или б, называется “бабочкой”, поскольку диаграмма похожа на стилизованное изображение этого насекомого. Целиком восьмиточечное БПФ можно изобразить так, как показано на рис. 3.5.

Пример 3.5

Сейчас будет полезно найти ДПФ последовательности $\{1, 0, 0, 1\}$, ранее определенное в разделе 3.2 с помощью алгоритма БПФ с временной децимацией. Это четырехточечное ДПФ с $x_0 = 1, x_1 = 0, x_2 = 0, x_3 = 1$ и $X_1(k) = X_{11}(k) + W_N^k X_{12}(k), k = 0, 1, 2, 3$. Переупорядоченная последовательность — x_0, x_2, x_1, x_3 .

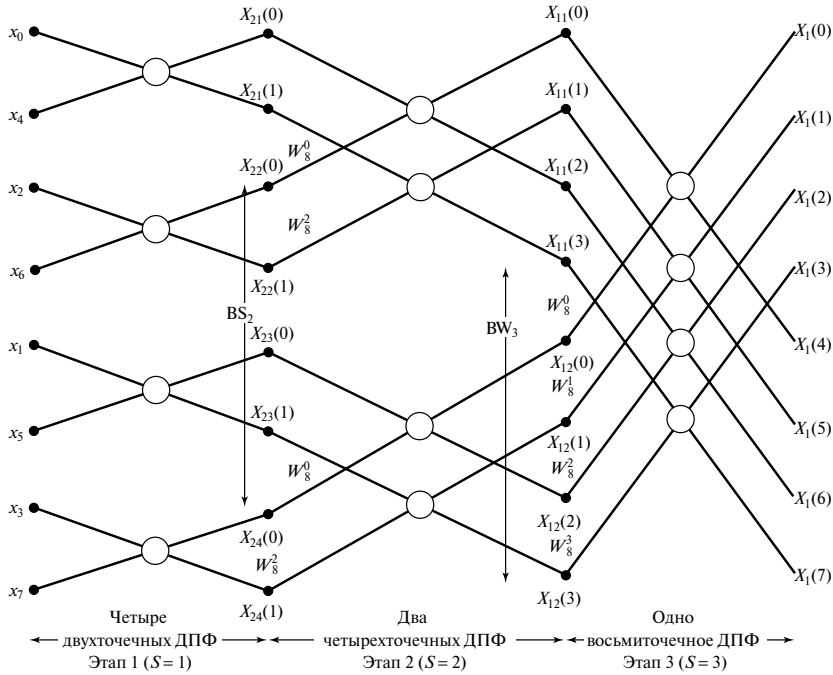


Рис. 3.5. “Бабочки” восьмиточечного ДПФ: BW_3 — память, выделенная точкам, дающим вклад в самую большую “бабочку” этапа 3; BS_2 — память, выделенная верхним точкам “бабочки” этапа 2 с одинаковыми весовыми коэффициентами

Теперь для выполнения ДПФ можно воспользоваться верхним левым углом рис. 3.5. Точки x_0, x_4, x_2, x_6 заменим точками x_0, x_2, x_1, x_3 , и искомыми значениями ДПФ будут $X_{11}(0), X_{11}(1), X_{11}(2), X_{11}(3)$. Следовательно,

$$\begin{aligned}
 X_{21}(0) &= x_0 + x_2 = 1, \\
 X_{21}(1) &= x_0 - x_2 = 1, \\
 X_{22}(0) &= x_1 + x_3 = 1, \\
 X_{22}(1) &= x_1 - x_3 = -1, \\
 X_{11}(0) &= X_{21}(0) + W_8^0 X_{22}(0) = 1 + 1 = 2, \\
 X_{11}(1) &= X_{21}(1) + W_8^2 X_{22}(1) = 1 + e^{-i\pi/2}(-1) = 1 + i, \\
 X_{11}(2) &= X_{21}(0) - W_8^0 X_{22}(0) = 1 - 1 = 0, \\
 X_{11}(3) &= X_{21}(1) - W_8^2 X_{22}(1) = 1 - i.
 \end{aligned}$$

Полученные значения идентичны величинам, которые были найдены в разделе 3.2, но с помощью алгоритма БПФ их получить намного проще. Это общий вывод, причем сокращение вычислений возрастает при увеличении числа исходных выборок.

Таблица 3.2. Переупорядочение последовательности путем обращения битов

<i>Последовательность, необходимая для вычисления “бабочки”</i>	<i>Бинарный адрес нужной последовательности данных</i>	<i>Адрес с обратным порядком битов</i>	<i>Соответствующая последовательность = исходная последовательность данных</i>
x_0	000	000	x_0
x_4	100	001	x_1
x_2	010	010	x_2
x_6	110	011	x_3
x_1	001	100	x_4
x_5	101	101	x_5
x_3	011	110	x_6
x_7	111	111	x_7

3.5.2. Выполнение алгоритма

Изучая рис. 3.5, видим, что для выполнения БПФ программа должна переупорядочить входные данные и выполнить вычисления по схеме “бабочка”. Обсудим эти операции по очереди (см. также [24]).

3.5.2.1. Переупорядочение входных данных

Хотя на первый взгляд может показаться, что у программы нет четкого способа переупорядочения данных, он есть. Секрет в том, что нужно перейти в двоичную систему счисления. В первом столбце табл. 3.2 показан нужный порядок входных данных для “бабочки”, изображенной на рис. 3.5. Предполагается, что каждое значение сохраняется в двоичной ячейке памяти. Адреса этих ячеек даны во втором столбце. В третьем столбце показаны эти же адреса, но с обратной последовательностью битов. Если в соответствии с этими обратными адресами взять двоичные адреса исходной последовательности данных, начав с $x(0)$ для 000, то получатся значения, которые записаны в четвертом столбце. Таким образом, видно, что адреса переупорядоченных данных — это записанные с обратной последовательностью битов адреса исходных данных. Следовательно, программа должна преобразовать номера заданных точек (от 0 до $N - 1$) в двоичные, обратить последовательность битов в этих номерах и преобразовать их снова в десятичные номера, которые и будут адресами переупорядоченных данных. Переход в двоичную систему осуществляется многократным делением на два, при этом остатки дают цифры соответствующего двоичного числа в обратном порядке, которое и является искомым бинарным адресом переупорядоченной последовательности. Эти остатки можно найти, воспользовавшись функцией MOD, реализованной в языках высшего уровня, например, функция $\text{MOD}(K, 2)$ выдаст остаток при делении десятичного числа K на 2. Целая часть $K/2$ находится путем целочисленного деления. Остальные цифры вычисляются путем повторения этого процесса до тех пор, пока не будет выполнено $\log_2 N$ операций деления. Это следует из того, что, поскольку данные должны состоять из $2^m = N$ точек, для каждого адреса нужно m цифр, и он должен делиться на два m раз, где $m = \log_2 N$. Отметим, что I -й бит нового адреса — это двоичный коэффициент числа 2^{m-1-I} , так что новый адрес (NADDR) можно найти с помощью цикла DO для определенного значения K ($K = \text{IADDR}$), которое приобретает значения от $I = 0$

до $I = m - 1$ и вычисляет остатки (RMNDR) от деления $K = IADDR$ последовательных целых чисел на два. Соответствующий псевдокод приводится ниже.

```
DO FOR I = 0 TO m-1
    RMNDR:=MOD(IADDR, 2)
    NADDR:=NADDR+RMNDR*2^(m-1-I)
    IADDR:=IADDR/2
END DO
```

Этот цикл DO должен находиться внутри еще одного цикла DO, задача которого — извлечь исходные данные из комплексного массива $DATA(K)$, K пробегает значения от 0 до $N - 1$, представляющие собой номер точки из набора данных (а также номер комплексного элемента массива и исходный адрес данных), и ввести переупорядоченные данные в массив $NEWDATA(NADDR)$. Данные в массиве $NEWDATA$ теперь стоят в последовательности, подходящей для вычисления “бабочки”. Полный псевдокод, таким образом, выглядит так:

```
DO FOR K = 0 TO N-1
    NADDR:=0
    IADDR:=K
    DO FOR I = 0 TO m-1
        RMNDR:=MOD(IADDR, 2)
        NADDR:=NADDR+RMNDR*2^(m-1-I)
        IADDR:=IADDR/2
    END DO
    NEWDATA(NADDR) := DATA(K)
END DO
```

3.5.2.2. Вычисление “бабочки”

Вычисление состоит из трех этапов:

- а) вычисление весовых коэффициентов $W_N^R = e^{-(2\pi i/N)/R}$;
- б) вычисление “бабочки” данного каскада (определение каскадов см. на рис. 3.5);
- в) вычисление всех каскадов “бабочки”.

Достаточно эффективная реализация вычислений по схеме “бабочка” включает вычисление весовых коэффициентов W_N^R , необходимых на данном каскаде, с последующим вычислением для каждого из них всех “бабочек” этого каскада, в которые входит данный коэффициент. После вычисления всех “бабочек” данного каскада процедура повторяется для всех каскадов. Итак, для каскада 2 на рис. 3.5 вычисляются W_8^0 и две “бабочки”, содержащие коэффициент W_8^0 . Затем вычисляются W_8^2 и две “бабочки”, содержащие этот коэффициент. Эта процедура выполняется для каждого из трех каскадов по очереди, при этом используются каскад 1 и переупорядоченные данные. Чтобы вычислить БПФ с помощью относительно короткой программы, алгоритм должен обладать рядом свойств. Пусть ширина “бабочки” $BWIDTH$ представляет расстояние в памяти между двумя точками, составляющими “бабочку”. Для нижней “бабочки” кас-

када 3 это BW_3 . Это равно расстоянию между четырьмя точками. Рассмотрев “бабочки” других каскадов, можно прийти к заключению, что в общем случае

$$BWIDTH = 2^{S-1}, \quad (3.59)$$

где S — номер этапа. $BWIDTH - 1$ — количество шагов, за которое изменяется показатель степени весового коэффициента. Пусть расстояние в памяти BSEP равно расстоянию в памяти между любыми точками ближайших “бабочек” данного каскада с одинаковым весовым коэффициентом. Для каскада 2 на рис. 3.5 BS_2 представляет собой BSEP для “бабочек” с весовым коэффициентом W_8^0 . Изучая рисунок, видим, что для каскада S

$$BSEP = 2^S. \quad (3.60)$$

Наконец, для N -точечного БПФ показатели степени весовых коэффициентов изменяются на

$$P = N/2^S \quad (3.61)$$

для этапа S . Это видно из рис. 3.5. Например, на этапе 2 $S = 2$, $P = 8/2^2 = 2$, а весовые коэффициенты равны W_8^0 и W_8^2 .

Каждую “бабочку” можно вычислить как

$$\begin{aligned} XNEW(TOP) &= XOLD(TOP) + W_N^R \times XOLD(BOTTOM) \\ XNEW(BOTTOM) &= XOLD(TOP) - W_N^R \times XOLD(BOTTOM), \end{aligned} \quad (3.62)$$

где левая сторона относится ко входу “бабочки”, а правая сторона — к выходу “бабочек”. Кроме того, для экономии памяти эти уравнения можно переписать в виде

$$TEMP = W_N^R \times X(BOTTOM),$$

где $X(BOTTOM)$ означает выход $XOLD(BOTTOM)$ “бабочки” на предыдущем этапе,

$$X(BOTTOM) = X(TOP) - TEMP, \quad (3.63, a)$$

где $X(BOTTOM)$ теперь означает искомый выход “бабочки”, $X(TOP)$ — предыдущее значение, а

$$X(TOP) = X(TOP) + TEMP, \quad (3.63, б)$$

где значение $X(TOP)$ в левой части — искомый выход “бабочки”.

Зная все это, можно записать псевдокод вычисления БПФ, как показано ниже.

```

10 PI:=3.141593
20 DO FOR S=1 TO m           рассчитать для m каскадов
30     BSEP:=2^S              расчет для каскада S
40     P:=N/BSEP              (P = N/2^S), изменение показателя степени
50     BWIDTH:=BSEP/2         (BWIDTH = 2^(S-1)), расстояние в памяти
                               между входами "бабочек"
60     DO FOR J=0 TO (BWIDTH-1) расчет весовых коэффициентов
                               для отдельного каскада
70         R:=P.J             расчет показателя степени WN
80         THETA:=2*PI*R/N     расчет степени e^(-J)
90         WN:=CMPLX{cos(THETA),-sin(THETA)} расчет (WN)^R
100        DO FOR TOPVAL=J STEP BSEP UNTIL N/2
                               для всех J-х "бабочек" данного каскада
110            BOTVAL:=TOPVAL+BWIDTH
120            TEMP:=X(BOTVAL)*WN
130            X(BOTVAL):=X(TOPVAL)-TEMP
140            X(TOPVAL):=X(TOPVAL)+TEMP
150        END DO
160    END DO
170 END DO

```

В строках 100–150 рассчитываются все “бабочки” данного каскада с одинаковыми весовыми коэффициентами. Каскад всегда содержит $N/2$ “бабочек”.

3.5.3. Вычислительные преимущества БПФ

Вычислительные преимущества БПФ можно проиллюстрировать, рассмотрев вначале алгоритм БПФ на рис. 3.5. На этом рисунке показано, что N -точечное БПФ содержит $N/2$ “бабочек” на каждом каскаде и $\log_2 N$ каскадов, т.е. всего в него входит $(N/2) \log_2 N$ “бабочек”. На рис. 3.4, а показано, что каждая “бабочка” содержит одну операцию комплексного умножения в виде $W_N R X_{ij}(k)$. Следовательно, БПФ содержит $(N/2) \log_2 N$ операций комплексного умножения в отличие от N^2 при ДПФ, как показано в разделе 3.4. Таким образом, вычислительная экономия при комплексном умножении составляет $N^2 - (N/2) \log_2 N$. Каждая “бабочка” содержит две операции комплексного сложения, так что для БПФ необходимо $N \log_2 N$ операций комплексного сложения в сравнении с $N(N-1)$ операциями ДПФ. Следовательно, экономия при комплексном сложении составляет $N(N-1) - N \log_2 N$. Эта экономия проиллюстрирована в табл. 3.3. При обычном 1024-точечном ДПФ видно, что время, необходимое для вычислений, можно снизить на два порядка, если воспользоваться алгоритмом БПФ.

3.6. Обратное быстрое преобразование Фурье

Алгоритм БПФ для определения обратного быстрого преобразования Фурье можно легко получить из алгоритма БПФ. Этот алгоритм нужен в основном для преобразования спектров в соответствующие им сигналы и для проверки правильности вычисления

Таблица 3.3. Экономия на операциях комплексного умножения и сложения при использовании БПФ вместо ДПФ

<i>N</i>	<i>ДПФ</i>		<i>БПФ</i>		<i>Отношение числа комплексного комплексного умножения при ДПФ и при БПФ</i>	<i>Отношение числа комплексного комплексного сложения при ДПФ и при БПФ</i>
	<i>Число операций комплексного умножения</i>	<i>Число операций комплексного сложения</i>	<i>Число операций операций умножения</i>	<i>Число операций операций сложения</i>		
2	4	2	1	2	4	1
4	16	12	4	8	4	1,5
8	64	56	12	24	5,3	2,3
16	256	240	32	64	8,0	3,75
32	1024	992	80	160	12,8	6,2
64	4096	4032	192	384	21,3	10,5
128	16 384	16 256	448	896	36,6	18,1
256	65 536	65 280	1024	2048	64,0	31,9
512	262 144	261 632	2304	4608	113,8	56,8
1024	1 048 576	1 047 552	5120	10 240	204,8	102,3
2048	4 194 304	4 192 256	11 264	22 528	372,4	186,1
4096	16 777 216	16 773 120	24 576	49 152	682,7	341,3
8192	67 108 864	67 100 672	53 248	106 496	1 260,3	630,0

БПФ с помощью (как правило) того самого алгоритма. Чтобы понять, как выводится ОБПФ, произведем такие замены в уравнении (3.20). Просуммируем по переменной λ , а не по n , переменную k переименуем в μ , а $\Omega = 2\pi/NT$, чтобы показатель экспоненты стал $-ik(2\pi/N)\lambda$. Снова воспользуемся выражением $x(\lambda T) = x(\lambda)$ и т.д. При этом уравнение (3.20) приобретет вид

$$X(\mu) = \sum_{\lambda=0}^{N-1} x(\lambda) e^{-i\mu(2\pi/N)\lambda} \quad \mu = 0, 1, \dots, N-1. \quad (3.64)$$

Теперь сделаем аналогичную замену в уравнении (3.23), т.е. положим $k = \lambda$ и $n = \mu$. Уравнение (3.23) приобретет вид

$$X(\mu) = \frac{1}{N} \sum_{\lambda=0}^{N-1} x(\lambda) e^{i\mu(2\pi/N)\mu} \quad \mu = 0, 1, \dots, N-1. \quad (3.65)$$

В двух последних уравнениях $X(\mu)$, $X(\lambda)$, $x(\mu)$ и $x(\lambda)$ — элементы массивов X и x одинаковой размерности, так что ОБПФ $x(\mu)$ отличается от БПФ $X(\mu)$ только коэффициентом $1/N$ и знаком показателя степени экспоненты. Следовательно, для вычисления ОБПФ можно воспользоваться БПФ с небольшими изменениями. Эти два преобразования могут составлять один алгоритм, если в приведенный выше псевдокод внести следующие изменения:

```

строка 5   К:=1 для БПФ, К:=-1 для ОБПФ
строка 80   ТНЕТА:=К*2*PI*R/N
строка 145   IF К=-1 DO

```



```
строка 146   X(BOTVAL) := X(BOTVAL) / N  
строка 147   X(TOPVAL) := X(TOPVAL) / N  
строка 148   END DO
```

3.7. Реализация БПФ

Теперь очевидно, что, в принципе, для вычисления БПФ и ОБПФ создается массив данных и с этими данными выполняются действия, определяемые алгоритмами БПФ и ОБПФ (в том числе и алгоритмом обращения битов). Однако остается еще несколько моментов, которые следует обсудить. До сих пор мы пренебрегали эффектами нарушения непрерывности, которые встречаются на краях последовательности данных, а также такими явлениями, которые называют *наложением* и *“частотолом”*. Чтобы хорошо оценить реальный спектр данных, необходимо учитывать эти эффекты, пользуясь методами, описанными в главе 11. Еще один аспект — до сих пор внимание в данной главе уделялось двоичному алгоритму временной децимации, но существуют и другие алгоритмы, в том числе и алгоритм частотной децимации (ЧД). Некоторые из этих пунктов будут обсуждаться далее.

3.7.1. БПФ с частотной децимацией

В разделе 3.5 описан алгоритм быстрого преобразования Фурье с временной децимацией, который заключался в многократном делении исходного дискретного преобразования Фурье в виде уравнения (3.43) на два преобразования, одно из которых состоит из четных членов, а другое — из нечетных, до тех пор, пока исходное преобразование не будет разложено на двухточечные преобразования исходных данных. Альтернативный подход состоит в разделении исходного преобразования на два, одно из которых содержит первую половину данных, а другое — вторую. Получающийся в результате алгоритм частотной децимации впервые был предложен в работе [12], хотя почему-то его часто называют алгоритмом Сэнда–Тьюки.

3.7.2. Сравнение алгоритмов ВД и ЧД

В алгоритме ВД порядок входных данных остается неизменным, но порядок выходной последовательности БПФ определяется обращенным порядком битов. Оба алгоритма (ВД и ЧД) плоские. При повторном их применении можно сохранить порядок как входных, так и выходных данных, но суммарные алгоритмы уже не будут плоскими, и для их выполнения нужна дополнительная память (см. главу 12). Количество операций комплексного умножения, необходимых для выполнения каждого алгоритма, одинаково. В целом, разница между ними небольшая.

3.7.3. Модификации для увеличения скорости

Несколько модернизировав алгоритм ВД, можно еще больше увеличить скорость вычислений. Например, для снижения количества операций комплексного умножения почти в два раза можно воспользоваться не БПФ по основанию 2, а БПФ по основанию 4. Количество операций сложения при этом также уменьшается. Кроме того, скорость можно увеличить, убрав ненужные операции умножения на коэффициент W_N (который часто называют *ленивым* коэффициентом!), если $W_N = \pm 1$ или $\pm i$. При этом также уменьшается число операций сложения. Для реализации этого при $W_N = \pm 1$ или $\pm i$ добавляется отдельная “бабочка”. В результате, например, получается двоичный плоский алгоритм ВД с двумя “бабочками”. Дополнительно можно сократить время вычислений, если вначале найти синусоидальные и косинусоидальные части W_N , а затем сохранить их значения в справочной таблице. Возможны и другие усовершенствования. Итак, ясно, что существует не одно БПФ, а много. Чтобы осветить эту тему полностью, нужны значительные затраты времени и описание громоздких математических аппаратов. Также неплохо было бы разбираться в современной теории матриц, многомерном индексном отображении и теории чисел. Но такой всесторонний подход к этой теме выходит за рамки цели, поставленной в этой главе, задача которой — объяснить принципы дискретных преобразований тем способом, который будет понятен большинству людей, занимающихся цифровой обработкой сигналов. Существует ряд других книг, например, [3] и [5] с более специализированным подходом, и именно они и рекомендуются читателям для дальнейшего изучения. Как уже говорилось, есть множество алгоритмов, кроме того, специалисты могут написать собственные алгоритмы, наиболее подходящие для данного приложения. Однако сокращение числа необходимых операций сложения и умножения не всегда приводит к пропорциональному увеличению скорости вычислений. Результатом снижения количества операций умножения может даже оказаться увеличение программного кода или числа операций сложения. Если используются специальные процессоры обработки сигналов, то они могут налагать свои собственные ограничения, сводящие на нет все попытки усовершенствования алгоритма. Поскольку коэффициент увеличения скорости редко превышает 2, авторы склонны рекомендовать для универсальных задач, в которых скорость не так критична, основное двоичное БПФ с ВД. В книге [5] представлено большое количество программ для вычисления БПФ на языке FORTRAN. Даются также программы, которые можно реализовать аппаратным способом. В приложении 3А рассматривается программа на языке С для вычисления двоичного БПФ с ВД.

3.8. Другие дискретные преобразования

Существует множество других преобразований. Преобразование Винограда–Фурье ([26]; см. также [3, 5, 15, 19, 22]) и алгоритм первоначального множителя ([3], см. также [15]) представляют собой оригинальные, но слишком сложные методы повышения скорости вычисления БПФ. Дискретное косинус-преобразование особенно подходит для сжатия данных (см. раздел 3.8.1). При преобразовании Уолша (раздел 3.8.2) сигнал

раскладывается на прямоугольные импульсы, а не на синусоиды, и оно вычисляется быстрее, чем БПФ. Преобразование Адамара (раздел 3.8.3), построенное с помощью перестановки последовательности Уолша, вычисляется еще быстрее. Хотя для некоторых целей преобразования Уолша и Адамара позволяют получить определенные преимущества, они имеют ряд недостатков, которые ограничивают область их применения (см. разделы 3.8.2 и 3.8.3). Наконец, преобразование Хаара особенно полезно для определения краев при обработке изображений [20] и в подобных приложениях. Вообще, стоит порекомендовать книгу [3] — хороший источник информации для начинающих, которые хотели бы больше знать о преобразованиях и их применении. В 1990-х годах возрос интерес к вейвлетному преобразованию [4, 6, 10, 11], поэтому оно также представлено ниже, в разделе 3.8.4.

3.8.1. Дискретное косинус-преобразование

Помимо того, что преобразования используются для ускорения вычисления корреляции и свертки, а также в спектральном анализе, эти методы применяются еще и для сжатия данных при, например, передаче речи или видеосигналов, а также для записи медицинских сигналов, таких как сигналы ЭКГ или ЭЭГ. Кроме того, они используются при распознавании шаблонов. В названных приложениях задействованы только самые значительные компоненты преобразований. В результате этого снижается количество битов, необходимое для их кодирования, что позволяет ускорить передачу, использовать линии передачи с более узкой полосой, а также облегчает распознавание шаблонов (благодаря уменьшению объема информации). Эти три немаловажных признака преобразования и определяют его эффективность с точки зрения сжатия данных, которая связана с концентрацией энергии на низких частотах, простотой вычислений и минимальной среднеквадратической ошибкой. С этой точки зрения идеальным преобразованием является преобразование Карунена–Лоэва, но его нельзя задать в виде алгоритма.² Впрочем, дискретное косинус-преобразование (ДКП) обладает фактически теми же свойствами, и его можно представить в виде алгоритма. По сути, оно представляет собой действительную часть ДПФ. Это определение справедливо, поскольку ряд Фурье из действительных и четных функций содержит только косинусоидальные члены, а при использовании, например, дискретных значений напряжения данные действительные, и их можно сделать симметричными — удвоить путем прибавления к ним их зеркального отображения. Итак, ДПФ задается следующим образом (уравнение (3.41)):

$$X(k) = \sum_{n=0}^{N-1} x_n e^{-2\pi i n k / N}, k = 0, 1, \dots, N-1.$$

Определив ДКП $X_c(k)$ как действительную часть этого преобразования, получим

$$X_c(k) = \operatorname{Re}[X(k)] = \sum_{n=0}^{N-1} x_n \cos\left(\frac{k2\pi n}{N}\right), k = 0, 1, \dots, N-1. \quad (3.66)$$

²См. книгу У. Прэтта. *Цифровая обработка сигналов*, т. 2. Там показано, что можно. Правда, вычислительная сложность алгоритма Карунена–Лоэва еще выше, чем у неоптимизированного ДПФ. — *Прим. ред.*

Это одна из нескольких форм ДКП. Более общий вид (см. [1, 3, 27]):

$$\begin{aligned} X_c(k) &= \frac{1}{N} \sum_{n=0}^{N-1} x_n \cos \left(\frac{k2\pi n + k\pi}{2N} \right) = \\ &= \frac{1}{N} \sum_{n=0}^{N-1} x_n \cos \left[\frac{k\pi(2n+1)}{2N} \right], \quad k = 0, 1, \dots, N-1. \end{aligned} \quad (3.67)$$

Существуют также и другие формы (см. [2, 17, 18, 27]).

Разработаны реализации ДКП, основанные на БПФ [16], кроме того, есть также быстрое ДКП, скорость которого в шесть раз выше указанных преобразований [7]. Еще одна версия — это С-матричное преобразование, которое намного проще реализовать на аппаратном уровне [23].

3.8.2. Преобразование Уолша

Преобразования, обсуждавшиеся выше, были основаны на функциях косинус и синус. Намного проще и быстрее считаются преобразования, основанные на импульсоподобных сигналах, которые принимают значения только ± 1 . Кроме того, они больше подходят для описания сигналов с нарушением непрерывности, которые встречаются, например, в изображениях. И, наоборот, они менее пригодны для описания непрерывных сигналов и могут не быть инвариантными по фазе, а если это так, полученный спектр может искажаться. Поэтому такие сигналы обычно используются при обработке изображений (астрономия и спектроскопия), кодировании сигналов и фильтрации.

Точно так же, как ДПФ основывается на наборе гармонических косинусоидальных и синусоидальных сигналов, дискретное преобразование Уолша (ДПУ) основано на наборе гармонических прямоугольных импульсов, которые называются *функциями Уолша*. Однако для прямоугольных импульсов частота не определена, поэтому используется аналоговый термин “последовательность”. *Последовательность* — это половина среднего числа переходов через нуль за единицу времени. На рис. 3.6 показаны функции Уолша до порядка $N = 8$, расположенные по возрастанию. В таком виде их называют *упорядоченными по Уолшу*. Функция Уолша со временем t и порядком n обозначается $WAL(n, t)$. Внимательно изучая рис. 3.6, видим, что существует равное количество четных и нечетных функций Уолша, точно так же, как и косинусоидальных и синусоидальных компонентов ряда Фурье. Четные функции $WAL(2k, t)$ записываются как $CAL(k, t)$, а нечетные функции $WAL(2k + 1, t)$ записываются как $SAL(2k + 1, t)$, где $k = 1, 2, \dots, N/2 - 1$.

Любой сигнал $f(t)$ можно разложить по набору функций Уолша (аналог разложения в ряд Фурье) как

$$f(t) = a_0 WAL(0, t) + \sum_{i=1}^{N/2-1} \sum_{j=1}^{N/2-1} [a_i SAL(i, t) + b_i CAL(j, t)], \quad (3.68)$$

где a_i и b_j — коэффициенты ряда.

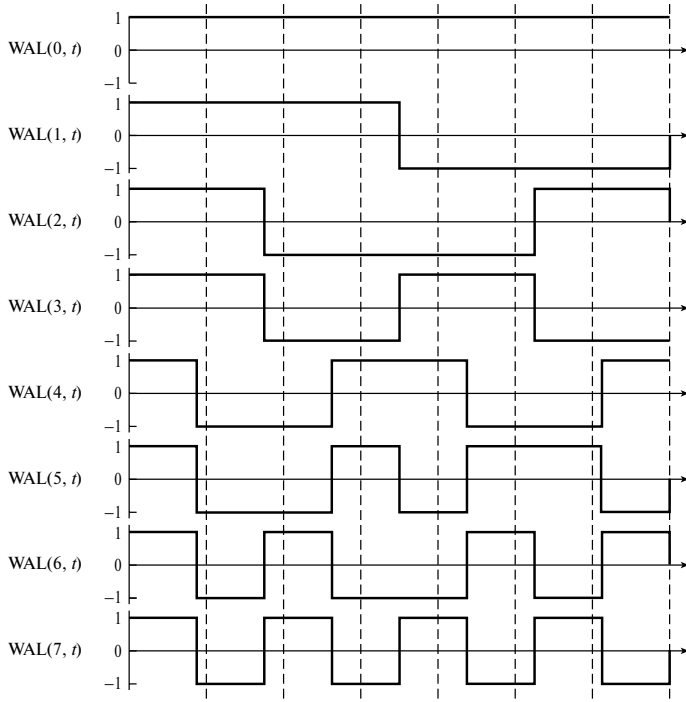


Рис. 3.6. Упорядоченные по возрастанию функции Уолша до $n = 7$, которые показывают времена дискретизации для матрицы преобразования Уолша порядка 8×8

Для любых двух функций Уолша

$$\sum_{t=0}^{N-1} \text{WAL}(m, t) \text{WAL}(n, t) = \begin{cases} N & \text{при } n = m \\ 0 & \text{при } n \neq m \end{cases},$$

т.е. функции Уолша ортогональны.

Пара дискретных Уолш-образов — это

$$X_k = \frac{1}{N} \sum_{i=0}^{N-1} x_i \text{WAL}(k, i) \quad k = 0, 1, \dots, N-1 \quad (3.69)$$

и

$$x_i = \sum_{k=0}^{N-1} X_k \text{WAL}(k, i) \quad k = 0, 1, \dots, N-1, \quad (3.70)$$

где следует заметить, что, не считая множителя $1/N$, обратное преобразование идентично прямому, и $\text{WAL}(k, i) = \pm 1$. Следовательно, пару образов можно найти простым

перемножением матриц с помощью цифровых методов. Однако отсутствие фазовой инвариантности означает, что ДПУ не подходит для вычисления быстрых корреляций или свертков.

Уравнение (3.69) показывает, что k -й элемент ДПУ можно получить, умножив каждый дискретный элемент сигнала x_i на функцию Уолша последовательности k и просуммировав по $k = 0, 1, \dots, N - 1$. Для всех элементов k это можно записать в матричном виде как

$$\mathbf{X}_k = \mathbf{x}_i \mathbf{W}_{ki}, \quad (3.71)$$

где $\mathbf{x}_i = [x_0 x_1 x_2 \dots x_{N-1}]$ — последовательность данных,

$$\mathbf{W}_{ki} = \begin{bmatrix} W_{01} & W_{02} & \dots & W_{0,N-1} \\ W_{11} & & & \\ \dots & & & \\ W_{N-1,1} & W_{N-1,2} & \dots & W_{N-1,N-1} \end{bmatrix} —$$

матрица преобразования Уолша, а $\mathbf{X}_k = [X_0 X_1 \dots X_{N-1}]$ — $(N - 1)$ компонентов ДПУ. Заметим, что $\mathbf{W}_{ki}$ — это матрица порядка $N \times N$, где N — количество заданных точек, т.е. точек дискретного сигнала. Следовательно, если есть N точек данных, то нужно рассматривать первые N упорядоченных функций Уолша. Каждая из них дискретизируется N раз, при этом k -я строка матрицы $\mathbf{W}_{ki}$ соответствует N дискретным значениям k -го компонента последовательности.

Пример 3.6

В качестве примера найдем ДПУ последовательности данных $(1, 2, 0, 3)$. Она состоит из $N = 4$ точек, так что $\mathbf{W}_{ki}$ — это матрица порядка 4×4 , которая получается из первых четырех строк рис. 3.6:

$$\mathbf{W}_{ki} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix}. \quad (3.72)$$

Следовательно, из уравнения (3.71) $\mathbf{X}(k)$ задается как

$$\mathbf{x}(k) = \frac{1}{4} \begin{bmatrix} 1 & 2 & 0 & 3 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 6 & 0 & 2 & -4 \end{bmatrix},$$

так что $X_0 = 1,5$, $X_1 = 0$, $X_2 = 0,5$ и $X_3 = -1$. Это вычисляется значительно проще, чем соответствующее ДПФ! Не говоря уже о том, что существуют быстрые ДПУ (БДПУ).

Дополнительно можно рассчитать соответствующий спектр с мощностными компонентами, которые задаются как

$$P(k) = [|CAL(k, t)|^2 + |SAL(k, t)|^2]^{1/2},$$

где

$$\begin{aligned} P(0) &= X_c^2(0), \\ P(k) &= X_c^2(k, t) + X_s^2(k, t), \\ P\left(\frac{N}{2}\right) &= X_s^2\left(\frac{N}{2}, t\right), \end{aligned} \quad (3.73)$$

где $k = 1, 2, \dots, N/2 - 1$, и фазовыми компонентами

$$\begin{aligned} \phi(0) &= 0, \pi, \\ \phi(k) &= \arctg \left[\frac{X_s(k)}{X_c(k)} \right], k = 1, 2, \dots, N/2 - 1, \end{aligned} \quad (3.74)$$

и

$$\phi\left(\frac{N}{2}\right) = 2k\pi \pm \pi/2, k = 0, 1, 2, \dots$$

Итак, для вышеописанного ДПУ мы получили

$$\begin{aligned} P(0) &= 1, 5^2 = 2, 25; \phi(0) = 0, \pi, \\ P(1) &= 0^2 + 0, 5^2 = 0, 25; \phi(1) = \arctg \left(\frac{0}{0, 5} \right) = 0, \\ P(2) &= (-1)^2 = 1; \phi(2) = \frac{\pi}{2} + 2k\pi, k = 0, 1, 2. \end{aligned}$$

3.8.3. Преобразование Адамара

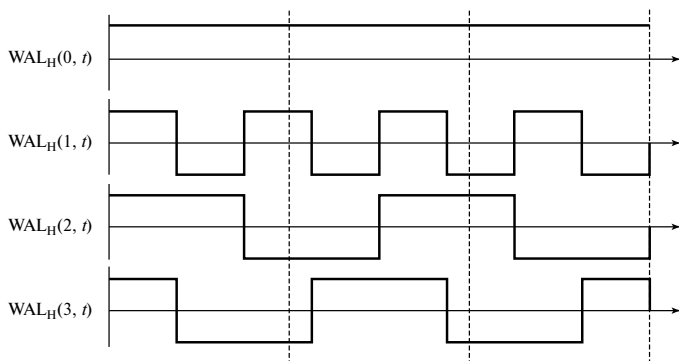
Преобразование Адамара, или преобразование Уолша–Адамара, — это, по сути, то же преобразование Уолша, но с другим порядком функций Уолша и, следовательно, строк матрицы преобразования. Получающаяся при такой перестановке матрица Адамара содержит подмассивы матриц второго порядка. На рис. 3.7 показана матрица Адамара порядка 8×8 (обозначается как ${}^8\mathbf{H}$). Видно, что ее можно записать через матрицы

$${}^2\mathbf{H} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad \text{и} \quad -{}^2\mathbf{H} = \begin{bmatrix} -1 & -1 \\ -1 & 1 \end{bmatrix}.$$

Любую матрицу Адамара порядка $2N$ можно рекурсивно получить из ${}^2\mathbf{H}$ как

$${}^{2N}\mathbf{H} = \begin{bmatrix} {}^N\mathbf{H} & {}^N\mathbf{H} \\ {}^N\mathbf{H} & -{}^N\mathbf{H} \end{bmatrix}. \quad (3.75)$$

		$i \rightarrow$							
		0	1	2	3	4	5	6	7
$k \downarrow$	0	1	1	1	1	1	1	1	1
	1	1	-1	1	-1	1	-1	1	-1
	2	1	1	-1	-1	1	1	-1	-1
	3	1	-1	-1	1	1	-1	-1	1
	4	1	1	1	1	-1	-1	-1	-1
	5	1	-1	1	-1	-1	1	-1	1
	6	1	1	-1	-1	-1	-1	1	1
	7	1	-1	-1	1	-1	1	1	-1

Рис. 3.7. Матрица преобразования Адамара порядка 8×8 Рис. 3.8. Функции Уолша, расположенные в порядке Адамара, до $n = 7$, показывающие времена дискретизации для матрицы преобразования Адамара порядка 4×4

Из этого рекурсивного свойства следует, что получающееся при расположении функций Уолша в порядке, определенном Адамаром, быстрое преобразование Уолша–Адамара можно вычислить намного быстрее, чем ДПУ. Расположенные в порядке Адамара (или в естественном порядке) функции Уолша показаны на рис. 3.8. Последовательность Адамара получается из последовательности Уолша следующим образом:

- а) порядок расположенных по Уолшу функций Уолша представляется в двоичной системе;
- б) биты переставляются в обратном порядке;
- в) двоичные значения преобразуются в код Грея;
- г) полученные значения преобразуются в десятичные.

Пример 3.7

В качестве примера рассчитаем дискретное преобразование Уолша–Адамара (ДПУА) последовательности (1, 2, 0, 3). Матрица Адамара порядка 4×4 $\mathbf{H}_{ki}$ равна

$$\mathbf{H}_{ki} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \quad (3.76)$$

согласно свойствам матрицы Адамара. Следовательно, ДПУА-образ последовательности (1, 2, 0, 3) задается как

$$\mathbf{x}_k^{WH} = \frac{1}{4} \begin{bmatrix} 1 & 2 & 0 & 3 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 6 & -4 & 0 & 2 \end{bmatrix},$$

так что $X_0^{WH} = 1,5$, $X_1^{WH} = -1$, $X_2^{WH} = 2$ и $X_3^{WH} = 0,5$. Значения этих величин такие же, как и рассчитанные ранее компоненты преобразования Уолша, только стоят в другом порядке.

3.8.4. Вейвлетное преобразование

Физический принцип неопределенности Гейзенберга говорит о том, что нельзя одновременно точно знать и положение x частицы, и ее импульс p . Фактически

$$xp \geq h = 6,626 \times 10^{-34} \text{ Дж с}, \quad (3.77)$$

где h — постоянная Планка. С помощью уравнения Эйнштейна $E = mc^2$ этот принцип можно перенести в область обработки сигналов, где он будет формулироваться так: нельзя одновременно с любой точностью определить время и частоту. Следовательно,

$$\Delta f T \geq 1, \quad (3.78)$$

где Δf и T описывают разрешение по частоте и по времени. Если разрешение по времени высокое, то частота будет определяться менее точно, и наоборот. Следовательно, может оказаться достаточно сложно одновременно измерить с необходимой степенью точности частоту компонента сигнала и время его появления или разделить во времени различные частотные компоненты. Это может произойти, если сигнал содержит кратковременные высокочастотные компоненты, расположенные слишком близко к более продолжительным компонентам во временной области, которые также близко расположены в частотной области и возникают в различные моменты времени. Такие сигналы непериодичны. Для решения этой общей проблемы частотно-временного анализа используется вейвлетное преобразование (wavelet transform), которое предоставляет средства для анализа нестационарных сигналов. Вейвлетное преобразование применяется

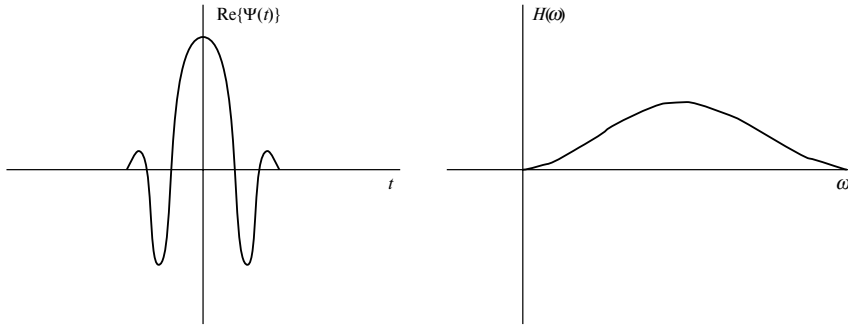


Рис. 3.9. Модифицированная гауссова, или морлетовская, материнская вейвлетная функция $\Psi(t)$ и ее Фурье-образ $H(\omega)$

также для фильтрации сигналов, устранения шумов, определения местонахождения сингулярностей и их распределения.

В то время как в преобразованиях Фурье значения сигнала имеют весовые коэффициенты, в показателе степени которых стоит мнимая часть, а аргумент — гармонический и зависит от частоты, т.е., по сути, является синусоидальным членом, в вейвлетном преобразовании в качестве весовых коэффициентов значений сигнала выступают вейвлетные функции.

Все вейвлетные функции получаются из основной (материнской, базовой) вейвлетной функции. Существует ряд возможных материнских функций, выбранных для получения следующих свойств. Они должны: осциллировать, не содержать компонентов постоянного тока, быть полосовыми, быстро спадать во времени до нуля и быть обратимыми. Последнее свойство гарантирует, что вейвлетное преобразование сигнала будет однозначным. Основную функцию можно записать как $\Psi(t)$. Например, морлетовская, или модифицированная гауссова материнская вейвлетная функция (*вейвлет Морле*), — это

$$\Psi(t) = e^{i\omega_0 t} e^{-t^2/2}, \quad (3.79)$$

Фурье-образ которой —

$$H(\omega) = \sqrt{2\pi} e^{-(\omega - \omega_0)^2/2}. \quad (3.80)$$

Эти два сигнала изображены на рис. 3.9, и видно, что $\Psi(t)$ удовлетворяет вышеизложенным требованиям, т.е. осциллирует и спадает до нуля.

Остальные (дочерние) функции получают путем такого изменения масштаба материнской, чтобы образовалось семейство функций. Каждую дочернюю функцию можно записать как

$$\frac{1}{\sqrt{a}} \Psi\{(t - \tau)/a\},$$

где a — переменный коэффициент масштабирования, а τ — константа переноса. Если масштаб a увеличивается, то амплитуда и аргумент функции уменьшаются. Уменьше-

ние аргумента при заданной амплитуде означает, что уменьшается частота. Следовательно, увеличение масштаба a соответствует уменьшению частоты, и поэтому функция расширяется во временной области по горизонтали. Положительные значения константы переноса приводят к переносу функции вдоль положительной временной оси. Итак, с помощью коэффициента масштабирования a и константы переноса τ можно создавать функции с большими или меньшими амплитудами, с высшими или низшими частотами и размещать их в различные моменты времени. Таким образом, нестационарные сигналы с различными частотными компонентами, расположенными в различных промежутках времени, можно описывать как сумму различных вейвлетных функций. Для этого и используется вейвлетное преобразование.

Есть несколько вариантов вейвлетного преобразования, определения которых представлены ниже. Они образуют естественную прогрессию с возрастающей дискретизацией. В этих определениях предполагается, что сигнал $s(t)$ квадратично интегрируем, т.е.

$$\int s^2(t)dt < \infty. \quad (3.81)$$

Это предположение справедливо для всех кратковременных сигналов с конечной амплитудой. Синусоидальные сигналы и сигналы постоянного тока не удовлетворяют этому условию, поэтому они исключаются из последующих рассуждений.

Непрерывное вейвлетное преобразование НВП(a, τ) можно определить как

$$\text{НВП}(a, \tau) = (1/\sqrt{a}) \int s(t) \Psi\{(t - \tau)/a\} dt. \quad (3.82)$$

Параметры этого уравнения можно дискретизовать, что даст вейвлетное преобразование с дискретными параметрами ВПДП(m, n), которое определяется как

$$\text{ВПДП}(m, n) = a_0^{-m/2} \int s(t) \Psi\{(t - n\tau_0 a_0^m)/a_0^m\} dt, \quad (3.83)$$

где были сделаны такие замены: $a = a_0^m$, $\tau = n\tau_0 a_0^m$. В этих заменах a_0 и τ_0 — интервалы дискретизации для a и τ , а m и n — целые числа. Довольно часто выбирают $a_0 = 2$, а $\tau_0 = 1$. Тогда

$$\begin{aligned} \text{ВПДП}(m, n) &= 2^{-m/2} \int s(t) \Psi\{(t - n2^m)/2^m\} dt = \\ &= 2^{-m/2} \int s(t) \Psi\{2^{-m}t - n\} dt. \end{aligned} \quad (3.84)$$

Это расширяет временную ось в 2^{-m} раза, а вейвлетная функция переносится в положительную сторону по времени на $2^m n$.

Дискретизация по времени дает вейвлетное преобразование с дискретным временем ВПДВ(m, n), которое определяется как

$$\text{ВПДВ}(m, n) = a_0^{-m/2} \sum_k s(k) \Psi(a_0^{-m}k - n\tau_0). \quad (3.85)$$

Если снова положить $a_0 = 2$, а $\tau_0 = 1$, то ИМДВ(m, n) станет таким:

$$\text{ВПДВ}(m, n) = 2^{-m/2} \sum_k s(k) \Psi(2^{-m}k - n), \quad (3.86)$$

что известно как *дискретное вейвлетное преобразование*. Итак, дискретное вейвлетное преобразование получается из непрерывного вейвлетного преобразования путем дискретизации масштабного параметра a , параметра переноса τ и времени с последующей подстановкой $a_0 = 2$ и $\tau_0 = 1$.

Кроме того что вейвлетные преобразования используются для изучения частотно-временного содержания сигналов, их можно использовать для фильтрации сигналов, т.е. для удаления какой-то части присутствующего шума. Сперва сигналы раскладывают на их компоненты. Затем идентифицируются и устраняются компоненты шума. Наконец, очищенный от шумов сигнал восстанавливают по вейвлетным функциям компонентов. При использовании непрерывного вейвлетного преобразования формула для восстановления (обратного преобразования) имеет вид

$$s(t) = \frac{1}{C_\Psi} \int_{-\infty}^{\infty} \int_{a>0} \text{НВП}(a, \tau) \left\{ \frac{1}{\sqrt{a}} \right\} \Psi\{(t - \tau)/a\} \left\{ \frac{1}{\sqrt{a^2}} \right\} da d\tau, \quad (3.87)$$

где

$$C_\Psi = \int_0^\infty \{|H(\omega)|^2 / \omega\} d\omega < \infty,$$

а $H(\omega)$ — Фурье-образ основного импульса $\Psi(t)$.

Читателям, которым нужно больше информации о восстановительных формулах, рекомендуется обратиться к работам [4, 6, 8, 10, 11, 14].

Для более глубокого понимания вейвлетного преобразования рассмотрим возможные интерпретации непрерывного вейвлетного преобразования (формула (3.82)). Видно, что НВП(a, τ) описывает взаимную корреляцию $s(t)$ и $\Psi(t/a)/\sqrt{a}$ с задержкой $-\tau/a$. Точно такой же операцией описывается взаимная корреляция масштабированного сигнала $s(at)$ и $\sqrt{a}\Psi(t)$ с задержкой $-\tau/a$. НВП(a, τ) может также описывать выход полосового фильтра с импульсной характеристикой НВП(a, τ) и временем τ/a при входном сигнале $s(t)$. Это же преобразование описывает выход полосового фильтра с импульсной характеристикой $\sqrt{a}\Psi(-t)$ со временем τ/a при входном сигнале $s(at)$. Таким образом, вейвлетное преобразование можно рассматривать как действие полосового фильтра или взаимную корреляцию. Отметим также, что изменение масштаба a позволяет отфильтровывать различные частотные компоненты.

3.8.5. Вейвлетный метод кратномасштабного анализа

Кратномасштабным анализом (КА) называют разделение компонентов сигнала на ряд частотных полос. Оно осуществляется с помощью фильтров высоких и низких частот и субдискретизации. Возможна и обратная процедура, которая позволяет восстанавливать сигнал. АПР выполняется с помощью дискретного вейвлетного преобразования и дискретных сигналов. Он применяется при анализе информационного содержания

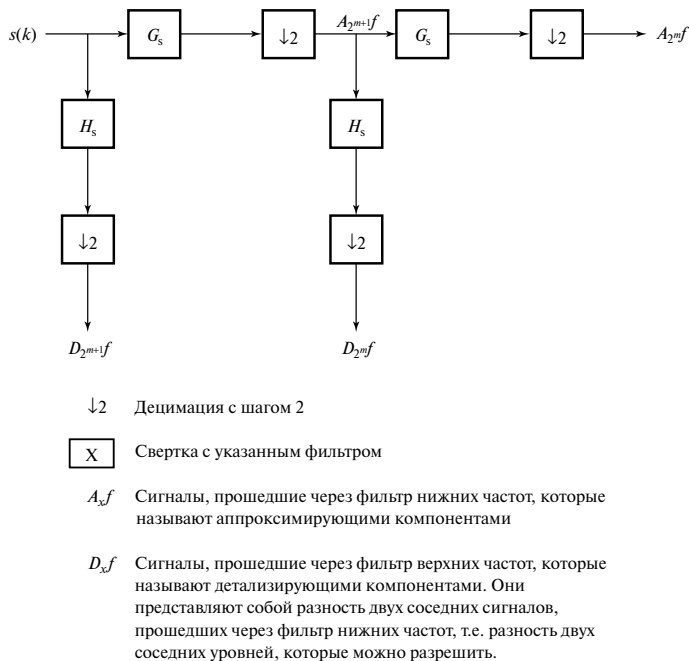


Рис. 3.10. Разложение сигнала для КА вейвлетным методом

изображений [13] и наведенных потенциалов электроэнцефалограммы [25]. В работе [13] описывается вейвлетный вариант КА. Положив $a = 2^m$ и $\tau = 2^m n$, можно создать фильтр нижних частот $G_s(\omega)$ и фильтр верхних частот $H_s(\omega)$, которые при многократном использовании в пирамидальной структуре разделяют сигнал на частотные полосы с возрастающей разрешающей способностью. Этот пирамидальный алгоритм показан на рис. 3.10. $G_s(\omega)$ и $H_s(\omega)$ симметричны соответственно фильтру низких частот $G(\omega)$ и фильтру высоких частот $H(\omega)$. Это квадратурные зеркальные фильтры, связанные с подходящей вейвлетной функцией. Для восстановления сигнала воспользуйтесь пирамидальным алгоритмом, показанным на рис. 3.11.

КА можно использовать для изучения компонентов сигнала, а также для фильтрации сигналов. После разложения сигнала нежелательные компоненты убираются, а отфильтрованный сигнал восстанавливается.

На рис. 3.12 показаны характерные результаты применения КА для извлечения потенциала CNV, зависящего от событий, из зашумленного фона электроэнцефалограммы (ЭЭГ), отношение сигнал-шум которой равно $-14,3$ дБ. При моделировании задавались определенные переменные потенциалы, которые прибавлялись к ЭЭГ. К данным эксперимента применялся и адаптивный КА [21], и его модифицированная версия. Видно, что сигнал CNV без шума совершенно не похож на реальный потенциал CNV. Поскольку отношения сигнал-шум относительно больших потенциалов, зависящих от событий, составляют приблизительно -15 дБ, из результатов видно, что описанные методы больше подходят для очистки от шумов сигналов с намного большими отношениями сигнал-шум.

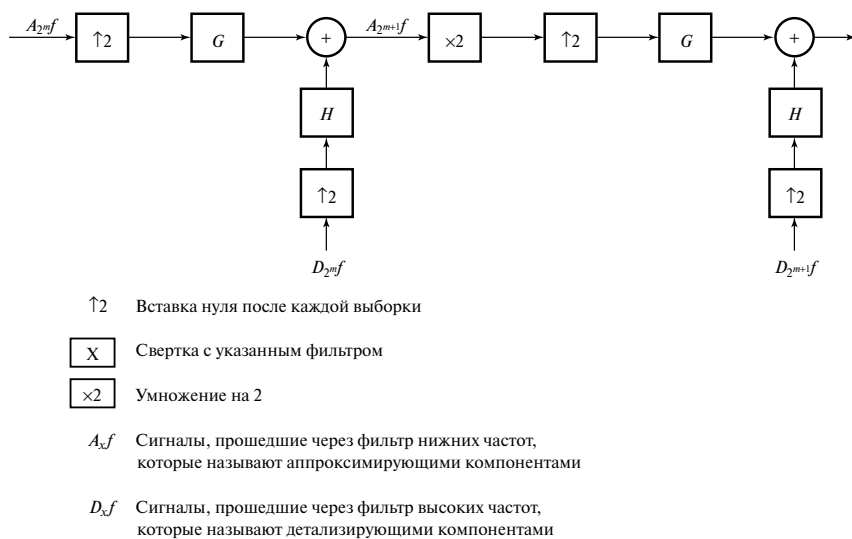


Рис. 3.11. Восстановление сигнала с помощью вейвлетного КА

3.8.6. Представление сигнала с помощью сингулярностей: метод вейвлетного преобразования

В работе [14] было показано, что все сигналы и шумы можно полностью описать через их сингулярности. Сингулярности определяются через их показатели степени Липшица. Если некоторая функция $f(t)$ непрерывно дифференцируема в точке t_0 , то она не сингулярна и относится к типу Липшиц-1. Сигналы, которые не относятся к типу Липшиц-1, — сингулярны. Если $f(t)$ дифференцируема в точке t_0 n раз, то ее n -я производная сингулярна и относится к типу Липшиц- α , где $\alpha > n$. Показатели степени Липшица могут быть также отрицательными и показывают величину и знак сингулярности. Описывая сигналы через их сингулярности, а затем убирая нежелательные сингулярности, по оставшимся сингулярностям можно восстанавливать отфильтрованные сигналы. Этот метод применяется, например, при определении краев на изображениях [14] и фильтрации сигналов от шумов [14, 28]. *Фильтрацией* сигнала от шумов называется устранение шумов из сигнала, позволяющее увеличить отношение сигнал-шум.

Многие сигналы имеют сингулярности с положительными показателями Липшица, тогда как шум характеризуется отрицательными показателями Липшица. Если сингулярности, связанные с шумами, можно найти и устранить, то это, следовательно, позволяет отделять сигналы от шума. В работе [14] показано, что сингулярности и их положение могут характеризоваться амплитудами и знаками максимумов вейвлетных преобразований на графиках их зависимости от времени в разных масштабах, как показано на рис. 3.13. Следовательно, каждый максимум на графике соответствует сингулярности. Сравнивая максимумы при различных масштабах, можно определить те из них, которые связаны с шумами, и найти значения показателей Липшица.

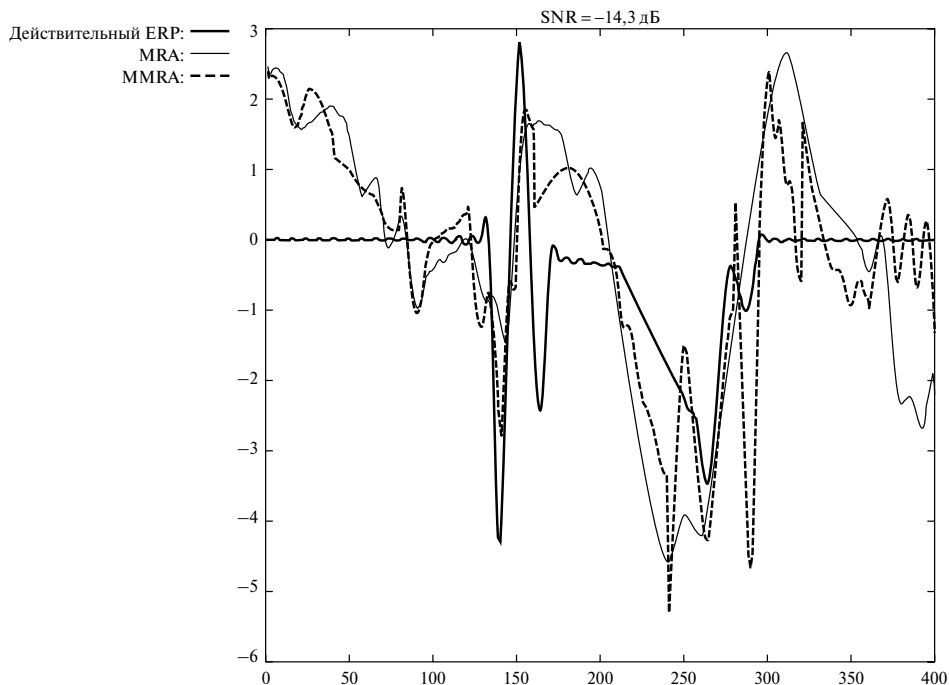


Рис. 3.12. Реальные потенциалы ERP и потенциалы без шума, извлеченные методом анализа с переменной разрешающей способностью при отношении сигнал-шум в $-14,3$ дБ

В дальнейшем предполагается, что у сигналов нет локальных осцилляций. Если бы они были, анализ был бы намного сложнее (см. [14]).

Рассмотрим для начала сигналы, значения максимумов вейвлетного преобразования которых лежат в пределах конуса $|(t - t_0)| \leq Ca$, где C — произвольная константа (рис. 3.13). Теперь рассмотрим те максимумы внутри конуса, которые лежат на линии, соединяющей различные масштабы (*линия максимумов*). Тогда вейвлетные преобразования $W(a, \tau)$, связанные с максимумами на линии максимумов, меняются с изменением масштаба a как

$$|W(a, \tau)| \leq Aa^\alpha, \quad (3.88)$$

где A — константа, $a > 0$ и $0 < \alpha < 1$.

Следовательно, с увеличением a модуль вейвлетного преобразования увеличивается. Другими словами, модули увеличиваются при переходе к более низкочастотному масштабу. Таким образом, энергетическое содержание сигнала увеличивается на более низких частотах или (эквивалентно) уменьшается на более высоких частотах. Прологарифмировав обе части уравнения (3.88), получим уравнение (3.89):

$$\log |W(a, \tau)| \leq \log A + \alpha \log a. \quad (3.89)$$

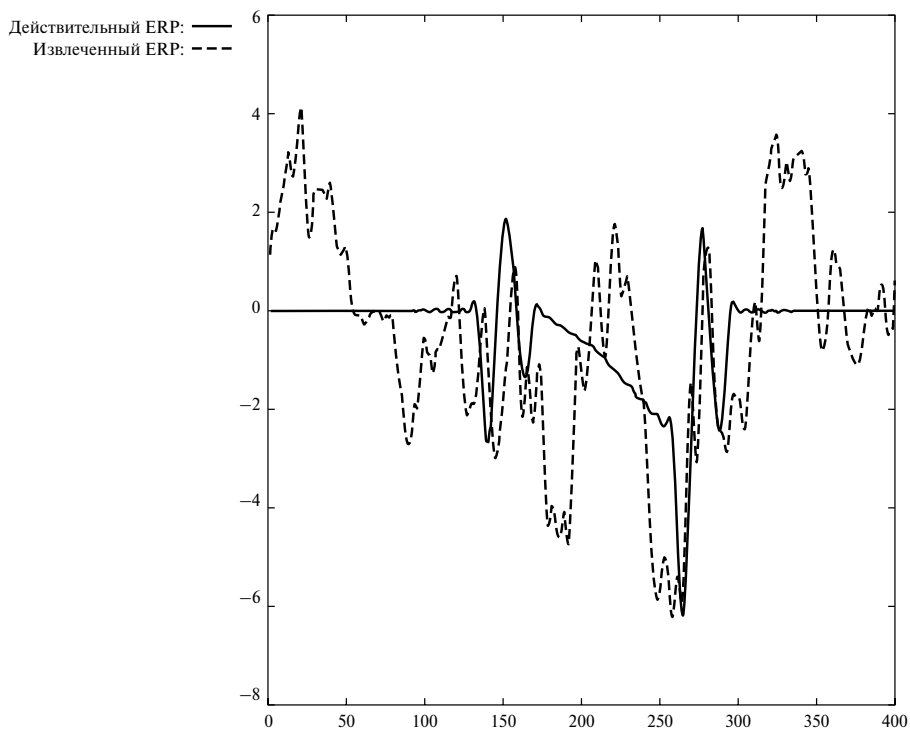


Рис. 3.14. Потенциал ERP, очищенный методом определения сингулярностей, и реальный потенциал ERP (ЭЭГ, отношение сигнал-шум $-5,3$ дБ)

увеличиваются с частотой. Отметим, что это противоположно изменениям величины сигналов с положительными показателями Липшица, причем то же справедливо и для других шумоподобных сигналов.

На рис. 3.13 (не приведен к соответствующему масштабу) схематически показаны максимумы зашумленного сигнала. Видно, что амплитуда максимумов сигнала с заданным положением увеличивается при повышении масштаба (при понижении частоты), тогда как величина максимумов шума увеличивается при понижении масштаба (при повышении частоты) и, кроме того, возрастает их количество. В общем случае, при уменьшении масштаба в два раза число максимумов шума удваивается. Таким образом, наблюдая за изменениями максимумов в одних и тех же точках при переходе от одного масштаба к другому, можно определить максимумы, связанные с шумом. Следовательно, их можно убрать и не учитывать при восстановлении чистого сигнала.

Этот метод (см. [14, 28]) применялся к модельным ERP данных ЭЭГ из раздела 3.8.5. Очищенный и реальный потенциалы ERP для эксперимента с однократным проходом показаны на рис. 3.14 при отношении сигнал-шум $-5,3$ дБ. Это достаточно большое отношение сигнал-шум по сравнению с -15 дБ, и извлеченный сигнал CNV не имеет ничего общего с реальным. Почему это так? Метод определения сингулярностей основан на предположении о белом шуме, а сигнал ЭЭГ не является белым. Если повторить моделирование, воспользовавшись белым шумом вместо ЭЭГ, то получится результат,

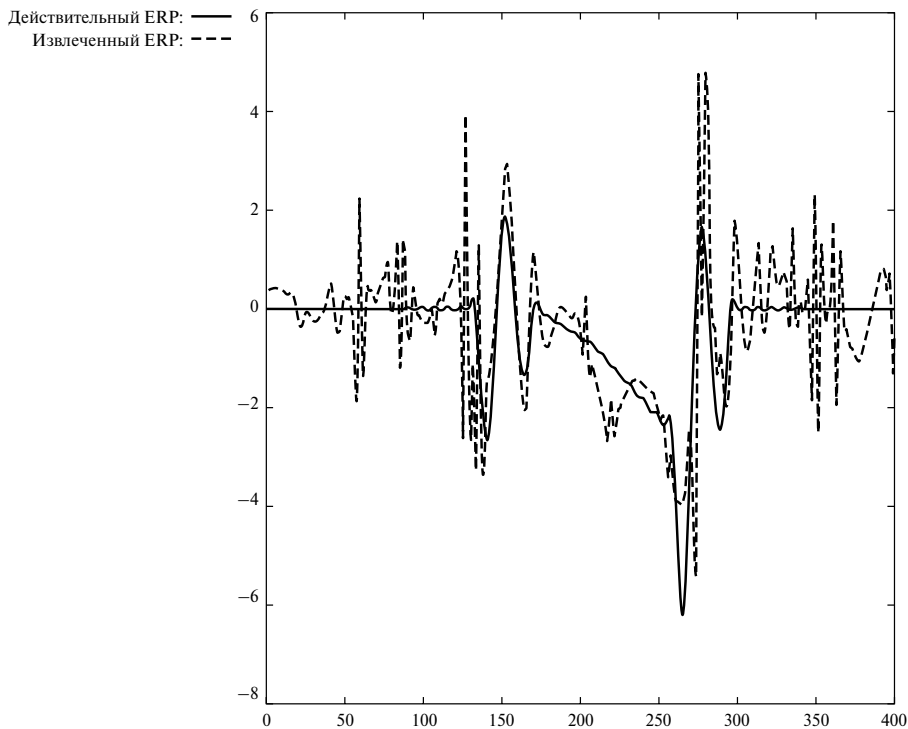


Рис. 3.15. Потенциал ERP, очищенный методом определения сингулярностей, и реальный потенциал ERP (белый шум, отношение сигнал-шум $-5, 67$ дБ)

показанный на рис. 3.15. На этот раз извлеченный сигнал CNV намного больше похож на реальный. Следовательно, для удовлетворительной работы метода шум должен быть белым.

Когда положения максимумов сигнала и шума совпадают, возникает проблема. Если максимум не ликвидировать, то шум в сигнале сохранится. Если максимум убрать, то исчезнет и часть сигнала, что приведет к восстановлению искаженного сигнала. Для решения этой проблемы можно искать высокие масштабы, где преобладает сигнал (т.е. относительно большое отношение сигнал-шум), и использовать уравнение (3.89) в пределах конуса $|(t - t_0)| \leq Ca$ для определения α . В этом случае отношение значений максимумов в одних и тех же точках при ближайших более мелких масштабах будет равно 2^α . Следовательно, в этих мелких масштабах можно найти действительные величины максимумов и использовать их при восстановлении сигнала. Разумеется, подобный конус $|(t - t_0)| \leq Ca$ можно построить для любого максимума t_0 .

Еще одно ограничение этого метода устранения шумов заключается в точности, с которой можно найти и устранить максимумы шума. Это зависит от точности определения α , которая, в свою очередь, зависит от выбора материнской вейвлетной функции и от алгоритма устранения сингулярностей (максимумов), а также от алгоритма восстановления. Более подробно об этом можно прочитать в литературе, указанной в конце этой главы.

3.9. Применение ДКП: сжатие изображений

Как говорилось в разделе 3.8.1, дискретное косинус-преобразование (ДКП) применяется для сжатия данных сигналов, а это имеет особое значение при хранении и передаче изображений, поскольку каждое изображение состоит из большого количества данных. Например, элемент (пиксель) изображения на рисунке размером 320×240 , который описывается восемью бит на пиксель, будет занимать 76,8 Кбайт, что эквивалентно приблизительно 25 страницам текста. Делаем вывод, что сжатие изображений — это очень полезная штука. Сжатые изображения, как правило, кодируют для передачи, что еще больше их сжимает. Ранее различные организации разрабатывали свои собственные подходы и стандарты, пока в 1986 году группа специалистов не организовала Объединенную группу экспертов по обработке фотоизображений (Joint Photographic Experts Group — JPEG), которая попыталась стандартизировать сжатие и передачу неподвижных черно-белых и цветных изображений. Комитет JPEG был одним из отделов Международной организации по стандартизации (International Organization for Standardization — ISO), кроме того, в него входили члены МККТТ (Международный консультативный комитет по телеграфии и телефонии, International Telegraph and Telephone Consultative Committee — ССИТТ) и МЭК (Международная электротехническая комиссия, International Electronic Commission — IEC). Стандартизация необходима для того, чтобы позволить взаимный обмен изображениями между различными устройствами, такими как персональные компьютеры, локальные сети, компакт-диски и цифровые камеры. Стандарт JPEG устанавливает набор базовых функций сжатия изображений, которые могут отличаться деталями. Есть основная структура, которой должны придерживаться все системы, но кроме этого существуют расширенные и иерархические структуры. Успех JPEG способствовал появлению MPEG (Moving Pictures Expert Group — Экспертная группа по вопросам движущегося изображения) и JBIG (Joint Bi-level Image Expert Group — Объединенная группа по вопросам двухуровневых изображений). В этом разделе мы сосредоточим внимание на некоторых основных понятиях стандарта JPEG. Полностью стандарт JPEG описан в книге [17]. Там же в качестве приложения приведены все требования и руководства, опубликованные в виде проекта международного стандарта Draft International Standard 10918-1. Краткие описания можно найти в работах [2, 18].

На рис. 3.16 представлена основная блок-схема системы сжатия стандарта JPEG для передачи данных. Для приема и восстановления сжатых данных нужна обратная система. Итак, сперва вычисляется двумерное ДКП данного изображения. Затем коэффициенты ДКП квантуются, и находится их пороговое значение. Далее коэффициенты нулевой частоты, или постоянного тока, последовательно подвергаются дифференциальной импульсно-кодовой модуляции, а получающийся в результате поток битов кодируется либо с помощью кода Хаффмана, либо с помощью арифметического кода. Остальные частотные коэффициенты (коэффициенты переменного тока) также кодируются либо с помощью кода Хаффмана, либо с помощью арифметического кода. Длинные серии нулей кодируются методом серийного кодирования. В результате получается два потока сжатых данных, состоящих из закодированных коэффициентов постоянного и переменного тока.

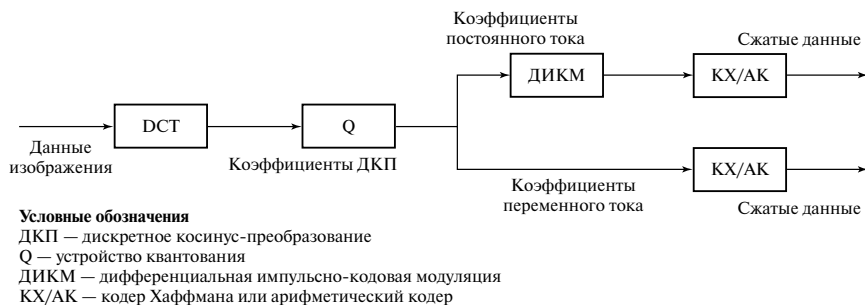


Рис. 3.16. Сжатие данных в стандарте JPEG

3.9.1. Дискретное косинус-преобразование

Любое прямоугольное изображение можно представить как массив численных значений, которые каким-то образом передают такие признаки изображения, как интенсивность, тон и цвет (более подробно см., например, [17]). Каждое из этих значений называют элементом изображения, или *пикселем*, и, как мы видели, изображение, чаще всего, состоит из очень большого их количества. Статистика изображения в различных областях может быть совсем разной, поэтому для преобразования предпочтительнее разделить изображение на ряд прилегающих друг ко другу меньших участков с более или менее похожими статистическими характеристиками. Кроме того, меньшие участки позволяют получить большее сжатие из-за большей корреляции между соседними пикселями. Более того, преобразования меньших участков вычислить намного легче. Следовательно, базовый блок стандарта JPEG состоит из квадрата размером 8×8 пикселей, следовательно, все изображение делится на блоки размером 8×8 пикселей, непосредственно соседствующих друг с другом.

Эти блоки размером 8×8 пикселей двумерны (2D), поэтому для их преобразования подходит двумерное ДКП. Его можно получить, если сперва вычислить ДКП каждой горизонтальной строки пикселей, затем заменить горизонтальные строки пикселей компонентами ДКП (горизонтальное ДКП), вычислить ДКП столбцов и заменить каждый столбец его ДКП (вертикальное ДКП). Поскольку частоты компонентов горизонтального ДКП увеличиваются слева направо, а вертикального — сверху вниз, то в получающемся в результате двумерном ДКП самые низкие частоты находятся в его верхней левой части, а самые высокие частоты — в нижней правой. Поскольку амплитуда низкочастотных компонентов, как правило, больше, чем у высокочастотных, то в верхней левой части отображения содержатся относительно большие значения, а в нижней правой — малые. В стандарте JPEG двумерное ДКП задается так [17]:

$$S(v, u) = \frac{1}{4} C(v) C(u) \sum_{x=0}^7 s(y, x) \cos\{(2x+1)u\pi/16\} \cos\{(2y+1)v\pi/16\}, \quad (3.91)$$

где $S(v, u)$ — коэффициенты двумерного ДКП,

$$C(v) = \begin{cases} 1/\sqrt{2} & \text{для } v = 0 \\ 1 & \text{для } v > 0 \end{cases},$$

$$C(u) = \begin{cases} 1/\sqrt{2} & \text{для } u = 0 \\ 1 & \text{для } u > 0 \end{cases},$$

а $s(y, x)$ — значение пикселя из блока размером 8×8 пикселей.

Обратное двумерное ДКП, необходимое для восстановления изображений, задается как

$$s(y, x) = \frac{1}{4} \sum_{u=0}^7 \sum_{v=0}^7 C(u)C(v)S(v, u) \cos\{(2x+1)u\pi/16\} \cos\{(2y+1)v\pi/16\}. \quad (3.92)$$

Как и для ДКП, существуют быстрые двумерные ДКП [17].

3.9.2. Квантование коэффициентов двумерного ДКП

Теперь каждый из 64 коэффициентов ДКП $S(v, u)$ квантуется отдельно с помощью устройства однородного квантования (см. раздел 2.4.2). У каждого из 64 устройств квантования разный шаг квантования. Каждый коэффициент нормируется на шаг своего устройства квантования, а результат округляется до ближайшего целого числа. В результате этой процедуры определяется массив целых чисел, содержащий некоторое количество нулей, преимущественно в нижней правой части массива.

3.9.3. Кодирование

Верхний левый коэффициент в массиве соответствует члену постоянного тока в двумерном ДКП. Он описывает средний уровень сигнала для блока 8×8 , и его значение обычно не очень быстро изменяется при переходе к соседнему блоку. Следовательно, этот коэффициент рассматривается отдельно от других коэффициентов переменного тока (см. рис. 3.16). Он кодируется с помощью дифференциальной импульсно-кодовой модуляции (ДИКМ) и ему присваивается значение разности между его значением и значением коэффициента постоянного тока предыдущего блока размером 8×8 пикселей. В результате, как правило, получаются относительно небольшие значения.

Коэффициенты переменного тока расположены последовательно в зигзагообразном порядке в массиве размером 8×8 пикселей (этот зигзаг показан на рис. 3.17). Последовательность строится согласно номерам в массиве с расположением двумерных коэффициентов в нисходящем порядке.

После этого оба набора коэффициентов кодируются (см. рис. 3.16), в результате чего данные сжимаются еще больше. Если попадаются длинные серии нулей, их можно сжать с помощью серийного кодирования. При таком кодировании просто показывается, сколько нулей в серии. Если подобная серия встречается в конце зигзага или блока, используется кодовое слово. Остальные коэффициенты кодируются с помощью кода Хаффмана по стандартной последовательной схеме или с помощью арифметического

$$\begin{pmatrix} 0 & 1 & 5 & 6 & 14 & 15 & 27 & 28 \\ 2 & 4 & 7 & 13 & 16 & 26 & 29 & 42 \\ 3 & 8 & 12 & 17 & 25 & 30 & 41 & 43 \\ 9 & 11 & 18 & 24 & 31 & 40 & 44 & 53 \\ 10 & 19 & 23 & 32 & 39 & 45 & 52 & 54 \\ 20 & 22 & 33 & 38 & 46 & 51 & 55 & 60 \\ 21 & 34 & 37 & 47 & 50 & 56 & 59 & 61 \\ 35 & 36 & 48 & 49 & 57 & 58 & 62 & 63 \end{pmatrix}$$

Рис. 3.17. Зигзагообразная последовательность двумерного ДКП

кода по схеме расширенного ДКП. Оба кода содержат кодовые слова различной длины, причем чаще всего встречаются самые короткие кодовые слова. Это снижает число битов, которые будут передаваться, или, другими словами, увеличивает степень сжатия. Эти два кода самые эффективные, т.е. они передают максимум информации с помощью наименьшего количества битов. В коде Хаффмана используется оптимальный набор кодовых слов, которые содержат целое число информационных битов. При использовании арифметического кодирования сжатие увеличивается приблизительно на 10%. Используется один из видов однопроходного адаптивного кодирования, при котором список кодовых слов динамически адаптируется к данным, которые нужно закодировать. Впрочем, темы, касающиеся теории информации и кодирования, в данной книге рассматриваться не будут.

3.10. Примеры

Пример 3.8

Первая четверка дискретных значений сигнала напряжения с шириной полосы 10 Гц, дискретизованного с частотой 125 Гц, — $(0, 5; 1; 1; 0, 5)$. Покажите, как можно получить ДКП-образ этой последовательности с помощью быстрого преобразования Фурье, и, следовательно, найдите Фурье-образ этих данных.

Решение

На рис. 3.18 приведена функциональная схема БПФ. Имеем

$$\begin{aligned} X(0) &= G(0) + W_4^0 H(0) = G(0) + H(0), \\ G(0) &= x(0) + W_2^0 x(2) = x(0) + x(2), \\ H(0) &= x(1) + W_2^0 x(3) = x(1) + x(3). \end{aligned}$$

Поскольку $W^0 = 1$, подставив соответствующие значения, получаем

$$\begin{aligned} X(0) &= x(0) + x(2) + x(1) + x(3) = \\ &= 0,5 + 1 + 1 + 0,5 = 3, \\ X(1) &= G(1) + W_4^1 H(1), \\ G(1) &= x(0) - W_2^0 x(2) = x(0) - x(2), \\ H(1) &= x(1) - W_2^0 x(3) = x(1) - x(3). \end{aligned}$$

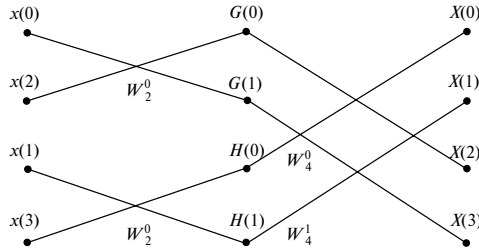


Рис. 3.18. Функциональная схема БПФ (пример 3.8)

Теперь $W_N = e^{-2\pi i/N}$ и, следовательно, $W_N^4 = e^{-2\pi i/4} = e^{-\pi i/2}$. При подстановке этих выражений получаем

$$\begin{aligned}
 X(1) &= x(0) - x(2) + e^{-i\pi/2}[x(1) - x(3)] = \\
 &= 0,5 - 1 + \left[\cos\left(\frac{\pi}{2}\right) - i \sin\left(\frac{\pi}{2}\right) \right] (1 - 0,5) = \\
 &= 0,5 - 1 + (0 - i)0,5 = -0,5 - i0,5 = -0,5(1 + i), \\
 X(2) &= G(0) - W_4^0 H(0) = \\
 &= G(0) - H(0) = x(0) + x(2) - [x(1) + x(3)] = \\
 &= 0,5 + 1 - (1 + 0,5) = 0, \\
 X(3) &= G(1) - W_4^1 H(1) = \\
 &= x(0) - x(2) - e^{-i\pi/2}[x(1) - x(3)] = \\
 &= 0,5 - 1 - \left[\cos\left(\frac{\pi}{2}\right) - i \sin\left(\frac{\pi}{2}\right) \right] (1 - 0,5) = \\
 &= -0,5 - (-i)0,5 = -0,5 + i0,5 = 0,5(-1 + i).
 \end{aligned}$$

Следовательно,

$$X(\Omega) = \{3; -0,5(1 + i); 0; 0,5(-1 + i)\}.$$

Если интервал дискретизации T маленький, то

$$\text{ПФ} = T\text{ДПФ},$$

где ПФ — это преобразование Фурье. Здесь $T = 1/125 \text{ с} = 0,008 \text{ с}$. Период сигнала равен $1/10 \text{ с} = 0,1 \text{ с}$; следовательно,

$$\frac{T}{\text{период}} = \frac{0,008}{0,1} = 0,08 \ll 1,$$

и ПФ = $T\text{ДПФ}$ будет хорошим приближением, так что

$$\text{ПФ} = \{0,024; -0,004(1 + i); 0; 0,004(-1 + i)\}.$$

Пример 3.9

В системе сжатия данных информация вначале преобразовывается, а затем преобразованные значения ограничиваются пороговым значением 0,375. Рассматриваются два преобразования: дискретное косинус-преобразование $X_c(k)$, которое задается как

$$X_c(k) = \frac{1}{N} \sum_{n=0}^{N-1} x_n \cos\left(\frac{k2\pi n}{N}\right), \quad k = 0, 1, \dots, N-1,$$

и преобразование Уолша X_k , которое задается как

$$X_k = \frac{1}{N} \sum_{i=0}^{N-1} x_i \text{WAL}(k, i), \quad k = 0, 1, \dots, N-1.$$

Предположив, что в результате получается последовательность данных $\{1; 2; 0; 3\}$, определите

- какое из двух преобразований в этом случае эффективнее для сжатия данных;
- максимальное достижимое сжатие данных (в процентах).

Найдите обратное преобразование данных, сжатых с помощью преобразования Уолша, и сравните с исходной последовательностью данных.

Решение

- Для ДКП с $x_0 = 1$, $x_1 = 2$, $x_2 = 0$, $x_3 = 3$.

$$\begin{aligned} X_c(0) &= \frac{1}{4}(x_0 \cos 0 + x_1 \cos 0 + x_2 \cos 0 + x_3 \cos 0) = \\ &= \frac{1}{4}(1 + 2 + 0 + 3) = \frac{6}{4} = 1,5, \\ X_c(1) &= \frac{1}{4} \sum_{n=0}^3 x_n \cos\left(\frac{2\pi n}{4}\right) = \\ &= \frac{1}{4} \sum_{n=0}^3 x_n \cos\left(\frac{\pi n}{2}\right) = \\ &= \frac{1}{4} \left[x_0 + x_1 \cos\left(\frac{\pi}{2}\right) + x_2 \cos\left(\frac{2\pi}{2}\right) + x_3 \cos\left(\frac{3\pi}{2}\right) \right] = \\ &= \frac{1}{4}[1 + 2 \times 0 + 0 \times (-1) + 3 \times 0] = 0,25, \\ X_c(2) &= \frac{1}{4} \left[x_0 \cos\left(\frac{4\pi \times 0}{4}\right) + x_1 \cos\left(\frac{4\pi \times 1}{4}\right) + x_2 \cos\left(\frac{4\pi \times 2}{4}\right) + \right. \\ &\quad \left. + x_3 \cos\left(\frac{4\pi \times 3}{4}\right) \right] = \\ &= \frac{1}{4}(x_0 - x_1 + x_2 - x_3) = \frac{1}{4}(1 - 2 + 0 - 3) = -1, \end{aligned}$$

$$\begin{aligned}
X_c(3) &= \frac{1}{4} \left[x_0 \cos \left(\frac{6\pi \times 0}{4} \right) + x_1 \cos \left(\frac{6\pi}{4} \right) + x_2 \cos \left(\frac{6\pi \times 2}{4} \right) + \right. \\
&\quad \left. + x_3 \cos \left(\frac{6\pi \times 3}{4} \right) \right] = \\
&= \frac{1}{4} [1 + 2 \times 0 + 0 \times (-1) + 3 \times 0] = 0,25.
\end{aligned}$$

Следовательно,

$$\text{ДКП} = \{1, 5; 0, 25; -1; 0, 25\}.$$

Значения, которые останутся после сравнения с пороговым значением ($|\text{значения}| > 0,375$), — это 1, 5 и -1 .

Для преобразования Уолша, которое рассчитывалось в разделе 3.8.2, $X_k = \{1, 5; 0; 0, 5; -1\}$, так что в этом случае после сравнения с порогом останутся значения 1, 5, 0, 5 и -1 . Следовательно, в этом случае ДКП обеспечивает более эффективное сжатие данных.

2. Если принять, что эффективность сжатия данных η задается как

$$\eta = \frac{(\text{ИП} - \text{ПП}) \times 100\%}{\text{ИП}},$$

где ИП — число элементов в исходной последовательности, ПП — число элементов данных в преобразованной последовательности, то

$$\eta = \frac{4 - 2}{4} \times 100\% = 50\%.$$

Наконец, данные, сжатые с помощью преобразования Уолша, — это последовательность $\{1, 5; 0; 0, 5; -1\}$. Обратное преобразование задается уравнением (3.70):

$$\begin{aligned}
x_i &= \sum_{k=0}^{N-1} X_k \text{WAL}(k, i), i = 0, 1, \dots, N-1, \\
x_i &= \begin{bmatrix} 1, 5 & 0 & 0, 5 & -1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} = \begin{bmatrix} 1 & 2 & 0 & 3 \end{bmatrix}.
\end{aligned}$$

Результат идентичен исходной последовательности. Это объясняется тем, что $X_1 = 0 < 0,375$, и, хотя этот элемент не передается, в обратном преобразовании он появляется как 0 — точное значение. Как правило, восстановленная последовательность — это только приближение исходной.

Задачи

- 3.1. Найдите представление периодических сигналов, изображенных на рис. 3.19, в виде ряда Фурье.
- 3.2. Найдите представление периодических сигналов, изображенных на рис. 3.19, в виде комплексного ряда Фурье.
- 3.3. Покажите, что амплитуды, найденные в задаче 3.1, согласуются с амплитудами, найденными в задаче 3.2.
- 3.4. Постройте графики амплитудного и фазового спектров компонентов Фурье для сигналов, изображенных на рис. 3.19.
- 3.5. Рассчитайте амплитуду и спектральную плотность энергии сигнала напряжения $v(t)$, который задается как

$$v(t) = \begin{cases} \frac{A}{\tau}t + A & -\tau \leq t \leq 0 \\ -\frac{A}{\tau}t + A & 0 \leq t \leq \tau \\ 0 & \tau \leq t \leq -\tau \end{cases},$$

где $A = 5$ В, а $\tau = 20$ мс.

- 3.6. Найдите спектральную плотность энергии и фазовый спектр сигнала $v(t)$, который задается как

$$v = \begin{cases} 2 \sin \left[\frac{2\pi}{T} \left(t + \frac{T}{4} \right) \right] & -\frac{T}{4} \leq t \leq \frac{T}{4} \\ 0 & \text{в остальных случаях} \end{cases},$$

где $T = 0,0167$ с.

- 3.7. Постройте график спектральной плотности энергии функции

$$w(t) = \begin{cases} \sin \left[\frac{2\pi}{T} \left(t + \frac{3T}{4} \right) \right] & -\frac{3T}{4} \leq t \leq -\frac{T}{2} \\ 1, 0 & -\frac{T}{2} \leq t \leq \frac{T}{2} \\ \sin \left[\frac{2\pi}{T} \left(t - \frac{3T}{4} \right) \right] & \frac{T}{2} \leq t \leq \frac{3T}{4} \\ 0 & \frac{3T}{4} \leq t \leq -\frac{3T}{4} \end{cases},$$

где $T = 4$ с.

- 3.8. Найдите ДПФ последовательности данных $\{0, 1, 1, 0\}$ и проверьте правильность вашего ответа по ОДПФ.
- 3.9. Вычислите размерности $X(k)$ и $X^2(k)$. Затем найдите и постройте график энергетического спектра последовательности данных $\{0, 1, 1, 0\}$, ДПФ которой было найдено в задаче 3.8.
- 3.10. Пусть последовательность $\{0, 1, 1, 0\}$ из задачи 3.8 представляет собой оцифрованные элементы выборки, взятые из сигнала напряжения, дискретизованного с частотой 125 Гц. Найдите спектральную плотность энергии и фазовый спектр Фурье-образа этой последовательности данных.

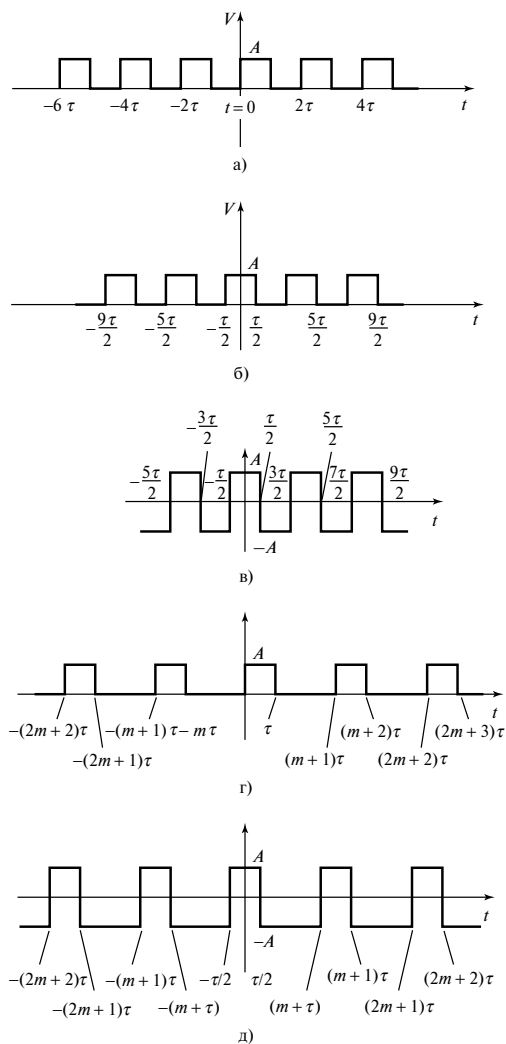


Рис. 3.19. Периодические сигналы для задач 3.1–3.4

- 3.11. Воспользуйтесь свойством сдвига по времени ДПФ и решением задачи 3.8, чтобы найти амплитудный и фазовый спектры временного ряда $\{0, 0, 0, 0, 0, 1, 1, 0\}$ для данных, дискретизованных в моменты времени $t = 0, 1, 2, \dots, 7$ мс.
- 3.12. Воспользуйтесь результатом задачи 3.9 и докажите теорему Парсеваля для данных $\{0, 1, 1, 0\}$.
- 3.13. Воспользовавшись теоремой о корреляции, найдите круговую корреляцию последовательностей данных $\{1, 1, 0, 1\}$ и $\{1, 0, 0, 1\}$. Постройте график зависимости корреляционной функции от номера задержки j .
- 3.14. Воспользуйтесь теоремой о корреляции для вычисления линейной корреляции последовательностей данных $\{1, 1, 0, 1\}$ и $\{1, 0, 0, 1\}$. Постройте график зависимости корреляционной функции от номера задержки и сравните результат с решением задачи 3.13. Объясните различия, если они будут.
- 3.15. Найдите ДПФ последовательности данных $\{0, 1, 1, 0\}$, воспользовавшись алгоритмом БПФ с временной децимацией (алгоритм Кули–Тьюки). Сравните ответ с результатом задачи 3.8. Сравните число операций комплексного сложения и умножения в этих двух методах.
- 3.16. Найдите ОБПФ результата задачи 3.15 и убедитесь, что получится последовательность данных $\{0, 1, 1, 0\}$.
- 3.17. Найдите БПФ последовательности данных $\{0, 0, 1, 1, 1, 1, 0, 0\}$ и постройте график амплитудного и фазового спектров. Проверьте ответ, вычислив ОБПФ и получив исходную последовательность.
- 3.18. Напишите программу для вычисления БПФ и ОБПФ. Проверьте БПФ путем вычисления ДПФ последовательностей $\{0, 1, 1, 0\}$ из задачи 3.8 и $\{1, 1, 0, 1\}$ и $\{1, 0, 0, 1\}$ из задачи 3.13. Проверьте ОБПФ путем вычисления ОДПФ ДПФ-образов.
- 3.19. Воспользуйтесь программой для вычисления БПФ и найдите 1024-точечное ДПФ сигналов из задач 3.5 и 3.7. Постройте их энергетические и фазовый спектры и сравните с графиками, полученными в решениях задач 3.5 и 3.7.
- 3.20. Воспользовавшись 1024-точечным БПФ, найдите и постройте энергетический спектр прямоугольного импульса с амплитудой 5 В и шириной $\tau = 6$ с. Сравните полученный результат с результатом задачи 3.7.
- 3.21. 1. Воспользуйтесь теоремой о свертке (уравнение (3.37)) и найдите свертку спектров двух пар сигналов:
 - а) $v_s = \sin(2\pi \times 100t)$ и импульса с единичной высотой v_w с центром в точке $t = 0$ и с шириной $2s$;
 - б) $v_s = \sin(2\pi \times 100t)$ и

$$v_w = \begin{cases} \cos(2\pi \times 0, 25t) & \text{при } 1 \leq t \leq -1 \text{ с} \\ 0 & \text{в остальных случаях.} \end{cases}$$
2. Если Фурье-компоненты сигнала получены с помощью ДПФ дискретных данных, то в действительности используется элемент выборки сигнала длиной $(N-1)T$, где N — количество данных, а T — промежуток времени между

элементами выборки. Говорят, что сигнал был вырезан с помощью окна длиной $(N - 1)T$. Тогда искомым спектр задается сверткой спектра сигнала со спектром окна. Обозначив для сигналов из пункта 1 через v_s сигнал, а через v_w — окно данных, прокомментируйте, насколько эти два окна подходят для определения элементов выборки сигнала.

- 3.22. Найдите дискретное косинус-преобразование, дискретное преобразование Уолша и дискретное преобразование Адамара последовательности данных $\{0, 1; -0, 2; 0, 3; -0, 4; 0, 5; 1, 5; 2; 1, 5; 0, 5; -0, 4; 0, 3; -0, 2; 0, 1\}$. Расположите их по степени предпочтения с точки зрения эффективности сжатия данных с выбранным пороговым значением 0,35.
- 3.23. Дискретные значения напряжения, полученные путем сканирования распределения интенсивности на фотоизображении, равны $\{3, 2; 3, 6; 3, 3; 2, 9; 1, 7; 1, 6; 1, 8; 1, 5\}$. Обсудите преимущества преобразования этих данных с помощью БПФ и ДПУ.
- 3.24. Дополните рассуждения задачи 3.23, включив число возможных способов сжатия (включая применение ДКП).
- 3.25. Создайте таблицу, показывающую преимущества, недостатки и области применения быстрого преобразования Фурье, дискретного преобразования Уолша, дискретного косинус-преобразования и дискретного преобразования Адамара.

Задачи для решения с помощью MATLAB

- 3.26. 1. Воспользуйтесь соответствующей функцией MATLAB, чтобы найти прямым методом коэффициенты ДПФ следующей восьмиточечной последовательности с дискретизацией во времени

$$x(n) = \{4, 2, 1, 4, 6, 3, 5, 2\}.$$

2. С помощью соответствующей функции MATLAB найдите дискретную во времени последовательность, которая соответствует следующим коэффициентам ДПФ:

$$27 + 0i$$

$$-4, 12132 + 3, 292893i$$

$$4 + i$$

$$0, 12132 - 4, 707107i$$

$$5 + 0i$$

$$0, 12132 + 4, 707107i$$

$$4 - i$$

$$-4, 12132 - 3, 292893i$$

- 3.27. 1. С помощью MATLAB найдите 32-точечное БПФ дискретной во времени последовательности, которая задается как

$$x(n) = \begin{cases} 1, & n = 0, 1, \dots, 15 \\ 0, & n = 16, 17, \dots, 31 \end{cases}.$$

2. Найдите с помощью MATLAB 64-точечное БПФ последовательности данных из п. 1.
3. Сравните результаты пп. 1 и 2.
- 3.28. Объясните, почему двоичным алгоритмом БПФ можно пользоваться для оценки частотной характеристики дискретной по времени системы с функцией z -переноса в виде рационального полинома. Проиллюстрируйте ответ, воспользовавшись функцией БПФ MATLAB, чтобы найти частотную характеристику дискретного во времени фильтра со следующей передаточной функцией, записанной на z -плоскости:

$$H(z) = \frac{1 - 1,618z^{-1} + z^{-2}}{1 - 1,516z^{-1} + 0,87z^{-2}}.$$

Назовите все практические следствия, которые могут иметь к этому отношение.

Литература

1. Ahmed N. and Rao K. R. (1975) *Orthogonal Transforms for Digital Signal Processing*. Berlin: Springer.
2. Bailey D. J. and Birch N. (1989) Image compression using a discrete cosine transform image processor. *Electronic Engineering*, July, 9–44.
3. Beauchamp K. G. (1987) *Transforms for Engineers. A Guide to Signal Processing*. Oxford: Clarendon.
4. Burrus C. S. (1998) *Introduction to Wavelets and Wavelet Transforms: A Primer*. Englewood Cliffs NJ: Prentice-Hall.
5. Burrus C. S. and Parks T. W. (1985) *DFT/FFT and Convolution Algorithms. Theory and Implementation*. New York: Wiley.
6. Chan Y. T. (1995) *Wavelet Basics*. Boston MA: Kluwer Academic.
7. Chen W., Smith C. H. and Fialick S. C. (1977) A fast computational algorithm for the discrete cosine transform. *IEEE Trans. Communications*, **25**, 1004–1009.
8. Chui C. K. (1992) *An Introduction to Wavelets*. Boston MA: Academic Press.
9. Cooley J. W. and Tukey J. W. (1965) An algorithm for the machine calculation of complex Fourier series. *Mathematics Computation*, **19**, 297–301.
10. Daubechies I. (1990) The wavelet transform, time-frequency localisation and signal analysis. *IEEE Trans. Information Theory*, **36**(5), 961–1005.
11. Daubechies I. (1992) *Ten Lectures on Wavelets*. Philadelphia: The Society for Industrial and Applied Mathematics.
12. Gentleman W. M. and Sande G. (1966) Fast Fourier transforms for fun and profit. In *Fall Joint Computing Conf., AFIPS Proc.*, **29**, 563–578.
13. Mallat S. G. (1989) A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, **11**(7), 674–693.
14. Mallat S. and Hwang W. L. (1992) Singularity detection and processing with wavelets. *IEEE Trans. Information Theory*, **38**(2), 617–643.
15. McClellan J. H. and Rader C. M. (1979) *Number Theory in Digital Signal Processing*. Englewood Cliffs NJ: Prentice-Hall.
16. Narasinka M. J. and Petersen A. M. (1978) On the computation of the discrete cosine transform. *IEEE Trans. Communications*, **26**, 934–936.
17. Pennebaker W. B. and Mitchell J. L. (1993) *JPEG Still Image Data Compression Standard*. New York: Van Nostrand Reinhold.

18. Pitas I. (1993) *Digital Image Processing Algorithms*. New York: Prentice-Hall.
19. Rader C. M. (1968) Discrete Fourier transform when the number of data samples is prime. *IEEE Proc.*, **56**, 1107, 1108.
20. Rosenfield A. and Thurston M. (1971) Edge and curve detection for visual scene analysis. *IEEE Trans. Computing*, **20**, 562–569.
21. Saatchi M. R., Gibson C. and Rowe J. K. W. (1997) Adaptive multiresolution analysis based evoked potential filtering. *IEE Proc.-Sci. Meas. Technol.*, **144**(4), July, 149–155.
22. Signal Processing Committee (ed.) (1979) *Programs for Digital Signal Processing*. New York: IEEE.
23. Srinivassan R. and Rao K. R. (1983) An approximation to the discrete cosine transform. *Signal Processing*, **5**, 81–85.
24. Strum R. D. and Kirk D. E. (1988) *First Principles of Discrete Systems and Digital Signal Processing*. Reading MA: Addison-Wesley.
25. Thakor N. V., Xin-Rong G., Yi-Chun S. and Hanley D. F. (1993) Multiresolution wavelet analysis of evoked potentials. *IEEE Trans. Biomedical Engineering*, **40**(11), 1085–1093.
26. Winograd S. (1978) On computing the discrete Fourier transform. *Mathematics Computation*, **32**, 175–199.
27. Yip P. and Ramamohan K. (1987) In *Handbook of Digital Signal Processing Engineering Applications* (Elliott D. E. (ed.)). New York: Academic Press.
28. Zhang J. and Zheng C. (1997) Extracting evoked potentials with the singularity detection technique. *IEEE Engineering in Medicine and Biology*. 155–161.

Приложения

3.A. Программа на С для прямого вычисления ДПФ

Приведенная ниже программа на языке С предназначена для прямого вычисления ДПФ или ОДПФ дискретной во времени последовательности $x(n)$:

$$X(k) = \sum_{n=0}^{N-1} x(n)W^{nk}, k = 0, 1, \dots, N-1 \quad \text{ДПФ}, \quad (3A.1, a)$$

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k)W^{-nk} \quad \text{ОДПФ}, \quad (3A.1, б)$$

где $W = e^{i-2\pi/N}$, а n — длина последовательности.

Входная последовательность $x(n)$ должна представляться в комплексной форме (действительная и мнимая части). Для последовательности действительных чисел мнимые части полагаются равными нулю. Основная функция DFTD.c приведена в программе 3A.1, а функция, которая вычисляет ДПФ или ОДПФ, — в программе 3A.2. Для чтения входной последовательности данных и сохранения преобразованных данных нужны две функции: `read_data()` и `save_data()` (программа 3A.3). Входные данные хранятся во входном файле `coeff.dat`, а выходные записываются в файл `dftout.dat`.

Программа 3А.1. Основная функция dftd.c для вычисления ДПФ

```

/*-----*/
/* Программа для прямого вычисления коэффициентов ДПФ */
/* используются 3 другие функции */
/* */
/* Э. Айфичер, июль 1992 года */
/* */
/*-----*/
#include "dsp1.h"
#include "dft.h"
main()
{
    extern long npt;
    extern int inv;
    printf("выберите тип преобразования\n");
    printf("n");
    printf("0 для прямого ДПФ\n");
    printf("1 для обратного ДПФ\n");
    scanf("%d", &inv);
    read__data();
    dft();
    save__data();
    exit();
}
#include "dft.c";
#include "rdata.c";
#include "sdata.c";

```

Программа 3А.2. Функция языка С для прямого вычисления ДПФ дискретной во времени последовательности. Функция записана в отдельном файле

```

/*-----*/
/* Функция для прямого вычисления ДПФ дискретной во времени */
/* последовательности */
/* */
/* Э. Айфичер, 31.10.91 */
/* */
/*-----*/
void dft()
{
    extern int inv;
    extern long npt;
    long k, n;
    double WN, wk, c, s, XR[size], XI[size];
    extern complex x[size];
    WN=2*pi/npt;
    if(inv==1)
        WN=-WN;
    for(k=0; k<npt; ++k){
        XR[k]=0.0; XI[k]=0.0;
    }
}

```



```

        wk=k*WN;
        for(n=0;n<npt;++n){
            c=cos(n*wk);s=sin(n*wk);
            XR[k]=XR[k]+x[n+1].real*c+x[n+1].imag*s;
            XI[k]=XI[k]-x[n+1].real*s+x[n+1].imag*c;
        }
        if(inv==1){ /* разделить на N для ОДПФ */
            XR[k]=XR[k]/npt;
            XI[k]=XI[k]/npt;
        }
    }
    for(k=1;k<=npt;++k){ /* сохранить преобразованные данные в x */
        x[k].real=XR[k-1];
        x[k].imag=XI[k-1];
    }
}

```

Программа 3А.3. Функция для чтения данных, функция для записи преобразованных данных в файл на диске, заголовочный файл, содержащий постоянные структурные определения, и заголовочный файл, содержащий общие объявления и переменные

```

/*-----*/
/*  Функция для чтения данных в комплексном формате для ДПФ или БПФ */
/*-----*/
/*  Э. Айфичер, последняя модификация: июль 1992 года */
/*-----*/
void      read_data()
{
    extern long    npt;
    int          n;
    extern complex x[size];
    for(n=0;n<size;++n){
        x[n].real=0;
        x[n].imag=0;
    }
    if((in=fopen("coeff.dat","r"))==NULL){
        printf("невозможно открыть файл coeff.dat\n");
        exit(1);
    }
    fscanf(in,"%ld",&npt);
    for(n=1;n<=npt;++n){
        fscanf(in,"%f%f",&x[n].real,&x[n].imag);
    }
    fclose(in);
}

void      save_data() /*имя файла sdata.c */
{
    long    k;
    int     k1;
    extern long npt;

```

```

extern complex x[size];
if((out=fopen("dftout.dat", "w"))==NULL){
    printf("невозможно открыть файл dftout.dat\n");
    exit(1);
}
fprintf(out, "k\tXR(k)\t\tXI(k)\n");
fprintf(out, "\n");
for(k=1; k<=npt; ++k){
    k1=k-1;
    fprintf(out, "%d, \t%f\t%f\n", k1, x[k].real, x[k].imag);
}
fclose(out);
}
/* Этот файл содержит общие объявления и структуры. */
#include <stdio.h>
#include <math.h>
#include <dos.h>
#define size 600
#define pi 3.141592654
#define maxbits 30
typedef struct {
    double real;
    double imag;
    double modulus;
    double angle;
} complex;
/* имя файла: dft.h */
void dft();
void fft();
void real__data();
void save__data();
int inv;
long npt;
complex x[size];
FILE *in, *out, *fopen();

```

Пример 3А.1

Воспользуйтесь программой прямого вычисления ДПФ и найдите коэффициенты следующей восьмиточечной дискретной во времени последовательности:

$$x(n) = \{4, 2, 1, 4, 6, 3, 5, 2\}$$

Файл входных данных, созданный для этой задачи с помощью ПК программы edlin (для этой цели можно использовать и другие текстовые процессоры), имеет следующий вид:

```
8
4 0
2 0
1 0
4 0
6 0
3 0
5 0
2 0
```

В первой строке задается длина последовательности данных.

ДПФ данных, полученное с помощью приведенной выше программы, имеет такой вид:

k	$XR(k)$	$XI(k)$
0	27,000000	0,000000
1	-4,121320	3,292893
2	4,000000	1,000000
3	0,121320	-4,707107
4	5,000000	-0,000000
5	0,121320	4,707107
6	4,000000	-1,000000
7	-4,121320	-3,292893

Пример 3А.2

С помощью программы вычисления ДПФ найдите дискретную во времени последовательность, соответствующую приведенным выше коэффициентам ДПФ. Входные данные имеют следующий вид:

```
8
27,000000 0,000000
-4,121320 3,292893
4,000000 1,000000
0,121320 -4,707107
5,000000 -0,000000
0,121320 4,707107
4,000000 -1,000000
-4,121320 -3,292893
```

Используя программу, получаем на выходе последовательность, идентичную дискретной последовательности в примере 3А.1.

Таблица 3А.1. Входная последовательность комплексных чисел

n	$x(n)$	
	Действительная часть	Мнимая часть
0	0,100000E01	0,0
1	0,900000E00	0,300000E00
2	0,720000E00	0,540000E00
3	0,486000E00	0,702000E00
4	0,226800E00	0,777600E00
5	-0,291600E-01	0,767880E00
6	-0,256608E00	0,682344E00
7	-0,435650E00	0,537127E00
8	-0,553224E00	0,352719E00
9	-0,603717E00	0,151480E00
10	-0,588789E00	-0,447828E-01
11	-0,516476E00	-0,216941E00
12	-0,399746E00	-0,350190E00
13	-0,254714E00	-0,435095E00
14	-0,987144E-01	-0,467999E00
15	0,515569E-01	-0,450814E00
16	0,181645E00	-0,390265E00
17	0,280560E00	-0,296745E00
18	0,341528E00	-0,182903E00
19	0,362246E00	-0,621539E-01
20	0,344667E00	0,527352E-01
21	0,294380E00	0,150862E00
22	0,219684E00	0,224090E00
23	0,130488E00	0,267586E00
24	0,371637E-01	0,279974E00
25	-0,505440E-01	0,263125E00
26	-0,124428E00	0,221649E00
27	-0,178480E00	0,162156E00
28	-0,209279E00	0,923965E-01
29	-0,216070E00	0,203732E-01
30	-0,200575E00	-0,464851E-01
31	-0,166572E00	-0,102009E00

Пример 3А.3

В третьем примере используется последовательность комплексных чисел из работы [IEEE, 1979]:

$$x(n) = Q^n, n = 0, 1, \dots, 31,$$

где $Q = 0,9 + i0,3$.

Входная последовательность данных $x(n)$ и ее ДПФ $X(k)$, полученное с помощью программы прямого вычисления ДПФ, приведены в табл. 3А.1 и 3А.2 соответственно.

Таблица 3А.2. Преобразованный выход (пример 3А.2)

0,693972	3,499714
2,792268	8,050456
9,402964	-9,135013
1,866446	-3,833833
1,131822	-2,234158
0,904794	-1,534631
0,799557	-1,139607
0,739607	-0,882315
0,700858	-0,698566
0,673577	-0,558478
0,653112	-0,446244
0,636987	-0,352691
0,623790	-0,272085
0,612613	-0,200642
0,602885	-0,135703
0,594200	-0,075314
0,586276	-0,017948
0,578899	0,037651
0,571898	0,092607
0,565139	0,147983
0,558490	0,204882
0,551858	0,264523
0,545134	0,328363
0,538217	0,398257
0,531000	0,476679
0,523403	0,567133
0,515361	0,674850
0,506928	0,808100
0,498469	0,980906
0,491388	1,219210
0,490730	1,577083
0,517355	2,188832

3.Б. Программа на С для двоичного БПФ с временной децимацией

Приведенная ниже программа вычисления БПФ представляет собой реализацию на С двоичного БПФ с временной децимацией [9]. Программа вычисляет ДПФ и ОДПФ дискретной во времени последовательности согласно определению (3А.1). Программа состоит из главной функции `dftf.c` и трех функций: `fft()`, `read__data()` и `save__data()`. Как и при вычислении непосредственно ДПФ, все функции записаны в отдельных файлах и объединяются при компиляции с помощью вызова `(include)` из главной функции. Функции `read__data()` и `save__data()` используются для чтения данных и записи преобразованных данных в файл. Эти два файла идентичны файлам, которые применялись для прямого вычисления ДПФ. Главная программа `dftf.c` и функция `fft()` приведены в программах 3Б.1 и 3Б.2 соответственно.

Использование любого из примеров, приведенных в приложении 3А, с программой для вычисления БПФ дает такие же результаты, что и с программой для прямого вычисления ДПФ. Доказать это утверждение предлагается читателям в качестве самостоятельного упражнения.

Программа 3Б.1. Главная функция `dftf.c` для вычисления ДПФ с помощью БПФ с временной децимацией

```

/*-----*/
/*  Программа для вычисления коэффициентов ДПФ с помощью БПФ с ВД  */
/*  используются 3 другие функции                                     */
/*                                                                 */
/*  Э. Айфичер, июль 1992 года                                     */
/*-----*/
#include "dsp1.h"
#include "dft.h"
main()
{
    extern long npt;
    extern int inv;
    printf("выберите тип преобразования\n");
    printf("\n");
    printf("0 для прямого ДПФ\n");
    printf("1 для обратного ДПФ\n");
    scanf("%d",&inv);
    read__data();
    fft();
    save__data();
    exit();
}
#include "fft.c";
#include "rdata.c";
#include "sdata.c";

```

Программа 3Б.2. Реализация на С двоичного алгоритма БПФ с временной децимацией

```

/*-----*/
/*  имя файла: fft.c                                             */
/*                                                                 */
/*  Э. Айфичер, июль 1992 года                                     */
/*                                                                 */
/*  функция вычисляет ДПФ последовательности с                  */
/*  помощью двоичного БПФ                                       */
/*-----*/
void    fft()
{
    int    sign;
    long   m,irem,l,le,lel,k,ip,i,j;

```

```

double  ur,ui,wr,wi,tr,ti,temp;
extern  long npt;
extern  complex x[size];
/* перестановка с замещением данных, определяемая
   обратным порядком битов */
j=1;
for(i=1;i<npt;++i) {
    if(i<j) {
        tr=x[j].real;ti=x[j].imag;
        x[j].real=x[i].real;
        x[j].imag=x[i].imag;
        x[i].real=tr;x[i].imag=ti;
        k=npt/2;
        while(k<j) {
            j=j-k;
            k=k/2;
        }
    }
    else{
        k=npt/2;
        while(k<j) {
            j=j-k;
            k=k/2;
        }
    }
    j=j+k;
}
/* считается количество каскадов: m=log2(npt) и выбирается БПФ
   либо ОБПФ */
m=0;irem=npt;
while(irem>1) {
    irem=irem/2;
    m=m+1;
}
if(inv==1)
    sign=1;
else
    sign=-1;
/* вычисляется БПФ для каждого каскада*/
for(l=1;l<=m,++l) {
    le=pow(2,l);
    le1=le/2;
    ur=1.0; ui=0;
    wr=cos(pi/le1);
    wi=sign*sin(pi/le1);
    for(j=1;j<=le1;++j){
        i=j;
        while(i<=npt){
            ip=i+le1;
            tr=x[ip].real*ur-x[ip].imag*ui;
            ti=x[ip].imag*ur+x[ip].real*ui;

```

```

        x[ip].real=x[i].real-tr;
        x[ip].imag=x[i].imag-ti;
        x[i].real=x[i].real+tr;
        x[i].imag=x[i].imag+ti;
        i=i+le;
    }
    temp=ur*wr-ui*wi;
    ui=ui*wr+ur*wi;
    ur=temp;
}
/* если нужно найти БПФ, каждый коэффициент делится на npt */
if(inv=-1) {
    for(i=1;i<=npt;++i) {
        x[i].real=x[i].real/npt;
        x[i].imag=x[i].imag/npt;
    }
}
}

```

3.В. ДПФ и БПФ в программном пакете MATLAB

Ключевыми функциями в программном пакете MATLAB и MATLAB Signal Processing Toolbox для выполнения одномерных ДПФ и БПФ являются функции `dftmtx`, `fft` и `ifft`. В Toolbox есть также функции для дискретного косинус-преобразования и двумерного БПФ.

Для вычисления дискретного преобразования Фурье N -точечной последовательности данных в виде вектора x можно использовать функцию `dftmtx`, которая имеет такой синтаксис:

```
X=x*dftmtx(N)
```

Функция `dftmtx` вычисляет и возвращает коэффициенты поворота в виде комплексной матрицы порядка $N \times N$. Затем она умножается на последовательность данных x , что и дает дискретное преобразование Фурье этой последовательности.

Обратное ДПФ можно найти с помощью команды `conj`:

```
x=X*conj(dftmtx(N))/N
```

Функция `fft` вычисляет ДПФ одномерной последовательности данных с помощью двоичного алгоритма БПФ (если длина данных — 2 в какой-то степени), а функция `ifft` используется для нахождения обратного ДПФ.

Программу из программного пакета MATLAB (программа 3В.1) можно использовать для чтения данных из файла и непосредственного вычисления прямого и обратного ДПФ. Программу 3В.2 можно использовать для вычисления ДПФ или обратного ДПФ по БПФ.

Примеры, иллюстрирующие применение этих программ, можно найти в книге [Ifeachor, 2001] (подробности см. в предисловии).

Программа 3В.1. Реализация в MATLAB прямого вычисления ДПФ

```
function DFTD
clear all;
% программа для прямого вычисления коэффициентов ДПФ
direction=-1; %1 - прямое ДПФ, -1 - обратное ДПФ
in=fopen('datain.dat','r');
x=fscan(in,'%g %g', ([2,inf]));
fclose(in);
x=x(1,:)+x(2,:)*i; % формирование комплексных чисел
if direction==1
    y=x*dftmtx(length(x)); % вычисление ДПФ
else
    y=x*conj(dftmtx(length(x)))/length(x); % вычисление ОДПФ
end
% запись/печать результатов
out=fopen('dataout.dat','w');
fprintf(out,'%g %g\n',[real(y);imag(y)]);
fclose(out);
subplot(2,1,1),plot(1:length(x),x);title('Входной сигнал');
subplot(2,1,2),plot(1:length(y),y);title('Выходной сигнал');
```

Программа 3В.2. Программа MATLAB для вычисления ДПФ через БПФ

```
function DFTF
% программа для вычисления коэффициентов ДПФ с помощью БПФ с ВД
clear all;
direction=-1; %1 - прямое ДПФ, -1 - обратное ДПФ
in=fopen('datain.dat','r');
x=fscan(in,'%g %g', ([2,inf]));
fclose(in);
x=x(1,:)+x(2,:)*i; % формирование комплексных чисел
if direction==1
    y=fft(x,length(x)) % вычисление ДПФ
else
    y=ifft(x,length(x)) % вычисление ОБПФ
end
% запись/печать результатов
out=fopen('dataout.dat','w');
fprintf(out,'%g %g\n',[real(y);imag(y)]);
fclose(out);
subplot(2,1,1),plot(1:length(x),x);title('Входной сигнал');
subplot(2,1,2),plot(1:length(y),y);title('Выходной сигнал');
```

Литература для приложений

Cooley J.W. and Tukey J.W. (1965) An algorithm for the machine calculation of complex Fourier series. *Mathematics Computation*, **19**(90), April, 297–301.

IEEE (1979) *Programs for Digital Signal Processing*. New York: IEEE Press.

Применение z -преобразования в обработке сигналов

4

4.1. Сигналы и системы дискретного времени	208
4.2. z -преобразование	209
4.3. Обратное z -преобразование	214
4.4. Свойства z -преобразования	229
4.5. Некоторые области применения z -преобразования в обработке сигналов	232
4.6. Резюме	252
Задачи	252
Литература	260
Дополнительная литература	261
Приложения	261

Самое удобное средство описания, анализа и проектирования сигналов и систем дискретного времени — это z -преобразование. В системах дискретного времени оно играет такую же роль, как преобразование Лапласа в системах непрерывного времени.

В этой главе представлены важные аспекты z -преобразования, причем акцент делается на тех, которые будут использоваться в последующих главах. Особое внимание z -преобразованию уделяется при разработке систем дискретного времени. В число областей применения z -преобразования входят: описание сигналов и систем дискретного времени, позволяющее без труда регулировать степень их устойчивости и визуализировать их частотные характеристики; анализ ошибок квантования цифровых фильтров и вычисление частотных характеристик систем дискретного времени. Более подробно большая часть областей применения преобразования освещается в последующих главах.

Как и во всех остальных частях данной книги, в этой главе принят практический подход. Для того чтобы читатели могли глубже понять предмет, приводятся алгоритмы и программы на языке C и коды MATLAB. Значительная часть тем, обсуждаемых в этой главе, касается линейных сигналов и системы дискретного времени, поэтому мы начнем с очень краткого обзора основных характерных признаков этого класса сигналов и систем.

4.1. Сигналы и системы дискретного времени

Значения дискретного сигнала определяются только при дискретных значениях времени или какой-либо другой соответствующей переменной, например, пространства. Как уже говорилось в главах 1 и 2, такой сигнал может появиться в результате дискретизации сигнала непрерывного времени, взятого через равные отрезки времени nT , $n = 0, 1, \dots$, где T — период дискретизации. Его можно создать и искусственным путем с помощью компьютера и определенного алгоритма. Амплитуда сигнала дискретного времени может быть либо также дискретной (дискретное время, дискретная амплитуда), либо непрерывной.

По традиции сигнал дискретного времени представляют в виде последовательности чисел:

$$x(n), \quad n = 0, 1, \dots \quad (4.1, a)$$

$$x(nT), \quad n = 0, 1, \dots \quad (4.1, б)$$

$$x_n, \quad n = 0, 1, \dots, \quad (4.1, в)$$

где под $x(n)$, $x(nT)$ или x_n понимается значение сигнала в дискретный момент времени n (или nT). Для удобства мы будем пользоваться записью $x(n)$ как для значения последовательности в дискретный момент времени n , так и для обозначения самой последовательности, если разница между ними не важна (смысл будет понятен из контекста). Вообще, в ЦОС принято опускать символ T , поскольку последовательность не всегда является функцией времени (это может быть, например, функция пространства). Иногда T опускается по той причине, что частота дискретизации для удобства предполагается равной единице (т.е. нормированной).

Система дискретного времени — это, по сути, математический алгоритм, входом которого служит последовательность $x(n)$, а на выходе появляется последовательность $y(n)$. Примерами систем дискретного времени могут служить цифровые контроллеры, цифровые анализаторы спектров и цифровые фильтры. Система дискретного времени может быть линейной или нелинейной, инвариантной относительно времени или изменяющейся со временем. Отметим, что линейные инвариантные относительно времени (ЛИВ) системы формируют значительный класс систем, используемых в ЦОС. Пример — цифровые фильтры, которые подробно рассматриваются в главах 6–8.

Говорят, что система дискретного времени линейна, если она подчиняется принципу суперпозиции. Это означает, что отклик системы на несколько входов равен сумме откликов этой системы на каждый вход в отдельности при отсутствии всех остальных входных сигналов. Например, если вход $x_1(n)$ системы дает выход $y_1(n)$, а другой вход $x_2(n)$ дает выход $y_2(n)$, то отклик системы на оба входа будет равен

$$a_1 x_1(n) + a_2 x_2(n) \rightarrow a_1 y_1(n) + a_2 y_2(n), \quad (4.2)$$

где a_1 и a_2 — произвольные константы.

Говорят, что система дискретного времени инвариантна относительно времени (иногда ее называют *инвариантной относительно сдвига*), если ее выход не зависит от времени приложения входа. Например, если вход $x(n)$ дает выход $y_1(n)$, то вход $x(n - k)$ даст выход $y(n - k)$:

$$x(n) \rightarrow y(n) \quad (4.3, a)$$

$$x(n - k) \rightarrow y(n - k), \quad (4.3, б)$$

т.е. задержка входного сигнала приведет к такой же задержке выходного сигнала.

Взаимосвязь между входом и выходом ЛИВ-системы задается сверточной суммой

$$y(n) = \sum_{k=-\infty}^{\infty} h(k)x(n - k), \quad (4.4)$$

где $h(k)$ — импульсная характеристика системы. Значения $h(k)$ полностью определяют систему дискретного времени во временной области. ЛИВ-система устойчива, если ее импульсная характеристика удовлетворяет условию

$$\sum_{k=-\infty}^{\infty} |h(k)| < \infty. \quad (4.5)$$

Это условие справедливо, если $h(k)$ имеет конечную длину, или если $h(k)$ стремится к нулю при увеличении k . Более подробно устойчивость рассмотрена в разделе 4.5.7.

Физически реализуемой называется система, которая дает выход только при наличии сигнала на входе. В общем случае, физически реализуемая последовательность дискретного времени $x(n)$ или импульсная характеристика $h(k)$ системы дискретного времени равны нулю до нулевого момента времени, т.е. $x(n) = 0$, $n < 0$ или $h(k) = 0$, $k < 0$. Большая часть материала, изложенного в этой книге, касается практических, т.е. физически реализуемых, систем.

4.2. z-преобразование

Определим z -преобразование последовательности $x(n)$, действительное для всех n :

$$X(z) = \sum_{n=-\infty}^{\infty} x(n)z^{-n}, \quad (4.6)$$

где z — комплексная переменная.

В причинных системах $x(n)$ может быть ненулевым только в интервале $0 < n < \infty$, и уравнение (4.6) сводится к так называемому одностороннему z -преобразованию:

$$X(z) = \sum_{n=0}^{\infty} x(n)z^{-n}. \quad (4.7)$$

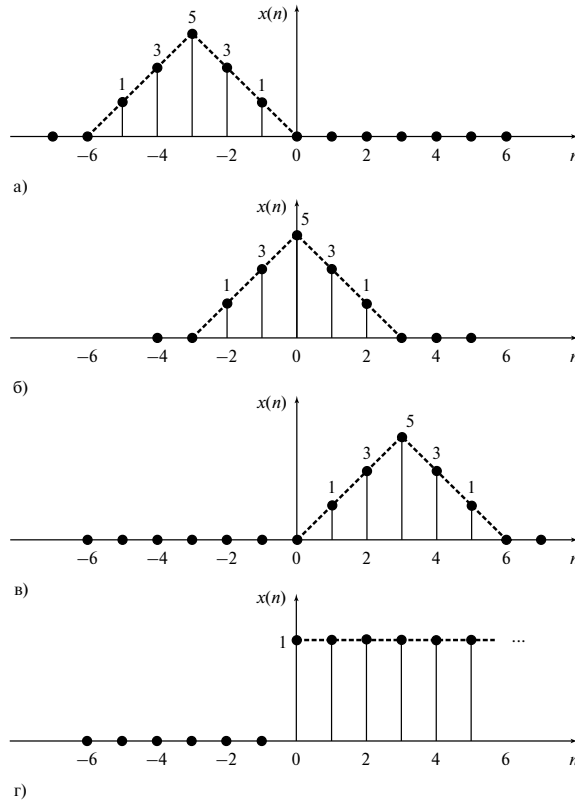


Рис. 4.1. Причинные и непричинные последовательности дискретного времени

Понятно, что z -преобразование — это степенной ряд с бесконечным количеством членов, поэтому он может не сходиться для всех значений z . Область, в которой z -преобразование сходится, называют *областью сходимости* (ОС), и в этой области значения $X(z)$ конечны. Не удивительно, что область сходимости определяется свойствами $x(n)$, или (эквивалентно) свойствами $X(z)$, что иллюстрируется ниже на примерах.

Пример 4.1

Найдите z -преобразование и область сходимости всех последовательностей дискретного времени, приведенных на рис. 4.1.

1. Последовательность на рис. 4.1, а непричинна, поскольку $x(n)$ не равно нулю при $n < 0$, но она имеет конечную длину. Значения последовательности $x(-6) = 0$,

$x(-5) = 1, x(-4) = 3, x(-3) = 5, x(-2) = 3, x(-1) = 1$ и $x(0) = 0$. Согласно уравнению (4.6) z -преобразование задается как

$$\begin{aligned} X_1(z) &= \sum_{n=-\infty}^{\infty} x(n)z^{-n} = \\ &= z^5 + 3z^4 + 5z^3 + 3z^2 + z. \end{aligned}$$

Несложно доказать, что значение $X(z)$ становится равным бесконечности при $z = \infty$. Следовательно, областью сходимости будет вся плоскость z , кроме $z = \infty$.

2. Последовательность на рис. 4.1, б также не причинна. Она двусторонняя и имеет конечную длину. Значения последовательности $x(-3) = 0, x(-2) = 1, x(-1) = 3, x(0) = 5, x(1) = 3, x(2) = 1$ и $x(3) = 0$. Согласно уравнению (4.6) z -преобразование задается как

$$\begin{aligned} X_2(z) &= \sum_{n=-\infty}^{\infty} x(n)z^{-n} = \\ &= z^2 + 3z + 5 + 3z^{-1} + z^{-2}. \end{aligned}$$

Очевидно, что значение $X(z)$ равно бесконечности при $z = 0$ или $z = \infty$. Следовательно, в область сходимости не входят только точки $z = 0$ и $z = \infty$.

3. На рис. 4.1, в представлена причинная последовательность конечной длины со значениями $x(0) = 0, x(1) = 1, x(2) = 3, x(3) = 5, x(4) = 3, x(5) = 1$ и $x(6) = 0$. Ее z -преобразование задается как

$$\begin{aligned} X_3(z) &= \sum_{n=-\infty}^{\infty} x(n)z^{-n} = \\ &= z^{-1} + 3z^{-2} + 5z^{-3} + 3z^{-4} + z^{-5}. \end{aligned}$$

В этом случае $X(n) = \infty$ при $z = 0$. Следовательно, в область сходимости не входит только точка $z = 0$.

4. Изображенная на рис. 4.1, г дискретная во времени последовательность математически определяется как

$$x(n) = \begin{cases} = 1 & 0 \leq n \leq \infty \\ = 0 & n < 0 \end{cases}.$$

Понятно, что это причинная последовательность бесконечной длины. Согласно уравнению (4.6) z -преобразование этой последовательности задается как

$$\begin{aligned} X(z) &= \sum_{n=-\infty}^{\infty} x(n)z^{-n} = \\ &= \sum_{n=0}^{\infty} z^{-n} = 1 + z^{-1} + z^{-2} + \dots \end{aligned}$$

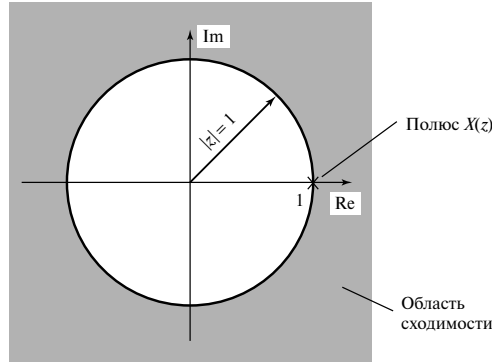


Рис. 4.2. Область сходимости сигналов (пример 4.1, п. 4)

Это геометрическая прогрессия с общим знаменателем z^{-1} . Прогрессия сходится, если $|z^{-1}| < 1$ или, что эквивалентно, если $|z| > 1$. Таким образом, $X(z)$ можно выразить в аналитическом виде, предположив, что $|z| > 1$:

$$X(z) = 1 + z^{-1} + z^{-2} + \dots = 1/(1 - z^{-1}) = z/(z - 1). \quad (4.8)$$

В этом случае z -преобразование действительно везде за пределами круга единичного радиуса с центром в начале координат (область сходимости; см. рис. 4.2). Можно легко доказать, что если $|z| > 1$ $X(z)$ сходится, а при $|z| < 1$ $X(z)$ расходится. Например, если $z = 2$ (за пределами единичного круга), то прогрессия из правой части уравнения (4.8) при сложении даст 2:

$$X(z) = 1 + 1/2 + (1/2)^2 + (1/2)^3 + \dots = 2/(2 - 1) = 2,$$

поскольку очевидно, что это геометрическая прогрессия с общим знаменателем $1/2$ и первым членом 1, сумма членов которой равна $2/(2 - 1) = 2$. С другой стороны, если $z = 1/2$ (внутри единичного круга), прогрессия уравнения (4.8) приобретает такой вид:

$$X(z) = 1 + 1/0,5 + (1/0,5)^2 + (1/0,5)^3 + \dots = 1 + 2 + 4 + 8 + \dots,$$

и видно, что она расходится. На рис. 4.2 видно, что область сходимости (заштрихована) ограничена окружностью $|z| = 1$, радиус которой равен значению полюса функции $X(z)$. Значения z , для которых $X(z) = \infty$, называются *полюсами* функции $X(z)$. Значения z , для которых $X(z) = 0$, называются *нулями* функции $X(z)$.

Из вышеприведенных примеров можно сделать вывод о том, что для причинных последовательностей конечной длины z -преобразование сходится везде, кроме точки $z = 0$. Для причинных последовательностей бесконечной длины z -преобразование сходится везде за пределами наибольшего круга, ограниченного радиусом, равным полюсу функции. Область сходимости устойчивых причинных систем всегда представляет собой круг единичного радиуса (это очень важно, поскольку позволяет описывать системы их частотными характеристиками).

В обычных последовательностях z -преобразования часто приводятся в аналитическом виде, например, в виде таблиц, таких как табл. 4.1. Подобные таблицы полезны также при поиске обратного z -преобразования.

Таблица 4.1. Примеры z -образов некоторых распространенных последовательностей

Номер	Последовательность дискретного времени $x(n), n \geq 0$	z -образ $X(z)$	Область сходимости $X(z)$
1	$k\delta(n)$	k	везде
2	k	$\frac{kz}{z-1}$	$ z > 1$
3	kn	$\frac{kz}{(z-1)^2}$	$ z > 1$
4	kn^2	$\frac{kz(z+1)}{(z-1)^3}$	$ z > 1$
5	$ke^{-\alpha n}$	$\frac{kz}{z-e^{-\alpha}}$	$ z > e^{-\alpha}$
6	$kne^{-\alpha n}$	$\frac{kze^{-\alpha}}{(z-e^{-\alpha})^2}$	$ z > e^{-\alpha}$
7	$1 - e^{-\alpha n}$	$\frac{z(1-e^{-\alpha})}{z^2 - z(1+e^{-\alpha}) + e^{-\alpha}}$	$ z > e^{-\alpha}$
8	$\cos(\alpha n)$	$\frac{z(z-\cos \alpha)}{z^2 - 2z \cos \alpha + 1}$	$ z > 1$
9	$\sin(\alpha n)$	$\frac{z \sin \alpha}{z^2 - 2z \cos \alpha + 1}$	$ z > 1$
10	$e^{-\alpha n} \sin(\alpha n)$	$\frac{ze^{-\alpha} \sin \alpha}{z^2 - 2ze^{-\alpha} z \cos \alpha + e^{-2\alpha}}$	$ z > e^{-\alpha}$
11	$e^{-\alpha n} \cos(\alpha n)$	$\frac{ze^{-\alpha} (ze^{-\alpha} - \cos \alpha)}{z^2 - 2ze^{-\alpha} z \cos \alpha + e^{-2\alpha}}$	$ z > e^{-\alpha}$
12	$\operatorname{ch}(\alpha n)$	$\frac{(z^2 - z \operatorname{ch} \alpha)}{z^2 - 2z \operatorname{ch} \alpha + 1}$	$ z > \operatorname{ch} \alpha$
13	$\operatorname{sh}(\alpha n)$	$\frac{(z \operatorname{sh} \alpha)}{z^2 - 2z \operatorname{ch} \alpha + 1}$	$ z > \operatorname{sh} \alpha$
14	$k\alpha^n$	$\frac{kz}{z-\alpha}$	$ z > \alpha$
15	$kn\alpha^n$	$\frac{k\alpha z}{(z-\alpha)^2}$	$ z > \alpha$
16	$2 c p ^n \cos(n\angle p + \angle c)$	$\frac{cz}{z-p} + \frac{c^*z}{z-p^*}$	

Здесь k и α — постоянные, c — комплексное число.

4.3. Обратное z-преобразование

Обратное z -преобразование (z^{-1}) позволяет восстанавливать последовательность дискретного времени $x(n)$ по ее z -образу. z^{-1} особенно полезно в операциях ЦОС, например, при поиске импульсной характеристики цифровых фильтров. В символической форме обратное z -преобразование можно определить как

$$x(n) = Z^{-1}[X(z)], \quad (4.9)$$

где $X(z)$ — это z -образ последовательности $x(n)$, а Z^{-1} — символ, обозначающий обратное z -преобразование.

Предположив, что последовательность причинна, z -образ $X(z)$ из уравнения (4.7) можно разложить в степенной ряд как

$$\begin{aligned} X(z) &= \sum_{n=0}^{\infty} x(n)z^{-n} = \\ &= x(0) + x(1)z^{-1} + x(2)z^{-2} + x(3)z^{-3} + \dots \end{aligned} \quad (4.10)$$

Видно, что значения последовательности $x(n)$ — это коэффициенты z^{-n} ($n = 0, 1, \dots$), и поэтому их можно найти непосредственно. На практике $X(z)$ часто выражается через отношение двух многочленов от z^{-1} или, что эквивалентно, от z :

$$X(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_N z^{-N}}{a_0 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_M z^{-M}}. \quad (4.11)$$

В этом виде обратное z -преобразование $x(n)$ можно найти с помощью одного из многих методов, например:

- а) метода разложения в степенной ряд;
- б) метода разложения на элементарные дроби;
- в) метода вычетов.

У каждого метода есть свои преимущества и недостатки. С точки зрения математической строгости метод вычетов, возможно, самый элегантный. Однако метод степенных рядов лучше всего подходит для компьютерных расчетов.

В следующих разделах с помощью численных примеров, иллюстрирующих принцип их действия, описаны все указанные методы. В приложении описывается ряд программ на языке С, которые применяются для оценки обратного z -преобразования для методов а и б. Приводятся иллюстративные численные примеры использования этих программ.

4.3.1. Метод степенных рядов

Если дано z -преобразование $X(z)$ причинной последовательности (уравнение (3.11)), его можно разложить в бесконечный ряд относительно z^{-1} или z путем деления в столбик (иногда его называют *синтетическим делением*):

$$\begin{aligned} X(z) &= \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_N z^{-N}}{a_0 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_M z^{-M}} = \\ &= x(0) + x(1)z^{-1} + x(2)z^{-2} + x(3)z^{-3} + \dots \end{aligned} \quad (4.12)$$

В этом методе числитель и знаменатель функции $X(z)$ сперва выражаются либо через уменьшающийся показатель степени z , либо через увеличивающийся показатель степени z^{-1} , а затем путем деления в столбик находится частное. Проиллюстрируем этот метод на примере.

Пример 4.2

Дан следующий z -образ причинной ЛИВ-системы:

$$X(z) = \frac{1 + 2z^{-1} + z^{-2}}{1 - z^{-1} + 0,3561z^{-2}}.$$

Найдите его z^{-1} , разложив в степенной ряд путем деления в столбик.

Решение

Вначале разложим $X(z)$ в степенной ряд с числителем и знаменателем в виде многочленов с увеличивающейся степенью z^{-1} , а затем выполним обычное деление в столбик.

$$\begin{array}{r} \begin{array}{r} 1 + 2z^{-1} + z^{-2} \\ 1 - z^{-1} + 0,3561z^{-2} \\ \hline 3z^{-1} + 0,6439z^{-2} \\ 3z^{-1} - 3z^{-2} + 1,0683z^{-3} \\ \hline 3,6439z^{-2} - 1,0683z^{-3} \\ 3,6439z^{-2} - 3,6439z^{-3} + 1,2975927z^{-4} \\ \hline 2,5756z^{-3} - 1,2975927z^{-4} \end{array} \quad \begin{array}{r} | 1 - z^{-1} + 0,3561z^{-2} \\ 1 + 3z^{-1} + 3,6439z^{-2} + 2,5756z^{-3} + \dots \end{array} \end{array}$$

Другой способ: выразить числитель и знаменатель через положительный показатель степени z в порядке уменьшения, а затем выполнить деление в столбик $\frac{z^2 + 2z + 1}{z^2 - z + 0,3561}$:

$$\begin{array}{r} \begin{array}{r} z^2 + 2z + 1 \\ z^2 - z + 0,3561 \\ \hline 3z + 0,6439 \\ 3z - 3 + 1,0683z^{-1} \\ \hline 3,6439 - 1,0683z^{-1} \\ 3,6439 - 3,6439z^{-1} + 1,2975927z^{-2} \\ \hline 2,5756z^{-1} - 1,2975927z^{-2} \end{array} \quad \begin{array}{r} | z^2 - z + 0,3561 \\ 1 + 3z^{-1} + 3,6439z^{-2} + 2,5756z^{-3} + \dots \end{array} \end{array}$$

В любом случае z -образ раскладывается в знакомый степенной ряд, т.е.

$$X(z) = 1 + 3z^{-1} + 3,6439z^{-2} + 2,5756z^{-3} + \dots$$

Теперь можно непосредственно записать обратное z -преобразование:

$$x(0) = 1; x(1) = 3; x(2) = 3,6439; x(4) = 2,5756; \dots$$

Метод деления в столбик можно сформулировать иначе (см. приложение 4А), так что значения последовательности $x(n)$ будут получаться рекурсивно:

$$\begin{aligned} x(0) &= b_0/a_0, \\ x(1) &= [b_1 - x(0)a_1]/a_0, \\ x(2) &= [b_2 - x(1)a_1 - x(0)a_2]/a_0, \\ &\dots \\ x(n) &= \left[b_n - \sum_{i=1}^n x(n-i)a_i \right] / a_0, \quad n = 1, 2, \dots \end{aligned} \tag{4.13, а}$$

где

$$x(0) = b_0/a_0. \tag{4.13, б}$$

Ниже данный пример повторяется для иллюстрации рекурсивного метода.

Пример 4.3

Рекурсивным способом найдите первые четыре члена обратного z -преобразования $x(n)$. Пусть z -образ $X(z)$ такой же, как и в примере 4.2, т.е.

$$X(z) = \frac{1 + 2z^{-1} + z^{-2}}{1 - z^{-1} + 0,3561z^{-2}}.$$

Решение

Сравнив вышеприведенные коэффициенты $X(z)$ с коэффициентами общего преобразования из уравнения (4.12), получим

$$a_0 = 1, a_1 = 2, a_2 = 1, b_0 = 1, b_1 = -1, b_2 = 0,3561, N = M = 2$$

Из уравнений (4.13) получим:

$$\begin{aligned}x(0) &= b_0/a_0 = 1, \\x(1) &= [b_1 - x(0)a_1]/a_0 = [2 - 1 \times (-1)] = 3, \\x(2) &= [b_2 - x(1)a_1 - x(0)a_2]/a_0 = 1 - 3 \times (-1) - 1 \times 0,3561 = 3,6439, \\x(3) &= [b_3 - x(2)a_1 - x(1)a_2 + x(0)a_3] = 0 - x(2)a_1 - x(1)a_2 = \\&= 0 - 3,6439 \times (-1) - 3 \times 0,3561 = 2,5756.\end{aligned}$$

Следовательно, имеем первые четыре значения обратного z -преобразования:

$$x(0) = 1; x(1) = 3; x(2) = 3,6439; x(4) = 2,5756.$$

Видно, что оба способа (рекурсивный и прямой) деления в столбик дают одинаковые решения.

Рекурсию в формуле (4.13) несложно реализовать на компьютере, как показано в следующем фрагменте кода на языке С.

```
x[0]=B[0]/A[0];
for(n=1;n<=npt;++n){
    sum=0;
    k=n;
    if(n>M)
        k=M;
    for(i=1;i<=k;++i){
        sum=sum+x[n-i]*A[i];
    }
    x[n]=(B[n]-sum)/A[0];
}
```

В этом коде M — порядок многочлена знаменателя, а npt — количество точек данных для z^{-1} . Предполагается, что в многочленах числителя и знаменателя степень z^{-1} возрастает. Программы на языке С, основанные на вышеприведенном коде, а также программы MATLAB для оценки z^{-1} даны в приложениях 4Б и 4Г соответственно.

4.3.2. Метод разложения на элементарные дроби

В этом методе z -преобразование вначале раскладывается на сумму простых дробей. Затем по таблицам, подобным табл. 4.1, находится обратное z -преобразование каждой элементарной дроби. Эти образы суммируются, и получается общее обратное z -преобразование. На практике во многих случаях z -преобразование задается как отношение многочленов по z или z^{-1} и имеет уже знакомый вид

$$\begin{aligned}X(z) &= \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_N z^{-N}}{a_0 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_M z^{-M}} = \\&= x(0) + x(1)z^{-1} + x(2)z^{-2} + x(3)z^{-3} + \dots\end{aligned}\tag{4.14}$$

Если полюсы функции $X(z)$ — первого порядка и $N = M$, то $X(z)$ можно разложить как

$$\begin{aligned} X(z) &= B_0 + \frac{C_1}{1 - p_1 z^{-1}} + \frac{C_2}{1 - p_2 z^{-1}} + \dots + \frac{C_M}{1 - p_M z^{-1}} = \\ &= B_0 + \frac{C_1 z}{z - p_1} + \frac{C_2 z}{z - p_2} + \dots + \frac{C_M z}{z - p_M} = B_0 + \sum_{k=1}^M \frac{C_k z}{z - p_k}, \end{aligned} \quad (4.15)$$

где p_k — полюсы функции $X(z)$, C_k — коэффициенты элементарных дробей, а

$$B_0 = b_N / a_N. \quad (4.16)$$

C_k также называют *вычетами* функции $X(z)$; см. раздел 4.3.3.

Если в уравнении (4.14) порядок числителя меньше, чем порядок знаменателя, т.е. $N < M$, то B_0 будет равно нулю. Если $N > M$, то $X(z)$ вначале нужно сократить, чтобы получить $N \leq M$, путем деления в столбик многочленов числителя и знаменателя, записанных через уменьшающийся показатель степени z^{-1} . Остаток можно выразить так, как это сделано в уравнении (4.15).

Коэффициент C_k , связанный с полюсом p_k , можно найти, умножив правую и левую части уравнения (4.15) на $(z - p_k)/z$, а затем сделав замену $z = p_k$:

$$C_k = \left. \frac{X(z)}{z} (z - p_k) \right|_{z=p_k}. \quad (4.17)$$

Если функция $X(z)$ имеет один или несколько полюсов больше первого порядка (т.е. совпадающих полюсов), то, чтобы учесть это, в уравнение (4.15) нужно добавить дополнительные члены. Например, если функция $X(z)$ имеет полюс m -го порядка в точке $z = p_k$, то в разложение на элементарные дроби должны входить члены вида

$$\sum_{i=1}^m \frac{D_i}{(z - p_k)^i}. \quad (4.18, a)$$

Коэффициенты D_i можно найти из зависимости

$$D_i = \frac{1}{(m - i)!} \frac{d^{m-i}}{dz^{m-i}} \left[(z - p_k)^m \frac{X(z)}{z} \right]_{z=p_k}. \quad (4.18, б)$$

Продemonстрируем оценку обратного z -преобразования методом разложения на элементарные дроби на примерах.

Пример 4.4

Функция $X(z)$ содержит простые полюсы первого порядка. Найдите ее обратное z -преобразование.

$$X(z) = \frac{z^{-1}}{1 - 0,25z^{-1} - 0,375z^{-2}}.$$

Решение

Для простоты вначале выразим z -преобразование через положительные показатели степени z , умножив числитель и знаменатель на z^2 (наивысшую степень z):

$$X(z) = \frac{z}{z^2 - 0,25z - 0,375} = \frac{z}{(z - 0,75)(z + 0,5)}.$$

$X(z)$ имеет полюсы первого порядка в точках $z = 0,75$ и $z = -0,5$ (т.е. в каждом положении полюса находится только один полюс). Поскольку порядок числителя меньше, чем порядок знаменателя ($N < M$), разложение на элементарные дроби выглядит как

$$X(z) = \frac{z}{(z - 0,75)(z + 0,5)} = \frac{C_1 z}{z - 0,75} + \frac{C_2 z}{z + 0,5}. \quad (4.19)$$

Чтобы упростить поиск значений C_k , разделим правую и левую стороны на z :

$$\frac{X(z)}{z} = \frac{z}{z(z - 0,75)(z + 0,5)} = \frac{C_1}{z - 0,75} + \frac{C_2}{z + 0,5}. \quad (4.20)$$

Чтобы найти C_1 , просто умножим правую и левую части уравнения (4.20) на $z - 0,75$ и сделаем замену $z = 0,75$:

$$\begin{aligned} \frac{(z - 0,75)X(z)}{z} &= \frac{(z - 0,75)}{(z - 0,75)(z + 0,5)} = C_1 + \frac{C_2(z - 0,75)}{z + 0,5} \\ C_1 &= \left. \frac{1}{z + 0,5} \right|_{z=0,75} = \frac{1}{0,75 + 0,5} = \frac{4}{5}. \end{aligned}$$

Аналогично находим C_2 :

$$\begin{aligned} C_2 &= \left. \frac{(z + 0,5)X(z)}{z} \right|_{z=-0,5} = \\ &= \left. \frac{(z + 0,5)}{(z - 0,75)(z + 0,5)} \right|_{z=-0,5} = \frac{1}{-0,5 - 0,75} = -\frac{4}{5}. \end{aligned} \quad (4.21)$$

Подставив значения C_1 и C_2 в уравнение (4.19), получим

$$X(z) = \frac{(4/5)z}{z - 0,75} + \frac{(-4/5)z}{z + 0,5}.$$

Согласно таблице z -преобразований (строка 14 табл. 4.1) обратное z -преобразование каждого члена правой части уравнения (4.21) задается как

$$\begin{aligned} Z^{-1} \left[\frac{(4/5)z}{z - 0,75} \right] &= \frac{4(0,75)^n}{5}, \\ Z^{-1} \left[-\frac{(4/5)z}{z + 0,5} \right] &= \frac{4(-0,5)^n}{5}. \end{aligned}$$

Искомое обратное z -преобразование $x(n)$ — это сумма двух обратных z -преобразований:

$$x(n) = \frac{4}{5}[(0,75)^n - (-0,5)^n], n > 0.$$

Пример 4.5

Функция $X(z)$ имеет комплексно-сопряженные полюсы первого порядка. Воспользовавшись методом разложения на элементарные дроби, найдите сигнал дискретного времени $x(n)$ по его z -образу

$$X(z) = \frac{1 + 2z^{-1} + z^{-2}}{1 - z^{-1} + 0,3561z^{-2}}.$$

Решение

Вначале $X(z)$ выражается через положительные степени z :

$$X(z) = \frac{N(z)}{D(z)} = \frac{z^2 + 2z + 1}{z^2 - z + 0,3561}.$$

Полюсы функции $X(z)$ находятся из решения квадратного уравнения $D(z) = z^2 - z + 0,3561 = 0$ с помощью формул

$$\begin{aligned} p_1 &= \frac{-b + (b^2 - 4ac)^{1/2}}{2a}, \\ p_2 &= \frac{-b - (b^2 - 4ac)^{1/2}}{2a}, \end{aligned} \quad (4.22)$$

где a и b — коэффициенты при z^2 и z соответственно, а c — постоянный член. При $a = 1$, $b = -1$ и $c = 0,3561$ полюсы будут равны:

$$\begin{aligned} p_1 &= \frac{-1 + (1 - 4 \times 0,3561)^{1/2}}{2} = \\ &= 0,5 + 0,3257i = re^{i\theta} \\ p_2 &= p_1^* = 0,5 - 0,3257i = re^{-i\theta}, \end{aligned}$$

где $r = 0,5967$, а $\theta = 33,08^\circ$. Теперь можно выразить функцию $X(z)$ через ее полюсы:

$$X(z) = \frac{z^2 + 2z + 1}{(z - p_1)(z - p_1^*)}.$$

Поскольку числитель и знаменатель функции $X(z)$ одного порядка, разложение на элементарные дроби будет выглядеть так:

$$\frac{X(z)}{z} = \frac{B_0}{z} + \frac{C_1}{z - p_1} + \frac{C_2}{z - p_1^*}. \quad (4.23)$$

Из уравнения (4.16) $B_0 = 1/0,3561 = 2,8082$. Чтобы найти C_1 , умножим обе стороны уравнения (4.23) на $(z - p_1)$, а затем сделаем замену $z = p_1$:

$$\frac{(z - p_1)X(z)}{z} = \frac{B_0(z - p_1)}{z} + C_1 + \frac{C_2(z - p_1)}{z - p_2} \Big|_{z=p_1}.$$

Таким образом,

$$\begin{aligned} C_1 &= \frac{(z - p_1)X(z)}{z} = \frac{(z - p_1)(z^2 + 2z + 1)}{z(z - p_1)(z - p_2)} \Big|_{z=p_1=re^{i\theta}} \\ &= \frac{(re^{i\theta})^2 + 2re^{i\theta} + 1}{re^{i\theta}(re^{i\theta} - re^{-i\theta})}, \end{aligned} \quad (4.24)$$

где $r = 0,5967$, $\theta = 33,08^\circ$. Выполнив некоторые преобразования и упростив выражение, получаем:

$$\begin{aligned} C_1 &= \frac{2,1439 + 0,97719i}{-0,2122 + 0,3257i} = \\ &= -0,9040999 - 5,992847i = \\ &= 6,06066 \angle -98,58^\circ. \end{aligned}$$

Поскольку p_1 и p_2 — комплексно-сопряженная пара, то

$$C_2 = C_1^* = -0,9040999 + 5,992847i = 6,06066 \angle 98,58^\circ.$$

Таким образом, z -преобразование можно выразить так (см. уравнение (4.23)):

$$X(z) = 2,8082 + \frac{C_1 z}{z - p_1} + \frac{C_2 z}{z - p_1^*}, \quad (4.25)$$

где

$$\begin{aligned} p_1 &= 0,5 + 0,3257ip_2 = 0,5 - 0,3257i, \\ C_1 &= -0,9041 - 5,59928iC_2 = -0,9041 + 5,59928i. \end{aligned}$$

Согласно таблице z -преобразований (строки 1 и 16 табл. 4.1) обратное z -преобразование членов правой части уравнения (4.25) имеет вид

$$\begin{aligned} Z^{-1}(2,8082) &= 2,8082u(n) \\ Z^{-1} \left[\frac{C_1 z}{z - p_1} + \frac{C_2 z}{z - p_1^*} \right] &= 2 \times 6,06066(0,5967)^n \cos(33,08n - 98,58^\circ) = \\ &= 12,1213(0,5967)^n \cos(33,08n - 98,58^\circ). \end{aligned}$$

Таким образом, сигнал дискретного времени выражается так:

$$x(n) = 2,8082u(n) + 12,1213(0,5967)^n \cos(33,08n - 98,58^\circ), \quad n \geq 0.$$

Полезно проверить результаты метода разложения на элементарные дроби, вычислив значения $x(n)$ для, скажем, $n = 0, 1, 2$, а затем сравнив их со значениями, полученными методом разложения в степенной ряд. Например, из выражения для $x(n)$ находим, что

$$x(0) = 2,8082 - 1,80838 = 1; x(1) = 2,99959 = 3; x(2) = 3,6436,$$

что совпадает с результатами, полученными в примере 4.3 методом разложения в степенной ряд.

Пример 4.6

Функция $X(z)$ содержит полюс второго порядка. Найдите дискретную во времени последовательность $X(n)$ со следующим z -образом:

$$X(z) = \frac{z^2}{(z - 0,5)(z - 1)^2}.$$

Решение

У функции $X(z)$ есть полюс первого порядка в точке $z = 0,5$ и полюс второго порядка в точке $z = 1$. В этом случае разложение на элементарные дроби выглядит как

$$X(z) = \frac{C}{z - 0,5} + \frac{D_1}{z - 1} + \frac{D_2}{(z - 1)^2}. \quad (4.26)$$

Чтобы найти C , будем действовать так же, как и ранее, и умножим обе стороны уравнения (4.26) на $z - 0,5$, положим $z = 0,5$ и оценим выражение

$$\begin{aligned} C &= \left. \frac{(z - 0,5)z^2}{z(z - 0,5)(z - 1)^2} \right|_{z=0,5} = \\ &= 0,5/(0,5 - 1)^2 = 2. \end{aligned}$$

Чтобы найти D_1 , воспользуемся уравнением (4.18, б) при $i = 1$ и $m = 2$. Таким образом,

$$\begin{aligned} D_1 &= \frac{d}{dz} \left[\frac{(z - 1)^2 X(z)}{z} \right]_{z=1} = \frac{d}{dz} \left[\frac{(z - 1)^2 z^2}{z(z - 0,5)(z - 1)^2} \right]_{z=1} = \\ &= \frac{d}{dz} \left(\frac{z}{z - 0,5} \right)_{z=1} = \left. \frac{z - 0,5 - z}{(z - 0,5)^2} \right|_{z=1} = -2. \end{aligned}$$

Аналогично D_2 находится из уравнения (4.18, б) при подстановке $i = 2$ и $m = 2$:

$$\begin{aligned} D_2 &= \left. \frac{(z - 1)^2 X(z)}{z} \right|_{z=1} = \left. \frac{(z - 1)^2 z^2}{z(z - 0,5)(z - 1)^2} \right|_{z=1} = \\ &= 1/(1 - 0,5) = 2. \end{aligned}$$

Объединив результаты, получим $X(z)$:

$$X(z) = \frac{2z}{z - 0,5} - \frac{2z}{z - 1} + \frac{2z}{(z - 1)^2}.$$

Обратное z -преобразование каждого члена из правой части уравнения можно найти по табл. 4.1, а затем просуммировать их, что даст $x(n)$:

$$x(n) = 2(0,5)^n - 2 + 2n = 2[(n - 1) + (0,5)^n], \quad n \geq 0. \quad (4.27)$$

Читатель может убедиться в правильности полученного результата, сравнив первые несколько значений последовательности $x(n)$ со значениями, найденными методом разложения в степенной ряд.

Согласитесь, метод разложения на элементарные дроби очень трудоемкий, за исключением простых случаев, и, применяя его, очень легко ошибиться. Для удобства в приложении приведены программы на языке C и MATLAB для вычисления обратного z -преобразования с помощью разложения на элементарные дроби функции $X(z)$ с полюсами первого порядка.

4.3.3. Метод вычетов

В этом методе z^{-1} находится путем вычисления контурного интеграла

$$x(n) = \frac{1}{2\pi i} \oint_C z^{n-1} X(z) dz, \quad (4.28)$$

где C — это контур интегрирования, охватывающий все полюсы функции $X(z)$. Для рациональных многочленов контурный интеграл из уравнения (4.28) находится с помощью фундаментального результата теории комплексных переменных, называемого *теоремой Коши о вычетах* (или *теоремой вычетов*) [3]:

$$\begin{aligned} x(n) &= \frac{1}{2\pi i} \oint_C z^{n-1} X(z) dz = \\ &= \text{сумма вычетов } z^{n-1} X(z) \text{ во всех полюсах внутри } C. \end{aligned} \quad (4.29)$$

В предыдущем разделе упоминалось, что коэффициенты разложения на элементарные дроби C_k еще называют *вычетами* функции $X(z)$, и приводился способ вычисления их значений. Главное, что нужно запомнить, — это то, что каждый вычет C_k связан с полюсом p_k . В настоящем методе вычеты функции $z^{n-1} X(z)$ в полюсе p_k (но не вычеты функции $X(z)$) задаются как

$$\text{Res}[F(z), p_k] = \frac{1}{(m-1)!} \frac{d^{m-1}}{dz^{m-1}} [(z - p_k) F(z)]_{z=p_k}, \quad (4.30)$$

где $F(z) = z^{n-1} X(z)$, m — порядок полюса в точке p_k , а $\text{Res}[F(z), p_k]$ — вычет $F(z)$ в точке $z = p_k$. Для простого (отдельного) полюса уравнение (4.30) сводится к

$$\text{Res}[F(z), p_k] = (z - p_k) F(z) = (z - p_k) z^{n-1} X(z) \Big|_{z=p_k}. \quad (4.31)$$

Пример 4.7

С помощью метода вычетов найдите сигнал дискретного времени, соответствующий следующему z -образу (тот же, что и в примере 4.4):

$$X(z) = \frac{z}{(z - 0,75)(z + 0,5)}.$$

Предположите, что C — окружность $|z| = 1$.

Решение

В разложенном виде $X(z)$ задается как

$$X(z) = \frac{z}{(z - 0,75)(z + 0,5)}.$$

Если положить $F(z) = z^{n-1}X(z)$, то

$$\begin{aligned} F(z) &= \frac{z^{n-1}z}{(z - 0,75)(z + 0,5)} = \\ &= \frac{z^n}{(z - 0,75)(z + 0,5)}. \end{aligned}$$

Функция $F(z)$ имеет полюсы в точках $z = 0,75$ и $z = -0,5$. Схема контура с обозначенными крестиками положениями полюсов дана на рис. 4.3. Оба полюса лежат внутри контура интегрирования (единичной окружности). Из уравнения (4.29) обратное z -преобразование задается как

$$x(n) = \text{Res}[F(z), 0,75] + \text{Res}[F(z), -0,5].$$

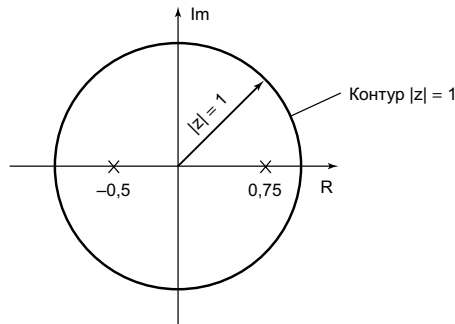
Поскольку функция имеет полюсы первого порядка, воспользуемся уравнением (4.31). Таким образом,

$$\begin{aligned} \text{Res}[F(z), 0,75] &= (z - 0,75)F(z)|_{z=0,75} = \frac{(z - 0,75)z^n}{(z - 0,75)(z + 0,5)} \Big|_{z=0,75} = \\ &= \frac{(0,75)^n}{0,75 + 0,5} = \frac{4}{5}(0,75)^n \\ \text{Res}[F(z), -0,5] &= (z + 0,5)F(z)|_{z=-0,5} = \\ &= \frac{(z + 0,5)z^n}{(z - 0,75)(z + 0,5)} \Big|_{z=-0,5} = -\frac{4}{5}(-0,5)^n. \end{aligned}$$

Обратное z -преобразование — это сумма вычетов в точках $z = 0,75$ и $z = -0,5$:

$$x(n) = (4/5)[(0,75)^n - (-0,5)^n],$$

что идентично результату, полученному методом разложения на элементарные дроби.

Рис. 4.3. Схема контура интегрирования, демонстрирующая роль $X(z)$ **Пример 4.8**

Если полюсы функции $X(z)$ — комплексно-сопряженные, найдите с помощью метода вычетов обратное z -преобразование по данному z -образу:

$$X(z) = \frac{z^2 + 2z + 1}{z^2 - z + 0,3561}.$$

Решение

В разложенном виде $X(z)$ записывается так:

$$X(z) = \frac{z^2 + 2z + 1}{(z - p_1)(z - p_2)},$$

где $p_1 = 0,5 + 0,3557i$, а $p_2 = 0,5 - 0,3557i$, т.е. $p_2 = p_1^*$. Чтобы найти обратное z -преобразование, найдем вычеты $F(z)$. В нашем случае это

$$F(z) = z^{n-1}X(z) = \frac{z^{n-1}(z^2 + 2z + 1)}{z^2 - z + 0,3561} = \frac{z^n(z^2 + 2z + 1)}{z(z^2 - z + 0,3561)}.$$

Полюсы $F(z)$ такие же, как и у $X(z)$, т.е. находятся в точках $z = p_1$ и $z = p_2$, плюс полюс в точке $z = 0$ при $n = 0$. Схема контура с обозначенными положениями полюсов показана на рис. 4.4. Все полюсы лежат внутри контура. Полюса в точке $z = 0$ при $n > 0$ не существует, поэтому нам придется рассмотреть два отдельных случая.

При $n = 0$ $F(z)$ сводится к

$$F(z) = \frac{z^2 + 2z + 1}{z(z^2 - z + 0,3561)}$$

и

$$x(0) = \text{Res}[F(z), 0] + \text{Res}[F(z), p_1] + \text{Res}[F(z), p_2].$$

Следовательно,

$$\begin{aligned}
 \operatorname{Res}[F(z), 0] &= zF(z)|_{z=0} = \\
 &= \frac{z(z^2 + 2z + 1)}{z(z^2 - z + 0,3561)} \Big|_{z=0} = \\
 &= 1/0,3561 = 2,8082 \\
 \operatorname{Res}[F(z), p_1] &= (z - p_1)F(z)|_{z=p_1} = \\
 &= \frac{(z - p_1)(z^2 + 2z + 1)}{z(z - p_1)(z - p_2)} = \\
 &= \frac{(re^{i\theta})^2 + 2re^{i\theta} + 1}{re^{i\theta}(re^{i\theta} - re^{-i\theta})},
 \end{aligned}$$

где $r = 0,5967$, а $\theta = 33,08^\circ$. Учитывая, что это выражение идентично уравнению (4.24), можем записать

$$\operatorname{Res}[F(z), p_1] = -0,9041 - 5,9928i.$$

Поскольку p_1 и p_2 — комплексно-сопряженная пара, то

$$\operatorname{Res}[F(z), p_2] = -0,9041 + 5,9928i.$$

Следовательно,

$$\begin{aligned}
 x(0) &= \operatorname{Res}[F(z), 0] + \operatorname{Res}[F(z), p_1] + \operatorname{Res}[F(z), p_2] = \\
 &= 2,8082 - 0,9041 - 5,9928i - 0,9041 + 5,9928i = \\
 &= 1.
 \end{aligned}$$

При $n > 0$ полюс в точке $z = 0$ исчезает, и мы имеем

$$\begin{aligned}
 F(z) &= \frac{z^n(z^2 + 2z + 1)}{z(z^2 - z + 0,3561)} \\
 x(n) &= \operatorname{Res}[F(z), p_1] + \operatorname{Res}[F(z), p_2] \\
 \operatorname{Res}[F(z), p_1] &= (z - p_1)F(z)|_{z=p_1} = \\
 &= \frac{(z - p_1)z^n(z^2 + 2z + 1)}{z(z - p_1)(z - p_2)} \Big|_{z=p_1} = \\
 &= \frac{(re^{i\theta})^n[(re^{i\theta})^2 + 2re^{i\theta} + 1]}{re^{i\theta}(re^{i\theta} - re^{-i\theta})},
 \end{aligned} \tag{4.32}$$

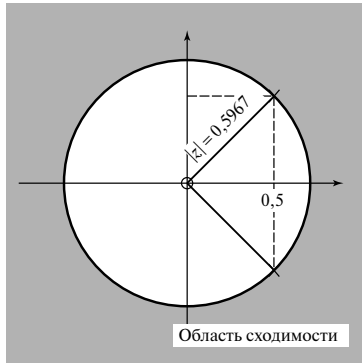


Рис. 4.4. Контур интегрирования, показывающий область сходимости (пример 4.8)

где $r = 0,5967$, а $\theta = 33,08^\circ$. Учитывая, что это выражение аналогично уравнению (4.24), можем записать

$$\begin{aligned}\text{Res}[F(z), p_1] &= (0,5967e^{i33,08})^n (6,06066e^{-i98,58}) = \\ &= 6,06066(0,5967)^n [\cos(33,08n - 98,58) + i \sin(33,08n - 98,58)].\end{aligned}$$

Поскольку p_1 и p_2 — комплексно-сопряженная пара, то

$$\text{Res}[F(z), p_2] = 6,06066(0,5967)^n [\cos(33,08n - 98,58) - i \sin(33,08n - 98,58)].$$

Следовательно,

$$\begin{aligned}x(0) &= \text{Res}[F(z), p_1] + \text{Res}[F(z), p_2] = \\ &= 12,1213(0,5967)^n \cos(33,08n - 98,58^\circ), n > 0,\end{aligned}$$

что совпадает с результатами, полученными при разложении на элементарные дроби.

Пример 4.9

Функция $X(z)$ содержит полюс второго порядка. Найдите последовательность дискретного времени $x(n)$ по следующему z -образу:

$$X(z) = \frac{z^2}{(z - 0,5)(z - 1)^2}.$$

Решение

Исходные данные задачи такие же, как в примере 4.6 разложения на элементарные дроби. Согласно методу вычетов последовательность дискретного времени задается как

$$x(n) = \sum_{k=1}^M \text{Res}[F(z), p_k],$$

где

$$F(z) = z^{n-1}X(z) = \frac{z^{n+1}}{(z - 0,5)(z - 1)^2}.$$

Функция $F(z)$ имеет простой полюс в точке $z = 0,5$ и полюс второго порядка в точке $z = 1$; таким образом, последовательность $x(n)$ задается как

$$\begin{aligned} x(n) &= \text{Res}[F(z), p_1] + \text{Res}[F(z), p_2] \\ \text{Res}[F(z), 0,5] &= \frac{(z - 0,5)z^{n+1}}{(z - 0,5)(z - 1)^2} = \\ &= \frac{z^{n+1}}{(z - 1)^2} \Big|_{z=0,5} = \\ &= 0,5(0,5)^n / (0,5)^2 = 2(0,5)^n \\ \text{Res}[F(z), 1] &= \frac{d}{dz} \left[\frac{(z - 1)^2 z^{n+1}}{(z - 0,5)(z - 1)^2} \right] = \\ &= \frac{(z - 0,5)(n+1)z^n - z^{n+1}}{(z - 0,5)^2} \Big|_{z=1} = \\ &= [(0,5)(n+1) - 1] / (0,5)^2 = 2(n-1). \end{aligned}$$

Объединив результаты, получим

$$x(n) = 2[(n-1) + (0,5)^n],$$

что аналогично результату, полученному методом разложения на элементарные дроби.

Возможно, вы заметили, что методы вычетов и разложения на элементарные дроби взаимосвязаны. И в том, и в другом методе нужно находить вычеты, хотя это делается различными способами. В методе разложения на элементарные дроби находятся вычеты функции $X(z)$, т.е. коэффициенты C_k , тогда как метод вычетов требует поиска вычетов функции $z^{n-1}X(z)$. Если функция $X(z)$ содержит полюсы первого порядка, то

$$\text{Res}[z^{n-1}X(z), p_k] = z^n \text{Res}[X(z), p_k] = z^n C_k. \quad (4.33)$$

Следовательно, программой на языке С для разложения на элементарные дроби, описанной в приложении, можно воспользоваться и для расчетов по методу вычетов.

4.3.4. Сравнение различных методов обратного z-преобразования

Мы более или менее подробно рассмотрели три метода вычисления обратного z -преобразования: разложение в степенной ряд, разложение на элементарные дроби и метод вычетов. Ограничение метода разложения в степенной ряд состоит в том, что он не дает решения в аналитическом виде (хотя в простых случаях его можно вывести),

но зато он прост и пригоден для вычислений с помощью компьютера. Однако из-за его рекурсивной природы нужно внимательно следить (например, используя двойную точность) за возможным нарастанием численных ошибок, при большом числе заданных точек обратного z -преобразования.

Как метод разложения на элементарные дроби, так и метод вычетов, дают результат в аналитическом виде. Главный недостаток этих методов — необходимость раскладывать на множители многочлен знаменателя, т.е. находить полюсы функции $X(z)$. Если порядок функции $X(z)$ высокий, то поиск полюсов $X(z)$, если функция не представлена в разложенном виде, — довольно трудная задача. Более глубоко эта тема рассмотрена в разделе 4.5.1. Если функция $X(z)$ имеет полюсы высокого порядка, оба метода могут также включать дифференцирование высокого порядка. Ясно, что если нужно найти решение в аналитическом виде, то лучше выбрать метод вычетов или разложения на элементарные дроби. Метод разложения на элементарные дроби особенно полезен для генерации коэффициентов параллельных структур для цифровых фильтров (см. раздел 4.5.11). Метод вычетов нашел широкое применение в анализе ошибок квантования в системах дискретного времени (см. главу 13).

Использование такого удобного инструмента, как программный пакет MATLAB или программы на языке C из данной книги, значительно упрощает операции вычисления z -преобразования и обратного z -преобразования. Для иллюстрации сказанного в приложениях дается несколько примеров.

4.4. Свойства z -преобразования

Ниже кратко описываются некоторые полезные свойства z -преобразования, которые нашли практическое применение в ЦОС. Доказательства некоторых из этих свойств предлагаются в качестве задач в конце этой главы.

1. *Линейность.* Если последовательности $x_1(n)$ и $x_2(n)$ имеют z -образы $X_1(z)$ и $X_2(z)$, то z -образ их линейной комбинации будет

$$ax_1(n) + bx_2(n) \rightarrow aX_1(z) + bX_2(z). \quad (4.34)$$

2. *Задержки или смещения.* Если z -образ последовательности $x(n)$ равен $X(z)$, то z -образ последовательности с задержкой на m элементов будет $z^{-m}X(z)$. Это свойство широко используется для превращения передаточной функции z систем дискретного времени в разностные уравнения во временной области, и наоборот; см. раздел 4.5.8.

$$x(n) \rightarrow X(z)$$

$$x(n - m) \rightarrow z^{-m}X(z)$$

3. *Свертка.* Дана ЛИВ-система дискретного времени со входом $x(n)$ и импульсной характеристикой $h(k)$. Выход системы задается как

$$y(n) = \sum_{k=-\infty}^{\infty} h(k)x(n - k). \quad (4.35, a)$$

Выраженные через z -образы вход и выход связаны соотношением

$$Y(z) = H(z)X(z), \quad (4.35, б)$$

где $X(z)$, $H(z)$ и $Y(z)$ — соответственно z -образы последовательностей $x(n)$, $h(k)$ и $y(n)$. При заданных $X(z)$ и $H(z)$ выход $y(n)$ можно найти с помощью обратного z -преобразования $Y(z)$.

Видно, что операция свертки в уравнении (4.35, а) превратилась в процесс умножения в z -области. Записанную выше функцию $H(z)$ часто называют *передаточной функцией системы*.

4. *Дифференцирование.* Если $X(z)$ — z -образ последовательности $x(n)$, то z -образ $nx(n)$ можно найти, продифференцировав $X(z)$:

$$\begin{aligned} x(n) &\rightarrow X(z), \\ nx(n) &\rightarrow -z \frac{dX(z)}{dz}. \end{aligned} \quad (4.36)$$

Это свойство полезно для вычисления обратного z -преобразования, когда $X(z)$ содержит полюсы высокого порядка.

5. *Взаимосвязь с преобразованием Лапласа.* Системы или сигналы непрерывного времени, как правило, описываются с помощью преобразования Лапласа. Если $z = e^{sT}$, где s — комплексная переменная Лапласа, которая задается как

$$s = d + i\omega,$$

то

$$z = e^{(d+i\omega)T} = e^{dT} e^{i\omega T}. \quad (4.37)$$

Следовательно,

$$|z| = e^{dT} \text{ и } \angle z = \omega T = 2\pi f / F_s = 2\pi\omega / \omega_s,$$

где ω_s (рад/с) — частота дискретизации. Поскольку ω пробегает значения от $-\infty$ до ∞ , s -плоскость отображается в z -плоскость так, как показано на рис. 4.5. Вся ось $i\omega$ на s -плоскости отображается в единичную окружность. Левая сторона s -плоскости отображается внутрь круга единичного радиуса, а правая сторона s -плоскости — на внешнюю сторону окружности единичного радиуса.

Если выражать это через частотную характеристику, то ось $i\omega$ имеет самое большое значение на s -плоскости. В этом случае $d = 0$ и частотные точки на s -плоскости связаны с точками единичной окружности на z -плоскости соотношением

$$z = e^{i\omega T}. \quad (4.38)$$

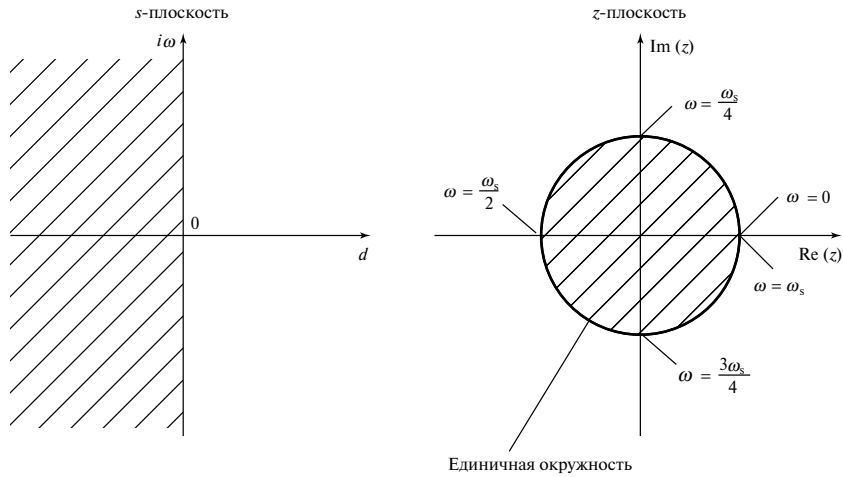


Рис. 4.5. Отображение s -плоскости в z -плоскость. Левая сторона s -плоскости отображается на внутреннюю часть z -плоскости, правая сторона отображается на внешнюю часть, а ось $i\omega$ отображается на единичную окружность

В табл. 4.2 показано, как некоторые особые частоты отображаются из s -плоскости в z -плоскость. Ясно, что такое отображение неоднозначно, поскольку, например, две частоты $\omega = \omega_s$ и $\omega = 2\omega_s$ на s -плоскости отображаются в одну и ту же точку единичной окружности.

Таблица 4.2. Отображение частот из s -плоскости в z -плоскость

<i>s-плоскость: ω (рад/с)</i>	<i>z-плоскость: ωT (рад)</i>
0	0
$\omega_s/4$	$\pi/2$
$\omega_s/2$	π
$3\omega_s/4$	$1,25\pi$
ω_s	2π
$1,25\omega_s$	$\pi/2$
$1,5\omega_s$	π
$1,75\omega_s$	$1,25\pi$
$2\omega_s$	2π

4.5. Некоторые области применения z -преобразования в обработке сигналов

Существует множество областей применения z -преобразования в ЦОС. Несколько из них подробно рассматривается в последующих главах, особенно в главе 8. Следующие несколько разделов посвящены выяснению некоторых фундаментальных понятий, общих для всех этих областей.

4.5.1. Описание систем дискретного времени с помощью полюсов и нулей

Для большинства практических систем дискретного времени z -преобразование, т.е. передаточную функцию системы $H(z)$, можно выразить через ее полюсы и нули. Рассмотрим, например, следующее z -преобразование, представляющее собой обычный фильтр дискретного времени N -го порядка (где $N = M$):

$$H(z) = \frac{N(z)}{D(z)}, \quad (4.39)$$

где

$$N(z) = b_0 z^N + b_1 z^{N-1} + b_2 z^{N-2} + \dots + b_N,$$

$$D(z) = a_0 z^N + a_1 z^{N-1} + a_2 z^{N-2} + \dots + a_N,$$

a_k и b_k — коэффициенты фильтра.

Если функция $H(z)$ имеет полюсы в точках $z = p_1, p_2, \dots, p_N$ и нули в точках $z = z_1, z_2, \dots, z_N$, то $H(z)$ можно разложить на множители и представить в виде

$$H(z) = \frac{K(z - z_1)(z - z_2) \dots (z - z_N)}{(z - p_1)(z - p_2) \dots (z - p_N)}, \quad (4.40)$$

где z_i — i -й нуль, p_i — i -й полюс, а K — коэффициент усиления. Напомним, что полюсы такого z -преобразования, как $H(z)$, — это значения z , в которых функция $H(z)$ равна бесконечности. Значения z , в которых $H(z)$ равна нулю, называют *нулями*. Полюсы и нули функции $H(z)$ могут быть действительными или комплексными. Если они комплексные, они идут комплексно-сопряженными парами, чтобы коэффициенты a_k и b_k были действительными. Из уравнения (4.40) видно, что если известны положения полюсов и нулей функции $H(z)$, то и саму функцию $H(z)$ можно легко восстановить с точностью до константы.

Информацию, содержащуюся в z -образе, можно удобно изобразить в виде диаграммы нулей и полюсов; см., например, рис. 4.6. На этой диаграмме крестиком (X) обозначены положения полюсов, а кружком (O) — положения нулей. В данном примере полюсы находятся в точках $z = 0,5 \pm 0,5i$ и $z = 0,75$, единственный нуль — в точке $z = -1$. Важной особенностью диаграммы нулей и полюсов является единичная окружность, т.е. окружность, которая задается уравнением $|z| = 1$ (см. рис. 4.6). Как вскоре станет понятно, единичная окружность играет немаловажную роль в анализе и проектировании систем дискретного времени.

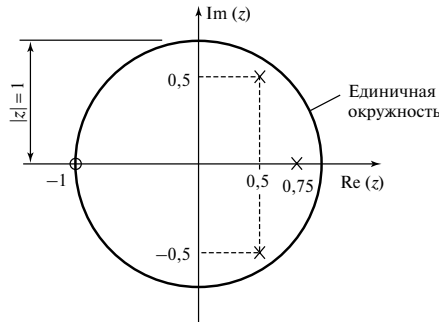


Рис. 4.6. Описание z -преобразования в виде диаграммы нулей и полюсов: X — полюс; O — нуль

Диаграмма нулей и полюсов раскрывает свойства данной системы дискретного времени. Например, из расположения полюсов и нулей можно получить частотную характеристику системы, а также ее степень устойчивости. Для устойчивых систем все полюсы должны лежать внутри единичной окружности (или совпадать с нулями на единичной окружности).

Часто z -преобразование нельзя представить в разложенном виде, его можно записать только как отношение многочленов, как в уравнении (4.39). В таких случаях для описания z -образа $H(z)$ через его нули и полюсы необходимо найти корни многочлена знаменателя $D(z)$ и многочлена числителя $N(z)$.

Корни многочлена второго порядка, который задается в виде $ax^2 + bx + c$, ищут по формуле

$$\frac{-b \pm (b^2 - 4ac)^{1/2}}{2a}. \quad (4.41)$$

Определение корней многочленов $N(z)$ и $D(z)$ более высоких порядков — задача не из легких. На практике это часто делают с помощью численных методов, в том числе, например, с помощью алгоритмов Ньютона и/или Бэйстоу (Bairstow) (см., например, [1]). Необходимость поиска нулей и полюсов часто возникает в связи с проектированием цифровых фильтров и анализом устойчивости. К счастью, при проектировании фильтров дискретного времени полюсы и нули автоматически выдаются программой проектирования фильтров, избавляя от необходимости находить корни многочленов вручную.

Пример 4.10

1. Выразите следующую передаточную функцию через ее полюсы и нули и постройте диаграмму нулей и полюсов:

$$H(z) = \frac{1 - z^{-1} - 2z^{-2}}{1 - 1,75z^{-1} + 1,25z^{-2} - 0,375z^{-3}}.$$

2. Найдите передаточную функцию $H(z)$ фильтра дискретного времени, диаграмма нулей и полюсов которого показана на рис. 4.7.

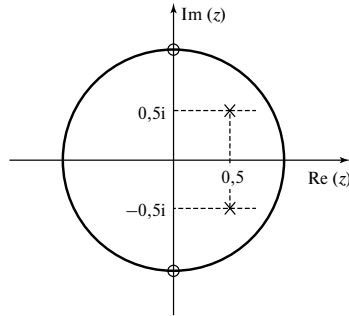


Рис. 4.7. Диаграмма нулей и полюсов для примера 4.10, п. 2

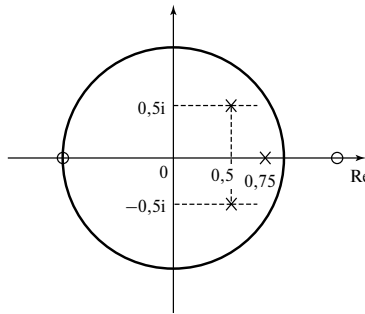


Рис. 4.8. Диаграмма нулей и полюсов для примера 4.10

Решение

- Во-первых, выразим $H(z)$ через положительные показатели степени z , а затем разложим ее таким образом, чтобы можно было найти полюсы и нули. Если умножить числитель и знаменатель на z^3 — самую высокую степень z , получится

$$H(z) = \frac{z^3 - 2z - 2z}{z^3 - 1,75z^2 + 1,25z - 0,375}.$$

В результате разложения получаем

$$H(z) = \frac{(z - 2)(z + 1)z}{(z - 0,5 + i0,5)(z - 0,5 - i0,5)(z - 0,75)}.$$

Следовательно, полюсы находятся в точках $z = 0,5 \pm 0,5i$ и в точке $z = 0,75$. Нули — в точках $z = 2$, $z = -1$ и $z = 0$. Соответствующая диаграмма нулей и полюсов изображена на рис. 4.8.

2. Согласно диаграмме нулей и полюсов нули передаточной функции находятся в точках $z = \pm i$, а полюсы — в точках $z = 0,5 \pm 0,5i$. Далее можно записать непосредственно саму передаточную функцию:

$$\begin{aligned} H(z) &= \frac{K(z-i)(z+i)}{(z-0,5-0,5i)(z-0,5+0,5i)} = \\ &= \frac{K(z^2+1)}{z^2-z+0,5} = \\ &= \frac{K(1+z^{-2})}{1-z^{-1}-0,5z^{-2}}. \end{aligned}$$

4.5.2. Оценка частотной характеристики

Существует множество случаев, когда нужно оценить частотную характеристику системы дискретного времени. Например, при проектировании дискретных фильтров часто приходится проверять спектр фильтра, чтобы убедиться в том, что он удовлетворяет искомым спецификациям. Отметим, что частотную характеристику системы можно запросто найти из ее z -преобразования.

Например, если взять $z = e^{i\omega T}$, т.е. найти z -преобразование по единичной окружности, получим Фурье-образ системы:

$$\begin{aligned} H(z) &= \sum_{n=-\infty}^{\infty} h(n)z^{-n} \Big|_{z=e^{-i\omega T}} = \\ &= H(e^{i\omega T}) = \sum_{n=-\infty}^{\infty} h(n)e^{-in\omega T}. \end{aligned} \quad (4.42)$$

$H(e^{i\omega T})$ называют частотной характеристикой системы. Мы воспользовались символом T , чтобы подчеркнуть зависимость частотной характеристики системы дискретного времени от частоты дискретизации. В общем случае $H(e^{i\omega T})$ — комплексная величина. Ее модуль дает амплитудную, а фаза — фазовую характеристику системы.

Частотную характеристику по z -преобразованию можно найти несколькими методами, три из них описаны ниже.

4.5.3. Геометрическая оценка

Это простой, но полезный метод, основанный на диаграмме нулей и полюсов, с помощью которого можно получить приблизительное представление о том, как выглядит частотная характеристика системы дискретного времени. Напомним, что z -преобразования ЛИВ-системы можно выразить через ее полюсы и нули:

$$H(z) = \frac{K(z-z_1)(z-z_2)\dots(z-z_N)}{(z-p_1)(z-p_2)\dots(z-p_N)} = \frac{\prod_{i=1}^N K(z-z_i)}{\prod_{i=1}^N K(z-p_i)}, \quad (4.43)$$

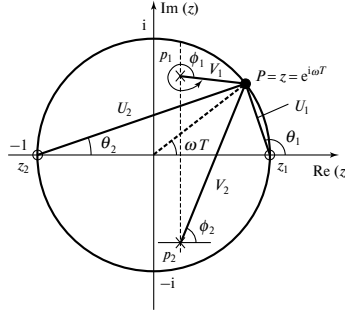


Рис. 4.9. Геометрическая оценка частотной характеристики по диаграмме нулей и полюсов

где для простоты предполагается, что порядки числителя и знаменателя совпадают. Чтобы найти частотную характеристику, подставим $z = e^{i\omega T}$ в уравнение (4.43) и вычислим $H(e^{i\omega T})$ на отрезке $(0 \leq \omega \leq \omega_s/2)$.

$$H(e^{i\omega T}) = \frac{\prod_{i=1}^N K(e^{i\omega T} - z_i)}{\prod_{i=1}^N K(e^{i\omega T} - p_i)}, \quad (4.44)$$

Геометрическая интерпретация уравнения (4.44) для z -преобразования со всего лишь двумя нулями и двумя полюсами показана на рис. 4.9. В этом случае частотная характеристика задается как

$$\begin{aligned} H(e^{i\omega T}) &= \frac{K(e^{i\omega T} - z_1)(e^{i\omega T} - z_2)}{(e^{i\omega T} - p_1)(e^{i\omega T} - p_2)} = \\ &= \frac{KU_1 \angle \theta_1 U_2 \angle \theta_2}{V_1 \angle \phi_1 V_2 \angle \phi_2}, \end{aligned} \quad (4.45)$$

где U_1 и U_2 представляют расстояния от нулей до точки $z = e^{i\omega T}$, а V_1 и V_2 — расстояния от полюсов до этой же точки, как показано на рис. 4.9. Следовательно, амплитудная и фазовая характеристики системы согласно уравнению (4.45) равны

$$\begin{aligned} |H(e^{i\omega T})| &= \frac{U_1 U_2}{V_1 V_2}, K = 1 \\ \angle[H(e^{i\omega T})] &= \theta_1 + \theta_2 - (\phi_1 + \phi_2). \end{aligned}$$

Для того чтобы найти частотную характеристику, вычислим $H(e^{i\omega T})$ при перемещении точки P от $z = 0$ до $z = -1$. Очевидно, что при перемещении точки P ближе к полюсу p_1 длина вектора V_1 уменьшается, следовательно, амплитудная характеристика растет. С другой стороны, при перемещении точки P ближе к нулю z_1 вектор нуля U_1 уменьшается, и поэтому амплитудная характеристика $|H(e^{i\omega T})|$ увеличивается. Таким

образом, в полюсе амплитудная характеристика имеет максимальное значение, тогда как в нуле амплитудная характеристика спадает до нуля.

В общем случае в геометрическом методе частотная характеристика при заданной частоте ω (при угле ωT) определяется отношением произведения векторов нулей $U_i \angle \theta_i$, $i = 1, 2, \dots$ к произведению векторов полюсов $V_i \angle \phi_i$, $i = 1, 2, \dots$

Пример 4.11

Найдите геометрическим методом частотную характеристику на постоянной составляющей и при частотах, равных $1/8$, $1/4$, $3/8$ и $1/2$ частоты дискретизации, причинной системы дискретного времени со следующим z -образом:

$$H(z) = \frac{z + 1}{z - 0,7071}.$$

Изобразите амплитудно-частотную характеристику на отрезке $0 \leq \omega \leq \omega_s$, где ω_s (рад/с) — частота дискретизации.

Решение

В этом примере функция $H(z)$ имеет единственный полюс и единственный нуль, как показано на диаграмме нулей и полюсов на рис. 4.10, а. Согласно уравнению (4.44) характеристика при частоте ω задается как

$$H(e^{i\omega T}) = \frac{U \angle \theta}{V \angle \phi} = \frac{e^{i\omega T} + 1}{e^{i\omega T} - 0,7071} = \frac{1 + \cos(\omega T) + i \sin(\omega T)}{\cos(\omega T) - 0,7071 + i \sin(\omega T)}. \quad (4.46)$$

На постоянной составляющей $\omega T = 0$, а векторы полюсов и нулей до точки $z = 0$ равны $2 \angle 0^\circ$ и $0,2929 \angle 0^\circ$. Следовательно, частотная характеристика задается как

$$H(e^{i\omega T}) = 2/0,2929 = 6,828 \angle 0^\circ.$$

При $\omega = \omega_s/8$ $\omega T = \omega_s/8F_s = \pi/4$. Векторы полюсов и нулей в этом случае показаны на рис. 4.10, б. Далее воспользуемся не действительными измерениями углов и длин векторов, а явным выражением из правой части уравнения (4.46). Таким образом,

$$\begin{aligned} H(e^{i\omega T}) &= \frac{1 + \cos(\pi/4) + i \sin(\pi/4)}{\cos(\pi/4) - 0,7071 + i \sin(\pi/4)} = \\ &= \frac{1,8477 \angle 22,5^\circ}{0,7071 \angle 90^\circ} = 2,6131 \angle -67,5^\circ. \end{aligned}$$

Ниже приведены характеристики при остальных частотах, полученные аналогичным образом, а соответствующие векторы даны на рис. 4.10, в–д.

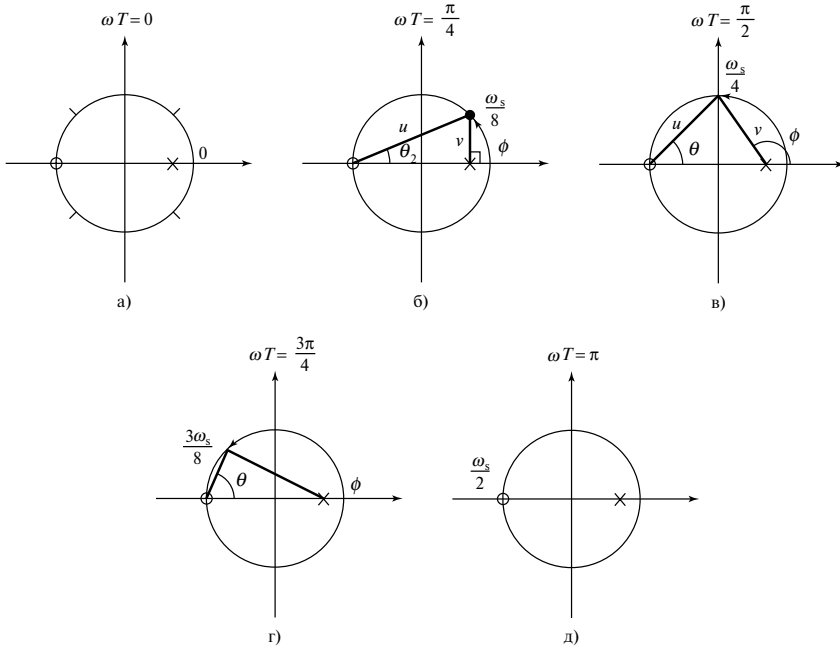


Рис. 4.10. Оценка частотной характеристики с помощью геометрического метода и диаграмма нулей и полюсов

ω (в рад/с)	ωT (в рад)	$ H(e^{j\omega T}) $	$\angle H(e^{j\omega T})$ (в градусах)
0	0	6,828	0
$\omega_s/8$	$\pi/4$	2,6131	-67,5
$\omega_s/4$	$\pi/2$	1,1547	-80,26
$3\omega_s/8$	$3\pi/4$	0,4840	-85,93
$\omega_s/2$	π	0	0

Схематическое изображение амплитудной и фазовой характеристики показано на рис. 4.11. Следует отметить один важный момент — амплитудная характеристика $|H(e^{j\omega T})|$ симметрична относительно половины частоты дискретизации (частоты Найквиста), а фазовая характеристика антисимметрична относительно этой же частоты. Это всегда так, когда коэффициенты a_k и b_k системы дискретного времени действительные. Более того, частотная характеристика таких систем периодична с периодом ω_s (частота дискретизации). Отметим, что такое поведение согласуется с теоремой о дискретном представлении.

4.5.4. Непосредственный компьютерный расчет частотной характеристики

Геометрический расчет частотной характеристики дает нам представление о частотной характеристике, но понятно, что он может оказаться очень трудоемким, если нужно знать точную характеристику на многих частотах. Несмотря на то что этот процесс

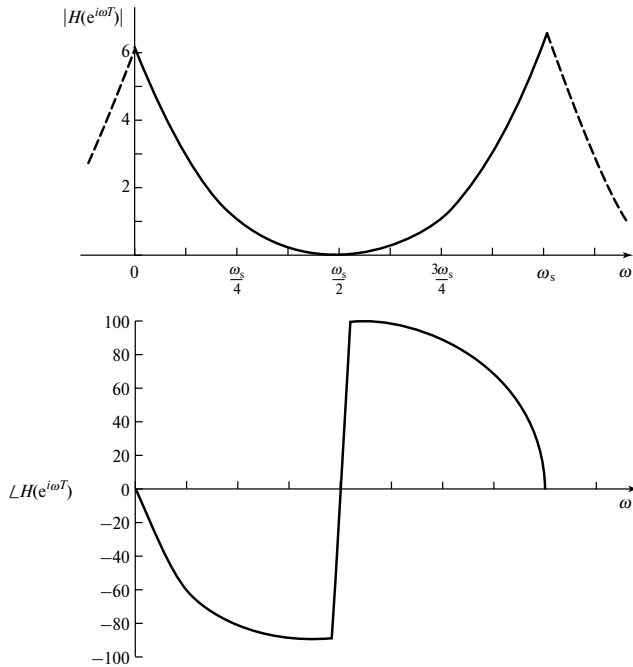


Рис. 4.11. Схематическое изображение частотной характеристики системы дискретного времени из примера 4.11

можно автоматизировать, его пригодность ограничивает сложность определения положения полюсов и нулей. Если необходимо знать полную частотную характеристику, как правило, в передаточную функцию непосредственно подставляют значение $z = e^{j\omega T}$ и вычисляют получающееся в результате этого выражение:

$$H(e^{j\omega T}) = \frac{b_0 + b_1 z^{-1} + \dots + b_N z^{-N}}{a_0 + a_1 z^{-1} + \dots + a_M z^{-M}} \Big|_{z=e^{j\omega T}} \quad (4.47)$$

$$= \frac{b_0 + b_1 e^{-j\omega T} + \dots + b_N e^{-jN\omega T}}{a_0 + a_1 e^{-j\omega T} + \dots + a_M e^{-jM\omega T}} =$$

$$= \frac{b_0 + b_1 [\cos(\omega T) - i \sin(\omega T)] + \dots + b_N [\cos(N\omega T) - i \sin(N\omega T)]}{a_0 + a_1 [\cos(\omega T) - i \sin(\omega T)] + \dots + a_M [\cos(M\omega T) - i \sin(M\omega T)]}. \quad (4.48)$$

Реализация уравнения (4.48) на языке С обсуждается в приложении 4Б. Программа вычисляет $H(e^{j\omega T})$ на отрезке $(0 \leq \omega \leq \omega_s/2)$. В приложении 4Г описано использование программного пакета MATLAB с примерами вычисления частотной характеристики.

4.5.5. Вычисление частотной характеристики с помощью БПФ

Для оценки частотной характеристики систем дискретного времени также можно использовать БПФ. Для БИХ-систем это делается следующим образом: сперва находится импульсная характеристика системы, например, методом разложения в степенной ряд, а затем вычисляется БПФ импульсной характеристики. Такая последовательность действий связана непосредственно с уравнением (4.42, б), которое показывает, что частотная характеристика системы дискретного времени — это просто Фурье-образ ее импульсной характеристики. Чтобы частотная характеристика получилась гладкой, прежде, чем искать БПФ, важно взять достаточное количество значений импульсной характеристики и/или дополнить значения импульсной характеристики нулями. Реализации этого алгоритма на языке С и в программном пакете MATLAB обсуждаются в приложениях к данной главе.

Альтернативный метод — вначале дополнить нулями числитель и знаменатель, например,

$$\begin{aligned}\{b(n)\} &= \{b_0, b_1, b_2, \dots, b_M, 0, 0, \dots, 0\}, \\ \{a(n)\} &= \{a_0, a_1, a_2, \dots, a_N, 0, 0, \dots, 0\},\end{aligned}\tag{4.49}$$

а затем найти БПФ последовательностей $\{b(n)\}$ и $\{a(n)\}$, $A(k)$ и $B(k)$ соответственно. Отношение этих двух БПФ и даст частотную характеристику:

$$H(e^{i\omega T}) = A(k)/B(k), k = 0, 1, \dots, N/2.\tag{4.50}$$

4.5.6. Единицы измерения частоты, которые используются в системах дискретного времени

Системы или сигналы непрерывного времени, как правило, описываются с помощью преобразования Лапласа. Следовательно, частотная характеристика системы непрерывного времени традиционно оценивается путем подстановки $s = i\omega$ в передаточную функцию системы $H(s)$, где s — комплексная переменная Лапласа. В ЦОС мы имеем дело с системами и сигналами дискретного времени. В этом случае частотную характеристику ищут путем подстановки $z = e^{i\omega T}$ и последующего вычисления функции z -преобразования $H(z)$ на отрезке $0 \leq \omega \leq \omega_s/2$. Здесь стоит остановиться на ключевом моменте систем дискретного времени — зависимости эффективного интервала частот от частоты дискретизации ω_s .

В табл. 4.3 показано, как изменяются ωT и z при изменении ω от 0 до ω_s . Можно показать, что если угол ωT пробегает значения от 0 до 2π , то значение z изменяется от 1 до i и обратно до 1. Эту информацию в графическом виде можно найти на рис. 4.12. Из рисунка также очевидно, что частотная характеристика системы дискретного времени циклична: если пройти полный круг один или несколько раз, то значения z будут просто повторяться.

Как правило, для описания частотной характеристики систем дискретного времени пользуются двумя единицами измерения частоты — ω (рад/с) и f (Гц). Если частота измеряется в рад/с, частотная характеристика пробегает значения от $\omega = 0$ до $\omega =$

Таблица 4.3. Единицы частоты, которые используются в системах дискретного времени, и их связь с точками на единичной окружности

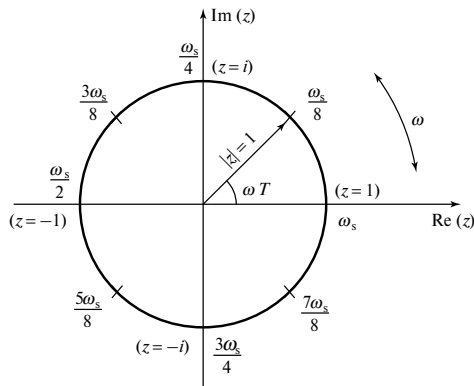
f (Гц)	ω (рад/с)	ωT (рад)	$z = e^{i\omega T}$
0	0	0	1
$\frac{F_s}{8}$	$\frac{\omega_s}{8}$	$\frac{\pi}{4}$	$\frac{\sqrt{2}}{2} + \frac{\sqrt{2}}{2}i$
$\frac{F_s}{4}$	$\frac{\omega_s}{4}$	$\frac{\pi}{2}$	i
$\frac{3F_s}{8}$	$\frac{3\omega_s}{8}$	$\frac{3\pi}{4}$	$-\frac{\sqrt{2}}{2} + \frac{\sqrt{2}}{2}i$
$\frac{F_s}{2}$	$\frac{\omega_s}{2}$	π	-1
$\frac{5F_s}{8}$	$\frac{5\omega_s}{8}$	$\frac{5\pi}{4}$	$-\frac{\sqrt{2}}{2} - \frac{\sqrt{2}}{2}i$
$\frac{3F_s}{4}$	$\frac{3\omega_s}{4}$	$\frac{3\pi}{2}$	$-i$
$\frac{7F_s}{8}$	$\frac{7\omega_s}{8}$	$\frac{7\pi}{4}$	$\frac{\sqrt{2}}{2} - \frac{\sqrt{2}}{2}i$
F_s	ω_s	2π	1

$F_s = 1/T$ — частота дискретизации в Гц; T — период дискретизации; $\omega_s = 2\pi/T$ — частота дискретизации в рад/с.

$\omega_s/2$ или, что эквивалентно, от $\omega = 0$ до $\omega = \pi/T$ (поскольку $\omega_s = 2\pi F_s = 2\pi/T$). Если пользоваться стандартной единицей измерения частоты, которая выражается в герцах, частотный диапазон будет от 0 до $F_s/2$, или от 0 до $1/2T$. Обе эти единицы измерения частоты можно записать в нормированном виде, т.е. при $T = 1$ или, что эквивалентно, $F_s = 1$. Взаимосвязь между этими двумя единицами измерения частоты показана в табл. 4.3. Следовательно, представляющие интерес частотные интервалы можно выразить одним из шести следующих эквивалентных способов:

$$\left. \begin{aligned} 0 \leq \omega \leq \omega_s/2 & \quad (\text{рад/с}) \\ 0 \leq \omega \leq \pi/T & \quad (\text{рад/с}) \\ 0 \leq \omega \leq \pi & \quad (\text{нормированный}) \end{aligned} \right\} \quad (4.51)$$

$$\left. \begin{aligned} 0 \leq f \leq F_s/2 & \quad (\text{Гц}) \\ 0 \leq f \leq 1/2T & \quad (\text{Гц}) \\ 0 \leq f \leq 1/2 & \quad (\text{нормированный}) \end{aligned} \right\} \quad (4.52)$$

**Рис. 4.12.** Единичная окружность z -плоскости, на которой показаны критические точки частоты

Измерение частоты в герцах больше привлекает (и меньше запутывает), если пользоваться графиками частотной характеристики или спецификацией системы дискретного времени. Однако при оценке численных математических формул в ЦОС удобнее пользоваться величинами, выраженными в рад/с.

Пример 4.12

Дано описание частотной характеристики полосового фильтра дискретного времени в виде

полоса пропускания 6–10 кГц,
 полосы подавления 0–4 и 12–16 кГц,
 частота дискретизации 32 кГц.

- А. Выразите спецификацию через нормированную частоту f .
- Б. Переведите спецификацию из стандартных единиц (Гц) в рад/с.
- В. Переведите спецификацию из рад/с (п. Б) в нормированную частоту ω .

Решение

1. Граничные частоты, которые измеряются в Гц, можно записать в нормированном виде, просто разделив каждую из них на частоту дискретизации. Таким образом, спецификация в нормированном виде будет выглядеть так:

полоса пропускания 0,1875–0,3125,
 полосы подавления 0–0,125 и 0,375–0,5,
 частота дискретизации 1.

2. Поскольку $\omega = 2\pi f$, для того, чтобы перейти к рад/с, каждая граничная частота просто умножается на 2π . Теперь спецификация частотной характеристики будет следующей:

полоса пропускания 12000π – 20000π рад/с,
 полосы подавления 0 – 8000π и 24000π – 32000π рад/с,
 частота дискретизации 64000π рад/с.

3. Граничные частоты из п. Б можно записать в нормированном виде, разделив каждую из них на 32 кГц (частоту дискретизации), например,

$$12000\pi \rightarrow \frac{12000\pi}{32000} = \frac{3\pi}{8}.$$

Таким образом, спецификация приводится к виду:

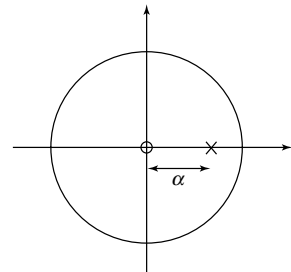
полоса пропускания $3\pi/8$ – $5\pi/8$,
 полосы подавления 0 – 4π и $3\pi/4$ – π ,
 частота дискретизации 2π .

4.5.7. Исследование устойчивости

Зачастую для разработки систем дискретного времени необходим анализ устойчивости. Полезный достаточный критерий устойчивости ЛИВ-систем можно сформулировать так: все ограниченные входные сигналы должны давать ограниченные выходные

	$h(n)$			
n	$\alpha = 0,5$	$\alpha = 0,99$	$\alpha = 1$	$\alpha = 1,5$
0	0,00000e+00	0,00000e+00	0,00000e+00	0,00000e+00
1	1,00000e+01	1,00000e+01	1,00000e+01	1,00000e+01
2	5,00000e+00	9,90000e+00	1,00000e+01	1,50000e+01
3	2,50000e+00	9,80100e+00	1,00000e+01	2,25000e+01
4	1,25000e+00	9,70299e+00	1,00000e+01	3,37500e+01
5	6,25000e-01	9,60596e+00	1,00000e+01	5,06250e+01
6	3,12500e-01	9,50990e+00	1,00000e+01	7,59375e+01
7	1,56250e-01	9,41480e+00	1,00000e+01	1,13906e+02
8	7,81250e-02	9,32065e+00	1,00000e+01	1,70859e+02
9	3,90625e-02	9,22745e+00	1,00000e+01	2,56289e+02

а)



б)

Рис. 4.13. Иллюстрация поведения импульсной характеристики системы при различных степенях устойчивости: *а* — импульсная характеристика; *б* — диаграмма нулей и полюсов в z -плоскости; z -образ системы равен $10z^{-1}/(1 - \alpha z^{-1})$. 1. При $\alpha = 0,5$ система устойчива. 2. При $\alpha = 0,99$ она минимально устойчива. 3. При $\alpha = 1$ система потенциально неустойчива. 4. При $\alpha = 1,5$ она неустойчива. Заметим, например, что при $\alpha = 0,5$ значения импульсной характеристики быстро спадают при увеличении n , тогда как при $\alpha = 1,5$ значения импульсной характеристики быстро возрастают

сигналы. Это так называемое условие ОВОВ (ограниченного входа, ограниченного выхода). Говорят, что ЛИВ-система ОВОВ-устойчива тогда и только тогда, когда она удовлетворяет условию

$$\sum_{k=0}^{\infty} |h(k)| < \infty, \quad (4.53)$$

где $h(k)$ — импульсная характеристика системы. Очевидно, что если импульсная характеристика конечна, то вышеприведенное условие удовлетворяется, поскольку сумма коэффициентов импульсной характеристики будет конечной. Таким образом, исследование устойчивости применимо только к системам с импульсной характеристикой бесконечной длины.

Чтобы выход был ограниченным, все полюсы должны лежать внутри единичной окружности. Если полюс лежит за пределами единичной окружности, система неустойчива. На практике система с полюсом, лежащим на единичной окружности, также считается неустойчивой или потенциально неустойчивой, поскольку незначительное возмущение или ошибка обязательно приведут систему в состояние неустойчивости. Исключение составляет тот случай, когда полюс на единичной окружности совпадает с нулем, так что его действие компенсируется. Импульсная характеристика неустойчивой системы будет бесконечно расти со временем.

В принципе, проверка на устойчивость очень проста: найти положение полюсов z -преобразования. Если какой-либо полюс находится на единичной окружности или за ее пределами (если только он не совпадает с нулем на единичной окружности), система неустойчива. На практике же определение положения полюсов может оказаться совсем не простой задачей.

Простая проверка, которой можно воспользоваться, если z -образ системы $H(z)$ нельзя разложить на множители, — это найти достаточное количество значений импульсной характеристики и построить их график, вычислив обратное z -преобразование. Если импульсная характеристика бесконечно увеличивается со временем или достаточно быстро падает до нуля, то система либо неустойчива, либо минимально устойчива. На рис. 4.13 показаны примеры поведения импульсной характеристики простой системы дискретного времени при различных степенях устойчивости. Другие, более изощренные, тесты на устойчивость можно найти в специализированных учебниках по z -преобразованию (например, [2, 4]). Далее тема исследования устойчивости для систем второго порядка развивается в главе 8.

4.5.8. Разностные уравнения

Разностное уравнение описывает реальные действия, которые система дискретного времени должна произвести над входными данными во временной области, чтобы получить необходимый выход. Разностное уравнение для большинства важных практических случаев можно записать в таком виде:

$$y(n) = \sum_{k=0}^N a_k x(n-k) - \sum_{k=1}^M b_k y(n-k), \quad (4.54)$$

где $x(n)$ — элемент входной последовательности, $y(n)$ — элемент выходной последовательности, $y(n-k)$ — предыдущий выход, а a_k , b_k — коэффициенты системы. Как следует из уравнения (4.54), текущий выход $y(n)$ получается из настоящего и прошлого элементов входной последовательности и предыдущего выхода $y(n-k)$.

Воспользовавшись свойством задержки z -преобразования, разностные уравнения для систем дискретного времени можно легко получить из передаточных функций, и наоборот:

$$\begin{aligned} a_k x(n) &\leftrightarrow a_k X(z), \\ a_k x(n-k) &\leftrightarrow a_k z^{-k} X(z). \end{aligned}$$

Таким образом, уравнение (4.54) можно записать как

$$Y(z) = \sum_{k=0}^N a_k z^{-k} X(z) - \sum_{k=0}^M b_k z^{-k} Y(z) \quad (4.55)$$

Упростив выражение, получим передаточную функцию $H(z)$ дискретной системы в z -области:

$$H(z) = \frac{Y(z)}{X(z)} = \sum_{k=0}^N a_k z^{-k} \left/ \left(1 + \sum_{k=0}^M b_k z^{-k} \right) \right. \quad (4.56)$$

Если все коэффициенты знаменателя b_k равны нулю, уравнения (4.54) и (4.55) сводятся к

$$y(n) = \sum_{k=0}^N a_k x(n-k), \quad (4.57, a)$$

$$H(z) = \frac{Y(z)}{X(z)} = \sum_{k=0}^N a_k z^{-k}. \quad (4.57, б)$$

Теперь выход системы $y(n)$ зависит только от настоящего и прошлого элементов входной последовательности, а не от предыдущего выхода, как в уравнении (4.54). Коэффициенты a_k в данном случае представляют собой импульсную характеристику системы и обычно обозначаются символом $h(k)$. Этот класс ЛИВ-систем называют *системами с конечной импульсной характеристикой* (КИХ), поскольку длина последовательности $h(k)$ определено конечна.

Системы, которые характеризуются уравнениями (4.54) и (4.56), где по меньшей мере один из коэффициентов знаменателя не равен нулю, называются *системами с бесконечной импульсной характеристикой* (БИХ). В системах БИХ по меньшей мере один из полюсов будет ненулевым, а системы КИХ обычно не имеют полюсов.

4.5.9. Оценка импульсной характеристики

При проектировании систем дискретного времени часто возникает необходимость вычисления значений импульсной характеристики. Например, при проектировании системы КИХ импульсная характеристика требуется для реализации системы, а при проектировании системы БИХ эти значения нужны для анализа устойчивости. Также импульсной характеристикой можно воспользоваться для оценки частотной характеристики системы.

Импульсную характеристику системы дискретного времени можно определить как результат применения обратного z -преобразования к передаточной функции системы $H(z)$:

$$h(k) = Z^{-1}[H(z)], k = 0, 1, \dots$$

Если z -преобразование $H(z)$ разложить в степенной ряд, т.е.

$$\begin{aligned} H(z) &= \sum_{n=0}^{\infty} h(n) z^{-n} = \\ &= h(0) + h(1)z^{-1} + h(2)z^{-2} + \dots, \end{aligned} \quad (4.58)$$

то коэффициенты z -преобразования дают непосредственно импульсную характеристику. $H(z)$ БИХ-систем часто выражается в виде отношения многочленов, как в уравнении (4.47). В этом случае для определения импульсной характеристики системы можно воспользоваться методами z^{-1} , описанными в разделе 4.3. Для этой цели также используются программы на языке С или коды MATLAB, которые приводятся в приложении.

Импульсную характеристику можно также рассматривать как отклик системы дискретного времени на единичный импульс $u(n)$, значение которого равно 1 при $n = 0$ и 0 при всех остальных значениях n . Такой подход оправдывается тем, что если вход системы сделать равным единичному импульсу, $x(n) = u(n)$, то выход системы будет

фактически равен $h(n)$ — импульсной характеристике системы (строго говоря, характеристике единичного элемента):

$$\begin{aligned} y(n) &= \sum_{k=0}^{\infty} h(k)x(n-k) = \sum_{k=0}^{\infty} h(k)u(n-k) = \\ &= h(0)u(n) + h(1)u(n-1) + h(2)u(n-2) + \dots = \\ &= h(n), n = 0, 1, \dots \end{aligned} \quad (4.59)$$

Это дает простой альтернативный метод вычисления $h(n)$ (в действительности получаем еще один метод вычисления обратного z -преобразования), что и проиллюстрировано с помощью следующего примера.

Пример 4.13

Найдите импульсную характеристику фильтра дискретного времени, который характеризуется приведенным ниже z -образом

$$H(z) = \frac{1 - z^{-1}}{1 + 0,5z^{-1}},$$

- 1) с помощью метода разложения в степенной ряд;
- 2) воспользовавшись единичным импульсом системы.

Решение

1. С помощью метода разложения в степенной ряд значения импульсной характеристики можно найти таким образом:

$$\begin{array}{r} 1 - z^{-1} \\ \underline{1 + 0,5z^{-1}} \\ -1,5z^{-1} \\ \underline{-1,5z^{-1} - 0,75z^{-2}} \\ 0,75z^{-2} \\ \underline{0,75z^{-2} + 0,375z^{-3}} \\ -0,375z^{-3} \end{array} \quad \frac{1 + 0,5z^{-1}}{1 - 1,5z^{-1} + 0,75z^{-2} - 0,375z^{-3} + \dots}$$

Исходя из коэффициентов, вычислим значения импульсной характеристики:

$$h(0) = 1; h(1) = -1,5; h(2) = 0,75; h(3) = -0,375$$

Конечно, значения импульсной характеристики можно найти и с помощью данной в приложении программы на языке С для разложения в степенной ряд.

2. Во-первых, исходя из передаточной функции нужно найти разностное уравнение фильтра:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1 - z^{-1}}{1 + 0,5z^{-1}}.$$

Выполнив перекрестное умножение и воспользовавшись свойством задержки z -преобразования, найдем разностное уравнение:

$$\begin{aligned} Y(z) + 0,5Y(z)z^{-1} &= X(z) - X(z)z^{-1}, \\ y(n) + 0,5y(n-1) &= x(n) - x(n-1). \end{aligned}$$

После упрощения получаем:

$$y(n) = x(n) - x(n-1) - 0,5y(n-1).$$

Теперь импульсную характеристику фильтра можно найти, сделав замену $x(n) = u(n)$, где

$$u(n) = \begin{cases} 1 & n = 0 \\ 0 & n \neq 0 \end{cases},$$

и предположив существование начального условия $y(-1) = 0$:

$$\begin{aligned} y(0) &= 1 \\ y(1) &= x(1) - x(0) - 0,5y(0) = 0 - 1 - 0,5 = -1,5 \\ y(2) &= x(2) - x(1) - 0,5y(1) = -0,5 \times -1,5 = 0,75 \\ y(3) &= x(3) - x(2) - 0,5y(2) = -0,5 \times 0,75 = -0,325 \\ &\dots \end{aligned}$$

Отсюда значения импульсной характеристики равны

$$h(0) = 1; h(1) = -1,5; h(2) = 0,75; h(3) = -0,325.$$

Видно, что оба метода приводят к одинаковым результатам.

4.5.10. Применение в проектировании цифровых фильтров

Одна из самых важных областей применения z -преобразования в ЦОС — это проектирование и анализ ошибок цифровых фильтров, особенно БИХ-фильтров. Это преобразование интенсивно используется для определения коэффициентов цифровых фильтров и анализа влияния различных ошибок квантования на работу цифрового фильтра. Например, хорошо известно, что системам дискретного времени, реализованным аппаратным или программным образом, всегда присущи ошибки квантования, вызванные конечной длиной регистра процессоров, применяющихся на практике, а z -преобразование предоставляет удобное средство анализа действия таких ошибок на работу системы. В частности, с помощью z -преобразования часто анализируются ошибки, возникающие в результате округления или усечения результата операций умножения, заложенных в разностных уравнениях. Более подробно анализ шумов фильтров дискретного времени обсуждается в главе 13.

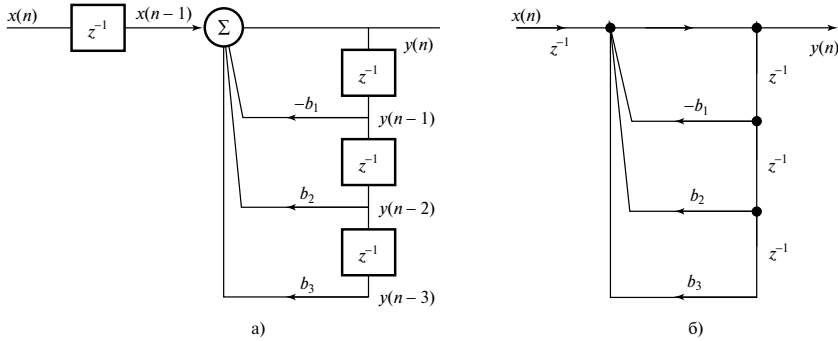


Рис. 4.14. Диаграммы реализации разностного уравнения: a — в виде блок-схемы; b — в виде функциональной схемы

Еще одна важная область применения z -преобразования при проектировании цифровых фильтров — это описание структуры цифрового фильтра. Рассмотрим этот вопрос подробнее, поскольку он требует использования программы разложения на элементарные дроби, которая упоминалась ранее.

4.5.11. Структуры реализации цифровых фильтров

Фильтры дискретного времени часто представляют в виде блок-схем или функциональных схем. Такие диаграммы — удобный способ представления разностных уравнений или, что эквивалентно, передаточных функций. Рассмотрим, например, простой дискретный фильтр со следующим разностным уравнением:

$$y(n) = x(n-1) - b_1 y(n-1) + b_2 y(n-2) + b_3 y(n-3) \quad (4.60)$$

Представление этого уравнения в виде блок-схемы показано на рис. 4.14, a . На этом рисунке символ z^{-1} обозначает задержку на 1 единицу времени. Стрелочками обозначены усилители, а постоянные, стоящие рядом с ними, — это коэффициенты усиления. Связь между разностным уравнением и блок-схемой очевидна. Представление того же разностного уравнения в виде функциональной схемы показано на рис. 4.14, b . В литературе блок-схему (или функциональную схему) принято называть *диаграммой реализации*.

Если $H(z)$ высокого порядка, фильтр дискретного времени редко реализуется непосредственно, как показано на рис. 4.14, поскольку при представлении коэффициентов и разностного уравнения небольшим числом битов будут получены большие ошибки (см. главы 8 и 13). Обычно передаточную функцию представляют в виде комбинации параллельно или последовательно соединенных z -образов первого и/или второго порядка. Для последовательного представления передаточная функция $H(z)$ раскладывается как

$$H(z) = H_1(z)H_2(z) \dots H_K(z) = \prod_{i=1}^K H_i(z), \quad (4.61)$$

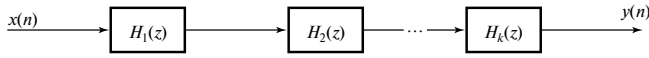


Рис. 4.15. Общая структура последовательной реализации

где $H_i(z)$ — звено второго либо первого порядка:

$$H_i(z) = \frac{b_0 + b_{1i}z^{-1} + b_{2i}z^{-2}}{1 + a_{1i}z^{-1} + a_{2i}z^{-2}} \quad \text{второй порядок}$$

$$H_i(z) = \frac{b_0 + b_{1i}z^{-1}}{1 + a_{1i}z^{-1}} \quad \text{первый порядок}$$

а K — целая часть $(M + 1)/2$. Все z -преобразование — это произведение отдельных z -преобразований (рис. 4.15).

При выборе параллельной реализации передаточная функция раскладывается на элементарные дроби, что дает

$$H(z) = B_0 + \sum_{i=1}^K H_i(z), \quad (4.62)$$

где, как и прежде, $H_i(z)$ — звено второго либо первого порядка, но на этот раз в виде

$$H_i(z) = \frac{a_0 + a_{1i}z^{-1}}{1 + b_{1i}z^{-1} + b_{2i}z^{-2}} \quad \text{— второй порядок}$$

$$H_i(z) = \frac{a_0}{1 + b_{1i}z^{-1}} \quad \text{— первый порядок}$$

где K — целая часть $(M + 1)/2$, а

$$B_0 = a_N/b_M.$$

Общая схема параллельной реализации показана на рис. 4.16.

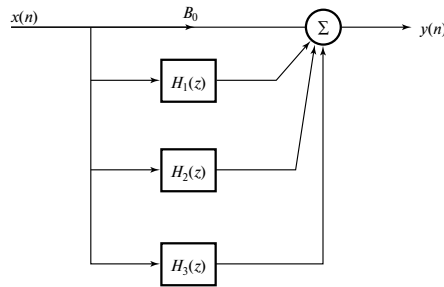


Рис. 4.16. Общая схема параллельной реализации

При разработке цифровых фильтров вышеприведенные коэффициенты a_k и b_k , как правило, определяют с помощью программных пакетов. К сожалению, большая часть программных пакетов выдает коэффициенты только для последовательной реализации. Коэффициенты же параллельной реализации можно найти по коэффициентам последовательной реализации с помощью метода разложения на элементарные дроби. Проиллюстрируем это на примере.

Пример 4.14

Система дискретного времени характеризуется следующим z -образом передаточной функции:

$$H(z) = \frac{1 - 2z^{-2} + z^{-4}}{1 - 0,41421z^{-1} + 0,08579z^{-2} + 0,292895z^{-3} + 0,5z^{-4}}.$$

1. Запишите $H(z)$ в виде, удобном для построения последовательной структуры с использованием звеньев второго порядка.
2. Повторите п. 1 для параллельной структуры.

Решение

1. В разложенном виде $H(z)$ задается как

$$H(z) = H_1(z)H_2(z),$$

где

$$H_1(z) = \frac{1 - 2z^{-1} + z^{-2}}{1 - 1,14121z^{-1} + z^{-2}}, \quad (4.63, a)$$

$$H_2(z) = \frac{1 - 2z^{-1} + z^{-2}}{1 + z^{-1} + 0,5z^{-2}}. \quad (4.63, б)$$

2. Чтобы записать $H(z)$ в виде, удобном для построения последовательной структуры, сперва разложим эту функцию с помощью метода разложения на элементарные дроби. Итак,

$$H(z) = B_0 + \frac{C_1}{z - p_1} + \frac{C_2}{z - p_2} + \frac{C_3}{z - p_3} + \frac{C_4}{z - p_4}. \quad (4.64)$$

Воспользовавшись программой для разложения на элементарные дроби (приложение 4Б), найдем полюсы от p_1 до p_4 и коэффициенты B_0 и C_1 – C_4 :

$$p_1 = 0,7071 + 0,7071i = e^{i0,785}; \quad p_2 = p_1^*$$

$$p_3 = -0,5 + 0,5i = 0,7071e^{i2,35619}; \quad p_4 = p_3^*$$

$$B_0 = 2$$

$$C_1 = 0,114383 + 0,666669i = 0,6764104 \angle 1,400877; \quad C_2 = C_1^* \quad (4.65)$$

$$C_3 = -0,61438276 - 0,58088079i =$$

$$= 0,8455108976 \angle 3,898969; \quad C_4 = C_3^*,$$

где углы измеряются в радианах. Найдя полюсы и коэффициенты C_k и B_0 , дроби из уравнения (4.64) нужно объединить так, чтобы функция $H(z)$ описывала сумму звеньев второго порядка в виде

$$H(z) = B_0 + \sum_{i=1}^2 H_i(z), \quad (4.66, a)$$

где

$$H_i(z) = \frac{a_0 + a_{1i}z^{-1}}{1 + b_{1i}z^{-1} + b_{2i}z^{-2}}. \quad (4.66, б)$$

Чтобы коэффициенты a_{ki} и b_{ki} из уравнения (4.65, б) были действительными, элементарные дроби из уравнения (4.64), содержащие C_1 и C_2 (комплексно-сопряженную пару), нужно объединить. Аналогично нужно объединить также дроби с коэффициентами C_3 и C_4 . Объединив дроби с коэффициентами C_1 и C_2 , получим:

$$\begin{aligned} \frac{C_1 z}{z - p_1} + \frac{C_2 z}{z - p_2} &= \frac{(C_1 + C_2)z^2 - (C_1 p_2 + C_2 p_1)z}{z^2 - (p_1 + p_2)z + p_1 p_2} = \\ &= \frac{C_1 + C_2 - (C_1 p_2 + C_2 p_1)z^{-1}}{1 - (p_1 + p_2)z^{-1} + p_1 p_2 z^{-2}}. \end{aligned} \quad (4.67)$$

Сравнив уравнения (4.65, б) и (4.67) при $i = 1$ в уравнении (4.65, б), находим, что

$$\begin{aligned} a_{01} &= C_1 + C_2, \quad a_{11} = -(C_1 p_2 + C_2 p_1) \\ b_{11} &= -(p_1 + p_2), \quad b_{12} = p_1 p_2 \end{aligned} \quad (4.68)$$

Если учесть тот факт, что $p_2 = p_1^*$, $C_2 = C_1^*$, и подставить значения p_1 и C_1 , то

$$\begin{aligned} a_{01} &= C_1 + C_1^* = 2 \times 0,114383 = 0,2288 \\ a_{11} &= -(C_1 p_1^* + C_1^* p_1) = \\ &= -(|C_1|e^{i\theta_1}|p_1|e^{-i\phi_1} + |C_1|e^{-i\theta_1}|p_1|e^{i\phi_1}) = \\ &= -|C_1||p_1|[e^{i(\theta_1 - \phi_1)} + e^{-i(\theta_1 - \phi_1)}] = \\ &= -2|C_1||p_1|\cos(\theta_1 - \phi_1) = \\ &= -2 \times 0,6764104 \times 1 \cos(1,400877 - 0,78540068) = \\ &= -1,1046, \end{aligned} \quad (4.69)$$

где $\theta_1 = \angle C_1$, $\phi_1 = \angle p_1$. Следовательно, мы получим:

$$H_1(z) = \frac{0,2288 - 1,1046z^{-1}}{1 - 1,4142z^{-1} + z^{-2}}, \quad (4.70)$$

где значения коэффициентов знаменателя берутся прямо из уравнения (4.63). Аналогично из элементарных дробей с коэффициентами C_3 и C_4 получаем

$$H_2(z) = \frac{-1,2288 - 0,0335z^{-1}}{1 + z^{-1} + 0,5z^{-2}}. \quad (4.71)$$

Объединив результаты, получим

$$H(z) = 2 + \frac{0,2288 - 1,1046z^{-1}}{1 - 1,4142z^{-1} + z^{-2}} + \frac{-1,2288 - 0,0335z^{-1}}{1 + z^{-1} + 0,5z^{-2}}.$$

Хотя вышеприведенный процесс по своей сути и прост, он очень трудоемкий, кроме того, существует большая вероятность допустить ошибку, особенно если вычислять коэффициенты элементарных дробей вручную. В приложении 4Б приводится стандартная программа на языке С, которую можно использовать для поиска коэффициентов параллельной структуры по известной передаточной функции, представленной в последовательной форме. В действительности эта программа представляет собой простое дополнение к программе разложения на элементарные дроби, описанной в том же приложении. В главе 8 мы несколько подробнее остановимся на областях применения последовательной и параллельной структур.

4.6. Резюме

При работе в области ЦОС очень важно разбираться в z -преобразовании — бесценном средстве описания, анализа и проектирования систем дискретного времени.

В главе показано, как находится z -образ последовательностей дискретного времени и как восстанавливаются последовательности по их z -образам. Предлагается несколько программ на языке С и кодов MATLAB, которые позволят читателям на практике изучить основные принципы и примеры применения z -преобразования в обработке сигналов.

Задачи

4.1. Найдите z -образы следующих последовательностей дискретного времени:

1.

$$x(n) = \sin(n\omega T), n = 0, 1, \dots$$

2.

$$x(n) = \begin{cases} a^n, & n \geq 0 \\ 0, & n < 0 \end{cases}$$

3.

$$x(n) = \begin{cases} 1, & 0 \leq n \leq N-1 \\ 0, & \text{в остальных случаях} \end{cases}$$

- 4.2. Экспоненциальная последовательность задается как

$$x(n) = e^{-kn}, n \geq 0.$$

Найдите ее z -образ и условие на z , при выполнении которого z -образ сходится, для каждого из следующих случаев:

- 1) k действительное;
 - 2) k комплексное.
- 4.3. Даны причинные последовательности $x(n)$ и $nx(n)$ с z -образами $X(z)$ и $X'(z)$. Покажите, что

$$X'(z) = -z \frac{dX(z)}{dz}.$$

- 4.4. Пусть z -образ последовательности дискретного времени задается как

$$X(z) = \sum_{n=0}^{\infty} x(n)z^{-n}.$$

Исходя из этого уравнения, покажите, оговорив все сделанные предположения, что обратное z -преобразование задается как

$$x(n) = \frac{1}{2\pi i} \oint z^{n-1} X(z) dz, n > 0.$$

Подумайте о роли теоремы о вычетах в вычислении этого интеграла.

- 4.5. 1. С помощью метода разложения в степенной ряд найдите первые пять значений причинной последовательности дискретного времени, соответствующей каждому из следующих z -образов:

а) $X(z) = \frac{z-1}{(z-0,7071)^2},$

б) $X(z) = \frac{1}{(z-0,5)(z+0,9)^3},$

в) $X(z) = \frac{z^4-1}{z^4+1},$

г) $X(z) = \frac{z^3-z^2+z-1}{(z+0,9)^3}.$

2. Повторите п. 1, воспользовавшись методом разложения на элементарные дроби.
 3. Повторите п. 1, воспользовавшись методом вычетов.
- 4.6. 1. Дан z -образ $X(z)$ в виде

$$X(z) = \frac{N(z)}{D(z)},$$

где $N(z)$ и $D(z)$ — многочлены. Предположив, что $X(z)$ имеет полюс в точке $z = p_k$, покажите, что

$$\text{Res}[X(z), p_k] = \frac{N(p_k)}{D'(p_k)}.$$

где

$$D'(p_k) = \frac{dD(z)}{dz}.$$

2. Воспользуйтесь этим результатом, чтобы найти обратное z -преобразование следующего выражения:

$$X(z) = \frac{1}{z^4 - 1}.$$

- 4.7. Дана следующая передаточная функция устойчивой причинной системы:

$$H(z) = \frac{1}{1 + b_1 z^{-1} + b_2 z^{-2}}.$$

С помощью метода вычетов найдите ее импульсную характеристику $h(n)$ в аналитическом виде. Предположите, что полюсы разные и комплексные.

- 4.8. Разложение системы дискретного времени N -го порядка на элементарные дроби выглядит так:

$$X(z) = \frac{N(z)}{D(z)} = B_0 + \sum_{k=1}^N \frac{C_k z}{z - p_k z^k},$$

где

$$\begin{aligned} N(z) &= a_0 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_N z^{-N} \\ D(z) &= b_0 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_M z^{-M}, \end{aligned}$$

p_k — полюсы функции $X(z)$ (предположим, что они отдельные), а C_k — коэффициенты элементарных дробей. Найдите общее выражение для C_k , $k = 1, 2, \dots$, записанное через полюсы p_k и $N(z)$. Предположив, что $N = 3$, покажите путем деления в столбик, что B_0 задается как

$$B_0 = a_3/b_3.$$

- 4.9. Дано разностное уравнение

$$y(n) + B_1 y(n-1) + B_2 y(n-2) = A, \quad n \geq 0,$$

где A , B_1 и B_2 — произвольные постоянные. Найдите z -образ $Y(z)$. Воспользовавшись подходящим методом вычисления обратного z -преобразования, найдите аналитическое выражение для $y(n)$.

- 4.10. Система дискретного времени второго порядка характеризуется следующим z -образом передаточной функции:

$$H(z) = \frac{1}{(z - 0,9)^2}, \quad |z| > 0,9.$$

С помощью метода вычетов найдите соответствующую последовательность дискретного времени $h(n)$.

- 4.11. Для системы дискретного времени, изображенной на рис. 4.17, найдите разностное уравнение, связывающее выход $y(n)$ и вход $x(n)$. Выведите ее передаточную функцию $H(z)$.

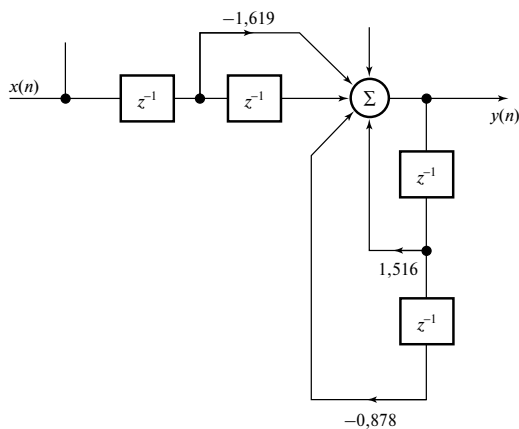


Рис. 4.17. Блок-схема системы дискретного времени из задачи 4.11

- 4.12. Передаточная функция системы дискретного времени имеет полюсы в точках $z = 0,5$, $z = 0,1 \pm 0,2i$ и нули в точках $z = -1$ и $z = 1$.
1. Изобразите диаграмму нулей и полюсов для этой системы.
 2. Выведите из диаграммы нулей и полюсов передаточную функцию $H(z)$ системы.
 3. Запишите разностное уравнение.
 4. Изобразите схему реализации в виде функциональной диаграммы.
- 4.13. На рис. 4.18 показана функциональная схема системы дискретного времени. Найдите двухэтапное разностное уравнение, связывающее выход $y(n)$ и вход $x(n)$. Из разностного уравнения выведите передаточную функцию $H(z)$.
- 4.14. Дана спецификация частотной характеристики полосового фильтра дискретного времени в нормированном виде:
- | | |
|------------------------|---------------------------------|
| полоса пропускания | $0,4\pi - 0,6\pi$, |
| полосы подавления | $0 - 0,3\pi$ и $0,7\pi - \pi$, |
| интервал дискретизации | $T = 100$ мкс. |

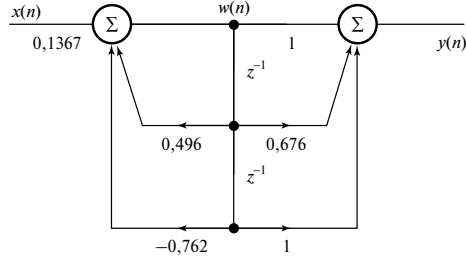


Рис. 4.18. Функциональная схема системы дискретного времени из задачи 4.13

1. Выразите эту спецификацию в рад/с (денормируйте ее).
 2. Переведите спецификацию из рад/с в стандартные единицы — герцы.
 3. Переведите спецификацию из герц (п. 2) в нормированный вид.
 4. Изобразите частотные характеристики для всех трех случаев на интервале от 0 до частоты дискретизации.
- 4.15. ЛИВ-система характеризуется следующим z -образом:

$$\frac{1 + z^{-2}}{1 + 1,81z^{-2}}.$$

Найдите частотную характеристику на постоянной составляющей, а также при $1/4$ и $1/2$ частоты дискретизации. Изобразите частотную характеристику на интервале $0 \leq \omega \leq \omega_s$, где ω_s — частота дискретизации в рад/с.

- 4.16. Требуется простой фильтр нижних частот со следующей спецификацией:

частота среза 1 кГц,
частота дискретизации 10 кГц.

Определите и изобразите подходящую диаграмму нулей и полюсов для этого фильтра.

Найдите передаточную функцию фильтра по его диаграмме нулей и полюсов. Определите амплитудную и фазовую характеристику при частоте 1, 2,5 и 5 кГц. Изобразите амплитудно-частотную характеристику.

- 4.17. Передаточная функция некоторой системы определяется как

$$H(z) = \frac{(1 - 1,094621z^{-1} - z^2)(1 - 0,350754z^{-1} + z^{-2})}{(1 - 1,340228z^{-1} + 0,796831z^{-2})(1 - 0,5z^{-1} - 0,5z^{-2})}.$$

1. Найдите полюсы и нули и изобразите диаграмму нулей и полюсов.
2. Определите, устойчива система или нет, и обоснуйте свой ответ.

Задачи для решения с помощью языковых средств и MATLAB

- 4.18. Передаточная функция системы дискретного времени задается как

$$H(z) = \frac{z^2 - z}{z^2 - 0,9051z + 0,4096}.$$

1. Определите положения полюсов и нулей с помощью команды MATLAB `roots` (нули должны быть в точках $z = 0$, $z = \pm 1$; полюсы — на радиусе $0,64 \angle 45^\circ$).
2. Повторите п. 1 для случая, когда многочлены в числителе и знаменателе записаны через возрастающие отрицательные показатели степени:

$$H(z) = \frac{1 - z^{-1}}{1 - 0,9051z^{-1} + 0,4096z^{-2}}.$$

3. Сравните результаты п. 1 и 2 и прокомментируйте разницу, если она есть. Что нужно сделать, чтобы ответы были одинаковыми?
 4. С помощью команды MATLAB `zplane` постройте диаграмму нулей и полюсов для $H(z)$ из п. 1 для следующих случаев:
 - а) в качестве входных данных используются коэффициенты многочленов числителя и знаменателя $b(z)$ и $a(z)$;
 - б) в качестве входных данных используются положения полюсов и нулей функции $H(z)$.
- 4.19. Передаточная функция узкополосного режекторного фильтра дискретного времени задается как

$$H(z) = \frac{1 - 2 \cos \theta + z^{-2}}{1 - 2r \cos \theta z^{-1} + r^2 z^{-2}},$$

где r — радиус полюсов, а θ — угол полюсов и нулей.

1. С помощью программы MATLAB постройте график амплитудно-частотной характеристики фильтра и оцените степень режекции (относительно амплитудной характеристики при частоте 0 кГц) для каждого из следующих случаев:
 - а) $r = 0,5; \theta = \pm 15^\circ$;
 - б) $r = 0,5; \theta = \pm 60^\circ$;
 - в) $r = 0,5; \theta = \pm 90^\circ$;
 - г) $r = 0,5; \theta = \pm 120^\circ$.
 2. Постройте график амплитудно-частотной характеристики фильтра и оцените степень режекции (относительно амплитудной характеристики при частоте 0 кГц) для каждого из следующих случаев:
 - а) $r = 0,5; \theta = \pm 45^\circ$;
 - б) $r = 0,8; \theta = \pm 45^\circ$;
 - в) $r = 0,9; \theta = \pm 45^\circ$;
 - г) $r = 0,99; \theta = \pm 45^\circ$.

Объясните, как положения (радиусы и углы) полюсов и нулей влияют на частотную характеристику узкополосного режекторного фильтра.
- 4.20. Пусть z -образ системы дискретного времени задается как

$$H(z) = \sum_{k=0}^8 a_k z^{-k} \bigg/ \sum_{k=0}^8 b_k z^{-k},$$

где

$$\begin{aligned}
 a_0 &= 2,740584 \times 10^{-2} & b_0 &= 1 \\
 a_1 &= 2,825341 \times 10^{-3} & b_1 &= 2,233030 \times 10^{-1} \\
 a_2 &= -2,932353 \times 10^{-2} & b_2 &= 2,353762 \\
 a_3 &= 3,563199 \times 10^{-4} & b_3 &= 4,369285 \times 10^{-1} \\
 a_4 &= 4,924136 \times 10^{-2} & b_4 &= 2,712411 \\
 a_5 &= 3,563226 \times 10^{-4} & b_5 &= 3,571619 \times 10^{-1} \\
 a_6 &= -2,932353 \times 10^{-2} & b_6 &= 1,593957 \\
 a_7 &= 2,825337 \times 10^{-3} & b_7 &= 1,141820 \times 10^{-1} \\
 a_8 &= 2,740582 \times 10^{-2} & b_8 &= 4,143201 \times 10^{-1}
 \end{aligned}$$

Найдите и постройте график импульсной характеристики системы, воспользовавшись программой для обратного z -преобразования (метод разложения в степенной ряд), описанной в приложении, или подходящей программой из пакета MATLAB. Опираясь на полученный график, определите, устойчива эта система, минимально устойчива или неустойчива.

- 4.21. Передаточная функция БИХ-системы третьего порядка в разложенном виде выглядит как

$$H(z) = \frac{N_1(z)N_2(z)}{D_1(z)D_2(z)},$$

где

$$\begin{aligned}
 N_1(z) &= 1 - 0,971426z^{-1} + z^{-2}, \\
 N_2(z) &= 1 + z^{-1}, \\
 D_1(z) &= 1 - 0,935751z^{-1} + 0,726879z^{-2}, \\
 D_2(z) &= 1 + 0,18311z^{-1}.
 \end{aligned}$$

1. Изобразите диаграмму реализации системы с последовательной структурой, используя звенья первого и второго порядка.
2. Выразите $H(z)$ в виде суммы элементарных дробей

$$H(z) = B_0 + \sum_{k=1}^3 \frac{C_k}{z - p_k}$$

и воспользуйтесь для вычисления коэффициентов B_0 и C_k программой разложения на элементарные дроби из приложения 4В или подходящей командой MATLAB.

3. Объедините элементарные дроби так, чтобы систему можно было реализовать в виде параллельной структуры с помощью звеньев первого и второго порядка.
4. С помощью результата п. 3 изобразите схему реализации параллельной структуры.

- 4.22. Цифровой узкополосный режекторный фильтр характеризуется следующим z -образом:

$$\frac{z^2 + 1}{z^2 + r^2}.$$

Найдите частотную характеристику, воспользовавшись методом БПФ и частотой дискретизации в 1 кГц, для каждого из нижеследующих случаев: а) $r = 0,8$; б) $r = 0,95$; в) $r = 1$. Объясните полученный результат.

- 4.23. Передаточная функция фильтра нижних частот дискретного времени имеет следующий вид:

$$H(z) = \frac{a_0 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_4 z^{-4}}{b_0 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_4 z^{-4}},$$

где

$$\begin{aligned} a_0 &= 0,193441 & b_0 &= 1 \\ a_1 &= 0,378331 & b_1 &= -2,516884 \\ a_2 &= 0,52414 & b_2 &= 1,054118 \\ a_3 &= 0,378331 & b_3 &= -0,240603 \\ a_4 &= 0,193441 & b_4 &= 0,1985861 \end{aligned}$$

Найдите частотную характеристику фильтра, воспользовавшись

- а) программой непосредственного вычисления частотной характеристики, описанной в приложении 4В;
б) методом разложения в степенной ряд и БПФ.

Сравните эти два результата и прокомментируйте различия, если они есть.

- 4.24. В табл. 4.4 приведены коэффициенты простой полосовой КИХ-системы. Предположив, что частота дискретизации равна 10 кГц, рассчитайте амплитудно-частотную характеристику системы, воспользовавшись программой, предложенной в приложении 4В.
- 4.25. Дан z -образ

$$H(z) = \frac{1 + 3z^{-1} + z^{-2} + z^{-3}}{1 + (1 - k)z^{-1} + (k + 0,3561)z^{-2} + 0,3561k}.$$

С помощью программы разложения в степенной ряд из приложения 4Б найдите достаточное количество значений импульсной характеристики системы для каждого из следующих случаев:

- а) $k = -1$;
б) $k = 1$;
в) $k = 2$;
г) $k = 0,9$.

Таблица 4.4. Коэффициенты полосового КИХ-фильтра из задачи 4.24

$H(1) =$	$-0,67299600E - 02$	$= H(35)$
$H(2) =$	$0,16799420E - 01$	$= H(34)$
$H(3) =$	$0,17195700E - 01$	$= H(33)$
$H(4) =$	$-0,27849080E - 01$	$= H(32)$
$H(5) =$	$-0,17486810E - 01$	$= H(31)$
$H(6) =$	$0,13515580E - 01$	$= H(30)$
$H(7) =$	$0,45570510E - 02$	$= H(29)$
$H(8) =$	$0,33293060E - 01$	$= H(28)$
$H(9) =$	$0,95162150E - 02$	$= H(27)$
$H(10) =$	$-0,68548560E - 01$	$= H(26)$
$H(11) =$	$-0,68992230E - 02$	$= H(25)$
$H(12) =$	$0,23802370E - 01$	$= H(24)$
$H(13) =$	$-0,11597510E - 01$	$= H(23)$
$H(14) =$	$0,12073780E + 00$	$= H(22)$
$H(15) =$	$0,23806900E - 01$	$= H(21)$
$H(16) =$	$-0,29095690E + 00$	$= H(20)$
$H(17) =$	$-0,12362380E - 01$	$= H(19)$
$H(18) =$	$0,36717700E + 00$	$= H(18)$

Для каждого случая постройте график импульсной характеристики и определите по нему, устойчива система, минимально устойчива или неустойчива.

- 4.26. Напишите на языке С программу проверки результата разложения на элементарные дроби. Программа должна принимать в качестве входа значения полюсов p_k ($k = 1, 2, \dots, M$) и связанные с ними коэффициенты C_k ($k = 1, 2, \dots, M$) и выдавать на выходе коэффициенты многочленов $A(z)$ и $B(z)$, где

$$X(z) = \frac{A(z)}{B(z)}$$

и

$$A(z) = a_0 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_N z^{-N},$$

$$B(z) = b_0 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_M z^{-M}.$$

Дополните вашу программу возможностью проверять результаты преобразования последовательных структур в параллельные.

- 4.27. Повторите задачу 4.26, воспользовавшись программным пакетом MATLAB.

Литература

1. Atkinson L. V. and Harley P. J. (1983) *An Introduction to Numerical Methods with Pascal*, Chapter 3. Wokingham: Addison-Wesley.
2. Jury E. I. (1964) *Theory and Applications of the z-transform Method*. New York: Wiley.
3. Mathews J. H. (1982) *Basic Complex Variables for Mathematics and Engineering*. Boston MA: Allyn and Bacon.
4. Proakis J. G. and Manolakis D. G. (1992) *Digital Signal Processing*, 2nd edn. New York: Macmillan.

Дополнительная литература

- Ahmed N. and Natarajan T. (1983) *Discrete-time Signals and Systems*. Reston VA: Reston Publishing Co. Inc.
- Churchhill R. V., Brown J. W. and Verhey R. F. (1976) *Complex Variables and Applications*. New York: McGraw-Hill.
- Jong M. T. (1982) *Methods of Discrete Signals and Systems Analysis*. New York: McGraw-Hill.
- Oppenheim A. V. and Schaffer R. W. (1975) *Digital Signal Processing*. Englewood Cliffs NJ: Prentice-Hall.
- Rabiner L. R. and Gold B. (1975) *Theory and Application of Digital Signal Processing*. Englewood Cliffs NJ: Prentice-Hall.
- Ragazzini J. R. and Zadeh L. A. (1952) *Analysis of sampled data systems*. Trans. AIEE, **71**(II), 225–234.
- Steiglitz K. (1974) *An Introduction to Discrete Systems*. New York: Wiley.
- Strum R. D. and Kirk D. E. (1988) *First Principles of Discrete Systems and Digital Signal Processing*. Reading MA: Addison-Wesley.

Приложения

4.A. Рекурсивный алгоритм вычисления обратного z -преобразования

В данной главе утверждалось, что метод деления в столбик можно переделать таким образом, чтобы он приобрел рекурсивный вид. Здесь, в частности, мы хотим показать, что при заданном z -образе $X(z)$, таком что

$$X(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{a_0 + a_1 z^{-1} + a_2 z^{-2}},$$

обратное z -преобразование $x(n)$ можно найти как [Jury, 1964]:

$$x(n) = \frac{1}{a_0} \left[b_n - \sum_{i=1}^n x(n-i) a_i \right], n = 1, 2, \dots$$

$$x(0) = \frac{b_0}{a_0}$$

Этот результат можно обобщить. С помощью деления в столбик можно выразить $X(z)$ в виде степенного ряда:

$$\begin{array}{l}
b_0 + b_1 z^{-1} + b_2 z^{-2} \\
\hline
b_0 + \left(\frac{b_0}{a_0} a_1\right) z^{-1} + \left(\frac{b_0}{a_0} a_2\right) z^{-2} \\
\hline
\left(b_1 - \frac{b_0}{a_0} a_1\right) z^{-1} + \left(b_2 - \frac{b_0}{a_0} a_2\right) z^{-2} \\
\hline
\left(b_1 - \frac{b_0}{a_0} a_1\right) z^{-1} + \frac{a_1}{a_0} \left(b_1 - \frac{b_0}{a_0} a_1\right) z^{-2} + \frac{a_2}{a_0} \left(b_1 - \frac{b_0}{a_0} a_1\right) z^{-3} \\
\hline
\left[\left(b_2 - \frac{b_0}{a_0} a_2\right) - \frac{a_1}{a_0} \left(b_1 - \frac{b_0}{a_0} a_1\right)\right] z^{-2} - \frac{a_2}{a_0} \left(b_1 - \frac{b_0}{a_0} a_1\right) z^{-3} \\
\hline
\left[\left(b_2 - \frac{b_0}{a_0} a_2\right) - \frac{a_1}{a_0} \left(b_1 - \frac{b_0}{a_0} a_1\right)\right] z^{-2} + \frac{a_1}{a_0} \left[\left(b_2 - \frac{b_0}{a_0} a_2\right) - \right. \\
\left. - \frac{a_1}{a_0} \left(b_1 - \frac{b_0}{a_0} a_1\right)\right] z^{-3} + \frac{a_2}{a_0} \left[\left(b_2 - \frac{b_0}{a_0} a_2\right) - \frac{a_1}{a_0} \left(b_1 - \frac{b_0}{a_0} a_1\right)\right] z^{-4} \\
\hline
\left\{\left[-\frac{a_2}{a_0} \left(b_1 - \frac{b_0}{a_0} a_1\right)\right] - \frac{a_1}{a_0} \left[\left(b_2 - \frac{b_0}{a_0} a_2\right) - \frac{a_1}{a_0} \left(b_1 - \frac{b_0}{a_0} a_1\right)\right]\right\} z^{-3} - \\
- \frac{a_2}{a_0} \left[\left(b_2 - \frac{b_0}{a_0} a_2\right) - \frac{a_1}{a_0} \left(b_1 - \frac{b_0}{a_0} a_1\right)\right] z^{-4}
\end{array}$$

Частное, полученное при делении в столбик, даст коэффициенты степенного ряда:

$$\begin{aligned}
X(z) = & \frac{b_0}{a_0} + \left[\left(b_1 - \frac{b_0}{a_0} a_1\right) / a_0\right] z^{-1} + \frac{1}{a_0} \left[\left(b_2 - \frac{b_0}{a_0} a_2\right) - \frac{a_1}{a_0} \left(b_1 - \frac{b_0}{a_0} a_1\right)\right] z^{-2} + \\
& + \frac{1}{a_0} \left\{\left[-\frac{a_2}{a_0} \left(b_1 - \frac{b_0}{a_0} a_1\right)\right] - \frac{a_1}{a_0} \left[\left(b_2 - \frac{b_0}{a_0} a_2\right) - \frac{a_1}{a_0} \left(b_1 - \frac{b_0}{a_0} a_1\right)\right]\right\} z^{-3} + \dots
\end{aligned}$$

Согласно определению z-преобразования для причинных систем $X(z)$ задается как

$$X(z) = \sum_{n=0}^{\infty} x(n) z^{-n} = x(0) + x(1) z^{-1} + x(2) z^{-2} + \dots$$

Следовательно, можно записать

$$\begin{aligned}
x(0) &= b_0 / a_0 \\
x(1) &= \frac{1}{a_0} \left(b_1 - \frac{b_0}{a_0} a_1\right) = \frac{1}{a_0} [b_1 - x(0) a_1] \\
x(2) &= \frac{1}{a_0} \left[\left(b_2 - \frac{b_0}{a_0} a_2\right) - \frac{a_1}{a_0} \left(b_1 - \frac{b_0}{a_0} a_1\right)\right] = \\
&= \frac{1}{a_0} [b_2 - x(0) a_2 - a_1 x(1)] \\
x(3) &= \frac{1}{a_0} \left\{\left[-\frac{a_2}{a_0} \left(b_1 - \frac{b_0}{a_0} a_1\right)\right] - \frac{a_1}{a_0} \left[\left(b_2 - \frac{b_0}{a_0} a_2\right) - \frac{a_1}{a_0} \left(b_1 - \frac{b_0}{a_0} a_1\right)\right]\right\} = \\
&= \frac{1}{a_0} [-a_2 x(1) - a_1 x(2)]
\end{aligned}$$

В общем виде запишем

$$x(n) = \frac{1}{a_0} \left[b_n - \sum_{i=1}^n x(n-i)a_i \right], n = 1, 2, \dots$$

$$x(0) = b_0/a_0.$$

4.Б. Программа на С для оценки обратного z -преобразования и превращения последовательной структуры в параллельную

Вычисление обратного z -преобразования с помощью метода разложения в степенной ряд или на элементарные дроби можно оформить в виде программы на языке С. Эта программа также используется для преобразования передаточной функции $H(z)$ системы дискретного времени из последовательной структуры в параллельную. Поскольку указанная программа достаточно велика, для эффективности было сформировано два программных модуля `izt.c` и `ltilib.c`, хранящихся в отдельных файлах, которые можно скомпилировать отдельно, а затем соединить:

`izt.c` программа вычисления обратного z -преобразования через разложение в степенной ряд или на элементарные дроби, и преобразования передаточной функции $H(z)$, представленной в виде последовательной структуры, в эквивалентную передаточную функцию, имеющую параллельную структуру, через разложение на элементарные дроби;

`ltilib.c` библиотека функций, в том числе функций `power_series` и `partial_fraction`.

Из-за недостатка свободного места сама программа и библиотека функций здесь не приводятся, но их можно найти на компакт-диске к книге [Ifeachor, 2001] (подробности см. в предисловии).

4.Б.1. Метод степенного ряда

Обратное z -преобразование $x(n)$ вычисляется рекурсивным способом с помощью функции `power_series()` (см. программу), в основе которой лежит следующее уравнение:

$$x(n) = \left[b_n - \sum_{i=1}^n x(n-i)a_i \right] / a_0, \quad n = 1, 2, \dots, \quad (4Б.1, a)$$

где

$$x(0) = b_0/a_0. \quad (4Б.1, б)$$

Для того чтобы можно было воспользоваться программой для вычисления обратного z -преобразования методом разложения в степенной ряд, z -образ должен быть представлен либо непосредственно, либо в виде последовательной структуры:

$$X(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_N z^{-N}}{a_0 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_M z^{-M}} \quad \text{непосредственный вид} \quad (4Б.2, а)$$

$$X(z) = \prod_{k=i}^K X_i(z) \quad \text{последовательная структура} \quad (4Б.2, б)$$

где $X_i(z)$ — звено второго порядка, которой задается как

$$X_i(z) = \frac{b_{0i} + b_{1i} z^{-1} + b_{2i} z^{-2}}{1 + a_{1i} z^{-1} + a_{2i} z^{-2}}. \quad (4Б.3)$$

Для работы программы нужно создать файл входных данных под названием `coeff.dat`. В этом файле записывается количество каскадов K (для непосредственного представления $K = 1$) и коэффициенты числителя и знаменателя z -образа. Пользоваться таким входным файлом очень удобно, поскольку это избавляет от необходимости вводить коэффициенты вручную, а при этом можно допустить ошибку. Более того, это позволяет использовать результаты одного исследования в качестве входных данных для другой программы. Ниже использование программы иллюстрируется на вычислении обратного z -преобразования через метод разложения в степенной ряд.

Пример 4Б.1

С помощью метода разложения в степенной ряд найдите первые пять значений обратного z -преобразования системы дискретного времени, которая характеризуется следующим z -образом:

$$X(z) = \frac{0,1833015 + 0,3419561z^{-1} + 0,3419561z^{-2} + 0,1833051z^{-3}}{1 - 0,3525182z^{-1} + 0,4194023z^{-2} - 0,016369z^{-3}}.$$

Видно, что $X(z)$ записана в непосредственном виде. Файл входных данных, созданный с помощью компьютерной команды `edlin`, имеет вид:

1			
1	-0,3525182	0,4194023	0,016369
0,1833015	0,3419561	0,3419561	0,1833015

Ниже приводится выход программы:

$$h(0) = -0,016369; \quad h(1) = 0,177531; \quad h(2) = 0,411404; \quad h(3) = 0,0705705;$$

$$h(4) = -0,1476666.$$

Пример 4Б.2

С помощью метода разложения в степенной ряд найдите первые пять значений обратного z -преобразования следующей системы:

$$X(z) = \frac{N_1(z)N_2(z)N_3(z)}{D_1(z)D_2(z)D_3(z)},$$

где

$$N_1(z) = 1 - 1,122346z^{-1} + z^{-2}$$

$$N_2(z) = 1 - 0,437833z^{-1} + z^{-2}$$

$$N_3(z) = 1 + z^{-1}$$

$$D_1(z) = 1 - 1,433509z^{-1} + 0,858110z^{-2}$$

$$D_2(z) = 1 - 1,293601z^{-1} + 0,556929z^{-2}$$

$$D_3(z) = 1 - 0,612159z^{-1}.$$

Видно, что передаточная функция состоит из трех частей: двух звеньев второго порядка и одного звена первого порядка. Звено первого порядка представляется как звено второго порядка с нулевым коэффициентом при z^{-2} . Ниже дан выходной файл данных программы.

3			/* количество этапов; максимум 5 */
1	-1,433509	0,858110	/* коэффициенты $D_1(z)$ */
1	-1,122346	1	/* коэффициенты $N_1(z)$ */
1	-1,293601	0,556929	/* коэффициенты $D_2(z)$ */
1	-0,437833	1	/* коэффициенты $N_2(z)$ */
1	-0,6121593	0	/* коэффициенты $D_3(z)$ */
1	1	0	/* коэффициенты $N_3(z)$ */

Комментарии в правой колонке не являются частью файла; они даны просто как пояснения. Ниже приводится выход программы:

$$x(0) = 1; x(1) = 2,77909; x(2) = 5,2725;$$

$$x(3) = 8,7218; x(4) = 11,7438; x(5) = 13,4723.$$

4.Б.2. Разложение на элементарные дроби

Дан z -образ N -го порядка с отдельными полюсами, т.е.

$$X(z) = \frac{N(z)}{D(z)} = \frac{b_0 z^N + b_1 z^{N-1} + \dots + b_{N-1} z + b_N}{a_0 z^N + a_1 z^{N-1} + \dots + a_{N-1} z + a_N}.$$

В таком случае $X(z)$ можно разложить на элементарные дроби следующим образом:

$$\frac{N(z)}{zD(z)} = \frac{N(z)}{z(z-p_1)(z-p_2)(z-p_3)\dots(z-p_N)} = \frac{B_0}{z} + \sum_{k=1}^M \frac{C_k}{z-p_k}, \quad (4Б.4)$$

где

$$\begin{aligned} N(z) &= b_0 z^N + b_1 z^{N-1} + \dots + b_{N-1} z + b_N, \\ D(z) &= a_0 z^N + a_1 z^{N-1} + \dots + a_{N-1} z + a_N, \end{aligned}$$

p_k — полюсы функции $X(z)$ (предположим, первого порядка), а C_k — коэффициенты элементарных дробей. Постоянная B_0 задается как

$$B_0 = b_N/a_N. \quad (4Б.5)$$

Коэффициенты элементарных дробей C_k , связанные с полюсами p_k , находят путем умножения обеих частей уравнения 4Б.4 на $z-p_k$ с последующей заменой $z=p_k$:

$$C_k = \frac{N(z)(z-p_k)}{zD(z)} = \left. \frac{N(z)}{zD_k(z)} \right|_{z=p_k}, \quad (4Б.6)$$

где

$$D_k(z) = \prod_{i=1, i \neq k}^M (z-p_i).$$

Например, чтобы найти C_1 , нужно умножить обе части уравнения (4Б.4) на $z-p_1$ и сделать замену $z=p_1$:

$$C_1 = \frac{N(z)(z-p_1)}{zD(z)} = \frac{N(z)(z-p_1)}{z(z-p_1)(z-p_2)(z-p_3)\dots(z-p_N)} \Big|_{z=p_1} = \frac{N(z)}{zD_1(z)} \Big|_{z=p_1},$$

где

$$D_1(z) = (z-p_2)(z-p_3)\dots(z-p_N).$$

Если полюсы выражены в полярных координатах, т.е. $p_k = r_k e^{i\theta_k}$, то коэффициенты задаются как

$$C_k = \frac{N(r_k e^{i\theta_k})}{r_k e^{i\theta_k} D_k(e^{i\theta_k})} \quad k = 1, \dots, N \quad (4Б.7)$$

С помощью функции разложения на элементарные дроби вначале находят положения полюсов p_k $k = 1, 2, \dots, N$, а затем для каждого полюса вычисляется выражение (4Б.7).

Когда найдены значения B_0 и C_k , z -образ можно записать как

$$X(z) = B_0 + \sum_{k=1}^N \frac{C_k z}{z-p_k}. \quad (4Б.8)$$

Для причинных последовательностей обратное z -преобразование — это сумма обратных z -преобразований каждого члена уравнения (4Б.8):

$$x(n) = B_0 u(n) + C_1(p_1)^n + C_2(p_2)^n + \dots + C_N(p_N)^n. \quad (4Б.9)$$

Чтобы воспользоваться программой для вычисления коэффициентов элементарных дробей, z -преобразование должно быть выражено в виде последовательной структуры, через множители второго порядка. Поясним это на примере.

Пример 4Б.3

С помощью метода разложения на элементарные дроби найдите обратное z -преобразование передаточной функции пятого порядка, приведенной в примере 4Б.2.

Разложение передаточной функции на элементарные дроби имеет вид

$$X(z) = B_0 + \sum_{k=1}^5 \frac{C_k z}{z - p_k}. \quad (4Б.10)$$

Файл входных данных *coeff.data* не отличается от входного файла, приведенного в примере 4Б.2. Выход программы дается ниже:

полюсы z -образа

pk	real	imag	mag	phase
1	0,716754	0,586833	0,926342	39,308436
2	0,716754	-0,586833	0,926342	-39,308436
3	0,646801	0,372261	0,746277	29,922232
4	0,646801	-0,372261	0,746277	-29,922232
5	0,612159	0,000000	0,612159	0,000000

коэффициенты элементарных дробей

$B_0 = -3,418163$

Ck	real	imag	mag	phase
1	1,611473	5,209672	5,453212	72,811944
2	1,611473	-5,209672	5,453212	-72,811944
3	-19,580860	-9,681908	21,843751	-153,689550
4	-19,580860	9,681908	21,843751	153,689550
5	40,356939	0,000000	40,356939	0,000000

Согласно этим данным обратное z -преобразование задается как

$$x(n) = B_0 u(n) + \sum_{k=1}^5 C_k (p_k)^n, \quad n \geq 0.$$

С помощью табл. 4.1 также можно найти обратное z -преобразование $x(n)$ в виде, который объединяет члены с комплексно-сопряженными полюсами. Это предлагается сделать читателям в качестве самостоятельного упражнения.

4.Б.3. Преобразование последовательной структуры в параллельную

Для перевода z -образа из последовательной в параллельную структуру можно также воспользоваться программой, в основе которой лежат принципы, описанные в примере 4.14.

Пример 4Б.4

Передающая функция системы дискретного времени четвертого порядка, реализованной в виде последовательной структуры, задается как

$$H(z) = \frac{N_1(z)N_2(z)}{D_1(z)D_2(z)},$$

где

$$\begin{aligned} D_1(z) &= 1 + 0,052921z^{-1} + 0,83173z^{-2}, \\ N_1(z) &= 1 + 0,481199z^{-1} + z^{-2}, \\ D_2(z) &= 1 - 0,304609z^{-1} + 0,238865z^{-2}, \\ N_2(z) &= 1 + 1,474597z^{-1} + z^{-2}. \end{aligned}$$

Используем программу преобразования передаточной функции из последовательной структуры в параллельную с таким файлом входных данных:

```
2
1  0,05292    0,83173
1  0,481199   1
1  -0,304609  0,238865
1  1,474597   1
```

В результате получим такой выход:

```
выбор желаемого действия
0  z-1 методом степенного ряда
1  оценка коэффициентов элементарных дробей
2  преобразование последовательной структуры в параллельную
2
```

полюсы z -преобразования:

pk	real	imag	mag	phase
1	-0,26460	0,911413	0,911797	91,662967
2	-0,26460	-0,911413	0,911797	-91,662967
3	0,152305	0,464401	0,488738	71,842631
4	0,152305	-0,464401	0,488738	-71,842631

коэффициенты элементарных дробей:

$B0 = 5,035604$

Ck	real	imag	mag	phase
1	-0,257338	0,421333	0,493705	121,415410
2	-0,257338	-0,421333	0,493705	-121,415410
3	-1,760464	-3,766287	4,157421	-115,052650
4	-1,760464	3,766287	4,157421	115,052650

Для продолжения нажмите <Enter>.

каскад	Ni(z)			
0	-0,514677	-0,781635		
1	-3,520927	4,034388		
2	0,000000	-0,000000		
каскад	Di(z)			
0	1,000000	0,52921	0,831373	
1	1,000000	-0,304609	0,238865	
2	0,000000	0,000000	0,000000	

4.B. Программа на языке C для вычисления частотной характеристики

С помощью этой программы вычисляется частотная характеристика либо путем непосредственного расчета, либо через БПФ, как описывается в разделе 4.5.5. Задать z -образ системы, частотную характеристику которой нужно найти, следует либо непосредственно, либо в виде последовательной структуры. Объясним сказанное на примере.

Пример 4B.1

Найдите частотную характеристику системы дискретного времени, передаточная функция которой задается в виде

$$H(z) = \frac{1 - 1,6180z^{-1} + z^{-2}}{1 - 1,5161z^{-1} + 0,878z^{-2}},$$

с помощью

- а) метода непосредственной оценки;
- б) метода БПФ.

Пусть частота дискретизации равна 500 Гц, а разрешающая способность меньше 1 Гц.

Решение

Чтобы достичь желаемой разрешающей способности, количество частотных точек, которое задается в программе `npt`, должно быть равным 512 для метода БПФ ($500/512 = 0,98$ Гц) и 256 для метода прямой оценки. Далее с помощью каждого метода для входных данных

```
1
1  -1,5161  0,878
1  -1,618   1
```

была найдена частотная характеристика. В любом случае характеристика записывается в формате ASCII в три файла, как показано ниже:

magn.dat содержит амплитудную характеристику в децибелах;
phase.dat содержит фазовую характеристику в радианах;
fresp.dat содержит частотную характеристику в прямоугольной форме.

Первые 10 значений амплитудной и фазовой характеристик приведены в табл. 4В.1, а амплитудная и фазовая характеристики для непосредственного метода изображены на рис. 4В.1, а и б соответственно.

Программа для вычисления частотной характеристики состоит из пяти функций, которые хранятся в отдельных файлах, как показано ниже:

freqresl.c главная функция;
fixdata.c вычисление амплитуды и фазовых углов;
freqd.c непосредственная оценка частотной характеристики;
fft.c алгоритм БПФ с двоичной временной децимацией;
ltilib.c набор общих функций ЦОС. Эта функция
 идентична функции ltilib.c, описанной в приложении 4Б,
 за исключением того, что она не требует файла .h.

Эти функции здесь не приводятся из соображений экономии места, но их можно найти на компакт-диске, прилагаемом к книге [Ifeachor, 2001] (подробнее см. предисловие).

Таблица 4В.1. Первые 10 значений амплитудной и фазовой характеристик (пример 4Б.1), если применяется метод непосредственной оценки или БПФ

<i>k</i>	<i>Непосредственная оценка</i>		<i>Оценка методом БПФ</i>	
	<i>Амплитуда (дБ)</i>	<i>Фаза (рад)</i>	<i>Амплитуда (дБ)</i>	<i>Фаза (рад)</i>
0	0,469496	0,000000	0,469496	0,000000
1	0,46939	−0,004155	0,469391	−0,004138
2	0,469073	−0,008318	0,469076	−0,008286
3	0,468541	−0,012500	0,468549	−0,012451
4	0,467791	−0,016710	0,467805	−0,016644
5	0,466817	−0,020956	0,466839	−0,020873
6	0,465612	−0,025249	0,465643	−0,025148
7	0,464165	−0,029599	0,464208	−0,029479
8	0,462466	−0,034016	0,462523	−0,033876
9	0,460501	−0,038511	0,460574	−0,038351

4.Г. Операции z-преобразования с помощью программного пакета MATLAB

В дополнение к использованию программ на языке С, описанных в предыдущих разделах, в нескольких следующих разделах будет проиллюстрировано использование функций программ MATLAB для выполнения ряда операций z-преобразования

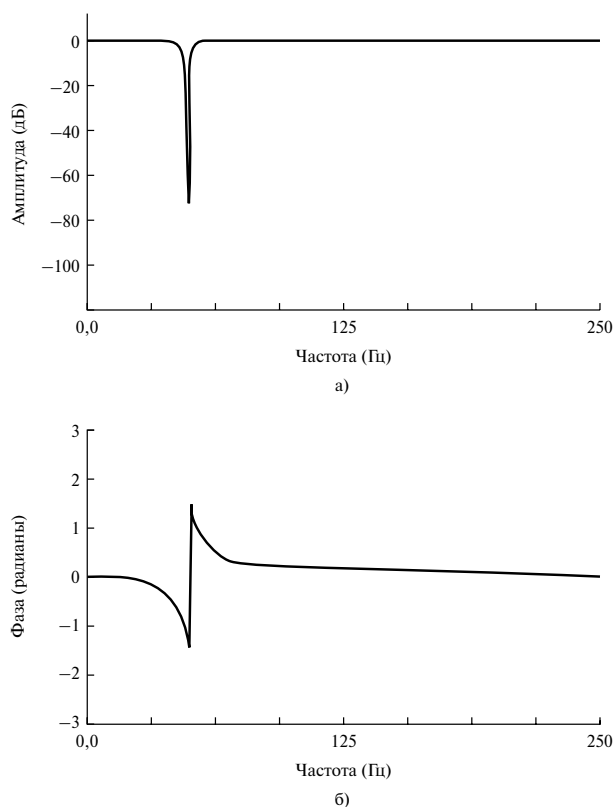


Рис. 4В.1. Амплитудно-частотная характеристика БИХ-системы из примера 4В.1 при непосредственной оценке (панель *а*). Фазовая характеристика БИХ-системы из примера 4В.1 при непосредственной оценке (панель *б*)

и обратного z -преобразования. Вообще, MATLAB и MATLAB Signal Processing Toolbox представляют собой быстрое и удобное средство для выполнения ряда операций z -преобразования и обратного z -преобразования, необходимых для проектирования и анализа систем ЦОС.

Дополнительные примеры, иллюстрирующие выполнение операций z -преобразования с помощью MATLAB, можно найти в книге [Ifeachor, 2001].

4.Г.1. Обратное z -преобразование

Ключевые функции MATLAB для выполнения операций обратного z -преобразования — это `deconv` и `residuez`. Функция `deconv` используется для выполнения деления в столбик, которое требуется для разложения в степенной ряд. Функция `residuez` используется при поиске коэффициентов элементарных дробей (вычетов) и полюсов z -преобразования.

4.Г.2. Разложение в степенной ряд с помощью программного пакета MATLAB

При разложении в степенной ряд ключевой операцией является деление многочленов, при котором используется функция MATLAB `deconv`. Вообще, в методе разложения в степенной ряд используется тот факт, что операция разложения эквивалентна делению многочленов. Следовательно, если дано z -преобразование $X(z)$ в виде:

$$X(z) = \frac{b_0 + b_1 z^{-1} + \dots + b_n z^{-n}}{a_0 + a_1 z^{-1} + \dots + a_m z^{-m}} = \frac{b(z)}{a(z)},$$

то формат команды разложения будет следующим:

$$[q, r] = \text{deconv}(b, a),$$

где b и a — векторы, представляющие многочлены числителя и знаменателя $b(z)$ и $a(z)$ соответственно при увеличении отрицательного показателя степени z . Результат деления многочленов возвращается в виде вектора q , а остаток записывается как r . Для реализации метода разложения в степенной ряд операция деления в столбик выполняется многократно, в зависимости от количества точек, необходимых для обратной операции.

Пример 4Г.1

Найдите первые пять членов обратного z -преобразования $x(n)$ с помощью метода разложения в степенной ряд (деления многочленов) и программного пакета MATLAB. Предположите, что z -образ $X(z)$ имеет следующий вид:

$$X(z) = \frac{1 + 2z^{-1} + z^{-2}}{1 - z^{-1} + 0,3561z^{-2}}.$$

Решение

Ниже приводится набор команд MATLAB и выход программы. Сначала образуются векторы коэффициентов для многочленов числителя и знаменателя, к вектору коэффициентов b прибавляются нули, чтобы обеспечить правильную размерность для программы MATLAB, а затем для вычисления обратного z -преобразования используется команда `deconv`.

```
>>
>> b = [1 2 1];
>> a = [ 1 -1 0,3561 ];
>> n=5;
>> b = [b zeros(1, n-1)];
>> [x,r] = deconv(b,a);
>> disp(x)

1,0000   3,0000   3,6439   2,5756   1,2780
```

Следовательно, $x(0) = 1$, $x(1) = 3$, $x(2) = 3,6439$, $x(3) = 2,5756$ и $x(4) = 1,2780$.

Пример 4Г.2

С помощью метода разложения в степенной ряд (деления многочленов) и программного пакета MATLAB найдите первые пять членов обратного z -преобразования следующего выражения

$$X(z) = \frac{N_1(z)N_2(z)N_3(z)}{D_1(z)D_2(z)D_3(z)},$$

где

$$N_1(z) = 1 - 1,22346z^{-1} + z^{-2}$$

$$N_2(z) = 1 - 0,437833z^{-1} + z^{-2}$$

$$N_3(z) = 1 + z^{-1}$$

$$D_1(z) = 1 - 1,433509z^{-1} + 0,85811z^{-2}$$

$$D_2(z) = 1 - 1,293601z^{-1} + 0,556929z^{-2}$$

$$D_3(z) = 1 - 0,612159z^{-1}$$

Решение

Представленный z -образ имеет три пары многочленов числителя и знаменателя. В коде MATLAB (программа 4Г.1) вначале образуются векторы, содержащие коэффициенты многочленов. Затем для преобразования трех пар многочленов в передаточную функцию с одной парой рациональных многочленов $b(z)/a(z)$ используется функция MATLAB `sos2tf` (функция преобразования звеньев второго порядка в полную передаточную функцию):

$$X(z) = \frac{b(z)}{a(z)} = \frac{b_0 + b_1z^{-1} + b_2z^{-2} + \dots + b_mz^{-m}}{a_0 + a_1z^{-1} + a_2z^{-2} + \dots + a_nz^{-n}}.$$

Чтобы найти коэффициенты обратного z -преобразования, используется функция `deconv`. Первые пять значений обратного z -преобразования равны:

$$x(0) = 1,00000; x(1) = 4,6915; x(2) = 11,4246; x(3) = 19,5863; x(4) = 27,0284$$

Программа 4Г.1

```
n=5; % количество точек степенного ряда
N1=[1 -1.122346 1]; D1 = [1 -1.433509 0.85811];
N2=[1 1.474597 1]; D2 = [1 -1.293601 0.556929];
N3=[1 1 0]; D3 = [1 -0.612159 0];
B=[N1; N2; N3]; A = [D1; D2; D3];
[b,a] = sos2tf([B A]);
b=[b zeros(1,n-1)];
[x,r] = deconv(b,a); % деление в столбик
disp(x);
```

4.Г.3. Разложение на элементарные дроби с помощью MATLAB

Для разложения на элементарные дроби z -образа $X(z)$, выраженного как отношение двух многочленов, можно воспользоваться функцией MATLAB `residuez`. Синтаксис команды `residuez`:

$$[r, p, k] = \text{residuez}(b, a),$$

где b и a — векторы, представляющие многочлены числителя и знаменателя $b(z)$ и $a(z)$ соответственно при возрастающем значении отрицательного показателя степени z :

$$H(z) = \frac{b(z)}{a(z)} = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_m z^{-m}}{a_0 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_n z^{-n}}.$$

Если полюсы функции $H(z)$ отдельные, ее разложение на элементарные дроби выглядит как:

$$\frac{b(z)}{a(z)} = \frac{r_1}{1 - p_1 z^{-1}} + \dots + \frac{r_n}{1 - p_n z^{-1}} + k_1 + k_2 z^{-1} + \dots + k_{m-n-1} z^{-(m-n)}.$$

Функция `residuez` заносит вычеты рационального многочлена $b(z)/a(z)$ в вектор r , положения полюсов — в вектор p и постоянные члены — в k .

Пример 4Г.3

Разложите на элементарные дроби следующий z -образ:

$$X(z) = \frac{1 + 2z^{-1} + z^{-2}}{1 - z^{-1} + 0,3561z^{-2}}.$$

Решение

В этом примере многочлены уже заданы в нужном виде, поэтому мы можем непосредственно применить команду:

```
>>[r,p,k] = residuez([1.2.1], [1,-1,0.3561])
```

```
r =
```

```
-0.9041 - 5.9928i
```

```
-0.9041 + 5.9928i
```

```
p =
```

```
0.5000 + 0.3257i
```

```
0.5000 - 0.3257i
```

```
k = 2.8082
```

Таким образом, z -образ, выраженный как разложение на элементарные дроби, приобретает вид:

$$X(z) = 2,8082 + \frac{r_1}{1 - p_1 z^{-1}} + \frac{r_2}{1 - p_2 z^{-1}},$$

где

$$\begin{aligned} r_1 &= -0,9041 - 5,9928i & r_2 &= -0,9041 + 5,9928i \\ p_1 &= 0,5 + 0,3257i & p_2 &= 0,5 - 0,3257i \end{aligned}$$

Пример 4Г.4

С помощью программного пакета MATLAB разложите на элементарные дроби следующий z -образ:

$$X(z) = \frac{N_1(z)N_2(z)N_3(z)}{D_1(z)D_2(z)D_3(z)},$$

где

$$\begin{aligned} N_1(z) &= 1 - 1,22346z^{-1} + z^{-2}, \\ N_2(z) &= 1 - 0,437833z^{-1} + z^{-2}, \\ N_3(z) &= 1 + z^{-1}, \\ D_1(z) &= 1 - 1,433509z^{-1} + 0,85811z^{-2}, \\ D_2(z) &= 1 - 1,293601z^{-1} + 0,556929z^{-2}, \\ D_3(z) &= 1 - 0,612159z^{-1}. \end{aligned}$$

Для преобразования многочленов числителя и знаменателя в единственную пару многочленов $b(z)/a(z)$ используется функция MATLAB `sos2tf`. Затем для разложения на элементарные дроби применяется функция `residuez`. Набор команд MATLAB для разложения $X(z)$ на элементарные дроби приведен в программе 4Г.2. При выполнении программы 4Г.2 получаем такие коэффициенты элементарных дробей:

```
r =
-1,9022 + 4,6797i
-1,9022 - 4,6797i
-9,0607 - 13,5515i
-9,0607 + 13,5515i
24,7049

p =
0,7168 + 0,5868i
0,7168 - 0,5868i
0,6468 + 0,3723i
0,6468 - 0,3723i
0,6122
k =1
```

Программа 4Г.2

```
N1 = [1 -1,122346 1];
N2 = [1 -0,437833 1];
N3 = [1 1 0];
D1 = [1 -1,433509 0,85811];
D2 = [1 -1,293601 0,556929];
D3 = [1 -0,612159 0];
```

```

sos = [N1 D1; N2 D2; N3 D3];
[b,a] = sos2tf(sos);
[r,p,k] = residuez(b,a)

```

4.Г.4. Преобразование одной структуры в другую — последовательно-параллельное преобразование

Программный пакет MATLAB предлагает набор функций, которые позволяют относительно легко выполнять преобразования между различными форматами и структурами, используемыми в ЦОС. Особенно полезна возможность преобразования последовательной структуры в параллельную.

Пример 4Г.5

Повторите пример 4Б.4, используя MATLAB.

Набор команд MATLAB для этого преобразования дан в программе 4Г.3.

Программа 4Г.3

```

nstage = 2;
N1 = [1 0,481199 1];
N2 = [1 1,474597 1];
D1 = [1 0,052921 0,83173];
D2 = [1 -0,304609 0,238865];
sos = [N1 D1; N2 D2];
[b,a] = sos2tf(sos);
[c,p,k] = residuez(b,a);
m = length(b);
b0 = b(m)/a(m);
j = 1;
for i=1:nstage
    bk(j) = c(j)+c(j+1);
    bk(j+1) = -(c(j)*p(j+1)+c(j+1)*p(j));
    ak(j) = -(p(j)+p(j+1));
    ak(j+1) = p(j)*p(j+1);
    j = j+2;
end
b0
ak
bk
c
p
k

```

```

cprealization
b0 =
    5,0334
ck =

```



```

-0,3766 - 0,2460i
-0,3766 + 0,2460i
1,4804 - 1,3903i
1,4804 + 1,3903i
pk =
-0,0265 + 0,9116i
-0,0265 + 0,9116i
0,1523 + 0,4644i
0,1523 - 0,4644i
ks = 1
b0 = 1,2023
ak = 1,4746 1,0000 0,0529 0,8317
bk = -0,0000 -0,0000 0,4283 0,1683
c =
-0,0000 + 0,0000i
-0,0000 - 0,0000i
0,2141 - 0,0861i
0,2141 + 0,0861i

```

4.Г.5. Диаграмма нулей и полюсов

Функция MATLAB `zplane` позволяет рассчитывать и выводить на экран диаграмму нулей и полюсов. Синтаксис этой команды следующий:

$$\text{zplane}(b, a),$$

где b и a — векторы коэффициентов многочленов числителя и знаменателя $b(z)/a(z)$. В таком формате эта команда сначала выполняет поиск положений полюсов (т.е. корней многочленов $b(z)$ и $a(z)$ соответственно), а затем — построение диаграммы в плоскости z .

Пример 4Г.6

Система дискретного времени характеризуется следующей передаточной функцией:

$$H(z) = \frac{1 - 1,6180z^{-1} + z^{-2}}{1 - 1,5161z^{-1} + 0,878z^{-2}}.$$

Найдите и изобразите ее диаграмму нулей и полюсов. При вычислении частотной характеристики предположите, что частота дискретизации равна 500 Гц, а разрешающая способность меньше 1 Гц.

Решение

Необходимый код MATLAB:

```

b = [1 -1.6180 1]; % формирование многочленов числителя и знаменателя
a = [1 -1.5161 0.878];

```

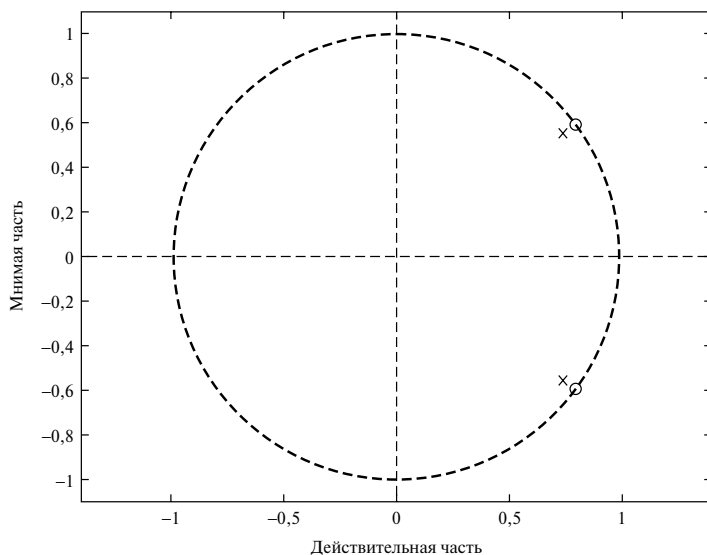


Рис. 4Г.1

`zplane(b,a)` % расчет и построение диаграммы нулей и полюсов

Искомая диаграмма нулей и полюсов показана на рис. 4Г.1.

Если положения полюсов и нулей известны, их можно использовать в качестве входа для команды `zplane`. Синтаксис этой команды в данном случае будет таким: `zplane(z,p)`, где z и p — нули и полюсы.

Положения нулей и полюсов можно найти непосредственно с помощью команды `roots`. Это полезно при переходе от описания через нули и полюсы к описанию через передаточную функцию.

Например, БИХ-фильтр описывается как

$$H(z) = \frac{1 - 1,6180z^{-1} + z^{-2}}{1 - 1,5161z^{-1} + 0,878z^{-2}}.$$

Полюсы и нули этого фильтра можно найти с помощью команды `roots`:

```
b = [1 -1,618 1];
a = [1 -1,5161 0,878];
zk = roots(b);
pk = roots(a)
```

Многочлены числителя и знаменателя $b(z)$ и $a(z)$ можно найти с помощью функции `poly`: `B=poly(zk); A = poly(pk)`.

4.Г.6. Оценка частотной характеристики

В Signal Processing Toolbox содержится множество полезных функций для вычисления и вывода на экран частотной характеристики систем дискретного времени. Самая распространенная функция — `freqz`.

Если передаточная функция системы задана в следующем виде:

$$X(z) = \frac{b_0 + b_1 z^{-1} + \dots + b_n z^{-n}}{a_0 + a_1 z^{-1} + \dots + a_m z^{-m}} = \frac{b(z)}{a(z)},$$

то для вычисления частотной характеристики используется функция `freqz`, в основе которой лежит метод БПФ. Удобнее всего использовать ее в таком формате — `[h, f] = freqz(b, a, npt, Fs)`, где переменные b и a — векторы многочленов числителя и знаменателя, Fs — частота дискретизации, а npt — количество частотных точек на отрезке между 0 и $F_s/2$. В MATLAB Toolbox единицей нормирования частоты является частота Найквиста (т.е. $F_s/2$). Использование команды `freqz` без выходного аргумента приводит к автоматическому построению графиков амплитудной и фазовой характеристик.

Пример 4Г.7

Система дискретного времени характеризуется следующей передаточной функцией:

$$H(z) = \frac{1 - 1,6180z^{-1} + z^{-2}}{1 - 1,5161z^{-1} + 0,878z^{-2}}.$$

Найдите и постройте график частотной характеристики системы, пользуясь программным пакетом MATLAB. Предположите, что частота дискретизации равна 500 Гц, а разрешающая способность меньше 1 Гц.

Решение

Команды MATLAB:

```
b = [1 -1,6180 1]; % формирование векторов многочленов числителя
% и знаменателя
a = [1 -1,5161 0,878];
freqz(b,a,256, 500) % расчет и построение графика частотной
% характеристики
```

Частотная характеристика данной системы дискретного времени показана на рис. 4Г.2.

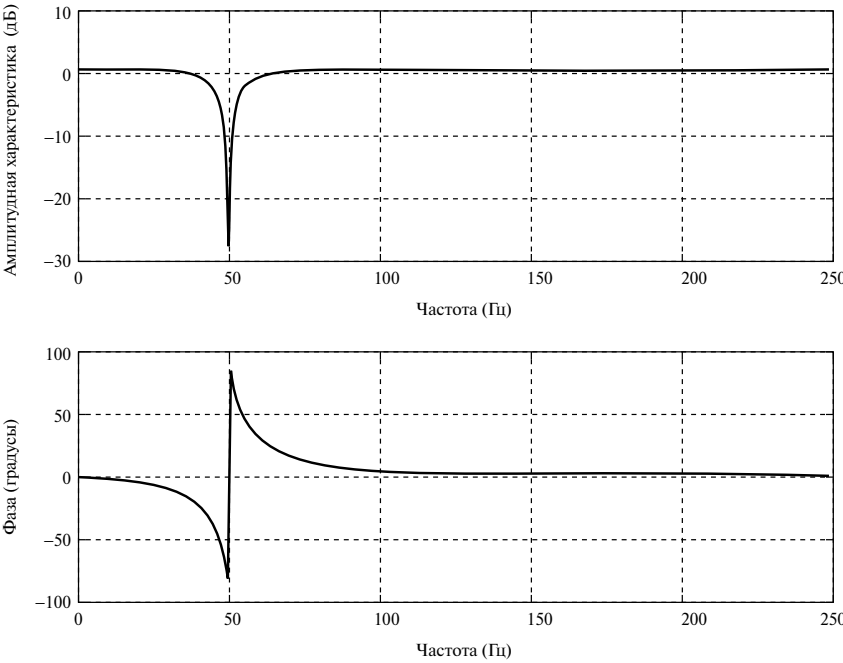


Рис. 4Г.2

Корреляция и свертка

5

5.1. Введение	281
5.2. Описание корреляции	282
5.3. Описание свертки	311
5.4. Реализация корреляции и свертки	339
5.5. Примеры применения	339
5.6. Резюме	347
Задачи	348
Литература	353
Приложение	354

В данной главе объясняется природа процесса корреляции, которая затем иллюстрируется на примерах вычисления функций взаимной корреляции и автокорреляции. Описано подавление шумовой составляющей сигнала с помощью корреляции и техника быстрой корреляции с использованием БПФ. Тема свертки рассматривается примерно по той же схеме, что и тема корреляции. В описание включены круговая и линейная свертка, быстрая линейная свертка и метод сегментов (наложение-сложение, наложение-запись), необходимый при обработке значительных массивов входных данных. Кроме того, рассматривается обращенная свертка, устанавливается связь между корреляцией и сверткой. В завершение главы рассказано о реализации корреляции и свертки и разобрано несколько примеров применения этих процедур.

5.1. Введение

Часто бывает необходимо определить степень независимости одного процесса от другого или установить сходство одного набора данных с другим. Другими словами, искомой является корреляция процессов или данных, которую можно определить математически и измерить. Вообще, процесс корреляции занимает значительное место в обработке сигналов. Этот математический аппарат нашел применение в обработке изображений в сфере компьютерного зрения или дистанционного зондирования со спутников, в которых сравниваются данные с различных изображений, в радарных или гидроакустических установках для дальнометрии и местоопределения (пеленгации), в которых сравниваются переданные и отраженные сигналы. Он используется в детектировании и идентификации сигналов в шуме, в организации технического контроля для наблюдения за влиянием входа на выход, в идентификации двоичных кодовых

слов в системе с импульсно-кодовой модуляцией, в обычных схемах оценки по методу наименьших квадратов и во многих других областях, например, в климатологии. Корреляция также является неотъемлемой частью процесса свертки, который, по сути, — та же корреляция двух последовательностей данных, при вычислении которой одна из последовательностей обращена во времени. Это означает, что для вычисления корреляции и свертки могут использоваться одни и те же алгоритмы, только в случае свертки одна из последовательностей обращается. Отметим также, что спектр записанного сигнала состоит из свертки спектра сигнала со спектром вырезающей функции.

Определение импульсной характеристики неизвестной системы называется *идентификацией системы*. Определение неизвестного выхода по импульсной характеристике системы и выходу называется *обращением свертки* (нахождением оригинала функции). Если импульсная характеристика неизвестна, определение искомого входного сигнала называется *слепым обращением свертки* (blind deconvolution). Все эти важные темы рассмотрены ниже.

5.2. Описание корреляции

Рассмотрим, как можно сравнить две последовательности данных, состоящие из значений, одновременно выбираемых из двух соответствующих сигналов. Если два сигнала похоже меняются при переходе от точки к точке, то меру их *корреляции* можно вычислить, взяв сумму произведений соответствующих пар точек. Данное предложение становится более аргументированным, если рассмотреть две независимые и случайные последовательности данных. В этом случае сумма произведений стремится к исчезающе малому случайному числу по мере увеличения пар точек. Это объясняется тем, что все числа, положительные и отрицательные, равновероятны, так что пары произведений компенсируются при сложении. В то же время, если сумма конечна, это указывает на наличие корреляции. Отрицательная сумма указывает на отрицательную корреляцию, т.е. увеличение одной переменной связано с уменьшением другой. Таким образом, взаимную корреляцию $r_{12}(n)$ двух последовательностей данных $x_1(n)$ и $x_2(n)$, содержащих по N элементов, можно записать как

$$r_{12} = \sum_{n=0}^{N-1} x_1(n)x_2(n).$$

Впрочем, такое определение взаимной корреляции дает результат, который зависит от числа взятых точек. Чтобы это исправить, результат нормируется на число точек (делится на N). Данную операцию можно также рассматривать как усреднение суммы произведений. Итак, получаем следующее улучшенное определение:

$$r_{12} = \frac{1}{N} \sum_{n=0}^{N-1} x_1(n)x_2(n).$$

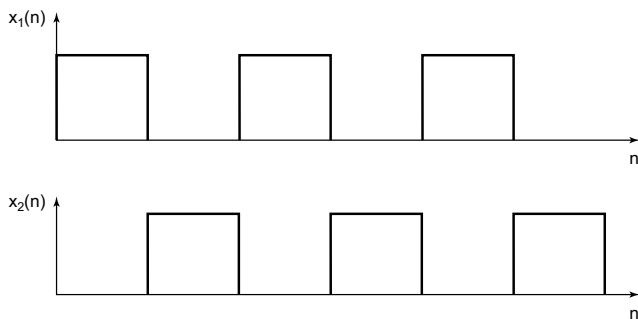


Рис. 5.1. Сигналы со 100%-ной корреляцией, идущие не в фазе, — при нулевой задержке корреляция равна нулю

Пример 5.1

Ниже иллюстрируется расчет r_{12} . Число точек данных в последовательности равно n , используются последовательности x_1 и x_2 .

n	1	2	3	4	5	6	7	8	9
x_1	4	2	-1	3	-2	-6	-5	4	5
x_2	-4	1	3	7	4	-2	-8	-2	1

$$\begin{aligned}
 r_{12} &= \frac{1}{9} (4 \times (-4) + 2 \times 1 + (-1) \times 3 + 3 \times 7 + (-2) \times 4 + (-6) \times (-2) + \\
 &\quad + (-5) \times (-8) + 4 \times (-2) + 5 \times 1) = \\
 &= 5.
 \end{aligned}$$

Впрочем, чтобы данное определение можно было использовать, его также нужно модифицировать. В некоторых случаях корреляция, определенная указанным выше способом, может быть нулевой, хотя две последовательности коррелируют на 100%. Это может произойти, например, когда два сигнала идут не в фазе (как часто и бывает). Данная ситуация иллюстрируется сигналами на рис. 5.1. На рисунке показано, что каждая пара произведений в функции корреляции равна нулю, следовательно, вся корреляция равна нулю, поскольку нулю всегда равно одно из значений x_1 или x_2 . Впрочем, очевидно, что сигналы сильно коррелируют, хотя и идут не в фазе. Разность фаз может, например, объясняться тем, что x_1 — некий эталонный сигнал, а x_2 — запаздывающий выход схемы. Чтобы преодолеть подобный сдвиг фаз, необходимо сдвинуть (или задержать) один из сигналов относительно другого. Обычно, чтобы выровнять сигналы перед определением корреляции, x_2 смещается влево. Как показано на рис. 5.2, это эквивалентно замене $x_2(n)$ на $x_2(n + j)$, где j представляет величину задержки — число точек выборки, на которое x_2 смещается влево. Альтернативной и эквивалентной процедурой является смещение x_1 вправо. В результате получаем такую формулу для взаимной корреляции:

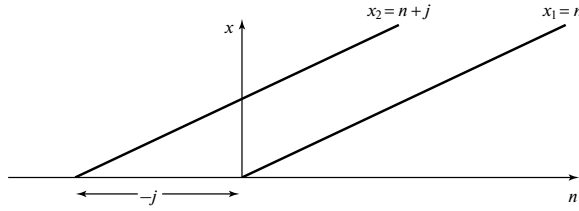


Рис. 5.2. Сигнал $x_2 = x_1 + j$, смещенный на j промежутков времени влево от сигнала x_1

$$\begin{aligned}
 r_{12}(j) &= \frac{1}{N} \sum_{n=0}^{N-1} x_1(n)x_2(n+j) = \\
 &= r_{12}(-j) = \frac{1}{N} \sum_{n=0}^{N-1} x_2(n)x_1(n-j).
 \end{aligned} \tag{5.1}$$

На практике, когда два сигнала коррелируют, их фазовая связь скорее всего неизвестна, так что корреляцию нужно находить для нескольких различных задержек, чтобы установить наибольшее значение корреляции, которое затем считается истинным.

Пример 5.2

Рассмотрим взаимную корреляцию указанных выше двух последовательностей $x_1(n)$ и $x_2(n)$ с задержкой $j = 3$, т.е. $r_{12}(3)$. Итак, используются следующие последовательности:

n	1	2	3	4	5	6	7	8	9
x_1	4	2	-1	3	-2	-6	-5	4	5
x_2	7	4	-2	-8	-2	-1			

так что

$$\begin{aligned}
 r_{12}(3) &= \frac{1}{9} (4 \times 7 + 2 \times 4 + (-1) \times (-2) + 3 \times (-8) + (-2) \times (-2) + \\
 &\quad + (-6) \times (-1)) = 2,667.
 \end{aligned}$$

Разумеется, также можно рассмотреть корреляцию в непрерывной временной области, и некоторые аналоговые схемы корреляции организованы именно так. В непрерывной области $n \rightarrow t$ и $j \rightarrow \tau$ и

$$r_{12}(\tau) = \lim_{T \rightarrow \infty} \frac{1}{T} \int_{-T/2}^{T/2} x_1(t)x_2(t+\tau)dt. \tag{5.2}$$

В то же время, если $x_1(t)$ и $x_2(t)$ — периодические с периодом T , формула (5.2) упрощается до

$$r_{12}(\tau) = \frac{1}{T_0} \int_{-T_0/2}^{T_0/2} x_1(t)x_2(t+\tau)dt. \tag{5.3}$$

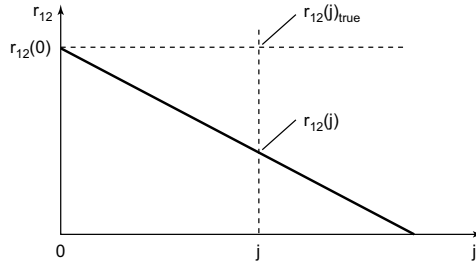


Рис. 5.3. Влияние краевого эффекта на взаимную корреляцию $r_{12}(j)$

Если данные — это сигналы с конечной энергией, например, непериодические импульсообразные сигналы, то усреднение по времени T при $T \rightarrow \infty$ не выполняется, поскольку в этом случае было бы $1/T \rightarrow 0$ и $1/T \rightarrow 0$, и $r_{12}(\tau)$ было бы исчезающе малой величиной. В таком случае используется следующая формула:

$$r_{12}(\tau) = \int_{-\infty}^{\infty} x_1(t)x_2(t+\tau)dt. \quad (5.4)$$

На практике обрабатываться будут записи конечной длины, так что применяются формулы (5.1) или (5.5):

$$r_{12}(\tau) = \frac{1}{T} \int_0^T x_1(t)x_2(t+\tau)dt. \quad (5.5)$$

Есть и другая сложность, связанная с нахождением взаимной корреляции последовательностей данных конечной длины. Это иллюстрирует приведенный выше пример, в котором было найдено значение $r_{12}(3) = 2,667$. При смещении x_2 влево сигналы уже не перекрываются, и данные в конце последовательностей не формируют парные произведения — возникает так называемый *краевой эффект*. В рассмотренном примере число пар при задержке 3 уменьшилось с девяти до шести. В результате наблюдалось линейное уменьшение $r_{12}(j)$ при увеличении j , и получены спорные значения $r_{12}(j)$. Одно из возможных решений возникшей проблемы заключается в том, чтобы длину одной последовательности сделать в два раза больше длины, необходимой для нахождения корреляции. Для этого можно записать больше данных или, если одна из последовательностей периодична, повторить последовательность (особое внимание следует обратить на согласование краев). Другое возможное решение — скорректировать все рассчитанные значения. На рис. 5.3 показано уменьшение r_{12} с ростом j в результате краевого эффекта, т.е. реальное изменение $r_{12}(j)$ не показано. При $j = 0$, $r_{12}(j) = r_{12}(0)$, что можно вычислить. При $j = N$, $r_{12}(N) = 0$, поскольку сигналы уже не перекрываются. В промежуточных случаях при некоторых значениях задержки j истинное значение $r_{12}(j)$ равно $r_{12}(j)_{\text{true}}$, тогда как действительное значение, искаженное краевым эффектом, равно $r_{12}(j)$. Далее на основе рисунка получаем

$$\frac{r_{12}(j)_{\text{true}} - r_{12}(j)}{j} = \frac{r_{12}(0)}{N},$$

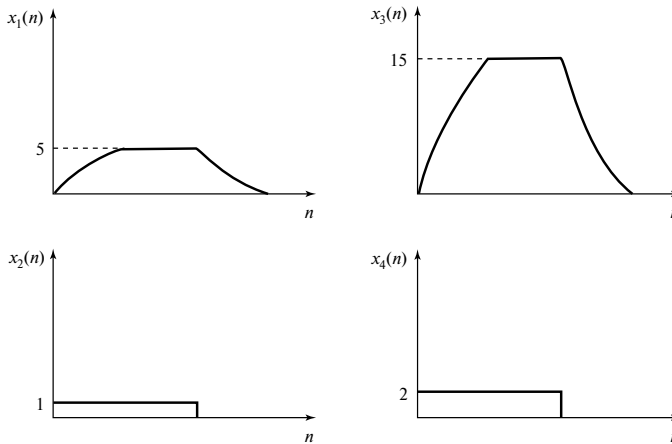


Рис. 5.4. Пары сигналов $\{x_1(n), x_2(n)\}$ и $\{x_3(n), x_4(n)\}$ различных амплитуд, но с равными взаимными корреляциями

откуда

$$r_{12}(j)_{\text{true}} = r_{12}(j) + \frac{j}{N} r_{12}(0). \quad (5.6)$$

Таким образом, вычисленные значения взаимной корреляции легко скорректировать для учета краевых эффектов, прибавив к $r_{12}(j)$ величину $j r_{12}(0)/N$.

Значения функции взаимной корреляции вычисляются согласно приведенным выше формулам в зависимости от абсолютных значений данных. Часто бывает необходимо измерить взаимную корреляцию в фиксированном масштабе между -1 и $+1$. Чтобы определить значения из указанного диапазона, полученные величины нормируют на величину, зависящую от энергетического содержания данных. Например, рассмотрим две пары сигналов $x_1(n)$, $x_2(n)$ и x_3 , $x_4(n)$. Значения элементов данных приведены в таблице ниже.

n	0	1	2	3	4	5	6	7	8
$x_1(n)$	0	3	5	5	5	2	0,5	0,25	0
$x_2(n)$	1	1	1	1	1	0	0	0	0
$x_3(n)$	0	9	15	15	15	6	1,5	0,75	0
$x_4(n)$	2	2	2	2	2	0	0	0	0

Как показано на рис. 5.4, сигналы $x_1(n)$ и $x_3(n)$ подобны и отличаются только амплитудой. То же справедливо для пары $x_2(n)$ и $x_4(n)$. Таким образом, корреляция между $x_1(n)$ и $x_2(n)$ равна корреляции между $x_3(n)$ и $x_4(n)$. В то же время, параметры корреляции r_1 и $r_{34}(1)$ равны соответственно 1,47 и 8,83. Они отличаются, поскольку зависят от абсолютных значений элементов данных. Чтобы поправить эту ситуацию, нормируем взаимную корреляцию $r_{12}(j)$ на коэффициент

$$\left[\frac{1}{N} \sum_{n=0}^{N-1} x_1^2(n) \times \frac{1}{N} \sum_{n=0}^{N-1} x_2^2(n) \right]^{1/2} = \frac{1}{N} \left[\sum_{n=0}^{N-1} x_1^2(n) \sum_{n=0}^{N-1} x_2^2(n) \right]^{1/2} \quad (5.7)$$

и подобным образом нормируем $r_{34}(j)$. В результате нормированное выражение для $r_{12}(j)$ приобретает такую форму:

$$\rho_{12}(j) = \frac{r_{12}(j)}{\frac{1}{N} \left[\sum_{n=0}^{N-1} x_1^2(n) \sum_{n=0}^{N-1} x_2^2(n) \right]^{1/2}}. \quad (5.8)$$

Величина $\rho_{12}(j)$ известна как *коэффициент взаимной корреляции*. Значение этого коэффициента всегда лежит между -1 и $+1$, причем “ $+1$ ” означает 100%-ную корреляцию в прямом смысле, “ -1 ” — 100%-ную корреляцию в противоположном смысле, например, как для сигналов в противофазе. Значение “0” указывает на нулевую корреляцию. Это означает, что сигналы совершенно независимы, например, если один из сигналов абсолютно случаен. Малые значения $\rho_{12}(j)$ указывают на незначительную корреляцию. Нормировочный коэффициент для $r_{12}(j)$ из приведенного выше примера равен

$$\frac{1}{N} \left[\sum_{n=0}^{N-1} x_1^2(n) \sum_{n=0}^{N-1} x_2^2(n) \right]^{1/2} = \frac{1}{9} (88,31 \times 6)^{1/2} = 2,56,$$

а для $r_{34}(j)$ —

$$\frac{1}{N} \left[\sum_{n=0}^{N-1} x_1^2(n) \sum_{n=0}^{N-1} x_2^2(n) \right]^{1/2} = \frac{1}{9} (794,8 \times 24)^{1/2} = 15,35.$$

Следовательно,

$$\rho_{12}(1) = \frac{r_{12}(1)}{2,56} = \frac{1,47}{2,56} = 0,57$$

и

$$\rho_{34}(1) = \frac{r_{34}(1)}{15,34} = \frac{8,83}{15,35} = 0,58.$$

Теперь $\rho_{12}(1) = \rho_{34}(1)$, откуда видно, что данный процесс нормировки действительно позволяет независимо сравнивать взаимные корреляции абсолютных значений данных.

Рассмотрим частный случай $x_1(n) = x_2(n)$, т.е. найдем корреляцию сигнала с самим собой. Данный процесс называется *автокорреляцией*. Автокорреляционная функция сигнала определяется как

$$r_{11}(j) = \frac{1}{N} \sum_{n=0}^{N-1} x_1(n) x_1(n+j).$$

Автокорреляционная функция имеет одно весьма полезное свойство:

$$r_{11}(0) = \frac{1}{N} \sum_{n=0}^{N-1} x_1^2(n) = S,$$

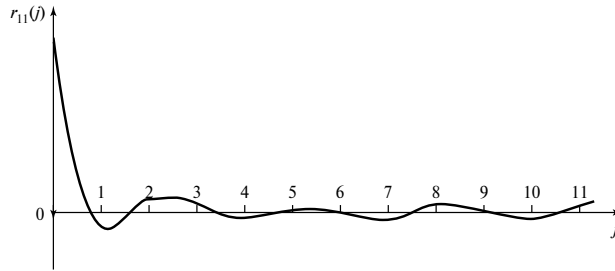


Рис. 5.5. Автокорреляционная функция случайного сигнала

где S — нормированная энергия сигнала. В результате получаем метод расчета энергии сигнала. Если сигнал абсолютно случаен, например, сигнал, соответствующий белому гауссову шуму в электрической системе, его автокорреляция будет максимальной при нулевой задержке и уменьшаться до случайных флуктуаций малой амплитуды возле нуля для задержек, превышающих единицу (см. рис. 5.5). Кроме того, справедливо следующее соотношение:

$$r_{11}(0) \geq r_{11}(j).$$

5.2.1. Взаимная корреляция и автокорреляция

Определение взаимной корреляции двух периодических последовательностей неравной длины требует аккуратности. Это объясняется тем, что результат корреляции будет повторяться с периодом более короткой последовательности. Этот результат не отражает полной периодичности более длинной последовательности, следовательно, неверен. Продемонстрируем это, найдя взаимную корреляцию $r_{ab}(j)$ последовательностей $a = \{4, 3, 1, 6\}$ и $b = \{5, 2, 3\}$. Последовательность b записывается под a и поэтапно смещается на одну позицию влево, в последнем столбце записываются соответствующие значения взаимной корреляции.

Последовательность				Задержка	$r_{ab}(j)$	$r_{ab}(j)$ повторяется
4	3	1	6			
3	5	2	3	0	47	
5	2	3	5	1	59	
2	3	5	2	2	34	
3	5	2	3	3	47	
5	2	3	5	4	59	
и т.д.						

Результат показывает, что $r_{ab}(j)$ циклично с периодом в три задержки, т.е. период $r_{ab}(j)$ равен периоду более короткой последовательности b . Описанная процедура называется *циклической корреляцией*. Чтобы получить правильное значение, в котором каждое значение a умножается на каждое значение b , все элементы b нужно последовательно сместить под каждым значением a :

```

      4  3  1  6
          5  2  3
              5  2  3
                  5  2  3
                      5  2  3
                          5  2  3
                              5  2  3

```

Видно, что для того, чтобы последовательность b стала повторяться, требуется 6 задержек. Длины последовательностей равны 4 и 3, а число требуемых задержек — $4 + 3 - 1 = 6$. Таким образом, получаем общее правило нахождения линейной взаимной корреляции двух периодических последовательностей длины N_1 и N_2 : дополнить нулями каждую последовательность, чтобы их длины были равны $N_1 + N_2 - 1$ (т.е. добавить $N_2 - 1$ нулей к последовательности длиной N_1 и $N_1 - 1$ нулей к последовательности длиной N_2). Ниже сказанное иллюстрируется для указанных выше последовательностей a и b .

Последовательность	Задержка	$r_{ab}(j)$
4 3 1 6 0 0		
5 2 3 0 0 0	0	29
2 3 0 0 0 5	1	17
3 0 0 0 5 2	2	12
0 0 0 5 2 3	3	30
0 0 5 2 3 0	4	17
0 5 2 3 0 0	5	35
5 2 3 0 0 0	6	29
		$r_{ab}(j)$ повторяется

Следовательно, искомая взаимная корреляция a и b равна

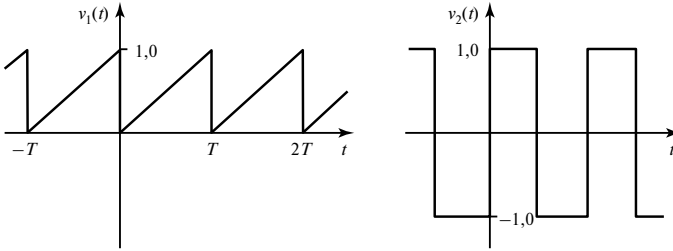
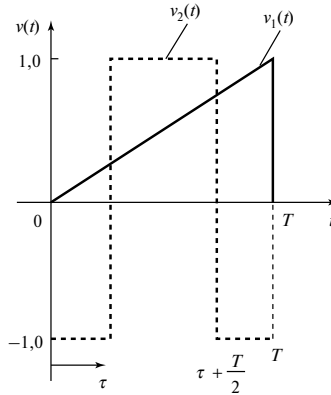
$$r_{ab}(j) = \{29, 17, 12, 30, 17, 35\}.$$

Выше предполагалось, что в моменты нахождения взаимной корреляции используются чисто цифровые данные, но взаимную корреляцию можно также посчитать аналитически, если сигнал записывается в явной (аналитической) форме. На практике эквивалент описанной аналитической процедуры применяется в аналоговых схемах взаимной корреляции. Ниже приводится пример аналитического расчета взаимной корреляции.

Пример 5.3

Найдите взаимную корреляцию $r_{12}(-\tau)$ двух сигналов $v_1(t)$ и $v_2(t)$, изображенных на рис. 5.6.

Требуемые сигналы легко записать аналитически, разделив их на прямолинейные сегменты. Это достаточно сделать для одного периода сигнала T , поскольку $r_{12}(-\tau)$ будет периодическим по τ с периодом T . Следовательно, для $0 \leq t \leq T$, $v_1(t) = t/T$ и для $0 \leq t \leq T/2$, $v_2(t) = 1, 0$, а для $T/2 \leq t \leq T$, $v_2(t) = -1, 0$. Далее требуется получить выражение для $r_{12}(-\tau)$, т.е. $v_2(t)$ (прямоугольный сигнал) необходимо сместить вправо относительно $v_1(t)$. Для $0 \leq \tau \leq T/2$ соответствующая иллюстрация приведена

Рис. 5.6. Сигналы $v_1(t)$ и $v_2(t)$ (пример 5.3)Рис. 5.7. Сегменты $v_2(t)$ для $\theta \leq \tau \leq T$

на рис. 5.7, из которого видно, что $v_1(t)$ нужно умножить на три последовательных сегмента $v_2(t)$, в которых значения $v_2(t)$ равны $-1, 1, -1$. Для $T/2 \leq \tau \leq T$ необходимая иллюстрация приведена на рис. 5.8, на котором последовательные значения $v_2(t)$ изменились на $1, -1, +1$. Это означает, что решение состоит из двух частей, которые нужно согласовать в точке $\tau = T/2$.

Разобьем взаимную корреляцию на три сегмента с границами в точках $t = \tau$, $t = \tau + T/2$ и $t = T$ (см. рис. 5.7). Далее получаем

$$\begin{aligned}
 r_{12}(-\tau) &= \frac{1}{T} \int_0^T v_1(t) v_2(t - \tau) dt = \\
 &= \frac{1}{T} \int_0^T \frac{t}{T} (-1) dt + \frac{1}{T} \int_{\tau}^{\tau+T/2} \frac{t}{T} (1) dt + \frac{1}{T} \int_{\tau+T/2}^T \frac{t}{T} (-1) dt = \quad (5.9) \\
 &= \frac{-1}{T^2} \left[\frac{t^2}{2} \right]_0^{\tau} + \frac{1}{T^2} \left[\frac{t^2}{2} \right]_{\tau}^{\tau+T/2} - \frac{1}{T^2} \left[\frac{t^2}{2} \right]_{\tau+T/2}^T,
 \end{aligned}$$

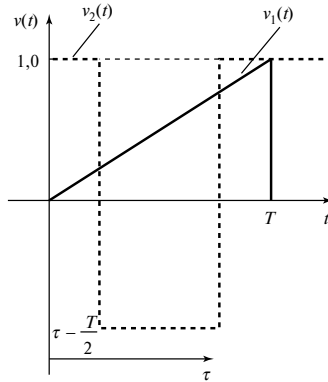


Рис. 5.8. Сегменты $v_2(t)$ для $T/2 \leq \tau \leq T$

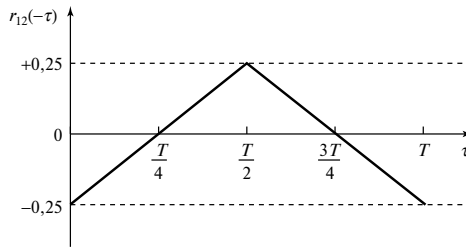


Рис. 5.9. Величина $r_{12}(-\tau)$ как функция τ

$$r_{12}(-\tau) = -\frac{1}{4} + \frac{\tau}{T} \text{ для } 0 \leq \tau \leq \frac{T}{2}.$$

Для $T/2 \leq \tau \leq T$, используя рис. 5.8, получаем

$$r_{12}(-\tau) = \frac{1}{T} \int_0^{\tau-T/2} \frac{t}{T}(1)dt + \frac{1}{T} \int_{\tau-T/2}^{\tau} \frac{t}{T}(-1)dt + \frac{1}{T} \int_{\tau}^T \frac{t}{T}(1)dt \quad (5.10)$$

$$r_{12}(-\tau) = \frac{3}{4} - \frac{\tau}{T} \text{ для } \frac{T}{2} \leq \tau \leq T.$$

Подставляя $\tau = T/2$ в формулы (5.9) и (5.10), определяем, что в обоих случаях $r_{12}(-\tau) = 1/4$, откуда следует, что функции согласованы правильно. График зависимости $r_{12}(-\tau)$ от τ для $0 \leq \tau \leq T$ изображен на рис. 5.9.

Отметим некоторые моменты, связанные с последствиями использования при вычислении корреляции данных конечной длины. Другими словами, чем отличается использование формулы (5.5.), в которой T конечно, вместо формулы (5.2)?

Для ответа на этот вопрос можно рассмотреть всего один синусоидальный Фурье-компонент сигнала. Формула (5.2) дает верную автокорреляцию, где $T \gg T_p$ (T_p — период синусоиды). Следовательно,

$$\begin{aligned} r_{11}(\tau) &= \lim_{T \rightarrow \infty} \frac{1}{2T} \int_{-T}^T A \sin(\omega t) A \sin(\omega t + \tau) dt = \\ &= \lim_{T \rightarrow \infty} \frac{A^2}{2} \left[\cos(\omega \tau) - \frac{\cos(\omega T)}{2\omega T} \sin(\omega \tau) \right]. \end{aligned} \quad (5.11)$$

Изучая данное уравнение, находим, что второе слагаемое в скобках стремится к нулю при $T \rightarrow \infty$, так что если $T \neq \infty$ — это указывает на ошибку. Член $\cos(\omega T)$ представляет периодическое влияние ошибки, а член $1/2\omega T$ — тенденцию в ошибку. Следовательно, при рассмотрении длины корреляции T ошибки больше для более коротких последовательностей, кроме того, они максимальные для низкочастотных компонентов сигнала. Ошибки также периодичны по τ .

Член $\cos(\omega T)$ дает наименьшие ошибки при $\omega T = [(2n + 1)/2]\pi$. Поскольку $\omega = 2\pi/T_p$ и искомыми являются большие значения T , это соответствует

$$T \geq (2n + 1) \frac{T_p}{4}. \quad (5.12)$$

Член $\sin(\omega \tau)$ дает наименьшее значение при $\omega \tau = m\pi$, где m — целое. Следовательно,

$$\tau = \frac{m}{2} T_p. \quad (5.13)$$

Теперь следует сделать несколько разумных предположений. Допустим, что условию большого T удовлетворяет $N \geq 10$. Тогда $T \geq nT_p/2$ или

$$T \geq 5T_p. \quad (5.14)$$

Из уравнения (5.13) находим, что наибольшее разрешенное значение τ для самого низкочастотного компонента $m = 1$ удовлетворяет условию

$$\tau < T_p. \quad (5.15)$$

Объединяя неравенства (5.14) и (5.15), получаем

$$\tau \leq T/5.$$

Это означает, что при нахождении корреляции сигналов ошибки, вызванные конечными длинами последовательностей данных, можно минимизировать следующим образом:

- 1) гарантировать, что $T \geq 5T_p$, где T_p соответствует наименьшему значимому частотному компоненту;
- 2) совмещать данные не более чем на 20% их длины.

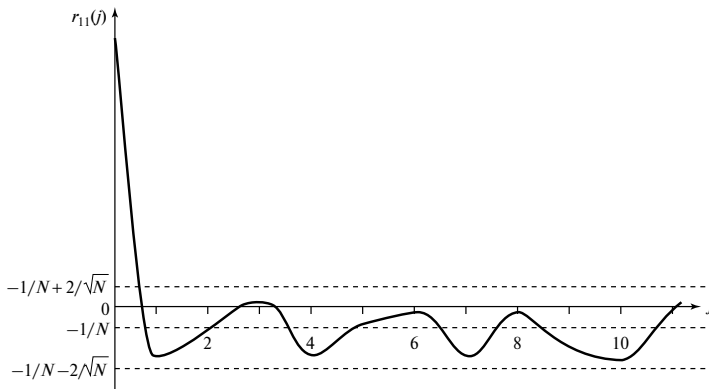


Рис. 5.10. Автокорреляционная функция случайного сигнала

Таким образом, например, если нужно найти корреляцию телефонного речевого сигнала с полосой 300–3400 Гц и того же сигнала, дискретизованного с частотой 40 кГц, $T_p = 1/300 = 3,3 \times 10^{-3}$ с. Наименьшая приемлемая длина данных составляет $5 \times 3,3 \times 10^{-3}$ с = 16,7 мс, а максимальное смещение при нахождении корреляции — 3,33 мс, или 133 точки данных.

На рис. 5.10 приведен график $\rho_{11}(j)$ коэффициента автокорреляции чисто случайного сигнала, например, белого шума. Можно показать, что математическое ожидание $r_{11}(j)$ равно $E[r_{11}] \approx -1/N$ [4], где N — число точек данных, а дисперсия равна $\text{var}[r_{11}(j)] \approx 1/N$. На рисунке показано математическое ожидание $-1/N$, а также доверительные границы по уровню 95% $-1/N$, равные $\pm 2/N^{1/2}$. Значения $r_{11}(j)$, которые не входят в эти доверительные пределы, могут быть значительными, т.е. они могут указывать, что сигнал не совсем случаен. При этом следует отметить, что одна точка из 20 может выходить за указанные пределы, даже если сигнал абсолютно случаен. Случайный сигнал $r_{11}(j)$ должен входить в доверительные границы по уровню 95% за одну-две задержки. Чтобы действительно гарантировать, что сигнал случайный, нужен определенный опыт и некоторые дополнительные действия. Например, в работе [7] рекомендуется использовать предварительное “отбеливание” данных (устранение белого шума).

Автокорреляционная функция периодического сигнала также является периодическим сигналом. Данное утверждение легко доказать. Периодический сигнал $x(t)$ с периодом T удовлетворяет условию

$$x(t) = x(t + nT),$$

так что

$$r_{11}(\tau) = \lim_{T \rightarrow \infty} \frac{1}{T} \int_{-T/2}^{T/2} x(t)x(t+\tau)dt =$$

$$= \lim_{T \rightarrow \infty} \frac{1}{T} \int_{-T/2}^{T/2} x(t)x(t + \tau + nT)dt \quad (5.16)$$

$$r_{11}(\tau) = r_{11}(\tau + nT).$$

Следовательно, сигнал $r_{11}(\tau)$ периодический по τ с периодом T . Данное свойство полезно, поскольку позволяет детектировать периодические сигналы в шуме при небольших отношениях сигнал-шум. Нахождение автокорреляции сигнала обычно снижает шум, при этом проявляя периодическую автокорреляционную функцию сигнала. При необходимости после детектирования можно выполнять дальнейшую обработку сигнала, чтобы определить его форму.

Из уравнения (5.11) видно, что автокорреляционная функция сигнала $A \sin(\omega t)$ равна $(A^2/2) \cos(\omega \tau)$. Поскольку амплитуда автокорреляционной функции связана просто с амплитудой сигнала, ее можно использовать для оценки амплитуды сигнала. Приведем другой распространенный пример — прямоугольный сигнал амплитуды A , амплитуда автокорреляционной функции которого равна A^2 , а сама функция имеет треугольную форму. Наконец, следует отметить, что автокорреляционные функции не уникальны. Это означает, что различные сигналы могут иметь одинаковые корреляционные функции. Таким образом, по найденной автокорреляционной функции нельзя определить форму сигнала.

Рассмотрим задачу, в которой сигнал $v(t)$ частично случайный. Итак, изучается зашумленный сигнал, который можно записать как сумму полезного сигнала $s(t)$ и шума $q(t)$. Таким образом,

$$v(t) = s(t) + q(t), \quad (5.17)$$

причем предполагается, что $s(t)$ и $q(t)$ не коррелируют. Выборочная автокорреляционная функция $v(t)$, равная $r_{vv}(j)$, записывается следующим образом:

$$r_{vv}(j) = \frac{1}{N} \sum_{n=0}^{N-1} [s(n) + q(n)][s(n+j) + q(n+j)] = \quad (5.18)$$

$$= \frac{1}{N} \sum_{n=0}^{N-1} s(n)s(n+j) + \frac{1}{N} \sum_{n=0}^{N-1} s(n)q(n+j) + \frac{1}{N} \sum_{n=0}^{N-1} q(n)s(n+j) + \quad (5.19)$$

$$\begin{aligned} &+ \frac{1}{N} \sum_{n=0}^{N-1} q(n)q(n+j) = \\ &= r_{ss}(j) + E[s(n)q(n+j)] + E[q(n)s(n+j)] + E[q(n)q(n+j)] = \\ &= r_{ss}(j) + E[s(n)]E[q(n+j)] + E[q(n)]E[s(n+j)] + E[q(n)]E[q(n+j)] = \\ &= r_{ss}(j) \overline{s(n)q(n)} + \overline{q(n)s(n)} + \overline{q(n)}^2 = \\ &= r_{ss}(j) + 2\overline{s} \overline{q} + \overline{q}^2. \end{aligned} \quad (5.20)$$

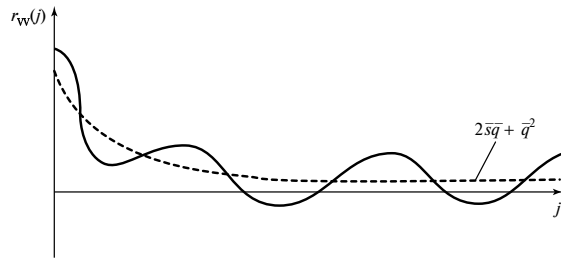


Рис. 5.11. Автокорреляционная функция зашумленного сигнала

Теперь $\bar{q} \rightarrow 0$ для больших N , для которых

$$r_{vv}(j) \rightarrow r_{ss}(j). \quad (5.21)$$

Для меньших N слагаемые взаимной корреляции в формуле (5.19) и автокорреляция шумовой составляющей стремятся к нулю с увеличением задержки j .

Таким образом, видно, что автокорреляционная функция частично случайного или зашумленного сигнала состоит из автокорреляционной функции сигнального компонента, на которую накладывается затухающая шумовая функция, зависящая от случайного компонента и полезного сигнала, и которая затухает до значения $2\bar{s}\bar{q} + \bar{q}^2$. Следовательно, график зависимости $r_{vv}(j)$ от j отражает периодичность $s(t)$ при условии $|r_{ss}(j)| > |(2\bar{s}\bar{q} + \bar{q}^2)|$ (см. рис. 5.11). В результате получаем метод определения периода сигнала в шуме (см. раздел 5.2.2).

Пример 5.4

Найдите функцию взаимной корреляции двух зашумленных сигналов.

Дано два сигнала $\{s_1(t) + q_1(t)\}$ и $\{s_2(t) + q_2(t)\}$. Их выборочная взаимная корреляция $r_{12}(j)$ записывается следующим образом:

$$\begin{aligned} r_{12}(j) &= \frac{1}{N} \sum_{n=0}^{N-1} [\{s_1(n) + q_1(n)\} \{s_2(n+j) + q_2(n+j)\}] = \\ &= \frac{1}{N} \sum_{n=0}^{N-1} [s_1(n)s_2(n+j) + s_1(n)q_2(n+j) + q_1(n)q_2(n+j)] = \end{aligned} \quad (5.22)$$

$$\begin{aligned} &= \frac{1}{N} \sum_{n=0}^{N-1} s_1(n)q_2(n+j) + \frac{1}{N} \sum_{n=0}^{N-1} q_1(n)s_2(n+j) + \\ &+ \frac{1}{N} \sum_{n=0}^{N-1} q_1(n)q_2(n+j) = \\ &= r_{s_1 s_2}(j) + r_{s_1 q_2}(j) + r_{q_1 s_2} + r_{q_1 q_2}(j). \end{aligned} \quad (5.23)$$

Как и в предыдущей задаче, последние три члена в правой части уравнения (5.23) затухают до нуля с увеличением задержки j . Для больших N уравнение (5.23) записывается следующим образом:

$$r_{12}(j) = r_{s_1 s_2}(j) + \overline{s_1 q_2} + \overline{q_1 s_2} + \overline{q_1 q_2}. \quad (5.24)$$

Таким образом, при увеличении N $r_{12}(j) \rightarrow r_{s_1 s_2}(j)$, функции взаимной корреляции двух сигналов.

Проведенный выше анализ демонстрирует, что функции взаимной корреляции и автокорреляции позволяют проявить свойства сигнала, снижая шумовую составляющую.

5.2.2. Применение корреляции

5.2.2.1. Расчет спектральной плотности энергии и энергетического содержимого сигнала

Можно показать, что

$$F[r_{11}(\tau)] = G_E(f), \quad (5.25)$$

где $G_E(f)$ — спектральная плотность энергии сигнала, т.е. спектральная плотность энергии и автокорреляционная функция являются Фурье-образами друг друга.

Далее можно показать, что

$$r_{11}(0) = E, \quad (5.26)$$

где E — общая энергия сигнала.

Пример 5.5

Найдите связь между корреляционными функциями с нулевой задержкой двух различных сигналов и их общим энергетическим содержанием.

Обозначим сигналы $v_1(n)$ и $v_2(n)$, а их сумму — $V(n) = v_1(n) + v_2(n)$. Автокорреляционная функция сигнала с нулевой задержкой $V(n)$ равна

$$r_{vv}(0) = E_v = \frac{1}{N} \sum_{n=0}^{N-1} V^2(n) = \frac{1}{N} \sum_{n=0}^{N-1} [v_1(n) + v_2(n)]^2,$$

где E_v — энергия сигнала $V(n)$.

$$\begin{aligned} E_v &= \frac{1}{N} \sum_{n=0}^{N-1} [v_1^2(n) + v_2^2(n) + 2v_1(n)v_2(n)] = \\ &= \frac{1}{N} \sum_{n=0}^{N-1} v_1^2(n) + \frac{1}{N} \sum_{n=0}^{N-1} v_2^2(n) + \frac{1}{N} \sum_{n=0}^{N-1} v_1(n)v_2(n), \end{aligned}$$

так что

$$E_v = r_{v_1}(0) = r_{v_2}(0) + 2r_{v_1 v_2}(0). \quad (5.27)$$

Полученное уравнение является одной из форм искомого результата. Альтернативным образом результат можно записать так:

$$E_V = E_{v_1} + E_{v_2} + 2r_{v_1 v_2}(0). \quad (5.28)$$

Итак, энергия сигнала $V(n)$ равна сумме энергий его компонентов плюс $2r_{v_1 v_2}(n)$, где $r_{v_1 v_2}(n)$ — функция взаимной корреляции с нулевой задержкой сигналов $v_1(n)$ и $v_2(n)$. Если $v_1(n)$ и $v_2(n)$ не коррелируют, общая энергия является просто суммой энергий компонентов.

Если сигналы $v_1(n)$ и $v_2(n)$ зашумленные и $v_1(n) = v'_1(n) + q_1(n)$ и $v_2(n) = v'_2(n) + q_2(n)$, легко показать, что

$$E_V = E_{v'_1} + E_{v'_2} + E_{q_1} + E_{q_2} + r_{v'_1 v'_2}(0). \quad (5.29)$$

5.2.2.2. Детектирование и оценка периодических сигналов в шуме

Рассмотрим использование взаимной корреляции для детектирования и оценки периодических сигналов в шуме. Известно, что сигнал, спрятанный в шуме, можно оценить, найдя его взаимную корреляцию с настраиваемым шаблонным (“эталонным”) сигналом. Шаблон настраивается методом проб и ошибок с использованием любых предварительных знаний, пока функция взаимной корреляции не достигнет максимального значения. Для подкрепления этого предложения можно обратиться к уравнению (5.22), предполагая, что для шаблона $q_2(n) = 0$. При таком условии уравнение (5.23) переходит в следующее:

$$r_{12}(j) = r_{s_1 s_2}(j) + r_{q_1 s_2}(j) = \quad (5.30)$$

$$= r_{s_1 s_2} + \bar{q}_1 \bar{s}_2. \quad (5.31)$$

Поскольку $\bar{q}_1 \rightarrow 0$ при увеличении N ,

$$r_{12}(j) \rightarrow r_{s_1 s_2}(j). \quad (5.32)$$

Очевидно, $r_{s_1 s_2}(j)$ имеет максимум при $s_2(n) = s_1(n)$, где $r_{s_1 s_2}$ — автокорреляционная функция сигнала $s_1(n)$. Таким образом, меняя форму шаблона $s_2(n)$ с целью максимизации функции взаимной корреляции, можно получить $s_2(n)$ как оценку $s_1(n)$.

Оценка сигнала методом подбора шаблона иногда удобна, например, когда форма сигнала приблизительно известна (например, для определенных биомедицинских потенциалов), но есть и более научный подход, который может быть предпочтительнее. В

этом методе вначале оценивается период сигнала через автокорреляционную функцию зашумленного сигнала, а затем находится взаимная корреляция зашумленного сигнала с периодической серией импульсов, период которой равен периоду сигнала. Получающаяся в результате функция взаимной корреляции считается оценкой сигнала.

Обозначим сигнал с периодом N_p точек ($N_p < N$) через $s(n)$, а шум через $q(n)$, так что зашумленный сигнал равен $S(n) = s(n) + q(n)$. Пусть $\delta(n - kN_p)$ — периодическая серия импульсов, используемая для нахождения взаимной корреляции, а N_δ — число импульсов, используемых при нахождении взаимной корреляции. Последняя величина также равна числу периодов сигнала, за которые находится взаимная корреляция сигнала с серией импульсов. Тогда

$$r_{s\delta}(-j) = \frac{1}{N} \sum_{n=0}^{N-1} [s(n) + q(n)] \delta(b - kN_p - j), \quad k = 0, 1, 2, \dots \quad (5.33)$$

Для $j = 0$ и учитывая, что $\delta(n - kN_p) = 0$ для всех $n \neq kN_p$,

$$r_{s\delta}(0) = \frac{1}{N_\delta} [s(0) + q(0) + s(N_p) + q(N_p) + s(2N_p) + q(2N_p) + \dots + s(N) + q(N)] \quad (5.34)$$

Далее из-за периодичности сигнала $s(n + kN_p) = s(n)$ уравнение (5.34) приводится к виду

$$r_{s\delta}(0) = \frac{1}{N_\delta} [Ns(0) + q(0) + q(N_p) + q(2N_p) + \dots + q(N)]$$

или

$$r_{s\delta}(0) = s(0) + \frac{1}{N_\delta} \sum_{k=0}^{N/N_p} q(kN_p). \quad (5.35)$$

При $N \rightarrow \infty$, $(1/N_\delta) \sum_{k=0}^{N/N_p} q(kN_p) \rightarrow 0$, следовательно, $r_{s\delta}(0) \rightarrow s(0)$. Подобным образом, для других значений j

$$r_{s\delta}(-j) = \frac{1}{N_\delta} \sum_{n=0}^{N-1} [s(n) + q(n)] \delta[(n - j) - kN_p], \quad k = 0, 1, 2, \dots,$$

в результате также подавляется шум и находятся значения $s(n)$ для $n = 1, 2, \dots$. Следовательно, из уравнения (5.33) получаем

$$r_{s\delta}(-j) = s(0), s(1), \dots, s(N - 1), \quad j = 0, 1, 2, \dots,$$

что и является требуемым сигналом. Таким образом, сигнал, затерянный в зашумленном сигнале, можно оценить следующим образом:

- 1) найти автокорреляцию сигнала и определить его период;
- 2) найти взаимную корреляцию сигнала с периодической серией импульсов, период которой равен периоду сигнала; в ходе этой процедуры серия импульсов смещается вправо относительно сигнала.

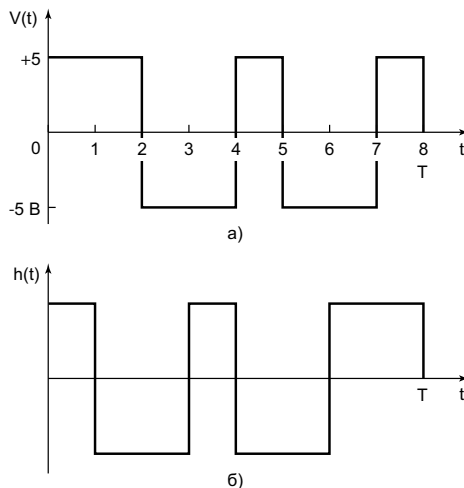


Рис. 5.12. Сигнал — 8-битовое кодовое слово PCM (панель *а*). Импульсная характеристика соответствующего согласованного фильтра (панель *б*)

5.2.2.3. Реализация корреляционного детектирования — согласованный фильтр

Корреляция также применяется для реализации корреляционного детектирования с помощью согласованного фильтра. *Согласованным* называется фильтр, максимизирующий отношение сигнал-шум на выходе. Импульсная характеристика согласованного фильтра записывается следующим образом¹ [13]:

$$h(t)cs_i(T-t), \quad (5.36)$$

где c — произвольная константа, $s_i(t)$ — входной сигнал (без шума), который выражается следующим образом:

$$s_i(t) = \begin{cases} s_i(t), & \text{для } 0 \leq t \leq T; \\ 0, & \text{для } T < t < 0. \end{cases}$$

Здесь T — момент времени, в который выполняется дискретизация выхода. Видно, что импульсная характеристика получена путем обращения сигнала во времени с последующим перемещением его на $T(s)$ вдоль временной оси. В качестве примера на рис. 5.12, *а* показан сигнал, являющийся 8-битовым словом в кодировке PCM, а на рис. 5.12, *б* показана импульсная характеристика согласованного фильтра, максимизирующего детектирование сигнала.

¹Формула (5.36) верна только для детерминированных сигналов, т.е. таких, о которых известно все, в том числе и момент времени T . Если сигнал имеет случайную компоненту (хотя бы фазовую, т.е. время его появления), то формула (5.36) принципиально неверна. В этом случае для построения согласованного фильтра следует воспользоваться теоремой Винера–Хинчина. — Прим. ред.

Теперь можно показать, что детектирование с помощью согласованного фильтра эквивалентно корреляции. Выход фильтра $y(t)$ вначале выражается через свертку его входа $s(t)$ с его импульсной характеристикой (свертка рассмотрена в разделе 5.3):

$$y(t) = \int_{-\infty}^{\infty} s(\tau)h(t - \tau)d\tau, \quad (5.37)$$

где

$$s(t) = s_1(t) + q(t). \quad (5.38)$$

Здесь τ — задержка, $q(t)$ обозначает шумовой компонент. Подставляя уравнение (5.38) в (5.37), получаем

$$\begin{aligned} y(t) &= \int_{-\infty}^{\infty} [s_1(\tau) + q(\tau)]h(t - \tau)d\tau = \\ &= \int_{-\infty}^{\infty} s_1(\tau)h(t - \tau)d\tau + \int_{-\infty}^{\infty} q(\tau)h(t - \tau)d\tau. \end{aligned}$$

Второй член правой части стремится к нулю, поскольку $q(\tau)$ — случайный сигнал, который не коррелирует с $h(t - \tau)$. Следовательно,

$$y(t) \approx \int_{-\infty}^{\infty} s_1(\tau)h(t - \tau)d\tau. \quad (5.39)$$

Далее из уравнения (5.36) получаем

$$h(t - \tau) = cs_1(T - t + \tau). \quad (5.40)$$

Объединяя уравнения (5.39) и (5.40), получаем

$$y(t) \approx \int_{-\infty}^{\infty} s_1(\tau)cs_1(T - t + \tau)d\tau. \quad (5.41)$$

Если этот выход дискретизуется в момент времени $t = T$, то

$$\begin{aligned} y(t) &\approx \int_{-\infty}^{\infty} s_1(\tau)cs_1(\tau)d\tau \approx \\ &\approx \int_{-\infty}^{\infty} s_1^2(\tau)d\tau = \int_{-\infty}^{\infty} s_1^2(t)dt = r_{11}(0), \end{aligned} \quad (5.42)$$

если $c = 1$.

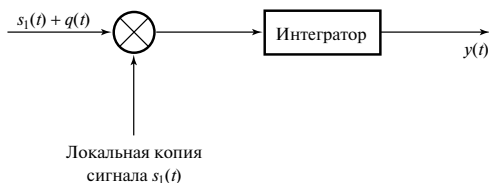


Рис. 5.13. Схематическое изображение корреляционного детектора

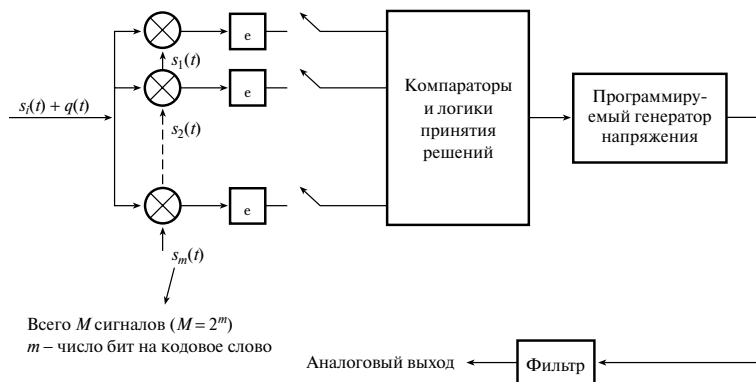


Рис. 5.14. Детектор кодовых слов PCM, основанный на корреляционном детекторе

Таким образом, $y(T)$ — автокорреляционная функция сигнала $s_1(t)$ с нулевой задержкой, которую можно получить, найдя взаимную корреляцию зашумленного входа с локально генерируемым бесшумным сигналом. На основе данного принципа и построен корреляционный детектор, схематическое изображение которого приведено на рис. 5.13. Например, на рис. 5.14 изображен детектор кодовых слов PCM, содержащий корреляционный детектор для каждого кодового слова.

В цифровом детекторе m -битовых кодовых слов хранятся кодовые слова, которые в процессе детектирования умножаются на m поступающих битов. Максимальное значение будет получено, если 1) m входных битов точно соответствуют m -битовому кодовому слову, или если 2) m входных битов случайно равны m -битовому кодовому слову. Второй вариант крайне нежелателен. Он возможен, если два соседних кодовых слова содержат последовательность битов, идентичную требуемому m -битовому кодовому слову, или если кодовое слово было повреждено. Следовательно, необходимо так упорядочить входной поток, чтобы была возможна синхронизация слов, а также битовая синхронизация в корреляционном приемнике.

Таким образом, нужна синхронизация кодовых слов, при которой выполняются следующие требования:

- 1) корреляция мала для моментов взятия выборок $t \neq T$;
- 2) корреляция велика для выборок, взятых в моменты $t = T$.

Кодовое слово с такими свойствами дает большую автокорреляцию при нулевой задержке и малую — при других задержках. Следовательно, детектирование больших

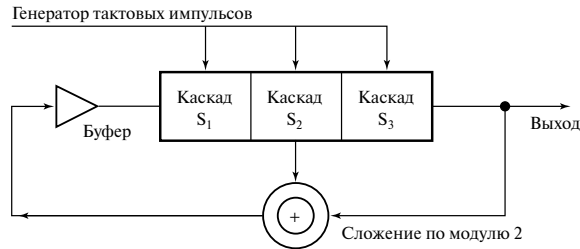


Рис. 5.15. Трехкаскадный генератор псевдослучайной последовательности

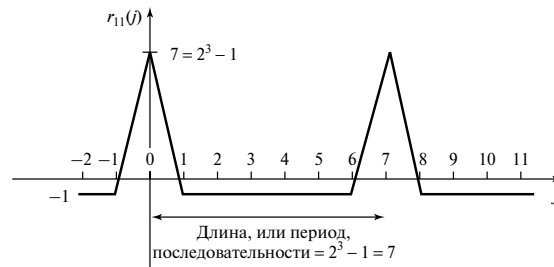


Рис. 5.16. Автокорреляционная функция биполярного сигнала трехкаскадного генератора псевдослучайной последовательности

значений функции взаимной корреляции в приемнике укажет расположение входных кодовых слов относительно записанного слова. Таким образом синхронизируется приемник. Далее отметим, что случайные сигналы имеют указанное выше автокорреляционное свойство, и их можно реализовать в цифровом приемнике с помощью псевдослучайной последовательности, которую легко генерировать, используя регистр сдвига с отводами. В качестве примера на рис. 5.15 показан трехкаскадный генератор псевдослучайной последовательности. Генератор производит последовательность 1, 1, 1, 0, 0, 1, 0, которая затем повторяется. Генерируемая автокорреляционная функция при биполярном сигнале на входе представлена на рис. 5.16.

Ниже указаны некоторые свойства псевдослучайных последовательностей.

1. Кодовые m -битовые слова порождают последовательность длины $2^m - 1$.
2. Максимальные значения равны $2^m - 1$.
3. Автокорреляционная функция равна -1 везде, кроме максимумов.
4. Выходная последовательность содержит 2^{m-1} единиц и $2^{m-1} - 1$ нулей.
5. Спектральная плотность мощности равномерна, так что описанные генераторы можно использовать как источники белого шума.

Последнее свойство означает, что псевдослучайные последовательности также применяются в качестве источников белого шума.

5.2.2.4. Определение импульсной характеристики электрических систем

Дальнейшее приложение корреляции и псевдослучайных последовательностей лежит в определении импульсной характеристики электрических систем. Например, рассмотрим системы импульсного тестирования. При наличии шума малые импульсы могут маскироваться шумом, а большие — вызывать переполнение системы. Кроме того, используя единственный импульс, сложно поддерживать равномерную спектральную плотность энергии по полосе. При этом, как объяснялось выше, псевдослучайная последовательность имеет равномерный энергетический спектр. Кроме того, если время измерения кратно длине последовательности, дисперсия измеренных значений будет равна нулю, в результате время измерения будет малым, а точность высокой.

Основная идея метода — подать псевдослучайную последовательность на вход системы. Далее через взаимную корреляцию поданной последовательности и выхода находится импульсная характеристика системы. Доказательство сказанного приводится ниже.

Пусть $q(t)$ — входная псевдослучайная последовательность, а $y(t)$ — выход системы с импульсной характеристикой $h(t)$. Тогда

$$r_{qy} = \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T q(t)y(t+\tau)dt = \quad (5.43)$$

$$= \lim_{t \rightarrow \infty} \frac{1}{T} q(t)dt \int_{-\infty}^{\infty} h(v)q(t-v+\tau)dv, \quad (5.44)$$

поскольку $y(t)$ — свертка входа с импульсной характеристикой:

$$y(t) = \int_{-\infty}^{\infty} h(v)q(t-v)dv. \quad (5.45)$$

Меняя в уравнении (5.44) порядок интегрирования, получаем

$$r_{qy}(\tau) = \int_{-\infty}^{\infty} h(v)dv \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T q(t)q(t-v+\tau)dt = \quad (5.46)$$

$$= \int_{-\infty}^{\infty} h(v)r_{qq}(\tau-v)dv. \quad (5.47)$$

Теперь $r_{qq}(\tau-v)$ аппроксимируется δ -функцией, поскольку это автокорреляционная функция псевдослучайной последовательности. Следовательно, уравнение (5.47) можно переписать в виде

$$r_{qy}(t) = K \int_{-\infty}^{\infty} h(v)\delta(\tau-v)dv = Kh(t), \quad (5.48)$$

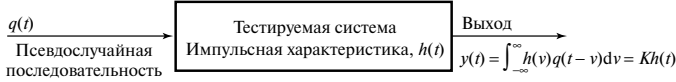


Рис. 5.17. Определение импульсной характеристики электрической системы

где K — площадь импульсной функции, равная среднеквадратическому значению шума [1]. Схема описываемой структуры изображена на рис. 5.17. Метод подвержен определенным ошибкам, и, чтобы их избежать, следует принимать определенные меры предосторожности.

5.2.2.5. Определение отношения сигнал-шум для периодического зашумленного сигнала

Измеряя коэффициент корреляции зашумленного периодического сигнала, можно определить его отношение сигнал-шум, а также мощность сигнала и шума [9]. Вывод соответствующих выражений приводится ниже. Для нахождения периода зашумленных сигналов можно использовать метод, описанный в подразделе 5.2.2.2.

Запишем сигнал как $V_s(i)$, шум — как $V_n(i)$, зашумленный сигнал $V(i)$ выражается следующим образом:

$$V(i) = V_s(i) + V_n(i). \quad (5.49)$$

В таком случае периодический сигнал с периодом n интервалов дискретизации запишется как

$$V(i) = V(i + n). \quad (5.50)$$

Ковариация (второй момент) $V(i)$ определяется следующим образом:

$$\text{cov}[V(i)] = \text{cov}[V(i), V(i + n)] = \frac{1}{N} \sum_{i=1}^N [\{V(i) - \bar{V}(i)\} \{V(i + n) - \bar{V}(i + n)\}], \quad (5.51)$$

где

$$\bar{V}(i) = \frac{1}{N} \sum_{i=1}^N V(i) -$$

среднее $V(i)$. Таким образом, видно, что $\text{cov}[V(i)]$ — автокорреляционная функция $V(i)$ с нулевым средним с задержкой n . Далее обычным образом определяется коэффициент автокорреляции $V(i)$:

$$\begin{aligned} \rho[V(i)] &= \frac{\text{cov}[V(i)]}{\sqrt{\left\{ \frac{1}{N} \sum_{i=1}^N (V(i) - \bar{V}(i))^2 \right\} \left\{ \frac{1}{N} \sum_{i=1}^N (V(i + n) - \bar{V}(i + n))^2 \right\}}} = \\ &= \frac{\text{cov}[V(i)]}{\sigma[V(i)]\sigma[V(i + n)]}, \end{aligned} \quad (5.52)$$

где

$$\sigma[V(i)] = \sqrt{\frac{1}{N} \sum_{i=1}^N (V(i) - \bar{V}(i))^2} \quad (5.53)$$

среднеквадратическое отклонение $V(i)$.

Отметим, что нормировка на коэффициент N в уравнениях (5.51)–(5.53) дает смещенные оценки. Несмещенные оценки получаются путем замены N на $N - 1$. Отметим также, что такие члены, как

$$\sigma^2[V_x(i)] = \frac{1}{N} \sum_{i=1}^N (\bar{V}_x(i))^2 = \text{var}[V_x(i)]$$

называются *дисперсией* и представляют мощность, соотнесенную с $V_x(i)$.

Учтем теперь в уравнении (5.51) тот факт, что корреляции между зашумленными сигналами стремятся к нулю при увеличении числа выборок. Таким образом,

$$\begin{aligned} \text{cov}[V(i), V(i+n)] &= \text{cov}[(V_s(i) + V_n(i)), (V_s(i+n) + V_n(i+n))] = \\ &= \text{cov}[V_s(i), V_s(i+n)] = \text{var}[V_s(i)]. \end{aligned} \quad (5.54)$$

Для достаточно длинных последовательностей данных $\sigma[V_n(i)] = \sigma[V - n(i+n)]$, поэтому

$$\begin{aligned} \sigma[V(i)]\sigma[V(i+n)] &= \sigma^2[V(i)] = \text{var}[V(i)] = \text{var}[V_s(i) + V_n(i)] = \\ &= \text{var}[V_s(i)] + \text{var}[V_n(i)] + 2\text{cov}[V_s(i), V_n(i)] = \text{var}[V_s(i)] + \text{var}[V_n(i)], \end{aligned} \quad (5.55)$$

поскольку $\text{cov}[V_s(i), V_n(i)] = 0$.

Чтобы получить выражение для коэффициента автокорреляции, подставим уравнения (5.54) и (5.55) в (5.52):

$$\rho[V(i)] = \frac{\text{var}[V_s(i)]}{\text{var}[V_s(i)] + \text{var}[V_n(i)]} = \frac{1}{1 + \frac{\text{var}[V_n(i)]}{\text{var}[V_s(i)]}}. \quad (5.56)$$

Отношение сигнал-шум S/N (дБ) определяется как $10 \lg \{(\text{мощность сигнала})/(\text{мощность шума})\}$, что в принятых обозначениях выглядит так:

$$\frac{S}{N}(\text{дБ}) = 10 \lg \left| \frac{\text{var}[V_s(i)]}{\text{var}[V_n(i)]} \right| \text{дБ}. \quad (5.57)$$

Объединяя уравнения (5.56) и (5.57), получаем

$$\frac{S}{N}(\text{дБ}) = 10 \lg \left| \frac{\rho[V(i)]}{1 - \rho[V(i)]} \right| \text{дБ}. \quad (5.58)$$

Таким образом, отношение сигнал-шум зашумленного периодического сигнала легко вычисляется по его коэффициенту автокорреляции.

Из уравнения (5.55) получаем

$$\text{var}[V_s(i)] + \text{var}[V_n(i)] \equiv S + N = \text{var}[V(i)], \quad (5.59)$$

где S и N представляют мощность сигнала и шума соответственно. Используя уравнения (5.56) и (5.59), можно получить выражения для S и N :

$$S = \rho[V(i)]\text{var}[V(i)] \quad (5.60)$$

и

$$N = (1 - \rho[V(i)])\text{var}[V(i)]. \quad (5.61)$$

Отметим, что уравнение (5.58) применяется для оценки производительности каналов магнитной записи с точки зрения их отношений сигнал-шум [9].

5.2.3. Быстрая корреляция

Расчет корреляции можно ускорить, используя теорему о корреляции, которая обычно формулируется следующим образом:

$$r_{12}(j) = F_D^{-1}[X_1^*(k)X_2(k)], \quad (5.62)$$

хотя корректной является такая формулировка:

$$r_{12}(j) = \frac{1}{N}F_D^{-1}[X_1^*(k)X_2(k)], \quad (5.63)$$

где F_D^{-1} обозначает обратное дискретное преобразование Фурье. Данный подход требует выполнения двух дискретных преобразований Фурье (ДПФ) и одного обратного ДПФ, что легче всего сделать, используя алгоритм БПФ (см. главу 3). Если число членов в последовательностях достаточно велико, данный метод БПФ дает результат быстрее, чем непосредственный расчет взаимной корреляции.

Доказательство теоремы о корреляции

Пусть $x_1(l)$, $x_2(r)$ и $x_3(n)$ — периодические последовательности длины N , и пусть их ДПФ-образы равны соответственно $X_1(k)$, $X_2(k)$ и $X_3(k)$. Более того, пусть

$$X_3(k) = X_1^*(k)X_2(k). \quad (5.64)$$

В этом случае

$$X_1^*(k) = \sum_{l=0}^{N-1} x_1(l)e^{(2\pi i/N)lk} \quad (5.65)$$

и

$$X_2^*(k) = \sum_{r=0}^{N-1} x_2(r)e^{(2\pi i/N)(-rk)}. \quad (5.66)$$

Подставляя уравнения (5.65) и (5.66) в (5.64), получаем

$$X_3(k) = \sum_{l=0}^{N-1} x_1(l)e^{(2\pi i/N)lk} \sum_{r=0}^{N-1} x_2(r)e^{(2\pi i/N)(-rk)} = \quad (5.67)$$

$$= \sum_{l=0}^{N-1} \sum_{r=0}^{N-1} x_1(l) x_2(r) e^{(2\pi i/N)(lk-rk)}. \quad (5.68)$$

Далее

$$x_2(n) = \frac{1}{N} \sum_{k=0}^{N-1} X_3(k) e^{(2\pi i/N)nk}. \quad (5.69)$$

Таким образом, подставляя уравнение (5.68) в (5.69), получаем

$$\begin{aligned} x_3(n) &= \frac{1}{N} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} \sum_{r=0}^{N-1} x_1(l) x_2(r) e^{(2\pi i/N)(lk-rk+nk)} = \\ &= \frac{1}{N} \sum_{l=0}^{N-1} x_1(l) \sum_{r=0}^{N-1} x_2(r) \left[\sum_{k=0}^{N-1} e^{(2\pi i/N)(l-r+n)l} \right]. \end{aligned} \quad (5.70)$$

При $r = n + l$ выражение в квадратных скобках равно N . При $r \neq n + l$ его можно рассматривать как геометрическую прогрессию вида

$$\sum a x^n,$$

сумма которой по N членам равна

$$\frac{a(1-x^N)}{1-x}.$$

В нашем случае сумма равна:

$$\frac{1[1 - e^{(2\pi i/N)(l-r+n)N}]}{1 - e^{(2\pi i/N)(l-r+n)}}. \quad (5.71)$$

Показатель экспоненты в числителе всегда кратен 2π , так что ее значение равно 1. Таким образом, сумма равна нулю при $r \neq n + l$. Следовательно, уравнение (5.70) можно переписать как

$$x_3(n) = \frac{1}{N} \sum_{l=0}^{N-1} x_1(l) \sum_{r=0}^{N-1} x_2(r) N \delta(l - r + n), \quad (5.72)$$

где $\delta(l - r + n) = 1$ при $r = n + l$ и $\delta(l - r + n) = 0$ при $r \neq n + l$. Положив $r = n + l$ и упростив, получаем:

$$x_3(n) = \sum_{l=0}^{N-1} x_1(l) x_2(l + n) \quad (5.73)$$

или

$$\frac{1}{N} x_3(n) = \frac{1}{N} \sum_{l=0}^{N-1} x_1(l) x_2(l + n). \quad (5.74)$$

Правая часть данного выражения эквивалентна взаимной корреляции $x(n)$ и $x_2(n)$, и, как легко видеть, она равна $(1/N)x_3(n)$. Из уравнения (5.69) получаем

$$x_3(n) = F_D^{-1}[X_3(k)]. \quad (5.75)$$

Далее, объединяя уравнения (5.74), (5.75) и (5.63), получаем

$$\frac{1}{N}F_D^{-1}[X_3(k)] = r_{12}(n) = \frac{1}{N}F_D^{-1}[X_1^*(k)X_2(k)]. \quad (5.76)$$

Наконец, заменяя n на j , получаем

$$r_{12}(j) = \frac{1}{N}F_D^{-1}[X_1^*(k)X_2(k)]. \quad (5.77)$$

Пример 5.6

Используя теорему о корреляции, найдем взаимную корреляцию двух последовательностей $x_1(n)$ и $x_2(n)$.

$$\begin{aligned} x_1(n) &= \{1; 0; 0; 1\}, \\ x_2(n) &= \{0, 5; 1; 1; 0, 5\}. \end{aligned}$$

Используем вначале теорему о корреляции (уравнение (5.77)). В разделе 3.5 было показано, что $X_1(k)$ равно

$$X_1(k) = 2; 1 + i; 0; 1 - i,$$

так что

$$X_1^*(k) = 2; 1 - i; 0; 1 + i.$$

Для получения $X_2(k)$ проще всего использовать алгоритм БПФ, приведенный в разделе 3.5. Таким образом, при $x_0 = 0, 5$, $x_2 = 1$, $x_1 = 1$ и $x_3 = 0, 5$ получаем

$$\begin{aligned} X_{21}(0) &= x_0 + x_2 = 1, 5, \\ X_{21}(1) &= x_0 - x_2 = -0, 5, \\ X_{22}(0) &= x_1 + x_3 = 1, 5, \\ X_{22}(1) &= x_1 - x_3 = 0, 5, \\ X_{11}(0) &= X_{21}(0) + X_{22}(0) = 3, \\ X_{11}(1) &= X_{21}(1) + (-i)X_{22}(1) = -0, 5 - 0, 5i, \\ X_{11}(2) &= X_{21}(0) - X_{22}(0) = 0, \\ X_{11}(3) &= X_{21}(1) - (-i)X_{22}(1) = -0, 5 + 0, 5i. \end{aligned}$$

Собирая значения БПФ, получаем

$$\begin{aligned} X_1^*(k) &= 2; 1 - i; 0; 1 + i, \\ X_2(k) &= 3; -0, 5 - 0, 5i; 0; -0, 5 + 0, 5i. \end{aligned}$$

Так что

$$\begin{aligned} X_1^*(0)X_2(0) &= 2 \times 3 = 6, \\ X_1^*(1)X_2(1) &= (1-i)(-0,5-0,5i) = -1, \\ X_1^*(2)X_2(2) &= 0 \times 0 = 0, \\ X_1^*(3)X_2(3) &= 0,5(1+i)(-1+i) = -1. \end{aligned}$$

Следовательно,

$$[X_1^*(k)X_2(k)] = 6; -1; 0; -1.$$

Теперь необходимо к этому результату применить обратное ДПФ. Как объясняется в разделе 3.6, обратное ДПФ получается путем замены знаков экспонент (в весовых коэффициентах W_N) приведенного выше алгоритма БПФ и деления результата на N . Следовательно, получаем такой результат:

$$\begin{aligned} X_{21}(0) &= x_0 + x_2 = 6, \\ X_{21}(1) &= x_0 - x_2 = 6, \\ X_{22}(0) &= x_1 + x_3 = -2, \\ X_{22}(1) &= x_1 - x_3 = 0, \\ X_{11}(0) &= X_{21}(0) + X_{22}(0) = 4, \\ X_{11}(1) &= X_{21}(1) + iX_{22}(1) = 6, \\ X_{11}(2) &= X_{21}(0) - X_{22}(0) = 8, \\ X_{11}(3) &= X_{21}(1) - iX_{22}(1) = 6. \end{aligned}$$

Компоненты $F_D^{-1}[X_1^*(k)X_2(k)]$ получаются делением значений $X_{11}(0)$, $X_{11}(1)$, $X_{11}(2)$ и $X_{11}(3)$ на $N = 4$. Таким образом,

$$F_D^{-1}[X_1^*(k)X_2(k)] = 1; 1,5; 2; 1,5.$$

Далее из уравнения (5.77) определяем

$$r_{12}(i) = \frac{1}{4}F_D^{-1}[X_1^*X_2(k)] = \{0,25; 0,375; 0,5; 0,375\}. \quad (5.78)$$

Эта корреляция будет круговой, поскольку все данные периодичны с периодом N . Если непосредственно посчитать взаимную корреляцию $r_{12}(i)$, то получатся такие значения:

$$\begin{aligned} r_{12}(0) &= (1 \times 0,5 + 0 + 0 + 1 \times 0,5)/4 = 0,25, \\ r_{12}(1) &= (1 \times 1 + 0 + 0 + 1 \times 0,5)/4 = 0,375, \\ r_{12}(2) &= (1 \times 1 + 0 + 0 + 1 \times 1)/4 = 0,5, \\ r_{12}(3) &= (1 \times 0,5 + 0 + 0 + 1 \times 1)/4 = 0,375. \end{aligned}$$

Следующее значение $r_{12}(4)$ снова равно 0,25, и последовательность периодически повторяется. Данный результат согласуется с данными, полученными выше с помощью теоремы о корреляции. Как объясняется в разделе 5.2.1, теорему о корреляции можно использовать для получения линейной корреляции путем добавления к двум последовательностям дополняющих нулей. Следовательно, если последовательность $x_1(n)$ имеет длину N_1 , а последовательность $x_2(n)$ — N_2 , то $x_1(n)$ дополняется $(N_2 - 1)$ нулями, а $x_2(n)$ — $(N_1 - 1)$ нулями. Далее на основе двух расширенных последовательностей рассчитывается взаимная корреляция. Этот метод вычисления взаимной корреляции с помощью теоремы о корреляции и БПФ называется *быстрой корреляцией*.

Расчет взаимной корреляции также можно ускорить, реализовав его рекурсивно. Проиллюстрируем это на примере с нулевой задержкой. Взаимная корреляция при нулевой задержке двух дискретных сигналов $x_1(n)$ и $x_2(n)$ равна

$$r_{12}(0) = \frac{1}{N} \sum_{n=0}^{N-1} x_1(n)x_2(n). \quad (5.79)$$

Эта операция включает вычисление N произведений, $N - 1$ суммы и одного деления и может требовать чрезмерного времени при реализации в реальном времени, когда пары новых данных поступают с частотой дискретизации. Расчет необходимо повторять при поступлении следующей пары данных. Новые вычисления будут отличаться от предыдущих только тем, что к сумме пар произведений будет прибавлено произведение новой пары и первое произведение вычтется. Итак, для каждой взаимной корреляции

$$\begin{aligned} (\text{новое значение}) = & (\text{предыдущее значение}) + \\ & + \frac{1}{N} (\text{произведение двух новых данных}) - \\ & - \frac{1}{N} (\text{произведение первых двух данных}). \end{aligned} \quad (5.80)$$

Это выражение — основа рекурсивного алгоритма. Каждая взаимная корреляция теперь требует только одного умножения, одного вычитания, одного сложения и одного деления при условии, что в памяти хранятся произведения пар данных. Для N -точечной корреляции рекурсивный подход дает правильные значения после вычисления первых $(N - 1)$ точек.

Во многих приложениях требуется, чтобы среднее данных было равно нулю, например, для устранения постоянного тока из электрических сигналов. При этом необходимо рассчитать среднее значение сигнала и вычесть его из всех дискретных значений. Это означает, что расчет среднего значения также можно провести рекурсивно, поскольку для каждой новой пары данных

$$\begin{aligned} \text{новое среднее} = & \text{предыдущее среднее} + \\ & + \frac{1}{N} (\text{новый элемент данных} - \text{первый элемент данных}). \end{aligned} \quad (5.81)$$

Кроме того, вычитание среднего уровня и расчет взаимной корреляции можно объединить в один рекурсивный алгоритм. Рассмотрим величины

$$\bar{x}_1(k) = \frac{1}{N} \sum_{n=0}^{N-1} x_1(n) \quad (5.82)$$

и

$$\bar{x}_2(k) = \frac{1}{N} \sum_{n=0}^{N-1} x_2(n). \quad (5.83)$$

Значение функции взаимной корреляции k -го набора из N точек равно

$$r_{12}(k) = \frac{1}{N} \sum_{n=0}^{N-1} x_1(n)x_2(n). \quad (5.84)$$

При удалении среднего значение функции взаимной корреляции становится равным $r_{12}^0(k)$, где

$$r_{12}^0(k) = \frac{1}{N} \sum_{n=0}^{N-1} [x_1(n) - \bar{x}_1(k)][x_2(n) - \bar{x}_2(k)], \quad (5.85)$$

что можно упростить до

$$r_{12}^0(k) = r_{12}(k) - \bar{x}_1(k)\bar{x}_2(k). \quad (5.86)$$

Объединяя уравнения (5.80) и (5.83), получаем

$$r_{12}(k) = r_{12}(k-1) + \frac{1}{N} [x_1(k)x_2(k) - x_1(k-N)x_2(k-N)]. \quad (5.87)$$

Из уравнения (5.81)

$$\bar{x}_1(k) = \bar{x}_1(k-1) + \frac{1}{N} [x_1(k) - x_1(k-N)] \quad (5.88)$$

и

$$\bar{x}_2(k) = \bar{x}_2(k-1) + \frac{1}{N} [x_2(k) - x_2(k-N)]. \quad (5.89)$$

Уравнения (5.86)–(5.89) составляют рекурсивный алгоритм, который объединяет вычитание среднего из данных с вычислением взаимной корреляции. Каждое вычисление требует только трех умножений, четырех вычитаний, трех сложений и четырех делений. Специалисты-практики отмечают, что при изменении средних значений данных следует аккуратно выбирать N , в противном случае можно получить неточные результаты.

5.3. Описание свертки

Свертка, помимо прочего, описывает, как выход системы определяется взаимодействием входа с самой системой. Обычно выход системы является запаздывающей и подавленной или усиленной версией входа. В этой связи особенно полезно рассмотреть

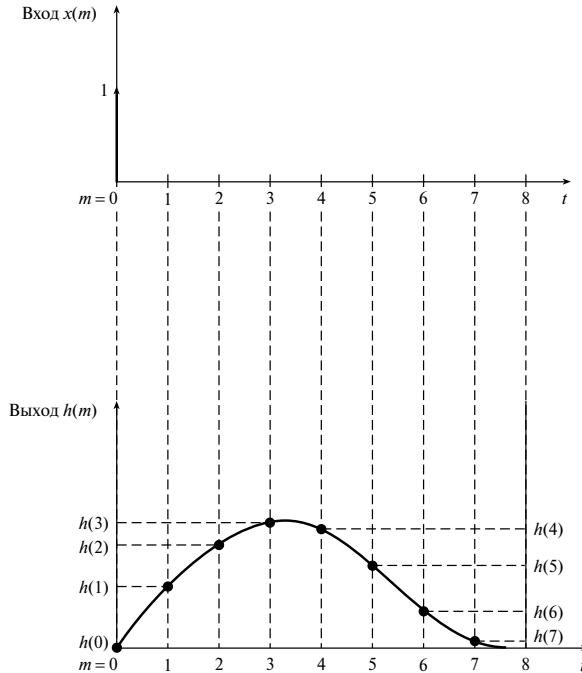


Рис. 5.18. Входной импульс и соответствующая импульсная характеристика системы

выход системы, порожденный импульсным входом. Это объясняется тем, что любой вход можно представить как последовательность импульсов разной мощности. Итак, выход системы, инициированный импульсным входом, не будет соответствующим импульсом, а будет меняться со временем, в некоторый момент достигая максимального значения, как показано на рис. 5.18. Из данного рисунка видно, что в момент взятия выборки m выход, порожденный единичным импульсом, поданным в момент 0, равен $h(m)$. Данная величина называется *импульсной характеристикой системы* или ее *импульсным откликом*.

Рассмотрим подачу на вход системы последовательности импульсов $x(m)$ в моменты времени m . В контексте рис. 5.19 выход в момент времени 0 равен $y(0)$, причем

$$y(0) = h(0)x(0).$$

В дискретный момент времени $m = 1$ выход равен $h(0)x(1)$ (влияние текущего входа $x(1)$) плюс $h(1)x(0)$ (запаздывающее влияние входа, поданного в момент $m = 0$). Следовательно,

$$y(1) = h(1)x(0) + h(0)x(1).$$

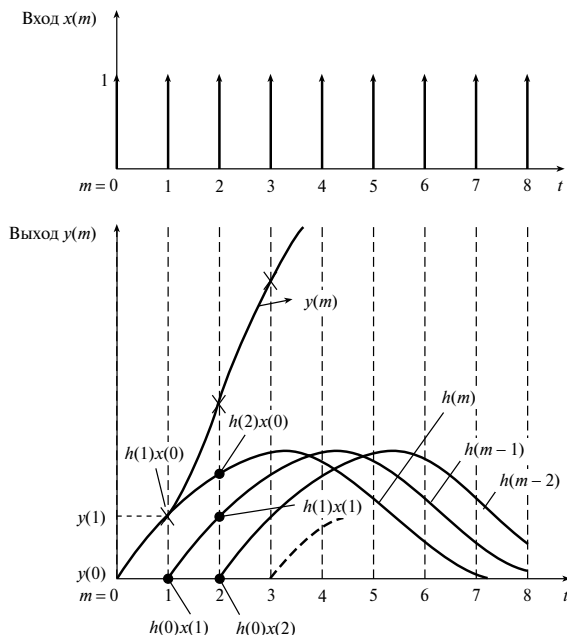


Рис. 5.19. Поданная на вход импульсная последовательность и характеристика системы, полученная из отдельных импульсных откликов

Таким образом, последующие выходы запишутся так:

$$\begin{aligned}
 y(2) &= h(2)x(0) + h(1)x(1) + h(0)x(2) \\
 y(3) &= h(3)x(0) + h(2)x(1) + h(1)x(2) + h(0)x(3) \\
 &\vdots \\
 y(n) &= h(n)x(0) + h(n-1)x(1) + \dots + h(0)x(n).
 \end{aligned} \tag{5.90}$$

Если система линейна, выход можно записать как линейную сумму влияния предыдущих входов. Выход линейной системы первого порядка описывается уравнением (5.90).

Изучая приведенные выражения, находим, что выход получается умножением входной последовательности на соответствующие точки обращенной во времени функции импульсной характеристики. Альтернативный вариант — записать уравнение (5.90) в таком виде:

$$y(n) = h(0)x(n) + h(1)x(n-1) + \dots + h(n)x(0) \tag{5.91}$$

и рассматривать выход как произведение соответствующих пар точек в функции импульсной характеристики и обращенной во времени входной последовательности. Следовательно, сверточная сумма эквивалентна взаимной корреляции одной последовательности и обращенной во времени другой.

Уравнения (5.90) и (5.91) можно записать компактно:

$$y(n) = \sum_{m=0}^n h(n-m)x(m) \quad (5.92)$$

и

$$y(n) = \sum_{m=0}^n h(m)x(n-m). \quad (5.93)$$

Данные функции называются *сверточными суммами входов* с импульсной характеристикой, а выход находится как *свертка входа с импульсной характеристикой системы*.

Уравнения (5.92) и (5.93) можно расширить на сигналы бесконечной длительности, записав их следующим образом:

$$y(n) = \sum_{m=-\infty}^{\infty} x(m)h(n-m) = x(n) \circledast h(n) \quad (5.94)$$

и

$$y(n) = \sum_{m=-\infty}^{\infty} h(m)x(n-m) = h(n) \circledast x(n). \quad (5.95)$$

В приведенных уравнениях символ “ $\circledast$ ” обозначает операцию свертки.

Если вход состоит из непрерывной последовательности импульсов, приведенную выше сумму можно заменить интегралом, например, уравнение (5.94) при этом приводится к виду

$$y(t) = \int_{-\infty}^{\infty} x(\lambda)h(t-\lambda)d\lambda, \quad (5.96)$$

что называется *интегралом свертки*.

До этого момента термином *свертка* назывался результат свертки импульсной характеристики системы со входом системы. В то же время, данную концепцию можно расширить на свертку любых двух наборов данных и рассматривать термин в более широком смысле.

В качестве примера свернем две периодические во времени последовательности $\{4, 3, 2, 1\}$ ($h(m)$) и $\{1, 2, 3, 4\}$ ($x(m)$). На рис. 5.20, *а* показана периодическая последовательность $\{4, 3, 2, 1\}$ ($h(m)$), а на рис. 5.20, *б* — обращенная во времени последовательность ($h(-m)$), $\{1, 2, 3, 4\}$. (Напомним, что сверточная сумма требует, чтобы одна из последовательностей умножалась на обращенную во времени вторую последовательность, т.е. свертка соответствует взаимной корреляции одной последовательности с обращенной во времени второй.) На рисунке также показано окно, ширина которого равна одному периоду, по которому выполняется свертка. Очевидно, что полученный результат будет периодическим, как и при нахождении взаимной корреляции (раздел 5.2.1), так что необходимо найти только свертку по выделенному интервалу. На рис. 5.20, *в* для справки показана вторая последовательность $\{1, 2, 3, 4\}$ ($x(m)$).

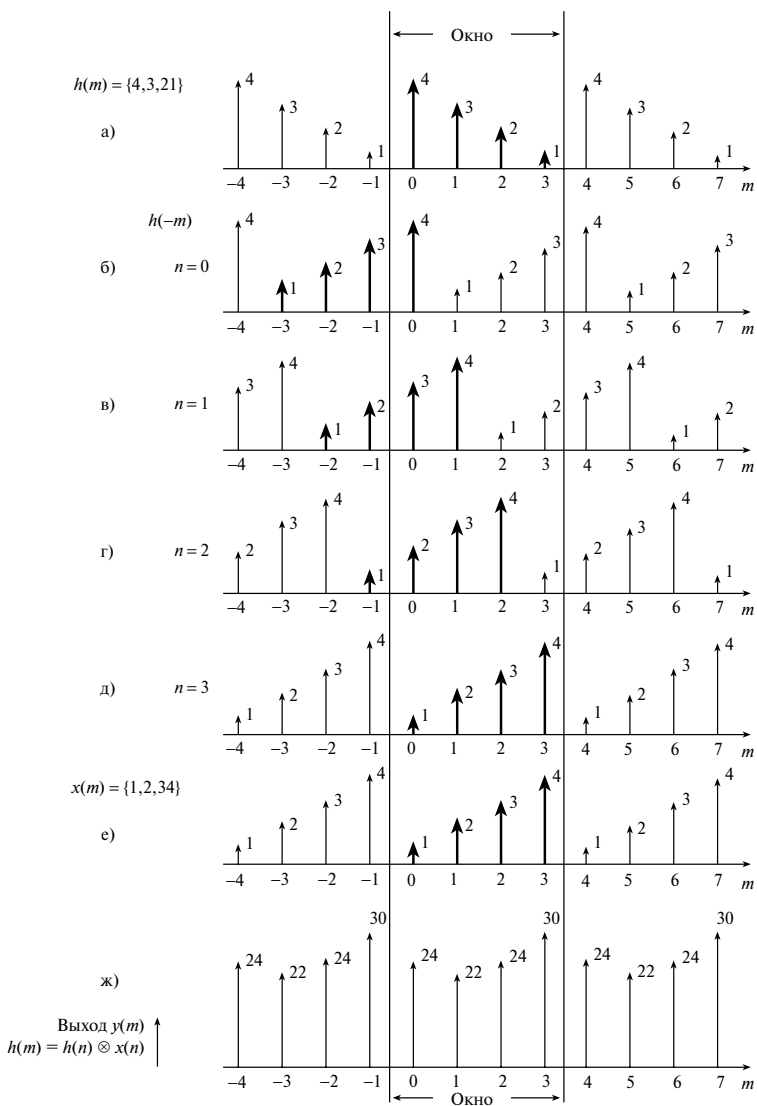


Рис. 5.20. Свертка $y(m)$ последовательностей $h(n)$ и $x(n)$: а) периодическая последовательность $h(m)$; б) обращенная во времени последовательность $h(-m)$; в)–д) копии $h(-m)$, последовательно смещаемые вправо; е) последовательность $x(m)$; ж) выходная последовательность $y(m) = h(n) \otimes x(n)$

При $n = 0$ уравнение (5.92) переходит в следующее:

$$y(0) = \sum_{m=0}^n h(-m)x(m).$$

Согласно этой формуле взаимная корреляция данных в окне на рис. 5.20, *б* и *е* равна

$$y(0) = 4 \times 1 + 1 \times 2 + 2 \times 3 + 3 \times 4 = 24.$$

При $n = 1$ уравнение (5.92) переходит в следующее:

$$y(1) = \sum_{m=0}^n h(1-m)x(m),$$

и согласно этой формуле взаимная корреляция данных в окне на рис. 5.20, *в* и *е* равна

$$y(1) = 3 \times 1 + 4 \times 2 + 1 \times 3 + 2 \times 4 = 22.$$

Следовательно,

$$y(2) = 2 \times 1 + 3 \times 2 + 4 \times 3 + 1 \times 4 = 24$$

и

$$y(3) = 1 \times 1 + 2 \times 2 + 3 \times 3 + 4 \times 4 = 30.$$

Далее выходная последовательность (см. рис. 5.20, *ж*) циклично повторяется.

Если сигналы поддаются строгому математическому описанию, свертку можно выполнить аналитически. Чтобы лучше понять процесс свертки, рассмотрим подобный пример и проиллюстрируем необходимые этапы графически.

Пример 5.7

Найдем аналитически свертку сигналов $x(t)$ и $h(t)$, изображенных на рис. 5.21, *а*.

Пусть интеграл свертки записывается как

$$y(t) = x(t) \otimes h(t) = \int_{-\infty}^{\infty} x(\tau)h(t - \tau)d\tau. \quad (5.97)$$

Уравнение (5.97) соответствует уравнению (5.96), в котором переменную λ заменили на τ , чтобы указать, что теперь рассматривается задача с задержками. Интеграл свертки зависит от переменной τ , так что рис. 5.21, *а* нужно заменить на рис. 5.21, *б*.

Далее необходимо обратить во времени сигнал $h(\tau)$, как показано на рис. 5.21, *в*. Далее $h(-\tau)$ смещается относительно $x(\tau)$ в положительном направлении τ . Получающийся в результате сигнал $h(t - \tau)$ накладывается на $x(\tau)$ за пять отдельных этапов, проиллюстрированных на рис. 5.21, *з-з*. Для каждого из этих этапов существует соответствующий интеграл свертки. Следовательно, $x(t) \otimes h(t)$ существует в пяти отдельных непрерывных областях.

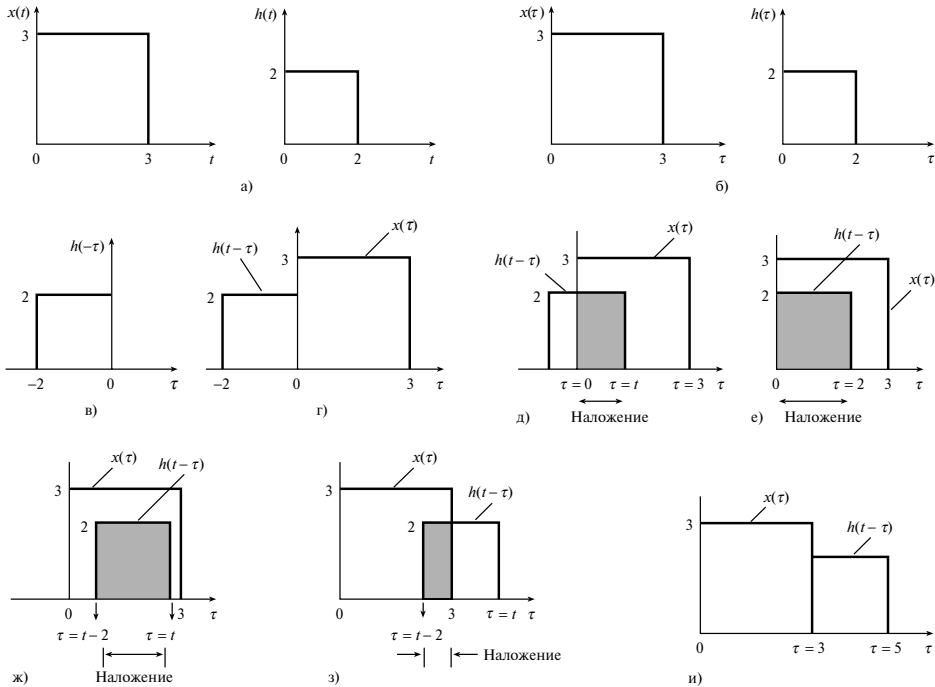


Рис. 5.21. Аналитическая свертка: а) сигналы $x(t)$ и $h(t)$; б) $x(\tau)$ и $h(\tau)$; в) $h(-\tau)$; г) $h(t-\tau)$ и $x(\tau)$; $t < 0$, сигнал $h(t-\tau)$ не накладывается на $x(\tau)$; д) $h(t-\tau)$ и $x(\tau)$; $0 < t \leq 2$, первое частичное наложение $h(t-\tau)$ и $x(\tau)$; е) $h(t-\tau)$ и $x(\tau)$; $t = 2$, конец первого частичного наложения; ж) $h(t-\tau)$ и $x(\tau)$; $2 \leq t \leq 3$, полное наложение $h(t-\tau)$ и $x(\tau)$; з) $h(t-\tau)$ и $x(\tau)$; $3 \leq t \leq 5$, второе частичное наложение $h(t-\tau)$ и $x(\tau)$; и) $h(t-\tau)$ и $x(\tau)$; $t > 5$, наложения $h(t-\tau)$ и $x(\tau)$ нет

- **Этап 1.** $t < 0$ и $h(t-\tau)$ не накладывается на $x(\tau)$ (рис. 5.21, з). Поскольку функции не перекрываются, $x(\tau)h(t-\tau) = 0$ для всех t и не дает вклада в интеграл свертки.
- **Этап 2.** $0 < t \leq 2$ и существует частичное наложение $h(t-\tau)$ и $x(\tau)$ (рис. 5.21, д). В этой области

$$y(t) = \int_{\tau=0}^{\tau=t} x(\tau)h(t-\tau)d\tau = \int_{\tau=0}^{\tau=t} (3) \times (2)d\tau \quad (5.98)$$

$$y(t) = 6[\tau]_0^t = 6t, \quad 0 < t \leq 2$$

Как показано на рис. 5.21, е, данный этап завершается в момент $t = 2$.

- **Этап 3.** $2 \leq t \leq 3$ и сигналы $h(t-\tau)$ и $x(\tau)$ полностью перекрываются (рис. 5.21, ж). В этой области

$$y(t) = \int_{\tau=t-2}^{\tau=t} (3) \times (2)d\tau = 6[\tau]_{t-2}^t \quad (5.99)$$

$$y(t) = 6(t - t + 2) = 12, \quad 2 \leq t \leq 3.$$

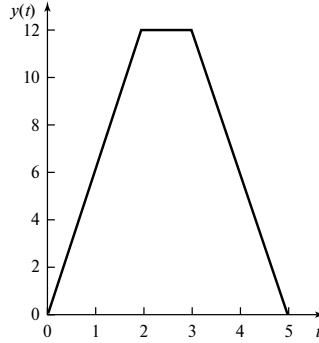


Рис. 5.22. Зависимость свертки $y(t) = x(t) \otimes h(t)$ от t

- *Этап 4.* $3 \leq t \leq 5$. Еще одна область частичного наложения, показанная на рис. 5.21, з:

$$y(t) = \int_{\tau=t-2}^{\tau=3} (3) \times (2) d\tau = 6[\tau]_{t-2}^3 = 6(5-t)3 - 6t. \quad (5.100)$$

- *Этап 5.* $t > 5$. Как показано на рис. 5.21, и, это вторая область без наложения, которая также не дает вклада в интеграл свертки.

Таким образом, вклад в интеграл свертки дают этапы 2–4, причем эти вклады выражаются следующим образом:

$$\begin{aligned} 0 < t \leq 2 \quad y(t) &= 6t, \\ 2 \leq t \leq 3 \quad y(t) &= 12, \\ 3 \leq t \leq 5 \quad y(t) &= 30 - 6t. \end{aligned}$$

Из этих выражений можно получить зависимость $y(t)$, изображенную на рис. 5.22.

Приведем повторно уравнения (5.94) и (5.96) и продолжим наше обсуждение.

$$y(n) = \sum_{m=-\infty}^{\infty} x(m)h(n-m) = x(n) \otimes h(n)$$

и

$$y(t) = \int_{-\infty}^{\infty} x(\lambda)h(t-\lambda)d\lambda.$$

Изучая эти уравнения, следует помнить, что свертка выполняется во времени — возникает так называемая *свертка во временной области*. Известно, что в частотной области выход системы на частоте f равен $Y(f)$:

$$Y(f) = H(f)X(f), \quad (5.101)$$

где $H(f)$ — частотная характеристика системы на частоте f , $X(f)$ — Фурье-образ входа $x(t)$. Кроме того, можно показать, что $H(f)$ — Фурье-образ $h(t)$. Применяя обратное преобразование Фурье к обеим частям уравнения (5.101), получаем

$$F^{-1}[Y(f)] = y(t) = F^{-1}[H(f)X(f)]. \quad (5.102)$$

Объединяя уравнения (5.96) и (5.102), получаем, что

$$y(t) = \int_{-\infty}^{\infty} x(\lambda)h(t - \lambda)d\lambda = x(t) \circledast h(t) = F^{-1}[H(f)X(f)]. \quad (5.103)$$

Таким образом, видно, что свертка двух сигналов во временной области эквивалентна применению обратного преобразования Фурье к произведению Фурье-образов двух сигналов. Данный полезный факт часто формулируют в сокращенной форме: *свертка во временной области эквивалентна умножению в частотной*.

Существует соотношение, дуальное приведенному, т.е. свертка в частотной области эквивалентна умножению во временной. Таким образом, можно показать [10], что

$$\begin{aligned} Y(\omega) &= \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\omega - u)H(u)du = X(f) \circledast H(f) = \\ &= F[y(t)] = F[x(t)h(t)]. \end{aligned} \quad (5.104)$$

Следовательно, Фурье-образ произведения двух временных последовательностей соответствует свертке Фурье-образов двух последовательностей. Данный результат обычно используется, чтобы объяснить взвешивание данных перед спектральным анализом (см. главу 11). В данной процедуре оцифрованная последовательность данных точка за точкой умножается на другую последовательность, которая состоит из дискретных значений весовой функции. Этот процесс называется *взвешиванием* и выполняется для уменьшения ошибок при вычислении энергетического спектра данных. После этого к взвешенным данным применяется дискретное преобразование Фурье, по результату которого вычисляется энергетический спектр. Задачей является получение энергетического спектра последовательности данных, но из сказанного выше следует, что в действительности получается спектр последовательности данных, свернутый со спектром весовой последовательности.

5.3.1. Свойства свертки

1. Закон коммутативности

$$x_1(t) \circledast x_2(t) = x_2(t) \circledast x_1(t). \quad (5.105)$$

Отметим, что данное выражение идентично следующему:

$$\int_{-\infty}^{\infty} x_1(\tau)x_2(t - \tau)d\tau = \int_{-\infty}^{\infty} x_2(\tau)x_1(t - \tau)d\tau.$$

2. Закон дистрибутивности

$$x_1(t) \circledast [x_2(t) + x_3(t)] = x_1(t) \circledast x_2(t) + x_1(t) \circledast x_3(t). \quad (5.106)$$

3. Закон ассоциативности

$$x_1(t) \circledast [x_2(t) \circledast x_3(t)] = [x_1(t) \circledast x_2(t)] \circledast x_3(t) \quad (5.107)$$

Данные свойства можно доказать, либо расписав соответствующие интегралы, либо рассмотрев свертку как взаимную корреляцию одной последовательности с обращенной во времени другой.

5.3.2. Круговая свертка

Из сказанного в разделе 5.2.1 следует, что результат корреляции двух периодических последовательностей неравной длины — это циклическая последовательность, период которой равен периоду меньшей последовательности, что, таким образом, является неверным результатом. Поскольку свертка эквивалента взаимной корреляции одной последовательности с обращенной во времени второй, сказанное справедливо и для свертки. Следовательно, как и в случае с корреляцией, при свертке необходимо, чтобы две последовательности были равной длины. Итак, если длины последовательностей равны N_1 и N_2 , то к первой из них следует добавить $N_2 - 1$ нулей, а ко второй — $N_1 - 1$ нулей. После этого обе последовательности получат равную длину $N_1 + N_2 - 1$, и будет вычислено правильное значение линейной свертки при выполнении других необходимых условий, указанных в разделе, посвященном корреляции.

5.3.3. Идентификация систем

В уравнении (5.95) представлена связь между входом системы $x(n)$ и ее выходом $y(n)$. Термином *идентификация системы* обозначают определение характеристики $h(n)$, если она неизвестна. Если на вход системы подать пробный сигнал $x(n)$ и измерить выход $y(n)$, характеристику $h(n)$ можно определить следующим образом (также можно использовать метод, описанный в разделе 5.2.2.4).

Из уравнения (5.91) следует, что $y(n) = h(0)x(n) + h(1)x(n-1) + \dots + h(n)x(0)$. При $n = 0$ $y(0) = h(0)x(0)$, поэтому

$$h(0) = \frac{y(0)}{x(0)}. \quad (5.108)$$

Далее, расписывая и переупорядочивая уравнение (5.93), получаем:

$$y(n) = h(n)x(0) + \sum_{m=0}^{n-1} h(m)x(n-m), \quad n \geq 1, \quad (5.109)$$

поэтому

$$h(n) = \frac{y(n) - \sum_{m=0}^{n-1} h(m)x(n-m)}{x(0)}, \quad n \geq 1, x(0) \neq 0. \quad (5.110)$$

Используя уравнения (5.108) и (5.110), можно вычислить $h(n)$.

Пример 5.8

Пробный сигнал $x(n) = \{1; 1; 1\}$ подается на вход системы с неизвестной импульсной характеристикой $h(n)$. На входе системы наблюдается последовательность $y(n) = \{1; 4; 8; 10; 8; 4; 1\}$. Определите $h(n)$.

Из уравнения (5.108)

$$h(0) = \frac{y(0)}{x(0)} = \frac{1}{1} = 1.$$

Используя уравнение (5.110), находим

$$h(n) = \frac{y(n) - \sum_{m=0}^{n-1} h(m)x(n-m)}{x(0)}.$$

Для $h(1)$ получаем следующее выражение:

$$h(1) = \frac{y(1) - \sum_{m=0}^0 h(m)x(1-m)}{x(0)} = \frac{y(1) - h(0)x(1)}{x(0)} = \frac{4 - 1 \times 1}{1} = 3.$$

Для $h(2)$ результаты таковы:

$$\begin{aligned} h(2) &= \frac{y(2) - \sum_{m=0}^1 h(m)x(2-m)}{x(0)} = \frac{y(2) - h(0)x(2) - h(1)x(1)}{x(0)} = \\ &= \frac{8 - 1 \times 1 - 3 \times 1}{1} = 4. \end{aligned}$$

Для $h(3)$ получаем следующее выражение:

$$\begin{aligned} h(3) &= \frac{y(3) - \sum_{m=0}^2 h(m)x(3-m)}{x(0)} = \frac{y(3) - h(0)x(3) - h(1)x(2) - h(2)x(1)}{x(0)} = \\ &= \frac{10 - 1 \times 0 - 3 \times 1 - 4 \times 1}{1} = 3. \end{aligned}$$

Для $h(4)$ вычисляем:

$$\begin{aligned} h(4) &= \frac{y(4) - \sum_{m=0}^3 h(m)x(4-m)}{x(0)} = \\ &= \frac{y(4) - h(0)x(4) - h(1)x(3) - h(2)x(2) - h(3)x(1)}{x(0)} = \\ &= \frac{8 - 1 \times 0 - 3 \times 0 - 4 \times 1 - 3 \times 1}{1} = 1. \end{aligned}$$

Для $h(5)$ получаем следующее выражение:

$$\begin{aligned} h(4) &= \frac{y(4) - \sum_{m=0}^3 h(m)x(4-m)}{x(0)} = \\ &= \frac{y(4) - h(0)x(4) - h(1)x(3) - h(2)x(2) - h(3)x(1)}{x(0)} = \\ &= \frac{8 - 1 \times 0 - 3 \times 0 - 4 \times 1 - 3 \times 1}{1} = 1. \end{aligned}$$

Фактически $h(n) = 0$ для $n \geq 5$. Следовательно, $h(n) = \{1; 3; 4; 3; 1\}$.

5.3.4. Обращение свертки

Если импульсная характеристика и выход системы известны, то для поиска неизвестного входа применяется *обращение свертки*. Обратить свертку можно с помощью процедуры, подобной описанной в разделе 5.3.3 для идентификации системы. Расписывая и переупорядочивая уравнение (5.93), получаем

$$y(n) = h(0)x(n) + \sum_{m=1}^n h(m)x(n-m). \quad (5.111)$$

При $n = 0$ $y(0) = h(0)x(0)$.

Следовательно,

$$x(0) = \frac{y(0)}{h(0)}. \quad (5.112)$$

Из уравнения (5.111) получаем

$$x(n) = \frac{y(n) - \sum_{m=1}^n h(m)x(n-m)}{h(0)}. \quad (5.113)$$

Уравнения (5.112) и (5.113) подобны уравнениям (5.108) и (5.110), так что процедура расчета $x(n)$ аналогична процедуре расчета $h(n)$.

Пример 5.9

Для системы, использованной в примере 5.8, вычислите вход $x(n)$ по данным $h(n) = \{1; 3; 4; 3; 1\}$ и $y(n) = \{1; 4; 8; 10; 8; 4; 1\}$.

Из уравнения (5.112) вычисляем:

$$x(0) = \frac{y(0)}{h(0)} = \frac{1}{1} = 1.$$

Из уравнения (5.113) получаем:

$$x(1) = \frac{y(1) - h(1)x(0)}{h(0)} = \frac{4 - 3 \times 1}{1} = 1$$

$$x(2) = \frac{y(2) - h(1)x(1) - h(2)x(0)}{h(0)} = \frac{8 - 3 \times 1 - 4 \times 1}{1} = 1$$

$$x(3) = \frac{y(3) - h(1)x(2) - h(2)x(1)}{h(0)} = \frac{10 - 3 \times 1 - 4 \times 1 - 3 \times 1}{1} = 0.$$

Для $n \geq 3$ $x(n) = 0$.

Следовательно, $x(n) = \{1; 1; 1\}$, что согласуется со значениями, использованными в примере 5.8.

5.3.5. Слепое обращение свертки

Процесс определения входного сигнала по выходному при неизвестной импульсной характеристике системы называется *слепым обращением свертки*. Описанный ниже метод его выполнения основан на разработке Белла и Сежновски [2]. Задача и ее решение иллюстрируется на рис. 5.23. На рис. 5.23, а требуемый неизвестный исходный сигнал $x(n)$ передается через систему с импульсной характеристикой $h(n)$, в результате чего получается измеренный выходной сигнал $f(n)$. Сигнал $f(n)$ представляет собой результат свертки $h(n)$ с $x(n)$ ($h(n) \otimes x(n)$), следовательно, искажается запаздывающей копией $x(n)$. В задаче требуется вычислить сигнал $u(n)$, являющийся хорошей аппроксимацией $x(n)$. Следовательно, как показано на рис. 5.23, б, требуется причинный фильтр $w(n)$, который при свертке входного сигнала с $f(n)$ даст необходимый выход $u(n)$. В качестве такого фильтра можно использовать трансверсальный фильтр, изображенный на рис. 5.24 (сравните с рис. 1.4, а). Выход этого фильтра равен

$$u(n) = \sum_{m=0}^{L-1} w(m)f(n-m),$$

что можно записать в альтернативной матричной форме

$$\mathbf{U} = \mathbf{W}\mathbf{F}, \quad (5.114)$$

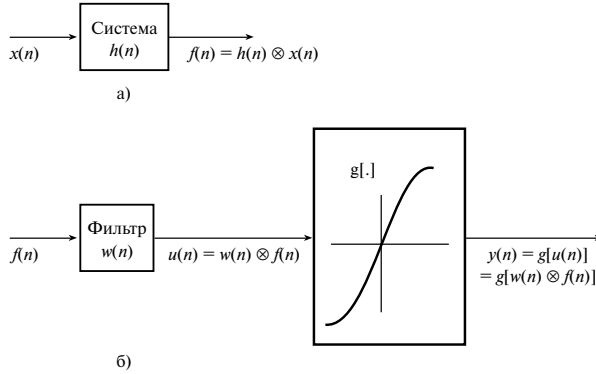
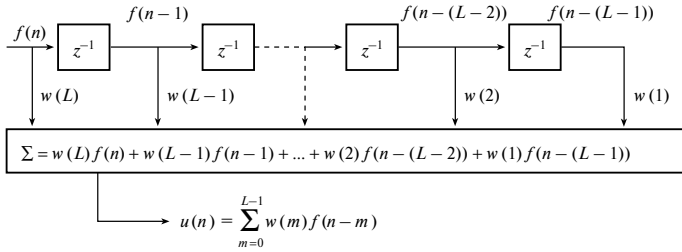


Рис. 5.23. Слепое обращение свертки

Рис. 5.24. Трансверсальный фильтр $w(n)$ для слепой свертки

где $\mathbf{U} = \{u(0), u(1), \dots, u(N)\}^T$,

$$\mathbf{W} = \begin{Bmatrix} w(L) & 0 & \dots & 0 & \dots & 0 \\ w(L-1) & w(L) & \dots & 0 & \dots & 0 \\ \vdots & \dots & \dots & \vdots & \dots & \vdots \\ w(1) & w(2) & \dots & w(L) & \dots & 0 \\ \vdots & \dots & \dots & \vdots & \dots & \vdots \\ 0 & \dots & \dots & w(1) & \dots & w(L) \end{Bmatrix}$$

$\mathbf{F} = \{f(0), f(1), \dots, f(N)\}^T$, N — число элементов во временном ряде.

В работе [2] использовался принцип максимизации информации, на основе которого выводился алгоритм адаптивного вычисления весовых коэффициентов $\mathbf{W}$. Следовательно, авторы проводили настройку, снижая статистическую корреляцию между точками $u(n)$. Этот подход известен как *отбеливание* $u(n)$, поскольку выборки в последовательности белого шума статистически независимы. Чтобы достичь этого, необходимо устранить статистические корреляции высоких порядков. Для этого $u(n)$ подается в систему с нелинейной передаточной функцией $g[u(n)]$ и максимизируется инфор-

мация на выходе системы $y(n) = g[u(n)]$. Обновление коэффициентов происходит по следующим формулам:

$$\Delta w(L-j) \propto \sum_{n=j}^N \left(\frac{1}{w(L)} - 2x(n)y(n) \right) \quad (5.115)$$

и

$$\Delta w(L-j) \propto \sum_{n=j}^N (-2x(n-1)y(n)). \quad (5.116)$$

Алгоритм продолжает выполняться, пока $\Delta w(L)$ и $\Delta w(L-j)$ не станет малым. Затем с использованием найденных весовых коэффициентов задержек и данных, подлежащих обработке, реализуется соответствующий фильтр.

5.3.6. Быстрая линейная свертка

В разделе 5.2.3, было показано, что вычисление свертки можно ускорить с помощью теоремы о корреляции. Существует также подобная *теорема о свертке*. Итак, используя дискретную терминологию и временную область, можно записать

$$x_1(l) \circledast x_2(r) = F_D^{-1} [X_1(k)X_2(k)]. \quad (5.117)$$

Уравнение (5.117) является формулировкой *теоремы о свертке*, где F_D^{-1} обозначает обратное дискретное преобразование Фурье, $X_1(k)$ — ДПФ-образ $x_1(l)$, $X_2(k)$ — ДПФ-образ $x_2(r)$. Как и в разделе 5.2.3, $x_1(l)$ и $x_2(r)$ — периодические последовательности длины N .

5.3.6.1. Доказательство теоремы о свертке

Доказательство данной теоремы практически идентично доказательству теоремы о корреляции, приведенному в разделе 5.2.3. При свертке одна из последовательностей данных обращена, так что вместо уравнения (5.65) применяется сопряженное:

$$X_1(k) = \sum_{l=0}^{N-1} x_1(l) e^{(2\pi i/N)(-lk)}, \quad (5.118)$$

при этом уравнение (5.66) остается без изменений:

$$X_2(k) = \sum_{r=0}^{N-1} x_2(r) e^{(2\pi i/N)(-rk)}. \quad (5.119)$$

Следовательно, повторно определив $x_3(n)$ как периодическую последовательность длины N с ДПФ-образом $X_3(l)$, $X_3(k)$ можно записать как

$$X_3(k) = X_1(k)X_2(k). \quad (5.120)$$

После этого применяется процедура, описанная в разделе 5.2.3, приводящая в конечном итоге к искомому результату:

$$x_1(l) \circledast x_2(r) = F_D^{-1}[X_1(k)X_2(k)] \quad (5.121)$$

для свертки во временной области. Для свертки в частотной области применяется аналогичное уравнение:

$$\frac{1}{N}[X_1(k) \circledast X_2(k)] = F_D[x_1(l)x_2(r)]. \quad (5.122)$$

Последние две формулы представляют периодические, или круговые, свертки, которые можно преобразовать в линейные с помощью дополняющих нулей, как описано в разделе 5.3.2.

5.3.7. Вычислительные преимущества быстрой линейной свертки

Метод быстрой линейной свертки предлагает преимущества меньшей вычислительной сложности по сравнению с прямым подходом, только если число значений, подлежащих свертке, достаточно велико. Ниже сравнивается число умножений, требуемых для выполнения свертки с помощью прямого и быстрого методов, выбранное в качестве меры их относительной вычислительной эффективности.

Необходимые расчеты для прямого метода выполняются согласно уравнению (5.90). Очевидно, что для получения линейной свертки двух N -точечных последовательностей $h(n-m)$ и $x(m)$ необходимо умножить каждое значение $h(n-m)$ на каждое значение $x(m)$. Следовательно, N значений $h(n-m)$ нужно перемножить с N значениями $x(m)$, в результате всего требуется $N \times N = N^2$ умножений.

Рассмотрим теперь линейную свертку тех же двух N -точечных последовательностей с помощью быстрого метода, записанного в уравнении (5.121). Прибавление необходимых дополняющих нулей означает, что каждая последовательность имеет длину $2N-1$ точек. Предположим, что $2N-1 \approx 2N$, например, $N \geq 8$, и что для использования БПФ по основанию 2 N — целая степень 2, т.е. $N = 2^d$, где d — целое. Можно показать, что число комплексных умножений для N -точечного БПФ равно $(N/2) \log_2 N$ (раздел 3.5.3), так что для $2N$ -точечного БПФ необходимо $(2N/2) \log_2 2N$ или $N \log_2 2N$ комплексных умножений. Согласно уравнению (5.121) требуется вычислить два ДПФ и одно обратное ДПФ. Для расчета обратного ДПФ можно использовать модифицированное ДПФ (раздел 3.6). Таким образом, необходимо вычислить три $2N$ -точечных БПФ, включающих в сумме $3N \log_2 2N$ комплексных умножений. Для каждого из $2N$ значений уравнения (5.121) необходимо выполнить комплексное умножение $X_1(k)X_2(k)$, таким образом общее число комплексных умножений возрастает до $3N \log_2 2N + 2N$. Далее каждое комплексное умножение вида $(A+iB)(C+iD)$ требует четырех действительных умножений AC , AD , BC и BD . Значит, в целом необходимо $12N \log_2 2N + 8N$ действительных умножений.

Таким образом, можно сделать вывод, что прямой метод требует N^2 действительных умножений, тогда как метод быстрой свертки требует $12N \log_2 2N + 8N$ умножений. В табл. 5.1 сравниваются числа действительных умножений, необходимые для различных значений N . Из таблицы видно, что быстрая свертка лучше прямого метода, если

длина последовательностей превышает 128 точек данных, причем для последовательностей из 1024 точек быстрая свертка дает результат за время, меньшее примерно в 10 раз. Тот же вывод справедлив и для прямой и быстрой корреляции.

Таблица 5.1. Число действительных умножений, требуемых для выполнения свертки двух N -точечных последовательностей

N	<i>Прямой метод</i>	<i>Быстрая свертка</i>	<i>(Быстрая свертка)/(прямой метод)</i>
8	64	448	7
16	256	1 088	4, 25
32	1 024	2 560	2, 5
64	4 096	5 888	1, 4375
128	16 384	13 312	0, 8125
256	65 536	29 696	0, 4531
512	262 144	65 536	0, 250
1024	1 048 576	143 360	0, 1367
2048	4 194 304	311 296	0, 0742

5.3.8. Свертка и корреляция путем сегментации

До этого момента предполагалось, что две функции, подлежащие свертке (или расчету корреляции), имеют конечную длительность. Впрочем, это справедливо не всегда. Например, можно считать, что входные данные имеют бесконечную длительность либо потому, что они поступают фактически непрерывно, либо (более вероятная причина) потому, что доступная память недостаточно велика, чтобы вместить их все. В подобных случаях свертку (или корреляцию) необходимо выполнять поэтапно, разделив входные данные на отдельные блоки, выполнив необходимые вычисления для каждого блока и затем объединив результаты. Для реализации подобной схемы используются два метода: наложения-сложения (overlap-add) и наложения-записи (overlap-save), которые будут описаны ниже. Перед этим, впрочем, необходимо рассмотреть, как повысить эффективность вычислений, если две функции не начинаются в нулевой момент времени.

На рис. 5.25 показаны два дискретных сигнала $x(n)$ и $h(n)$, а также их свертка $x(n) \otimes h(n) = y(n)$. Сигналы $x(n)$ и $h(n)$ начинаются соответственно в моменты a и b , поэтому если a и b велики по сравнению с количеством данных N_1 и N_2 в $x(n)$ и $h(n)$ соответственно, то во многих тактах вычислений будут фигурировать расчеты с нулевыми данными. Число этих вычислений можно уменьшить, сместив сигналы так, чтобы они выходили из начала координат, как показано на рис. 5.26. После этого данные необходимо дополнить нулями, чтобы оба сигнала содержали равное число точек $N = N_1 + N_2 - 1$, и их периодическая свертка соответствовала линейной свертке двух сигналов. Свертка выполняется согласно теореме о свертке (уравнение (5.117)) и алгоритму БПФ. Чтобы полученный результат был правильным, вычисленную свертку необходимо так сместить вдоль оси n , чтобы она начиналась в точке $n = a + b$ (рис. 5.26, z). На приведенном рисунке предполагается, что $N = 2^d$, где d — целое, поэтому можно использовать БПФ по основанию 2.

Аналогичная иллюстрация для корреляции $x(n)$ и $h(n)$, $r_{xh}(n)$, приведена на рис. 5.27. При трансляции указанных сигналов в начало координат вводятся дополнительные нули, так что $N = 2^d \geq N_1 + N_2 - 1$, и корреляция вычисляется согласно теореме о

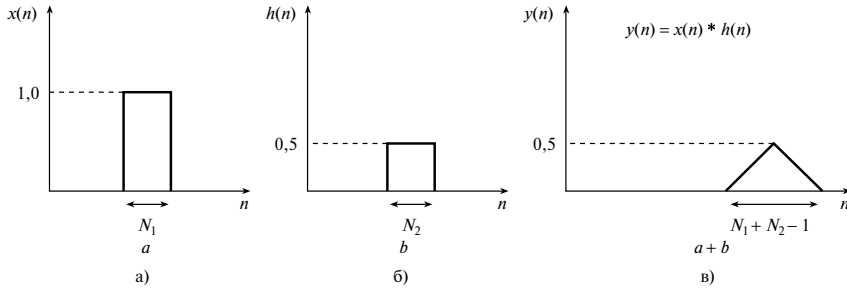


Рис. 5.25. Свертка $y(n) = x(n) \otimes h(n)$ двух сигналов $x(n)$ и $h(n)$, не выходящих из начала координат

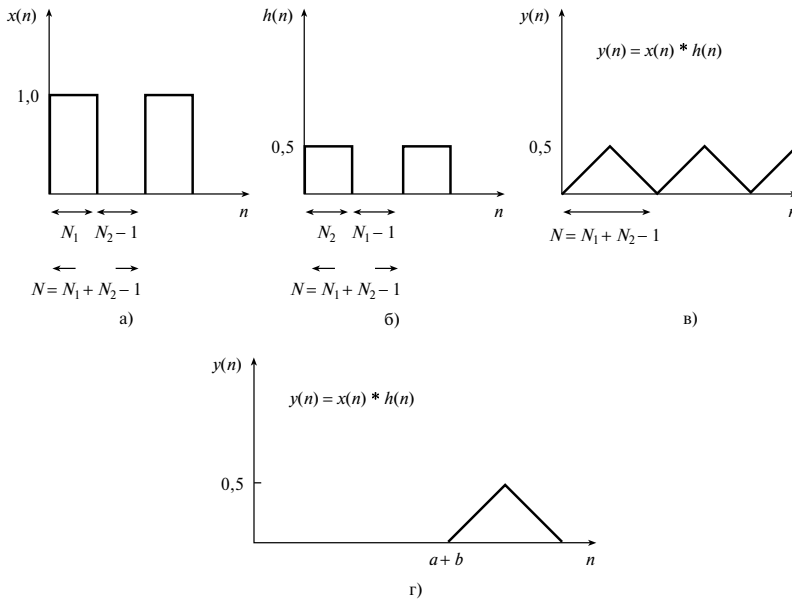


Рис. 5.26. Свертка двух сигналов, изображенных на рис. 5.25, для получения которой $x(n)$ и $h(n)$ транслировались в начало координат. Прибавление к $x(n)$ $N_2 - 1$ дополняющих нулей (панель а). Прибавление к $h(n)$ $N_1 - 1$ дополняющих нулей (панель б). Свертка $y(n) = x(n) \otimes h(n)$ (панель в). Правильная линейная свертка получается заменой $y(n)$ вдоль оси n функцией, начинающейся в точке $n = a + b$ (панель з)

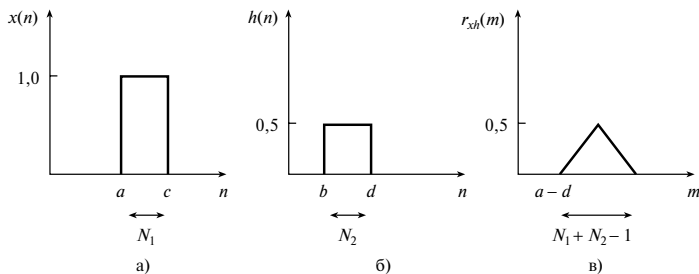


Рис. 5.27. Взаимная корреляция $r(m)$ двух сигналов $x(n)$ и $h(n)$, которые начинаются не в начале координат: а) $x(n)$; б) $h(n)$; в) $r_{xh}(m)$

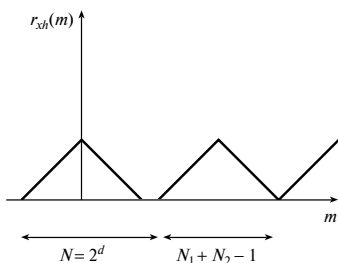


Рис. 5.28. Неверная взаимная корреляция, полученная при трансляции $x(n)$ и $h(n)$ в начало координат

корреляции (формула (5.77)). Получающийся результат представлен на рис. 5.28. Это не периодическая копия рис. 5.27, в, хотя и содержит правильную базовую форму сигнала. Искомый периодический результат можно получить, перенеся $x(n)$ в точку $n = N - N_1 + 1$, при этом расположив $h(n)$ в точке $n = 0$, как на рис. 5.29, где видно, что в результате указанных действий получается требуемая периодическая корреляционная функция (рис. 5.29, в). Полученный результат нужно сместить на $a - d - N + N_1 + N_2$ точек данных вправо, чтобы переместить в верное значение $a - d$ (сравните с рис. 5.29, в).

Распространим теперь приведенные соображения на задачу с бесконечной последовательностью $x(n)$, которую нужно свернуть с конечной последовательностью $h(n)$.

5.3.9. Метод наложения-сложения

Пусть $x(n)$ делится на сегменты равной длины, состоящие из N_1 точек данных. Далее предположим, что данные последовательности периодичны и требуется найти их свертку с N_2 точками данных (последовательность $h(n)$), дополненными $N_1 - N_2$ нулями, так что теперь обе последовательности периодичны и имеют длину N_1 . Результат этой свертки будет неверным, поскольку для получения правильного результата каждая последовательность должна иметь длину $N = N_1 + N_2 - 1$. Впрочем, каждый сегмент $x(n)$ имеет длину N_1 (и эту величину увеличить нельзя). Чтобы устранить эту проблему, можно рассмотреть сегменты $x(n)$ с длиной N и заменить последние $N_2 - 1$ точек данных нулями, расширив первоначальные $N - N_2 + 1 = N_1$ данных (рис. 5.30).

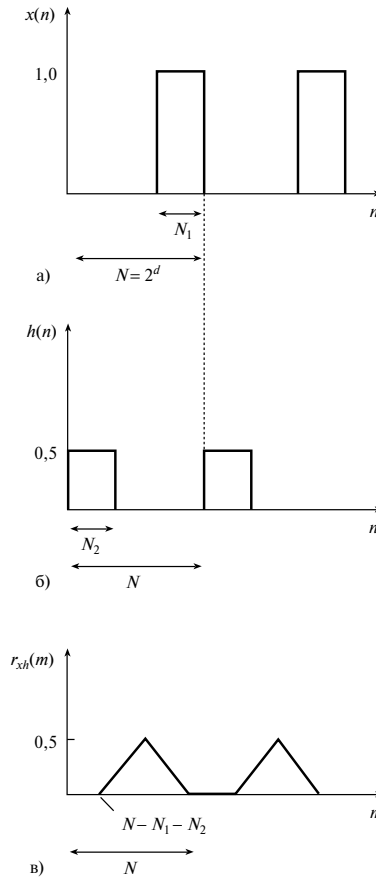


Рис. 5.29. Метод получения правильной взаимной корреляции двух последовательностей $x(n)$ и $h(n)$: а) $x(n)$ смещается в $N - N_1 + 1$; б) $h(n)$ смещается в начало координат; в) получающийся правильный периодический коэффициент взаимной корреляции $r_{xh}(m)$

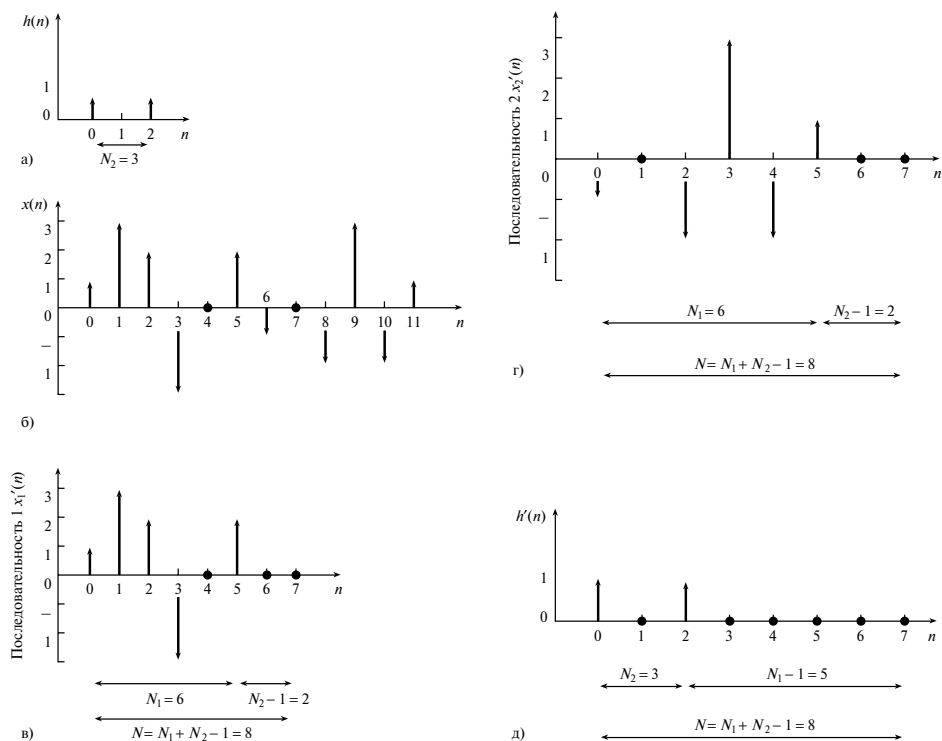


Рис. 5.30. Свертка методом наложения-сложения

Подобным способом последовательность N_1 данных $x(n)$ с $N_2 - 1$ дополняющими нулями сворачивается с N_2 данными последовательности $h(n)$ с $N_2 - 1$ дополняющими нулями. Обе последовательности содержат $N = N_1 + N_2 - 1$ точек данных, и их свертка находится правильно (рис. 5.31). Та же процедура выполняется для оставшихся последовательностей $x(n)$ длины N . Поскольку последние $N_2 - 1$ точек данных сегментов $x(n)$ были заменены нулями, получающиеся сверточные функции ошибочны в первых и последних $N_2 - 1$ точках каждой свертки, но все эти точки суммируются и дают правильную свертку при трансляции каждого свернутого сигнала в точку его истинного начала ($a + b$), а последние $N_2 - 1$ точек свертки находятся путем наложения одного сегмента на другой. Данный процесс иллюстрируется на рис. 5.31. Подытожим: вначале для устранения краевых эффектов к сегментам добавляется достаточное число нулей, затем результаты свертки накладываются друг на друга точно так, как последовательность длиной N_1 дополнялась нулями, а затем находится их сумма. Название метода — *наложение-сложение* — следует непосредственно из выполняемых действий.

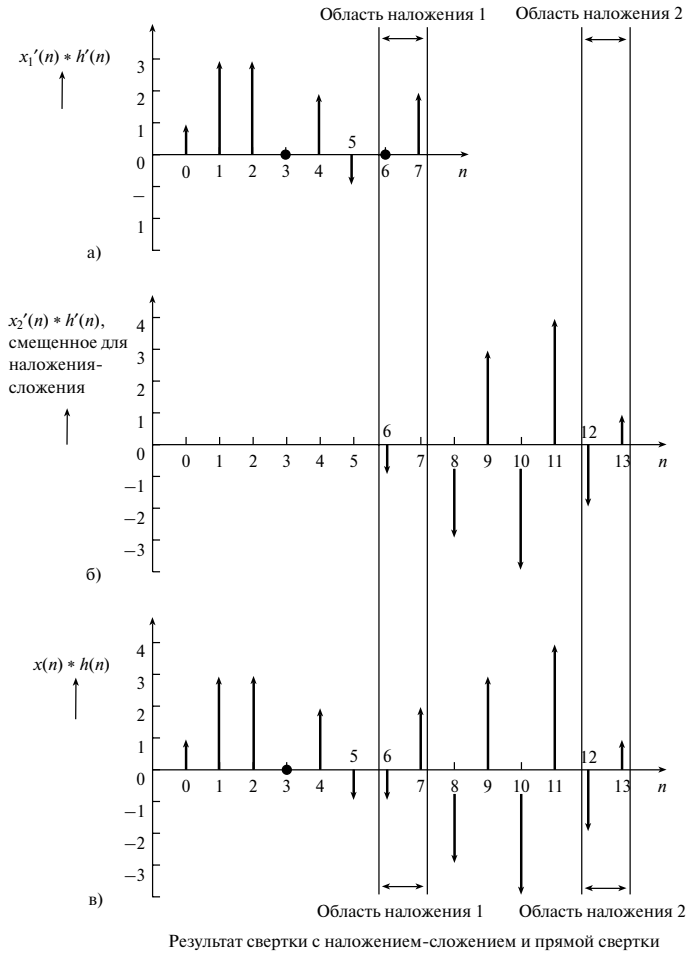


Рис. 5.31. Эквивалентность метода наложения-сложения и прямой свертки

Пример 5.10

Используйте процедуру наложения-сложения, чтобы найти свертку двух последовательностей $h(n) = \{1; 0; 1\}$ и $x(n) = \{1; 3; 2; -3; 0; 2; -1; 0; -2; 3; -2; 1; \dots\}$.

Решение

Пусть последовательность $x(n)$ разбита на сегменты длиной $N_1 = 6$, при этом N (число точек в ДПФ) равно $N_1 + N_2 - 1 = 6 + 3 - 1 = 8 = 2^d$, где $d = 3$, таким образом удовлетворяются требования к использованию линейной свертки и БПФ по основанию 2.

Дополняя $h(n)$ нулями, получаем последовательность $h'(n)$:

$$h'(n) = \{1; 0; 1; 0; 0; 0; 0; 0\}.$$

Первые две расширенные последовательности $x(n)$ имеют такой вид:

$$x'_1(n) = \{1; 3; 2; -3; 0; 2; 0; 0\}$$

и

$$x'_2(n) = \{-1; 0; -2; 3; -2; 1; 0; 0\}.$$

Сверточная сумма $x'_1(n) \circledast h'(n)$ содержит следующие члены:

$$\begin{aligned} y_{10} &= h'_0 x'_{10} = 1, \\ y_{11} &= h'_0 x'_{11} + h'_1 x'_{10} = 3 + 0 = 3, \\ y_{12} &= h'_0 x'_{12} + h'_1 x'_{11} + h'_2 x'_{10} = 2 + 0 + 1 = 3, \\ y_{13} &= h'_0 x'_{13} + h'_1 x'_{12} + h'_2 x'_{11} = -3 + 0 + 3 = 0, \\ y_{14} &= h'_0 x'_{14} + h'_1 x'_{13} + h'_2 x'_{12} = 0 + 0 + 2 = 2, \\ y_{15} &= h'_0 x'_{15} + h'_1 x'_{14} + h'_2 x'_{13} = 2 + 0 - 3 = -1, \\ y_{16} &= h'_0 x'_{16} + h'_1 x'_{15} + h'_2 x'_{14} = 0 + 0 + 0 = 0, \\ y_{17} &= h'_0 x'_{17} + h'_1 x'_{16} + h'_2 x'_{15} = 0 + 0 + 2 = 2. \end{aligned}$$

Сверточная сумма $x'_2(n) \circledast h'(n)$ состоит из таких компонентов:

$$\begin{aligned} y_{20} &= h'_0 x'_{20} = -1, \\ y_{21} &= h'_0 x'_{21} + h'_1 x'_{20} = 0 + 0 = 0, \\ y_{22} &= h'_0 x'_{22} + h'_1 x'_{21} + h'_2 x'_{20} = -2 + 0 - 1 = -3, \\ y_{23} &= h'_0 x'_{23} + h'_1 x'_{22} + h'_2 x'_{21} = 3 + 0 + 0 = 3, \\ y_{24} &= h'_0 x'_{24} + h'_1 x'_{23} + h'_2 x'_{22} = -2 + 0 - 2 = -4, \\ y_{25} &= h'_0 x'_{25} + h'_1 x'_{24} + h'_2 x'_{23} = 1 + 0 + 3 = 4, \\ y_{26} &= h'_0 x'_{26} + h'_1 x'_{25} + h'_2 x'_{24} = 0 + 0 - 2 = -2, \\ y_{27} &= h'_0 x'_{27} + h'_1 x'_{26} + h'_2 x'_{25} = 0 + 0 + 1 = 1. \end{aligned}$$

Приведенные выше сверточные суммы показаны на рис. 5.31, *а* и *б* соответственно. Если первые $N_2 - 1 = 2$ точки данных x'_2 наложить на последние $N_2 - 1$ точек данных x'_1 и просуммировать сверточные суммы, получатся первые 12 точек данных итогового сверточного сигнала, показанные на рис. 5.31, *в*.

Можно показать, что приведенный выше результат идентичен тому, который получается при прямом вычислении свертки. Исходная последовательность $x(n)$ содержит 12, а $h(n)$ — 3 точки данных. Чтобы получить линейную свертку этих двух последовательностей, их нужно дополнить нулями, чтобы длина обеих составляла $12+3-1 = 14$ точек данных. Таким образом, последовательности становятся равными

$$h'(n) = \{1; 0; 1; 0; 0; 0; 0; 0; 0; 0; 0; 0; 0; 0\}$$

и

$$x(n) = \{1; 3; 2; -3; 0; 2; -1; 0; -2; 3; -2; 1; 0; 0\}.$$

Ниже приводятся первые девять членов сверточной суммы.

$$y_0 = h'_0 x'_0 = 1,$$

$$y_1 = h'_0 x'_1 + h'_1 x'_0 = 3,$$

$$y_2 = h'_0 x'_2 + h'_1 x'_1 + h'_2 x'_0 = 2 + 0 + 1 = 3,$$

$$y_3 = h'_0 x'_3 + h'_1 x'_2 + h'_2 x'_1 = -3 + 0 + 3 = 0,$$

$$y_4 = h'_0 x'_4 + h'_1 x'_3 + h'_2 x'_2 = 0 + 0 + 2 = 2,$$

$$y_5 = h'_0 x'_5 + h'_1 x'_4 + h'_2 x'_3 = 2 + 0 - 3 = -1,$$

$$y_6 = h'_0 x'_6 + h'_1 x'_5 + h'_2 x'_4 = -1 + 0 + 0 = -1,$$

$$y_7 = h'_0 x'_7 + h'_1 x'_6 + h'_2 x'_5 = 0 + 0 + 2 = 2,$$

$$y_8 = h'_0 x'_8 + h'_1 x'_7 + h'_2 x'_6 = -2 + 0 - 1 = -3.$$

Члены этой сверточной суммы действительно идентичны величинам, полученным методом наложения-сложения, изображенным на рис. 5.31, в.

На основе сказанного можно вывести следующую процедуру наложения-сложения для быстрой свертки (или корреляции) с помощью сегментации.

1. Выбрать число N_1 данных последовательности $x(n)$ порядка числа N_2 данных последовательности $h(n)$ (причем $N_1 > N_2$) и число точек ДПФ в виде $N = 2^d$, где d — целое и $N \gg N_1 + N_2 - 1$. Для удовлетворения этим условиям последовательности данных при необходимости дополняются нулями.
2. Сместить расширенные сегменты данных $x(n)$ в начало координат.
3. Для каждого расширенного сегмента данных $x(n)$, $x'(n)$, найти быструю свертку $x'(n) \otimes h'(n)$, т.е. вычислить $X(k)H(k)$ и затем применить обратное преобразование.
4. Последовательно наложить полученные свертки, совмещая последние $N_2 - 1$ точек одного результата с $N_2 - 1$ первыми точками другого, и просуммировать свертки.

5.3.10. Метод наложения-записи

Рассмотрим еще раз свертку $x(n) \otimes h(n)$, проиллюстрированную на рис. 5.32, где к $h(n)$ добавлено $N_2 - 1$ нулей, так что обе последовательности имеют длину N_1 . Чтобы получить линейную свертку указанных последовательностей, $h(n)$ можно последовательно (по одной точке) смещать $h(n)$ вправо, проводить перекрестное умножение



Рис. 5.32. Последовательности $x(n)$ и $h(n)$, причем $h(n)$ дополнена $N_2 - 1$ нулями

соответствующих членов и суммировать результаты. В то же время, поскольку ни одна последовательность не имеет длины $N_1 + N_2 - 1$, результат не будет равен $x(n) \otimes h(n)$. Фактически, если длина последовательности $x(n)$ равна N_1 , пропущено $N_2 - 1$ нулей. Это означает, что первые $N_2 - 1$ членов сверточной суммы будут неверными, и их следует отбросить. Следовательно, если данные $x(n)$ делятся на смежные сегменты длины N_1 , первые $N_2 - 1$ значений каждой сверточной суммы следует отбрасывать. Значит, свертка $x(n) \otimes h(n)$ будет содержать периодическую последовательность пропусков длины $N_2 - 1$. Данные пропуски можно корректно заполнить, наложив последние $N_2 - 1$ точек данных каждой последовательности $x(n)$ длины N_1 на первые $N_2 - 1$ точек данных следующей последовательности, а затем отбросив эти первые $N_2 - 1$ точек данных. Описанная процедура известна как *метод наложения-записи*.

Пример 5.11

Используйте процедуру наложения-записи для свертки последовательностей, приведенных в разделе 5.3.9, т.е.

$$h(n) = \{1; 0; 1\}$$

и

$$x(n) = \{1; 3; 2; -3; 0; 2; -1; 0; -2; 3; -2; 1\}.$$

Решение

Поскольку для $h(n)$ $N_2 = 3$, величина наложения $N_2 - 1 = 2$. Само наложение выполняется так, как показано на рис. 5.33. Ниже приводится расчет сверток для каждого сегмента.

Для сегмента 1:

$$y_{10} = h'_0 x'_{10} = 1,$$

$$y_{11} = h'_0 x'_{11} + h'_1 x'_{10} = 3 + 0 = 3,$$

$$y_{12} = h'_0 x'_{12} + h'_1 x'_{11} + h'_2 x'_{10} = 2 + 0 + 1 = 3,$$

$$y_{13} = h'_0 x'_{13} + h'_1 x'_{12} + h'_2 x'_{11} + h'_3 x'_{10} = -3 + 0 + 3 + 0 = 0.$$

Следовательно,

$$y_1 = \{1; 3; 3; 0\}.$$

Для остальных сегментов следует помнить, что $h_1 = h_3 = 0$. Итак, для сегмента 2 получаем:

$$y_{20} = h'_0 x'_{20} = 2,$$

$$y_{21} = h'_0 x'_{21} = -3,$$

$$y_{22} = h'_0 x'_{22} + h'_2 x'_{20} = 2 + 0 = 2,$$

$$y_{23} = h'_0 x'_{23} + h'_2 x'_{21} + h'_2 x'_{21} = 2 - 3 = -1,$$

$$y_2 = \{2; -3; 2; -1\}.$$

Подобным образом, для сегмента 3 получаем:

$$y_3 = \{0; 2; -1; 2\}.$$

Для сегмента 4:

$$y_4 = \{-1; 0; -3; 3\}.$$

Наконец, для сегмента 5 находим:

$$y_5 = \{-2; 3; -4; 4\}.$$

Данные результаты иллюстрируются в табл. 5.2, из которой видно, что первые $N_2 - 1$ результатов каждой последовательности отбрасываются. Последняя строка кроме первой позиции содержит правильное значение свертки.

Таблица 5.2. Результаты решения примера 5.11

Сегмент 1	y_0	$\underbrace{1 \ 3}_{\text{отбрасывается}}$	$3 \ 0$						
Сегмент 2	y_1		$\underbrace{2 \ 3}_{\text{отбрасывается}}$	$2 \ -1$					
Сегмент 3	y_2			$\underbrace{0 \ 2}_{\text{отбрасывается}}$	$-1 \ 2$				
Сегмент 4	y_3				$\underbrace{-1 \ 0}_{\text{отбрасывается}}$	$-3 \ 3$			
Сегмент 5	y_4					$\underbrace{-2 \ 3}_{\text{отбрасывается}}$	$-4 \ 4$		
$x(n) \otimes h(n)$		$\underbrace{1 \ 3}_{\text{отбрасывается}}$	$3 \ 0$	$2 \ -1$	$-1 \ 2$	$\underbrace{-3 \ 3}_{\text{отбрасывается}}$	$-4 \ 4$		

Таким образом, получаем следующую процедуру наложения-записи.

1. Выбрать число точек данных последовательности $x(n)$ равным $N_1 = 2^d$ и добавить к $h(n)$ $N_2 - 1$ нулей, чтобы обе последовательности имели длину N_1 .
2. Разместить обе последовательности в начале координат.
3. Для каждой последовательности вычислить соответствующие значения $X(k)$ и $X(k)$, используя БПФ.

$h(n)$	1	0	1										
$x(n)$	1	3	2	-3	0	2	-1	0	-2	3	-2	1	
Участок 1	1	3	2	-3									
Участок 2			2	-3	0	2							
Участок 3					0	2	-1	0					
Участок 4							-1	0	-2	3			
Участок 5									-2	3	-2	1	

Рис. 5.33. Наложение сегментов для выполнения свертки методом наложения-записи

4. Вычислить $X(k)H(k)$ и обратное к нему, которое равно свертке каждой последовательности с $h(n)$.
5. Расположить все свертки так, чтобы каждая следующая совмещалась с предыдущей по $N_2 - 1$ точкам.
6. Отбросить первые $N_2 - 1$ точек данных каждой свертки и считать оставшиеся значения, которые соответствуют верной свертке.

5.3.11. Вычислительные преимущества быстрой свертки через сегментацию

В разделе 5.3.8 показано, что ненужных вычислений можно избежать, перенеся начало каждого сегмента в начало координат. Предположим, что это уже сделано. Затем допустим, что вычислительные требования методов наложения-сложения и наложения-записи подобны, поэтому достаточно рассмотреть только первый из этих методов. Предполагается, что последовательность $x(n)$ длины N делится на N/N_1 сегментов, каждый из которых имеет длину N_1 , что последовательность $h(n)$ имеет длину N_2 и что длины последовательностей, подлежащих линейной свертке, равны $N^1 = 2^d \geq N_1 + N_2 - 1$. Кроме того, в разделе 5.3.7 было показано, что для выполнения быстрой свертки двух N^1 -точечных последовательностей требуется $12N^1 \log_2 2N^1 + 8N^1$ действительных умножений. Следовательно, чтобы выполнить быструю свертку N -точечной последовательности $x(n)$ с помощью метода наложения-сложения, потребуется $(N/N^1)(12N^1 \log_2 2N^1 + 8N^1) = R_m(S)$ действительных операций умножения. Отсюда следует, что длина сворачиваемых последовательностей N^1 должна быть малой, а длины N_1 сегментов $x(n)$ — близкими к N^1 . В идеальном случае $N^1 = 2^d = N_1 + N_2 - 1$. Число действительных умножений, необходимых для обработки исходной N -точечной последовательности, равно $12N \log_2 2N + 8N = R_m(N)$. Из табл. 5.3 следует, что для примера из раздела 5.3.9 отношение $R_m(S)/R_m(N) \leq 1$, при этом экономия времени вычислений может составлять порядка 50%.

Таблица 5.3. Отношение $R_m(S)/R_m(N)$ (числа действительных умножений для метода сегментации к числу действительных умножений при быстрой свертке обычным образом)

N	N^1	N_1	N/N_1	N_2	$R_m(S)/R_m(N)$	<i>Комментарий</i>
1020	8	6	170	3	0,54	Наилучший результат при малом N^1 , $N_1 \approx N^1$
1024	256	254	4	3	0,83	$N_1 \approx N^1$
1020	128	102	10	3	0,93	
1020	256	204	5	3	1,04	

5.3.12. Связь между сверткой и корреляцией

При свертке значение n -го выхода определяется сверточной суммой, представленной в формуле (5.93):

$$y(n) = \sum_{m=0}^n h(m)x(n-m) = h(0)x(n) + h(1)x(n-1) + \dots + h(n)x(0). \quad (5.123)$$

Значение функции взаимной корреляции для сигналов $h(n)$ и $x(n)$ для i -й задержки представлено в формуле (5.1), которая в несколько модифицированном виде выглядит так:

$$\begin{aligned} r_{hx}(j) &= \frac{1}{N} \sum_{n=0}^{N-1} h(n)x(j+n) = \\ &= \frac{1}{N} [h(0)x(j) + h(1)x(j+1) + \dots + h(N-1)x(j+N-1)]. \end{aligned} \quad (5.124)$$

Сравнить $y(n)$ и $r_{hx}(j)$ проще всего, если $j = 0$, т.е. при нулевой задержке. В этом случае уравнение (5.124) приводится к такому виду:

$$\begin{aligned} r_{hx}(0) &= \frac{1}{N} \sum_{n=0}^{N-1} h(n)x(n) = \\ &= \frac{1}{N} [h(0)x(0) + h(1)x(1) + \dots + h(N-1)x(N-1)]. \end{aligned} \quad (5.125)$$

Сравнивая уравнения (5.123) и (5.125), находим, что они имеют сходную форму, за исключением того, что последовательность $x(n)$ в функции взаимной корреляции идет в обратном порядке по сравнению с тем, как она идет в свертке. Следовательно, свертка эквивалентна функции взаимной корреляции двух сигналов, в которой одна из исходных последовательностей обращена во времени, а нормировочный коэффициент $1/N$ равен единице. Это означает, что свертки и корреляции можно вычислять с помощью одной компьютерной программы, просто обращая одну из последовательностей.

5.4. Реализация корреляции и свертки

При рассмотрении реализации указанных операций следует помнить, что они тесно связаны. Для двух последовательностей данных можно легко найти их корреляцию или свертку, причем вычисления будут отличаться только порядком одной последовательности. Более того, для более длинных последовательностей данных необходимые операции можно ускорить, используя методы быстрого преобразования Фурье, чтобы получить быструю корреляцию или быструю свертку. Если одна последовательность имеет чересчур большую длину, можно использовать методы наложения-сложения или наложения-записи; см. разделы 5.3.9 и 5.3.10 и работы [3, 5, 14].

Свертку или корреляцию можно выполнять, используя, например, КИХ-фильтр, реализуемый с помощью БПФ, а также применяя согласованный фильтр, как показано в разделе 5.2.2 (рис. 5.13). В сфере цифровой обработки для реализации трансверсальных фильтров может использоваться технология приборов с зарядовой связью (ПЗС). Это позволяет получать линейную фазовую характеристику при скоростях передачи данных, превышающих 100 МГц при стандартных конфигурациях линий задержки [6]. Аналоговую обработку можно выполнять, реализовав линии задержки с отводами, используя устройства с поверхностными акустическими волнами (ПАВ) [6]. Эти устройства работают в диапазоне от 2 МГц до 2 ГГц. Другие реализации можно получить на основе конвольверов и корреляторов, универсальных цифровых процессоров сигналов, стандартных микропроцессоров и транспьютеров. В качестве примеров последних можно привести системы реального времени для устранения окулярных артефактов из всех 16 каналов ЭКГ человека [8].

Время вычислений, необходимых для быстрой корреляции и свертки, можно дополнительно сократить в два раза [3]. Рассмотрим свертку $x(n)$ и $h(n)$. При вычислении $X(k)$ действительная часть БПФ заполняется четными членами $x(n)$, а мнимая часть — нечетными, и длина БПФ вполнину сокращается. Далее действительная часть $(1/N)F_D^{-1}[X(k)H(k)]$ дает четные члены искомой сверткой, а мнимая часть — нечетные.

Аналогично свертку двух последовательностей данных $x_1(n)$ и $x_2(n)$ с $h(n)$ можно вычислить одновременно. Заполним действительную часть БПФ элементами последовательности $x_1(n)$, а мнимую часть — элементами последовательности $x_2(n)$ и вычислим $X^1(k)$. Действительная часть $(1/N)F_D^{-1}[X^1(k)H(k)]$ равна $x_1(n) \otimes h(n)$, а мнимая часть — $x_2(n) \otimes h(n)$.

5.5. Примеры применения

5.5.1. Корреляция

Пример 5.12

Данный упрощенный пример касается применения теории корреляции для управления положением в пространстве космического аппарата, чтобы солнечная батарея всегда была обращена к солнцу. Ошибки определения положения представлены как

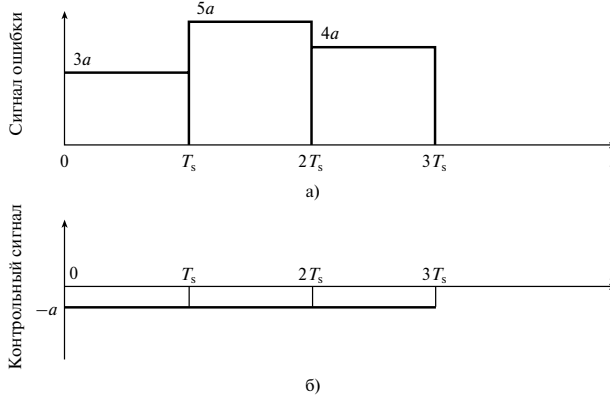


Рис. 5.34. Контроль за положением космического аппарата: а) сигнал ошибки; б) контрольный сигнал

многоуровневые импульсы с разрешением уровней $a = 0,2$ мВ, ширина импульса $T_s = 1$ мкс. Изначально контроль за ошибкой положения космического аппарата при наличии положительной ошибки выполняется путем передачи последовательности отрицательных импульсов высоты a . Система контроля считается удовлетворительной только в том случае, если коэффициент корреляции между ошибкой и контрольными сигналами меньше $-0,5$. На рис. 5.34. а показаны три импульса ошибки, а на рис. 5.34. б — соответствующие импульсы контрольного сигнала. В данном примере предполагается, что этих импульсов достаточно и что рассматривать задержку, превышающую T_s , не нужно. Таким образом задача состоит в том, чтобы определить, можно ли рассматривать систему как удовлетворительную.

Можно доказать, что $r_{12}(\tau) > 0,5$ для $0 \leq \tau \leq T_s$. Взаимную корреляцию можно найти, сместив контрольный сигнал вправо, не меняя при этом сигнал ошибки. Это означает, что требуется найти $r_{12}(-\tau)$.

Итак,

$$r_{12}(-\tau) = \int_{-\infty}^{\infty} v_1(t)v_2(t-\tau)d\tau,$$

где $v_1(t)$ — сигнал ошибки, $v_2(t)$ — контрольный сигнал.

$$\begin{aligned} r_{12}(-\tau) &= \frac{1}{3T_s} \int_{\tau}^{T_s} 3a(-a)dt + \frac{1}{3T_s} \int_{T_s}^{2T_s} 5a(-a)dt + \frac{1}{3T_s} \int_{2T_s}^{3T_s} 4a(-a)dt = \\ &= \frac{a^2}{3T_s} \{ [-3t]_{\tau}^{T_s} + [-5t]_{T_s}^{2T_s} + [-4t]_{2T_s}^{3T_s} \} = \\ &= \frac{a^2}{3T_s} (-3T_s + 3\tau - 10T_s + 5T_s - 12T_s + 8T_s) = \\ &= \frac{a^2}{3T_s} (-12T_s + 3\tau). \end{aligned}$$

Здесь $r_{12}(\tau)$ нормировано (принадлежит диапазону $-1 < r_{12}(r) < 1$) путем деления на следующий нормировочный коэффициент:

$$\frac{1}{3T_s} \left[\int_{-\infty}^{\infty} v_1^2(t) dt \int_{-\infty}^{\infty} v_2^2(t) dt \right]^{1/2}.$$

Теперь

$$\begin{aligned} \int_{-\infty}^{\infty} v_1^2(t) dt &= \int_0^{T_s} (3a)^2 dt + \int_{T_s}^{2T_s} (5a)^2 dt + \int_{2T_s}^{3T_s} (4a)^2 dt = \\ &= a^2 \{ [9t]_0^{T_s} + [25t]_{T_s}^{2T_s} + [16t]_{2T_s}^{3T_s} \} = \\ &= a^2 (9T_s + 25T_s + 16T_s) = 50a^2 T_s. \end{aligned}$$

Кроме того,

$$\int_{-\infty}^{\infty} v_2^2(t) dt = \int_0^{3T_s} (-a)^2 dt = a^2 [t]_0^{3T_s} = 3a^2 T_s.$$

Следовательно, нормировочный коэффициент равен

$$\frac{1}{3T_s} [(50a^2 T_s)(3a^2 T_s)]^{1/2} = \frac{1}{3T_s} 150^{1/2} a^2 T_s,$$

и нормированное выражение для $r_{12}(-\tau)$ приводится к такому виду:

$$\begin{aligned} r_{12}^N(-\tau) &= \frac{3\tau - 12T_s}{150^{1/2} T_s} = \frac{3\tau}{12, 25T_s} - \frac{12}{12, 25} \\ r_{12}^N(-\tau) &= 0, 245 \times 10^6 \tau - 0, 98. \end{aligned}$$

При $\tau = 0$

$$r_{12}^N(0) = -0, 98.$$

При $\tau = 1$ мкс (наибольшее разрешенное значение)

$$r_{12}^N(10^{-6}) = -0, 735.$$

Следовательно, в рассматриваемом диапазоне $|r_{12}^N(-\tau)| > |0, 5|$, что согласуется с критерием хорошего контроля за положением космического аппарата.

Пример 5.13

Для определения расстояния до источника звука требуется гидроакустический комплекс. Источник широкополосный и гауссов с нулевым средним. Система состоит из двух гидроакустических преобразователей (датчиков), разнесенных на расстояние d , и соответствующей системы обработки сигналов. Датчики T_1 и T_1 принимают широкополосные шумы $q_1(t)$ и $q_2(t) = Aq_1(t + \Delta t)$ соответственно, Δt — временная задержка, порожденная разным расстоянием от датчиков до источника, A — соответствующий коэффициент затухания (в данном случае предполагается, что $A = 1$). Система обработки сигналов вычисляет корреляционную функцию выходов двух датчиков, имеющих равные длины.

Изобразите и соответствующим образом подпишите блок-схему простой системы, разработанной для получения корреляции в кратчайшие возможные сроки, и объясните принципы, на которых он основан.

Изобразите выходные сигналы датчиков и их функцию взаимной корреляции, отметив важные особенности.

Если максимальное значение функции взаимной корреляции равно 10, а ширина полосы приемника — 1–10 Гц, чему равна принятая энергия?

Решение

Блок-схема системы изображена на рис. 5.35. Данная система ускоряет вычисление корреляции путем использования теоремы о корреляции и расчета соответствующих БПФ. Такая реализация будет быстрее прямого расчета корреляции, если число точек данных в последовательностях превышает 128. Таким образом, система вычисляет $r_{12}(\tau)$, которое выражается так:

$$r_{12}(j) = F_D^{-1}[F_1(k)F_2(*k)].$$

Выход системы $r_{12}(j)$ равен

$$r_{12}(j) = \frac{1}{N} \sum_{n=0}^{N-1} q_1(n)Aq_1(n + \Delta n + j).$$

Поскольку $q_1(n)$ и $q_2(n)$ случайные, система даст существенный выход только тогда, когда сигналы так смещены, что идут почти в фазе. Это происходит при $j = -\Delta n$. В этом случае выход системы равен

$$\frac{1}{N} \sum_{n=0}^{N-1} q_1^2(n) = p_{AV}, \quad \text{средней мощности}$$

Сигналы и их функция взаимной корреляции изображены на рис. 5.36 и 5.37.

Взаимная корреляция двух сигналов равна

$$r_{12}(\tau) = \frac{1}{T} \int_{-T/2}^{T/2} q_1(t)q_2(t + \tau)dt.$$

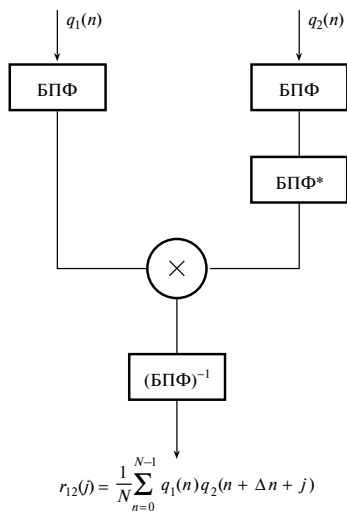
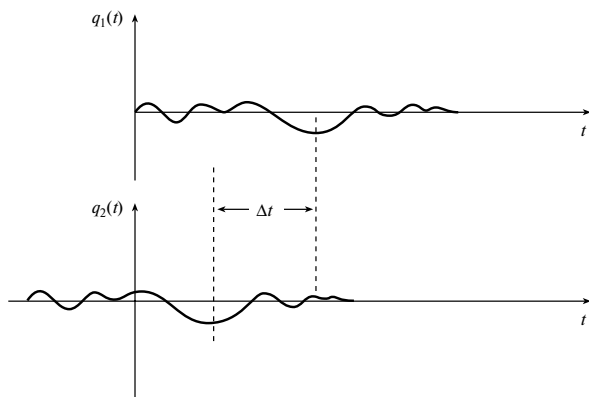


Рис. 5.35. Блок-схема гидроакустического комплекса



Δt – задержка во времени между двумя датчиками

Рис. 5.36. Широкополосный шум, детектируемый гидроакустической системой

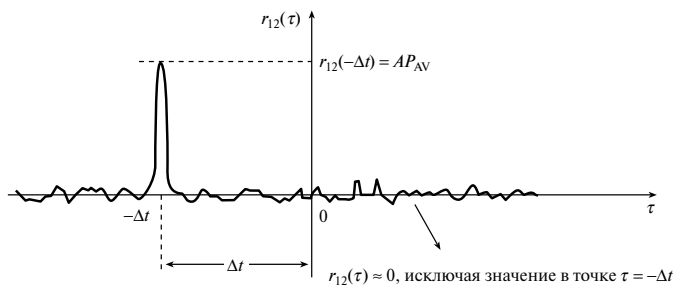


Рис. 5.37. Функция взаимной корреляции сигналов, детектируемых гидроакустической системой

Подставляя выражение для $q_2(t)$, получаем

$$r_{12}(\tau) = \frac{1}{T} \int_{-T/2}^{T/2} q_1(t) A q_1(t + \Delta t + \tau) dt,$$

что можно записать следующим образом:

$$r_{12}(\tau) = \frac{A}{T} \int_{-T/2}^{T/2} q_1(t) q_1(t + \tau') dt, \quad \text{где } \tau' = \Delta t + \tau.$$

Видно, что подынтегральное выражение эквивалентно по величине автокорреляционной функции $q_1(t)$ при нулевой задержке и, следовательно, представляет мощность этого сигнала P_{AV} . Значит,

$$r_{12}(\tau) = AP_{AV} \delta(t + \Delta t),$$

где δ представляет дельта-функцию. Видно, что амплитуда $r_{12}(\tau)$ равна AP_{AV} . Следовательно, $AP_{AV} = 10$.

Принятую энергию в требуемой полосе можно получить, применив вначале для получения спектральной плотности энергии теорему Винера–Хинчина (Wiener–Khinchine). Теорема формулируется следующим образом:

$$\begin{aligned} G_E(f) &= F_D[r_{12}(\tau)] = \\ &= F_D[AP_{AV} \delta(t + \Delta t)] = AP_{AV} e^{i\omega \Delta t}. \end{aligned}$$

Следовательно, $|G_E(f)| = AP_{AV} = 10$ Дж Гц⁻¹. Ширина полосы сигнала равна $10 - 1 = 9$ (Гц). Значит, принята энергия $10 \times 9 = 90$ (Дж).

5.5.2. Свертка

5.5.2.1. БИХ- и КИХ-фильтры

Работа трансверсальных фильтров (КИХ- и БИХ-) является хорошим практическим примером свертки [5, 13]. Эти фильтры можно спроектировать так, чтобы они выполняли свертку последовательностей или более общую цифровую фильтрацию, например, двумерную фильтрацию, используемую в сфере обработки изображений [6] с целью подавления шума, очистки изображений и распознавания шаблонов.

Рассмотрим линейную независимую от времени систему, которую можно описать следующим образом:

$$y(n) = \sum_{k=1}^N a_k y(n-k) + \sum_{k=0}^L b_k x(n-k), \quad (5.126)$$

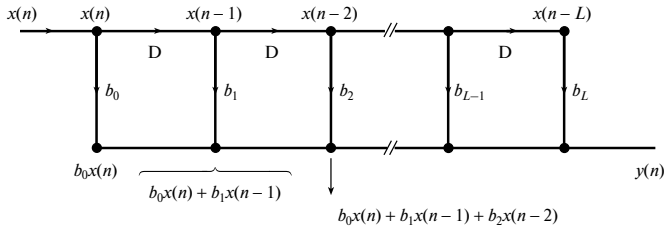


Рис. 5.38. Представление нерекурсивного фильтра в диаграммной форме

где $y(n)$ представляет выходную, а $x(n)$ — входную последовательность. Видно, что выход зависит от текущего входа, а также от предыдущих входов и выходов. В приведенном соотношении a_k и b_k — действительные константы, N — порядок уравнения, который указывает, какое число предыдущих выходов нужно рассматривать.

Поскольку текущий выход зависит от предыдущих выходов, система рекурсивна. Если выход системы зависит только от предыдущих входов, говорят, что система нерекурсивна, и ее можно описать следующим образом:

$$y(n) = \sum_{k=0}^L b_k x(n-k) \quad (5.127)$$

(данное уравнение описывает трансверсальный фильтр, или линию задержки с отводами).

На рис. 5.38 показано описание системы, определяемой уравнением (5.127), в форме диаграммы. Члены суммы, представляющие выход системы, получаются путем суммирования задержанных и взвешенных значений входов. Предположим, что вход $x(n)$ — это единичный импульс $\delta(n)$, где

$$x(n) = \delta(n) \begin{cases} 1, & n = 0, & \text{т.е. } x(0) = 1 \\ 0, & n \neq 0, & \text{т.е. } x(n \neq 0) = 0 \end{cases}.$$

Соответствующий выход — импульсная характеристика $h(n)$. Подставляя последовательные входные значения в уравнение (5.127), получаем

$$\begin{aligned} y(0) &= h(0) = b_0 x(0) + b_1 \times 0 = b_0 \times 1 = b_0, \\ y(1) &= h(1) = b_0 \times 0 + b_1 x(0) + b_2 \times 0 = b_1 \times 1 = b_1, \\ &\vdots \\ y(L) &= h(L) = b_0 \times 0 + 0 + 0 + \dots + 0 + b_L \times 1 = b_L. \end{aligned}$$

Следовательно,

$$h(n) = \{b_0, b_1, \dots, b_L\}, \quad (5.128)$$

откуда видно, что весовые коэффициенты на диаграмме системы соответствуют коэффициентам ее импульсной характеристики. Подобные системы называются *фильтрами с конечной импульсной характеристикой* (КИХ-фильтрами).

Рассмотрим теперь выход, соответствующий входной последовательности общего вида $x(n)$. Подставляя последовательные значения в уравнение (5.127), получаем

$$\begin{aligned} y(n) &= b_0x(n) + b_1x(n-1) + \dots + b_nx(0) \equiv \\ &\equiv h(0)x(n) + h(1)x(n-1) + \dots + h(n)x(0), \end{aligned} \quad (5.129)$$

что, как и следовало ожидать, можно записать как свертку входа с выходом. Следовательно, КИХ-фильтры можно также рассматривать как конвольверы (устройства выполнения свертки), в которых весовые коэффициенты соответствуют коэффициентам их импульсной характеристики.

Аналогичное соотношение существует и для БИХ-фильтров. Рассмотрим рекурсивный фильтр первого порядка, описываемый уравнением

$$y(n) = a_1y(n-1) + b_0x(n). \quad (5.130)$$

Легко показать, что для единичного импульса на входе

$$y(n) = h(n) = b_0a_1^n, \quad n \geq 0. \quad (5.131)$$

Для произвольной входной последовательности $x(n)$, предполагая, что $y(-1) = 0$, можно записать

$$\begin{aligned} y(0) &= b_0x(0), \\ y(1) &= a_1b_0x(0) + b_0x(1), \\ y(2) &= a_1^2b_0x(0) + a_1b_0x(1) + b_0x(2), \\ &\vdots \\ y(n) &= a_1^n b_0x(0) + a_1^{n-1} b_0x(1) + \dots + a_1 b_0x(n-1) + b_0x(n). \end{aligned}$$

Подставляя известные значения весовых коэффициентов из уравнения (5.131), получаем

$$y(n) = h(n)x(0) + h(n-1)x(1) + \dots + h(0)x(n). \quad (5.132)$$

Из уравнений (5.131) и (5.132) следует, что БИХ-фильтр, соответствующий системе первого порядка, — это конвольвер, коэффициенты импульсной характеристики которого равны $h(n) = b_0a_1^n$.

КИХ-фильтры используют в обработке речи, для получения кодов РСМ со сниженной шириной полосы, в субполосных кодерах, для параметрического спектрального анализа и в вокодерах с линейным предсказанием [6].

5.5.2.2. Сверточное кодирование

Сверточные коды позволяют исправлять пакеты ошибок путем введения в код цифр контроля четности [13, 15]. Выходы триггеров регистра сдвига обеспечивают задержки и подаются в отводы, после чего соответствующим образом объединяются с использованием сумматоров по модулю 2. В результате получается несколько выходов, которые последовательно считываются в каждом такте (рис. 5.39). Система является причинной и нерекурсивной, производит выход, который зависит от ее предыдущих входов, и сворачивает новые входные данные со своей импульсной характеристикой.

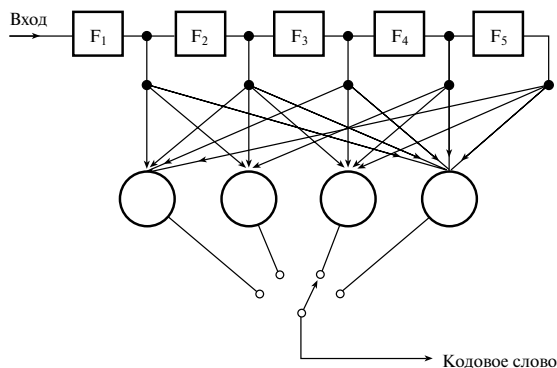


Рис. 5.39. Сверточный кодер

5.5.2.3. Обращение свертки

Вход всех систем сворачивается с импульсной характеристикой системы и может искажать выход. Это происходит, например, в системах телекоммуникаций и может принуждать использовать эквалайзер — линейный фильтр, который по выходу восстанавливает исходный сигнал. Перед тем как разрабатывать подобный фильтр, необходимо определить импульсную характеристику системы (*идентифицировать систему*). Тема идентификации системы подробно рассмотрена в работе [11].

5.5.2.4. Речь

Тема анализа и кодирования речи в сферах взаимодействия человека с машиной и сжатия данных довольно интересна. Иногда при этом используется тот факт, что речевой сигнал можно смоделировать как свертку серии импульсов, представляющих основные тоны, активирующих импульсов и импульсной характеристики речевого тракта [12]. Получающуюся тройную свертку можно легко обратить в форму, подходящую для линейной независимой от времени системы. Более подробно использование КИХ-фильтров в сфере обработки речи освещено в разделе 5.5.2.1.

5.6. Резюме

В данной главе подробно рассмотрены темы корреляции и свертки, а также их взаимосвязь. В рамках темы корреляции изучены процедуры нормировки и подходы, позволяющие избежать краевых эффектов. Описаны эффекты корреляции зашумленных сигналов и идентификация сигналов в шуме с помощью корреляции. Представлены техники быстрой корреляции и свертки, основанные на теоремах о корреляции и свертке и использовании БПФ, также показано, как получить линейную свертку. Выведены быстрые методы наложения-сложения и наложения-записи, позволяющие получать свертку больших последовательностей данных. Кроме того, описаны средства дальнейшего двукратного ускорения вычисления для действительных данных на основе использования действительной и мнимой частей БПФ.

Задачи

5.1. Две последовательности данных равной длины записаны по периодической серии импульсов, которая передана по зашумленному каналу. Мгновенные значения напряжения приведены в табл. 5.4.

1. Определите величину задержки между двумя записями и период сигнала.
2. Определите периодический сигнал.

Таблица 5.4. Мгновенные значения напряжения (вольт) двух разных записей состояния одного канала

Запись 1	6,02	-5,98	7,92	-7,96	-0,78
Запись 2	8,93	-7,20	-0,82	3,23	1,44
	-8,34	9,22	-2,65	-3,7	9,51
	5,43	-9,88	-1,13	0,79	9,83
	5,53	3,50	-3,18	-8,85	8,21
	-8,73	4,64	-8,49	-4,66	-8,84
	1,69	-0,06	6,65	-8,00	-9,21
	5,55	-8,24	-0,37	2,71	4,63
	-0,78	7,27	-5,98	-3,97	9,11
	1,88	-0,92	-5,33	9,01	9,23
	4,23	2,99	-1,85	-5,27	3,81
	-3,7	5,08	-0,72	-5,08	-2,6
	6,62	-2,64	2,08	-5,91	-3,58
	9,67	-8,55	-3,08	4,18	8,11
	-1,65	3,64	-8,19	-3,50	4,84
	0,74	-3,87	-4,09	8,03	6,91
	7,25	2,93	-4,42	-8,21	3,61
	-9,87	-3,62	-8,29	-5,8	-7,04

- 5.2. Определите функции взаимной корреляции записей 1 и 2 (табл. 5.4) с учетом и без учета краевого эффекта. Оцените ошибки, возникшие вследствие краевого эффекта.
- 5.3. Чему равна корреляция (в процентах) между записями 1 и 2 (табл. 5.4) при нулевой задержке? Корреляция в процентах определяется как коэффициент корреляции, умноженный на 100%.
- 5.4. В табл. 5.5 представлены зашумленные значения напряжения. Используя метод взаимной корреляции с шаблонным сигналом, определите точную форму присутствующего периодического сигнала. Проверьте результат с помощью другого метода.

Таблица 5.5. Значения напряжения для зашумленного сигнала

-7,37;	-7,99;	3,31;	-8,59;	-1,68;	3,01;	12,21;	-2,38;	7,46;
-9,84;	1,48;	1,1;	-1,8;	5,48;	8,93;	0;	-9,36;	-10,11;
1,61;	3,36;	-4,86;	6,27					

- 5.5. Рассчитайте автокорреляционную функцию периодического сигнала из примера 5.4 а) численно, б) аналитически. Сравните решения между собой и с автокорреляционной функцией зашумленного сигнала. Объясните отличия от предполагаемых результатов.
- 5.6. Цифровой сигнал выбирается и дискретизируется. Оцифрованные значения напряжения представлены в табл. 5.6. Определите, можно ли считать сигнал случайным. Предполагая, что интервал дискретизации равен 1 мс, и присутствует периодический компонент с периодом 4 мс, оцените и изобразите периодический сигнал.

Таблица 5.6. Оцифрованные значения напряжения

0,92;	−3,71;	3,11;	−0,24;	4,65;	0,84;	−2,98;	−3,94;	−4,03;	−2,51;	0,17;
3,85;	2,58;	0,38;	4,58;	3,4;	−3,46					

- 5.7. Сравните отношения сигнал-шум в следующих случаях:
- 1) зашумленный периодический сигнал — запись 1 в табл. 5.4;
 - 2) автокорреляционная функция записи 1 в табл. 5.4;
 - 3) функция взаимной корреляции записей 1 и 2 в табл. 5.4.
- 5.8. Рассчитайте теоретические отношения сигнал-шум в представленных ниже задачах.
1. Периодический сигнал, полученный по данным записи 1 в табл. 5.4 путем нахождения взаимной корреляции с подходящей серией импульсов.
 2. Автокорреляционная функция записи 1 в табл. 5.4, при условии, что, во-первых, отношение сигнал-шум $(S/N)_{r0}$ автокорреляционной функции зашумленного синусоидального сигнала выражается так:

$$(S/N)_{r0} = \frac{N}{1 + 8/\left(\frac{S_i}{N_i}\right) + 2/\left(\frac{S_i}{N_i}\right)^2},$$

где N — число данных, S_i — мощность сигнала, N_i — мощность шума, и что, во-вторых, отношение сигнал-шум $(S/N)_\delta$ взаимной корреляции зашумленного синусоидального сигнала с серией импульсов, имеющей такой же период, что и сигнал, равно

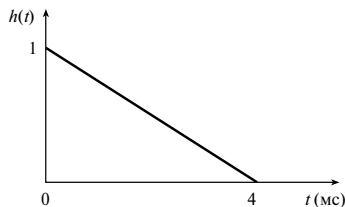
$$(S/N)_\delta = \frac{N}{1 + 1/\left(\frac{S_i}{N_i}\right)}.$$

- 5.9. Сравните результаты, полученные при решении задач 5.7 и 5.8.
- 5.10. Согласованный фильтр, используемый для детектирования поступления на приемник из зашумленного канала соответствующего сигнала, имеет импульсную характеристику $\{1, -1, -1, 1, 1, -1, 1\}$. В табл. 5.7 приведены значения выборок сигнала, представляющие элементы серии биполярных импульсов с амплитудами $\pm 1,5$ В и шириной 1 мкс. Определите время поступления сигнала и значение константы согласованного фильтра.

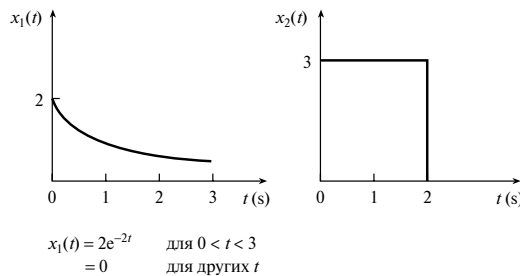
Таблица 5.7. Мгновенные значения напряжения зашумленного биполярного сигнала

t (мкс)	0	1	2	3	4	5	6
Напряжение	0,14	0,48	1,61	2,09	-2,40	0,40	2,35
t (мкс)	7	8	9	10	11	12	13
Напряжение	-0,59	-1,81	0,32	-0,47	1,81	-1,63	-2,28

- 5.11. Найдите импульсную характеристику системы, если ее отклик на псевдослучайную последовательность $\{1; 1; -1; 1; -1; -1; 1; -1\}$ равен $y(n) = \{0; 0; 0,5; 1,5; 1,5; 1,5; 1; -1; -1; -1; -1,5; -0,5; -0,5; -0,5\}$.
- 5.12. Импульс единичной амплитуды подается в цепь, импульсная характеристика которой изображена на рис. 5.40. Численно определите выходной сигнал. Сигналы дискретизируйте с интервалом 0,5 мс.

**Рис. 5.40.** Импульсная характеристика системы (задача 5.12)

- 5.13. 1. На рис. 5.41 показаны две функции $x_1(t)$ и $x_2(t)$. Оцените следующие величины:
- их свертку $x_3(t)$ численно, взяв дискретные значения в моменты времени $t = 0, 1, 2, 3, 4, 5$ с;
 - их свертку $x_3(t)$ аналитически.
2. Изобразите функции $x_3(t)$ и объясните, почему они отличаются.

**Рис. 5.41.** Функции $x_1(t)$ и $x_2(t)$ (задача 5.13)

- 5.14. Определите форму выходного импульса, если на вход однокаскадного RC-фильтра нижних частот с частотой среза 6 МГц подается прямоугольный им-

пульс с амплитудой 5 В и шириной 0,4 мкс. Предполагается, что импульсная характеристика фильтра выражается следующим образом:

$$h(t) = \frac{1}{CR} e^{-t/CR} u(t).$$

- 5.15. Прямоугольный импульс с высотой 5 В и шириной 1,0 мкс подается на вход системы с характеристикой $h(t)$:

$$h(t) = 0,1[1 - e^{-t/(1,09 \times 10^{-6})}] \quad 0 \leq t \leq 10 \text{ мкс} \\ = 0 \quad 10 \text{ мкс} < t < 0$$

Определите выход системы:

- а) аналитически;
 - б) путем дискретизации $h(t)$ с интервалом 1 мкс и представления импульса импульсной функцией, расположенной в точке $t = 0$ с.
- Критически сравните полученные результаты.
- 5.16. Найдите функцию взаимной корреляции двух наборов данных $\{1, 5; 2, 0; 1, 5; 2, 0; 2, 5\}$ и $\{0; 0, 33; 0, 67; 1, 0\}$
- 1) прямым расчетом взаимной корреляции;
 - 2) применяя теорему о корреляции.
- 5.17. Определите выход электрической системы с импульсной характеристикой $\{0; 0, 899; 0, 990; 0, 991; 1\}$ при подаче на вход сигнала $\{0; 2, 5; 5, 0; 0\}$ (вольт)
- 1) прямым расчетом свертки;
 - 2) применяя теорему о свертке.
- 5.18. Используйте метод наложения-сложения для расчета выхода системы с импульсной характеристикой $h(n) = \{0; 0, 899; 0, 990; 1\}$ с входными данными, приведенными в табл. 5.5 (последние два элемента данных игнорируйте). Предполагайте, что данные дискретизируются с интервалом 2,5 мкс, и разделите входные данные на пять сегментов равной длины. Рассчитайте сдвиг фаз между выходом и входом системы. Ответ проверьте, используя метод прямой свертки.
- 5.19. Решите задачу 5.18, используя метод наложения-сложения, в котором свертка вычисляется с использованием теоремы о свертке. Сравните результат с ответом задачи 5.18.
- 5.20. Найдите выход системы, описанной в задаче 5.18, имеющей импульсную характеристику $h(n) = \{0; 0, 899; 0, 990; 0, 999; 1\}$, используя метод наложения-записи и входные данные, приведенные в табл. 5.5 (последние два элемента данных игнорируйте). Результат сравните с ответом задачи 5.18.
- 5.21. Решите задачу 5.20, применяя для расчета сверток теорему о свертке. Сравните результат с ответами задач 5.18–5.20.
- 5.22. Используйте решения задач 5.18–5.21 для сравнения объемов вычислений, необходимых в различных схемах, включая расчет с помощью прямой свертки.

- 5.23. Напишите программу расчета свертки методом наложения-сложения. Используйте ее для проверки ответа задачи 5.18, а также нахождения выходов различных систем с входами по вашему выбору.
- 5.24. 1. Напишите программу расчета быстрой корреляции и используйте ее для расчета быстрой корреляции записей 1 и 2 в табл. 5.4.
2. Исследуйте функции взаимной автокорреляции и автокорреляции некоторых сигналов, таких как меандры, синусоиды, случайный шум и сигналы с различными отношениями сигнал-шум.
3. Сравните относительные возможности методов расчета корреляции и оценки спектра в сфере детектирования сигналов в шуме.

Задачи для решения с помощью MATLAB

- 5.25. Даны последовательные выборки двух дискретных сигналов:

$$x = 4, 2, -1, 3, -2, -6, -5, 4, 5,$$

$$y = -4, 1, 3, 7, 4, -2, -8, -2, 1.$$

1. Используя MATLAB, оцените нормированную и денормированную автокорреляционную функцию каждой последовательности данных.
 2. Вычислите и изобразите оценки смещенной и несмещенной автокорреляционной функции обеих последовательностей.
 3. Вычислите и изобразите оценки нормированной и денормированной функции взаимной корреляции двух последовательностей.
 4. Вычислите и изобразите оценку смещенной и несмещенной функции взаимной корреляции двух последовательностей.
 5. Оцените автокорреляционную функцию или функцию взаимной корреляции (исходя из условия) при нулевой задержке для п. 1–4.
 6. Определите длину функции взаимной корреляции или автокорреляционной функции (исходя из условия) для п. 1–4.
 7. Сравните результаты п. 1–6 и прокомментируйте отличия.
- 5.26. Сравните нормированные функции взаимной корреляции последовательностей x , y и последовательностей y , x из задачи 5.25.
- 5.27. 1. Сгенерируйте 1000-точечную последовательность данных со свойствами случайного белого гауссова шума (используя функцию `randn`).
2. Рассчитайте и изобразите оценки автокорреляционной функции последовательности из п. 1 для первых 30 задержек.
- 5.28. Непрерывный сигнал характеризуется следующим уравнением:

$$x(t) = A \cos(2\pi f_1 t) + B \cos(2\pi f_2 t).$$

1. Сгенерируйте с помощью MATLAB дискретный эквивалент сигнала. Частота дискретизации равна 1 кГц, $f_1 = 50$ Гц, $f_2 = 100$ Гц, а отношение амплитуд частотных компонентов $A/B = 1,5$.
 2. Вычислите и изобразите автокорреляционную функцию последовательности, полученной в п. 1.
- 5.29. 1. С помощью соответствующих функций MATLAB сгенерируйте и изобразите графики следующих сигналов:
- а) синусоида — используйте $\sin(2\pi t/100)$ при $t=0:1:1000$;
 - б) зашумленный сигнал — используйте функцию `randn`;
 - в) зашумленный синусоидальный сигнал — сложите сигналы, полученные в п. 1, а и б;
 - г) прямоугольный сигнал — используйте $\text{square}(2\pi t/100)$.
2. Рассчитайте и изобразите нормированную автокорреляционную функцию всех сигналов, фигурирующих в п. 1.
3. Кратко опишите общие и уникальные свойства автокорреляционных функций, вычисленных в п. 2.
- 5.30. Ниже моделируется задача оценки расстояния до объектов с помощью корреляции. Взаимная корреляция переданного сигнала и зашумленного отраженного сигнала должна иметь максимум при задержке, соответствующей удвоенному искомому расстоянию.
1. С помощью подходящих функций MATLAB сгенерируйте сигналы, изображенные на рис. 1.3, а и б (верхний и нижний следы представляют соответственно переданный и полученный сигналы).
 2. Вычислите функцию взаимной корреляции двух сигналов, оценив таким образом расстояния от передатчика до объекта.
- Предполагается, что радиоволны распространяются со скоростью 3×10^8 м/с, а частота дискретизации равна 4 МГц.
- 5.31. Решите задачу 5.25, используя функцию `xcov`. Прокомментируйте отличия результатов.

Литература

1. Beauchamp K. G. (1973) *Signal Processing Using Analog and Digital Techniques*. London: Allen and Unwin.
2. Bell A. J. and Sejnowski T. J. (1995) An information-maximisation approach to blind separation and blind deconvolution. *Neural Computation*, **7**, 1129–1159.
3. Brigham E. O. (1974) *The Fast Fourier Transform*, Sections 13.3 and 13.4. Englewood Cliffs NJ: Prentice-Hall.
4. Chatfield C. (1980) *The Analysis of Time Series*, p. 62. London: Chapman and Hall.
5. DeFatta D. J., Lucas J. G. and Hodgkiss W. S. (1988) *Digital Signal Processing: A System Design Approach*, Section 6.9, p. 306. New York: Wiley.

6. Grant P. M., Cowan C. F. N., Mulgrew B. and Dripps J. H. (1989) *Analogue and Digital Signal Processing and Coding*, Chapters 16, 17, 19 and 20. Bromley, UK: Chartwell-Bratt.
7. Jenkins G. M. and Watts D.G. (1968) *Spectral Analysis and its Applications*. San Francisco CA: Holden-Day.
8. Jervis B. W., Goude A., Thomlinson M., Mir S. and Miller G. (1990) Least squares artefact removal by transputer. In *IEE Colloquium on the Transputer and Signal Processing*, Savoy Place, London, 5 March 1990.
9. Main G. and Howell T. D. (1993) Determining a signal to noise ratio for an arbitrary data sequence by a time domain analysis. *IEEE Transactions on Magnetics*, **29**(6), November, 3999–4001.
10. McGillem C. D. and Cooper G. R. (1974) *Continuous and Discrete Signal and System Analysis*. New York: Holt, Rinehart, and Winston.
11. Proakis J. G. and Manolakis D. G. (1988) *Introduction to Digital Signal Processing*, p. 429. Basingstoke: Macmillan.
12. Rabiner L. R. and Gold B. (1975) *Theory and Application of Digital Signal Processing*, Chapters 12 and 13. Englewood Cliffs NJ: Prentice-Hall.
13. Stremler F. G. (1982) *Introduction to Communication Systems*, 2nd edn, Section 3.10 and p. 407. Reading MA: Addison-Wesley.
14. Strum R. D. and Kirk D. E. (1988) *First Principles of Discrete Systems and Digital Signal Processing*, Chapter 3. Reading MA: Addison-Wesley.
15. Taub H. and Schilling D. L. (1986) *Principles of Communication Systems*, 2nd edn, p. 562. New York: McGraw-Hill.

Приложение

5.A. Программа на языке C для расчета взаимной корреляции и автокорреляции

Программа на языке C для расчета взаимной корреляции и автокорреляции последовательностей данных приводится на компакт-диске к книге [Ifeachor, 2001] (подробности см. в предисловии).

Схема разработки цифровых фильтров

6

6.1. Введение в цифровые фильтры	355
6.2. Типы цифровых фильтров: КИХ- и БИХ-фильтры	357
6.3. Выбор между КИХ- и БИХ-фильтрами	359
6.4. Этапы разработки фильтра	362
6.5. Примеры	372
6.6. Резюме	377
Задачи	377
Литература	380
Дополнительная литература	380

Цель данной главы — сформулировать общие принципы проектирования цифровых фильтров. Глава содержит простое поэтапное руководство по разработке цифровых фильтров (от спецификации до реализации). Чтобы подчеркнуть основные альтернативы и факторы, влияющие на выбор фильтра, использованы показательные примеры. Стоит сказать, что в большинстве работ по ЦОС значительное место занимает теория цифровых фильтров, особенно методы аппроксимации. Это указывает на значительный объем исследований, проведенных в поисках полезных методов расчета коэффициентов фильтров, и существенные достижения в сфере разработки фильтров. В то же время, при таком изложении материала часто предлагается много несущественной для молодого разработчика информации и не говорится, как действительно стоит провести разработку фильтра. Схема, предложенная в данной главе, поможет не просто понять теоретические основы процесса проектирования фильтра, но и реализовать его практически. Кроме того, данная глава создает необходимый базис для досконально рассмотренной в главах 7 и 8 темы разработки цифровых фильтров.

6.1. Введение в цифровые фильтры

По сути, фильтр — это система или сеть, избирательно меняющая форму сигнала (амплитудно-частотную или фазово-частотную характеристику). Основными целями фильтрации являются улучшение качества сигнала (например, устранение или снижение помех), извлечение из сигналов информации или разделение нескольких сигналов, объединенных ранее для, например, эффективного использования доступного канала связи.

Как будет показано позже, цифровой фильтр — это математический алгоритм, реализованный на аппаратном и/или программном уровне, который с заданной целью действует на входной и генерирует выходной цифровой сигнал. Термином “цифровой

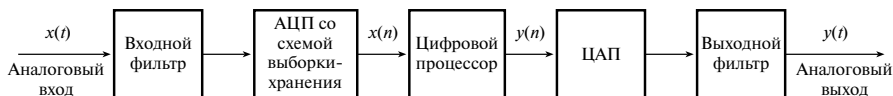


Рис. 6.1. Упрощенная блок-схема цифрового фильтра реального времени с аналоговым входным и выходным сигналами

фильтр” обозначают определенную аппаратную или программную процедуру, реализацию алгоритма фильтрации. В цифровых фильтрах часто используются оцифрованные аналоговые сигналы или просто хранящиеся в памяти компьютера числа, представляющие некоторые переменные.

Упрощенная блок-схема цифрового фильтра реального времени с аналоговым входом и выходом приведена на рис. 6.1. Узкополосный аналоговый сигнал периодически выбирается и конвертируется в набор цифровых выборок, $x(n)$, $n = 0, 1, \dots$. Цифровой процессор производит фильтрацию, отображая входную последовательность $x(n)$ в выходную $y(n)$ согласно вычислительному алгоритму фильтра. ЦАП конвертирует отфильтрованный цифровой образ выход в аналоговые значения, которые затем проходят аналоговую фильтрацию для сглаживания и устранения нежелательных высокочастотных компонентов.

Важную роль в цифровой обработке сигналов играют цифровые фильтры. По сравнению с аналоговыми фильтрами они предпочтительны во множестве областей (например, сжатие данных, биомедицинская обработка сигналов, обработка речи, обработка изображений, передача данных, цифровое аудио, телефонное эхоподавление), так как обладают рядом преимуществ, часть из которых описана ниже.

- Цифровые фильтры могут иметь характеристики, получить которые на аналоговых фильтрах невозможно, например, действительно линейную фазовую характеристику.
- В отличие от аналоговых, производительность цифровых фильтров не зависит от изменений среды, например, от колебаний температуры. Таким образом, цифровые фильтры не требуют периодической калибровки.
- Если фильтр построен с использованием программируемого процессора, его частотная характеристика может настраиваться автоматически (поэтому такие процессоры широко применяются в адаптивных фильтрах).
- Один цифровой фильтр может обрабатывать несколько входных сигналов или каналов без дублирования аппаратных блоков.
- Как фильтрованные, так и нефильтрованные данные можно сохранить для последующего использования.
- Можно легко использовать достижения из области технологий СБИС и получать небольшие цифровые фильтры с пониженной потребляемой мощностью и более низкой ценой.
- На практике точность, которой можно добиться при использовании аналоговых фильтров, ограничена; например, затухание в полосе подавления нельзя поднять выше 60–70 дБ (если использовать стандартные аналоговые компоненты). Точность цифровых фильтров ограничена только используемой длиной слова.
- Производительность цифровых фильтров одинакова для всех устройств серии.

- Цифровые фильтры могут использоваться при очень низких частотах, характерных, например, для многих биомедицинских приложений, где применять аналоговые фильтры непрактично. Кроме того, цифровые фильтры могут использоваться в большом диапазоне частот, для чего достаточно просто менять частоту дискретизации. Впрочем, по сравнению с аналоговыми цифровые фильтры имеют и ряд недостатков.
- *Ограничение скорости.* Максимальная ширина полосы сигналов, которые в реальном времени способны обработать цифровые фильтры, значительно уже, чем у аналоговых фильтров. В приложениях реального времени процесс преобразования “аналоговый-цифровой-аналоговый” вводит ограничение по скорости на производительность цифрового фильтра. Наивысшую частоту дискретизации, с которой может работать фильтр, ограничивает время конвертации АЦП и время установления сигнала ЦАП. Кроме того, скорость работы цифрового фильтра зависит от скорости работы используемого цифрового процессора и числа арифметических операций, которые надлежит выполнить в алгоритме фильтрации, и повышается, когда характеристика фильтра становится более сжатой.
- *Влияние конечной разрядности.* Цифровые фильтры подвержены шуму АЦП, исходящему от квантования непрерывного сигнала, и шуму округления, который вводится при вычислениях. При использовании рекурсивных фильтров высоких порядков накопление шума округления может привести к неустойчивости фильтра.
- *Значительное время разработки и внедрения.* Разработка и внедрение цифровых фильтров, особенно внедрение аппаратного обеспечения, могут выполняться гораздо дольше, чем подобные процедуры для аналоговых фильтров. В то же время, однажды разработанное аппаратное и/или программное обеспечение может использоваться в других задачах цифровой обработки сигналов с незначительной модификацией или вообще без изменений (соответствующие примеры приведены в последующих главах). Если при проектировании цифровых фильтров доступна хорошая компьютерная поддержка, эту задачу будет весьма интересно решать, хотя для того, чтобы эффективно и полно использовать такую поддержку, нужно определенное умение.

6.2. Типы цифровых фильтров: КИХ- и БИХ-фильтры

Цифровые фильтры разделены на два обширных класса: фильтры с бесконечной импульсной характеристикой (БИХ-фильтры) и фильтры с конечной импульсной характеристикой (КИХ-фильтры). Фильтр каждого типа (в стандартной форме) можно представить через коэффициенты его импульсной характеристики $h(k)$ ($k = 0, 1, \dots$), как показано на рис. 6.2. Входной и выходной сигналы фильтра связаны через операцию свертки, данная связь приведена в формуле (6.1) для БИХ-фильтра и в формуле (6.2) для КИХ-фильтра.

$$y(n) = \sum_{k=0}^{\infty} h(k)x(n-k) \quad (6.1)$$

$$y(n) = \sum_{k=0}^{N-1} h(k)x(n-k) \quad (6.2)$$

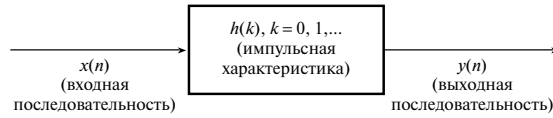


Рис. 6.2. Концептуальное представление цифрового фильтра

Из данных уравнений понятно, что для БИХ-фильтров импульсная характеристика имеет бесконечную длительность, тогда как для КИХ-фильтра она конечна, поскольку $h(k)$ для КИХ-фильтра может принимать всего N значений. На практике вычислить выход БИХ-фильтра с использованием уравнения (6.1) невозможно, поскольку длительность импульсного отклика слишком велика (теоретически — бесконечна). Взамен этого уравнение БИХ-фильтрации переписывается в рекурсивной форме

$$y(n) = \sum_{k=0}^{\infty} h(k)x(n-k) = \sum_{k=0}^N b_k x(n-k) - \sum_{k=1}^M a_k y(n-k), \quad (6.3)$$

где a_k и b_k — коэффициенты фильтра. Таким образом, уравнения (6.2) и (6.3) — это разностные уравнения КИХ- и БИХ-фильтров соответственно. Данные уравнения (и в частности значения $h(k)$ для КИХ-фильтра или a_k и b_k для БИХ-фильтра) используются во многих задачах разработки фильтров. Отметим, что в уравнении (6.3) текущая выходная выборка $y(n)$ является функцией предыдущих выходов, а также текущей и предыдущих входных выборок, так что БИХ-фильтр — это в некотором роде система с обратной связью. Сделав этот вывод, стоит внимательнее присмотреться к уравнению для КИХ-фильтра, в котором текущая выходная выборка $y(n)$ зависит только от прошедших и текущего входных значений. Отметим, впрочем, что если положить все b_k равными нулю, уравнение (6.3) сводится к уравнению (6.2).

В уравнениях (6.4, а и б) приведены альтернативные представления (через передаточные функции) КИХ- и БИХ-фильтров соответственно, причем такие описания очень удобны при оценке частотных характеристик фильтров (подробнее см. главы 4, 7 и 8).

$$H(z) = \sum_{k=0}^{N-1} h(k)z^{-k} \quad (6.4, a)$$

$$H(z) = \sum_{k=0}^N b_k z^{-k} \bigg/ \left(1 + \sum_{k=1}^M a_k z^{-k} \right) \quad (6.4, б)$$

Как будет понятно из следующих разделов, факторы, влияющие на выбор из альтернатив, доступных разработчику цифрового фильтра на каждом этапе процесса проектирования, значительно зависят от того, к какому классу относится фильтр — с конечной или бесконечной импульсной характеристикой. Следовательно, весьма важно различать эти классы, их специфические характеристики и, что еще более важно, понимать, фильтр какого типа выбрать.

6.3. Выбор между КИХ- и БИХ-фильтрами

Выбор между КИХ- и БИХ-фильтрами зависит от относительных преимуществ обоих типов.

1. КИХ-фильтры могут иметь строго линейную фазовую характеристику. Следовательно, фильтр не вводит фазового искажения в сигнал, что важно во многих сферах, например, передаче данных, биомедицине, цифровой аудиообработке или обработке изображений. Фазовая характеристика БИХ-фильтров нелинейна, особенно на краях полос.
2. КИХ-фильтры реализованы нерекурсивно, т.е. (что следует непосредственно из формулы (6.2)) они всегда устойчивы. Гарантировать устойчивость БИХ-фильтров удастся не всегда.
3. Для реализации фильтров используется ограниченное число битов. Практические последствия этого (например, шум округления и ошибки квантования) значительно менее существенны для КИХ-фильтров, чем для БИХ-фильтров.
4. Чтобы получить конечную импульсную характеристику с помощью фильтров с резкими срезами характеристики, потребуется больше коэффициентов, чем для получения бесконечной импульсной характеристики. Следовательно, для реализации предложенной спецификации амплитудной характеристики с КИХ необходимо больше вычислительной мощности и памяти, чем для реализации ее с БИХ. Впрочем, эффективность КИХ-реализаций можно значительно повысить, сыграв на вычислительной скорости БПФ и обработке при нескольких скоростях (см. главу 9).
5. Аналоговые фильтры легко преобразовать в эквивалентные цифровые БИХ-фильтры, удовлетворяющие сходным спецификациям. Для получения КИХ-фильтров такое преобразование невозможно, поскольку для них не существует аналоговых прототипов. Впрочем, получать произвольные частотные характеристики на КИХ-фильтрах легче.
6. Вообще, синтез КИХ-фильтров алгебраически сложнее, если не использовать компьютерную поддержку разработки.
7. БИХ-фильтры рекуррентны. Это означает, что, пропустив через фильтр один и тот же сигнал, но с “обратным ходом времени”, мы получим, вообще говоря, разные результаты. Если для речи временная анизотропия естественна, то, например, для изображений уже нет, поэтому БИХ-фильтры имеют ряд ограничений по применению. Учитывая приведенные соображения, общий принцип выбора между КИХ и БИХ можно сформулировать следующим образом.
 - Использовать БИХ, если единственными важными требованиями являются характеристика с резкими срезами и высокая пропускная способность, поскольку БИХ-фильтры (особенно те, в которых использованы эллиптические характеристики) требуют определения меньшего числа коэффициентов, чем КИХ-фильтры.
 - Использовать КИХ, если число коэффициентов фильтров не очень велико и, в частности, если нужно, чтобы фазовое искажение отсутствовало или было малым. Кроме того, можно добавить, что архитектуры новейших процессоров ЦОС приспособлены к КИХ-фильтрации, мало того, некоторые из них специально разработаны для КИХ-фильтров (см. главу 12).

Пример 6.1

Нижеследующие передаточные функции представляют два различных фильтра, удовлетворяющих идентичным спецификациям амплитудно-частотной характеристики.

1.

$$H(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}},$$

где

$$b_0 = 0,0981819,$$

$$b_1 = 0,9274777,$$

$$b_2 = 0,4981819,$$

$$a_1 = -0,6744878,$$

$$a_2 = -0,3633482.$$

2.

$$H(z) = \sum_{k=0}^{11} h(k) z^{-k},$$

где

$$h(0) = 0,54603280 \times 10^{-2} = h(11),$$

$$h(1) = -0,45068750 \times 10^{-1} = h(10),$$

$$h(2) = 0,69169420 \times 10^{-1} = h(9),$$

$$h(3) = -0,55384370 \times 10^{-1} = h(8),$$

$$h(4) = -0,63428410 \times 10^{-1} = h(7),$$

$$h(5) = 0,57892400 \times 10^0 = h(6).$$

Для каждого фильтра

- а) определите, является он КИХ- или БИХ-фильтром;
- б) представьте операцию фильтрации в виде блок-схемы и запишите соответствующие разностные уравнения;
- в) определите и прокомментируйте вычислительную сложность и требования к памяти.

Решение

6.1, а. Фильтры 1 и 2 являются соответственно БИХ- и КИХ-фильтром.

6.1, б. Блок-схема фильтра 1 представлена на рис. 6.3, а. Соответствующий набор уравнений имеет вид

$$w(n) = x(n) - a_1 w(n-1) - a_2 w(n-2),$$

$$y(n) = b_0 w(n) + b_1 w(n-1) + b_2 w(n-2).$$

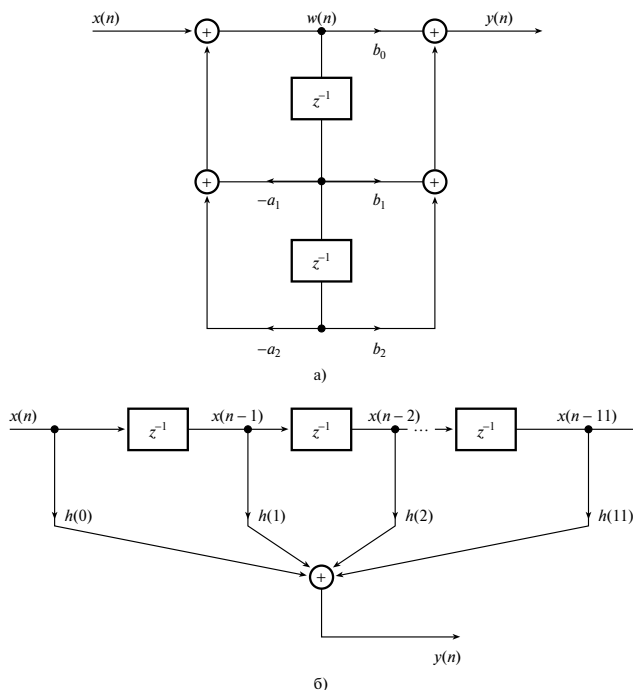


Рис. 6.3. Блок-схема БИХ-фильтра из примера 6.1 (панель а); блок-схема КИХ-фильтра из примера 6.1 (панель б)

Блок-схема фильтра 2 представлена на рис. 6.3, б. Соответствующее разностное уравнение выглядит так:

$$y(n) = \sum_{k=0}^{11} h(k)x(n-k).$$

6.1, в. Изучая два разностных уравнения, получаем следующие требования к объему вычислений и памяти.

	КИХ	БИХ
Число операций умножения	12	5
Число операций сложения	11	4
Количество ячеек памяти (коэффициенты и данные)	24	8

Очевидно, что БИХ-фильтры экономнее, чем КИХ-фильтры, как с точки зрения необходимых вычислений, так и с точки зрения требований к памяти. Впрочем, можно еще дополнительно учесть симметрию коэффициентов КИХ-фильтров и считать более эффективными КИХ-фильтры (правда, только с точки зрения очевидной простоты реализации). Еще один момент, который стоит отметить в связи с полученными результатами: при одинаковых спецификациях амплитудной характеристики число коэффициентов КИХ-фильтра (в данном примере — 12) обычно в шесть раз превышает порядок (наивысшая степень z в знаменателе) передаточной функции БИХ-фильтра (в данном примере — 2).

6.4. Этапы разработки фильтра

Разработка цифрового фильтра проходит в пять этапов.

1. Спецификация требований к фильтру.
2. Вычисление подходящих коэффициентов фильтра.
3. Представление фильтра подходящей структурой.
4. Анализ влияния конечной разрядности на производительность фильтра.
5. Реализация фильтра на программном и/или аппаратном уровне.

Названные пять этапов не всегда независимы; кроме того, они не всегда располагаются в указанном порядке. Фактически существуют методы, которые позволяют объединить второй этап и некоторые аспекты третьего и четвертого. Подход, изложенный в данной главе, — это простое пошаговое руководство, гарантирующее успешную структуру. Чтобы получить эффективный фильтр, иногда приходится проводить данный процесс в несколько итераций, особенно, если спецификации не являются совершенно определенными (как обычно и бывает), или же разработчик желает исследовать альтернативные структуры. Подробное описание названных пяти этапов приводится ниже.

6.4.1. Спецификация требований

Спецификация требований включает спецификации 1) характеристик сигналов (тип источника и получателя сигнала, интерфейс ввода-вывода, скорость передачи данных и ширина полосы, наивысшая частота, представляющая практический интерес); 2) характеристик фильтра (желаемая амплитудная и/или фазовая характеристика и то, насколько данные требования строги, скорость работы и режимы фильтрации (реальное или модельное время)); 3) принципа реализации (например, как компьютерной программы на языке высокого уровня или как системы ЦОС на базе процессора, здесь же выполняется выбор процессора сигналов) и 4) других требований к структуре (например, стоимость фильтра). Разработчик может не иметь достаточно информации, чтобы полностью определить фильтр на начальных этапах, но для упрощения процесса разработки следует сформулировать максимальное количество требований.

Хотя перечисленные выше требования уточняются в зависимости от конкретной задачи, некоторым аспектам п. 2 стоит уделить особое внимание. Характеристики цифровых фильтров часто задаются в частотной области. Для частотно-избирательных фильтров, таких как фильтры нижних частот и полосовые фильтры, спецификации часто формулируются в виде схем допусков. Пример подобной схемы для фильтра нижних частот приведен на рис. 6.4. Заштрихованные горизонтальные линии обозначают пределы допустимых отклонений. В полосе пропускания амплитудная характеристика имеет типовое отклонение δ_p , а в полосе подавления — максимальное отклонение δ_s .

Ширина полосы перехода определяет, насколько резким является характеристика фильтра. В этой области амплитудная характеристика монотонно уменьшается от по-

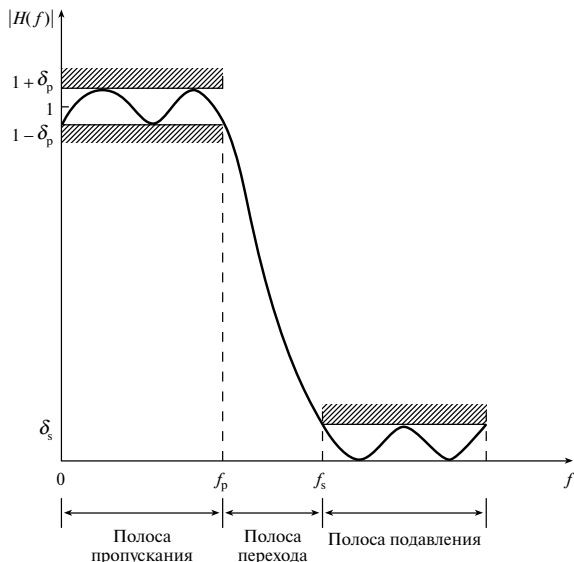


Рис. 6.4. Схема допусков для фильтра нижних частот

лосы пропускания до полосы подавления. Интерес представляют следующие ключевые параметры:

- δ_p — отклонение в полосе пропускания;
- δ_s — отклонение в полосе подавления;
- f_p — граничная частота полосы пропускания;
- f_s — граничная частота полосы подавления.

Граничные частоты часто представляются в нормированной форме, т.е. как доля частоты дискретизации (f/F_s), но часто более содержательными являются спецификации, в которых используются стандартные единицы частоты (герцы или килогерцы). Отклонения в полосе пропускания и полосе подавления могут выражаться как обычные числа или в децибелах, когда они выражают неравномерность в полосе пропускания и затухание в полосе подавления соответственно. Следовательно, минимальное затухание в полосе подавления A_s и максимальная неравномерность в полосе пропускания A_p в децибелах записываются следующим образом (для КИХ-фильтров):

$$A_s(\text{затухание в полосе подавления}) = -20 \lg \delta_s, \quad (6.5, a)$$

$$A_p(\text{неравномерность в полосе пропускания}) = 20 \lg(1 + \delta_p). \quad (6.5, б)$$

При спецификации фазовой характеристики цифровых фильтров во многих случаях достаточно указать, что фазовое искажение существенно или что желательна линейная фазовая характеристика. В то же время, в некоторых приложениях, где фильтры ис-

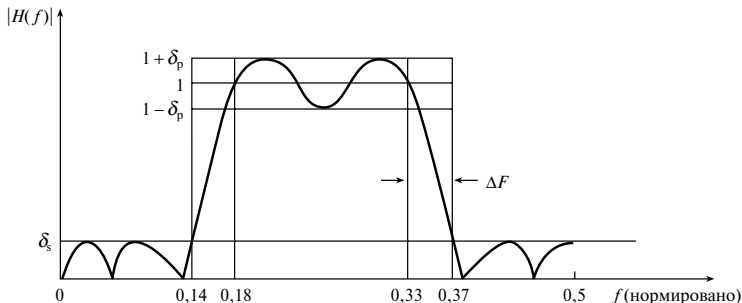


Рис. 6.5. Схема допусков для полосового фильтра из примера 6.2

пользуются для выравнивания или компенсации фазовой характеристики системы или как фазовращатели, фазовую характеристику задавать следует.

Пример 6.2

Требуется спроектировать полосовой КИХ-фильтр, соответствующий следующей спецификации частотной характеристики:

полоса пропускания	0,18–0,33 (нормированная);
ширина полосы перехода	0,04 (нормированная);
отклонение в полосе подавления	0,001;
отклонение в полосе пропускания	0,05.

1. Сформируйте схему допусков для этого фильтра.
2. Выразите граничные частоты полосы пропускания фильтра в стандартных единицах (килоггерцы), предполагая, что частота дискретизации равна 10 кГц, а отклонение в полосе подавления и полосе пропускания выражено в децибелах.

Решение

1. Схема допусков фильтра представлена на рис. 6.5.
2. Граничные частоты полосы пропускания при частоте дискретизации 10 кГц и отклонения в полосе подавления и полосе пропускания представлены ниже.

Полоса пропускания	1,8–3,3 кГц.
Полосы подавления	0–1,4 кГц и 3,7–5 кГц.
Затухание в полосе подавления	$-20 \lg(0,001) = 60$ дБ.
Неравномерность в полосе пропускания	$-20 \lg(1 + 0,05) = 0,42$ дБ.

6.4.2. Расчет коэффициентов

На этом этапе выбирается один из методов аппроксимации и вычисляются значения коэффициентов $h(k)$ (для КИХ-фильтра) или a_k и b_k (для БИХ-фильтра), при которых удовлетворяются условия, принципы определения которых представлены в разделе 6.4.1. Метод вычисления коэффициентов фильтра зависит от того, к какому классу относится фильтр — КИХ или БИХ.

Вычисление коэффициентов БИХ-фильтра традиционно основывается на преобразовании характеристик известных аналоговых фильтров в характеристики эквивалентных цифровых. При этом используются два основных подхода: метод инвариантного преобразования импульсной характеристики и метод билинейного преобразования. При использовании метода инвариантного преобразования импульсной характеристики после оцифровки аналогового фильтра сохраняется импульсная характеристика исходного аналогового фильтра, но не сохраняется амплитудно-частотная характеристика. Вследствие внутреннего наложения данный метод не подходит для фильтров верхних частот или режекторных фильтров. Билинейный метод, с другой стороны, обеспечивает весьма эффективные фильтры и хорошо подходит для вычисления коэффициентов частотно-избирательных фильтров. В результате можно создавать цифровые фильтры с известными классическими характеристиками, такими как в фильтрах Баттерворта, Чебышева или эллиптических. Цифровые фильтры, полученные методом билинейного преобразования, будут, в общем случае, иметь ту же амплитудную характеристику, что и аналоговые, но иные свойства во временной области. В настоящее время доступны эффективные компьютерные программы вычисления коэффициентов фильтров, в которых используется билинейный метод, и в которых требуется только задать нужные параметры фильтра (см. главу 8). Метод инвариантного преобразования импульсной характеристики хорош при моделировании аналоговых систем, но для частотно-избирательных БИХ-фильтров лучше использовать билинейный метод.

В качестве альтернативной схемы вычисления коэффициентов БИХ-фильтров применяется еще метод размещения нулей и полюсов — простой путь вычисления коэффициентов очень простых фильтров. В то же время, для фильтров с хорошей амплитудной характеристикой данный метод использовать не рекомендуется, поскольку в нем фигурирует перебор положений нулей и полюсов.

Коэффициенты КИХ-фильтров также можно вычислить несколькими различными способами. В данной книге подробно рассмотрено три метода: вырезания (взвешивания), частотной выборки и оптимальный (алгоритм Паркса–Мак-Клиллана (Parks–McClellan)). Метод взвешивания предоставляет очень простой и гибкий способ вычисления коэффициентов КИХ-фильтра, но не позволяет разработчику адекватно управлять параметрами фильтра. Самой привлекательной чертой метода частотной выборки является то, что он допускает рекурсивную реализацию КИХ-фильтров, что может быть весьма вычислительно выгодно. В то же время, этому методу недостает гибкости в плане управления или задания параметров фильтров. В настоящее время в промышленности широко используется оптимальный метод (в совокупности с дополняющей его эффективной и простой в использовании программе), который в большинстве случаев дает требуемый КИХ-фильтр. Следовательно, при проектировании таких фильтров вначале стоит попробовать оптимальный метод, если только конкретное приложение не предусматривает использования другого метода.

Подытожим: существует несколько методов вычисления коэффициентов фильтров, наиболее используемые из них перечислены ниже.

- Метод инвариантного преобразования импульсной характеристики (БИХ).
- Билинейное преобразование (БИХ).
- Размещение нулей и полюсов (БИХ).

- Метод взвешивания (КИХ).
- Частотная выборка (КИХ).
- Оптимизационные методы (КИХ).

На выбор метода, наиболее подходящего для конкретной задачи, влияют несколько факторов, в частности, критичные требования в спецификациях. Вообще, основным является выбор между КИХ и БИХ. Если наличие конечной импульсной характеристики существенно, то стоит использовать оптимальный метод, если же желательна бесконечная импульсная характеристика, то в большинстве случаев будет достаточно билинейного метода.

6.4.3. Представление фильтра подходящей структурой

Данный этап включает преобразование данной передаточной функции $H(z)$ в подходящую фильтрующую структуру. Для отражения структуры фильтра часто используются блок-схемы или функциональные схемы, на которых для облегчения реализации цифрового фильтра показывается ход вычислений. Используемая структура зависит от выбора КИХ- или БИХ-фильтра.

Для БИХ-фильтров широко используются три структуры — прямая, каскадная и параллельная формы. Прямая форма — это непосредственное представление передаточной функции БИХ-фильтра. В каскадной форме передаточная функция БИХ-фильтра (формула (6.4, б)) факторизуется и выражается как произведение звеньев второго порядка. В параллельной форме $H(z)$ раскладывается (с использованием элементарных дробей) на сумму звеньев второго порядка. В качестве иллюстрации (а также для сравнения) на рис. 6.6 приведена блок-схема БИХ-фильтра четвертого порядка, представленного в прямой, каскадной и параллельной формах. На рисунке также указан соответствующий набор передаточных функций и разностных уравнений, описывающих структуру фильтра.

При разработке БИХ-фильтров наиболее широко используются параллельная и каскадная структуры, поскольку они предоставляют более простые алгоритмы фильтрации и менее чувствительны к эффектам реализации с использованием конечного числа битов, чем фильтры с прямой структурой. Последние в подобных случаях весьма уязвимы, поэтому их следует избегать любой ценой.

Для КИХ-фильтров, наоборот, наиболее используемой является прямая структура (рис. 6.7, а), поскольку ее проще всего реализовать. В такой форме КИХ-фильтр иногда называется *линией задержки с отводами* (tapped delay line) или трансверсальным фильтром. Кроме того, еще используются две других структуры: структура частотной выборки и схема быстрой свертки (рис. 6.7, б, в). По сравнению с трансверсальной структурой, реализация по схеме частотной выборки может быть вычислительно более эффективной, поскольку она требует расчета меньшего числа коэффициентов. Однако ее бывает не так просто реализовать, и она может требовать больше памяти. При быстрой свертке используются вычислительные преимущества быстрого преобразования Фурье (БПФ), и она особенно привлекательна в ситуациях, когда дополнительно нужно вычислить спектр сигнала.

Существует множество других практических структур цифровых фильтров, но большинство из них популярны только в определенных сферах. Пример — решетчатая структура, которая используется в сферах обработки речи и линейного предсказания. Решетчатая структура может использоваться для представления как КИХ-, так и БИХ-фильтров, и в стандартной форме она характеризуется единственным входом и парой выходов, как показано на рис. 6.8, *а*. Выведенная из нее решетчатая структура, описывающая N -точечный КИХ-фильтр, показана на рис. 6.8, *б*, а структура, предназначенная для БИХ-фильтра второго порядка со всеми заданными полюсами (т.е. такого, для которого указаны только коэффициенты знаменателя), представлена на рис. 6.8, *в*. Более подробно решетчатая структура описана в примере 6.5.

Подытожим. КИХ- и БИХ-фильтры обычно реализуются со следующими структурами:

- трансверсальная (прямая структура) (КИХ);
- частотная выборка (КИХ);
- быстрая свертка (КИХ);
- прямая форма (БИХ);
- каскадная (БИХ);
- параллельная (БИХ);
- решетчатая (КИХ или БИХ).

Для данного фильтра выбор между структурами зависит от следующих факторов: 1) какая требуется характеристика (конечная или бесконечная); 2) простота реализации; 3) насколько структура чувствительна к эффектам конечной разрядности. Более подробно структуры реализации КИХ- и БИХ-фильтров описаны соответственно в главах 7 и 8.

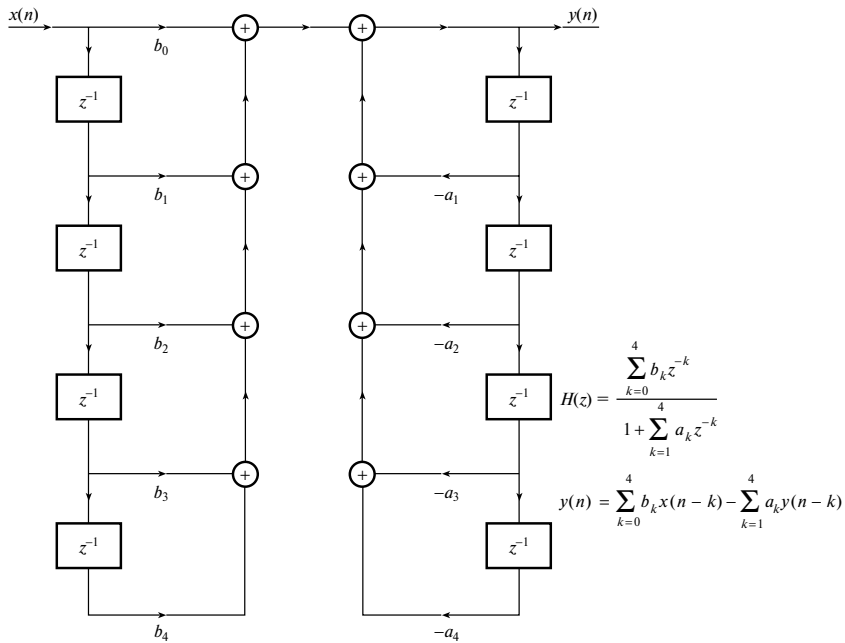
6.4.4. Анализ влияния конечной разрядности

Этапы аппроксимации и реализации предполагают работу с бесконечной или очень высокой точностью. В то же время, в настоящих реализациях часто требуется представить коэффициенты фильтра конечным числом битов (обычно от 8 до 16 бит), кроме того, арифметические операции, указанные в разностных уравнениях, выполняются с использованием арифметики конечной точности.

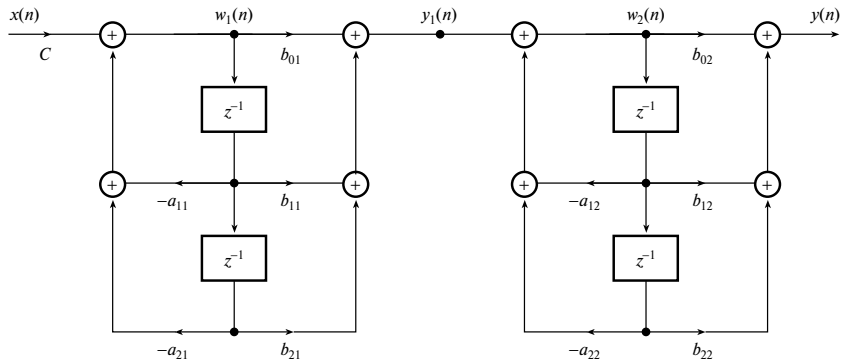
Влияние конечного числа битов проявляется в снижении производительности фильтра, и в некоторых случаях фильтр может стать неустойчивым. Разработчик должен проанализировать данные эффекты и выбрать подходящую длину слова (т.е. число битов) для представления коэффициентов фильтра, переменных фильтра (т.е. входных и выходных выборок) и выполнения арифметических операций в фильтре.

Перечислим основные источники ухудшения производительности фильтра.

- *Квантование сигнала на входе-выходе.* В частности, шум АЦП вследствие квантования входных выборок сигнала — это существенная величина (подробности см. в главе 2).
- *Квантование коэффициентов.* Данный фактор приводит к искажению частотных характеристик КИХ- и БИХ-фильтров и возможной неустойчивости БИХ-фильтров.



а)



$$H(z) = C \prod_{k=1}^2 \frac{1 + b_{1k} z^{-1} + b_{2k} z^{-2}}{1 + a_{1k} z^{-1} + a_{2k} z^{-2}}$$

$$w_1(n) = Cx(n) - a_{11}w_1(n-1) - a_{21}w_1(n-2)$$

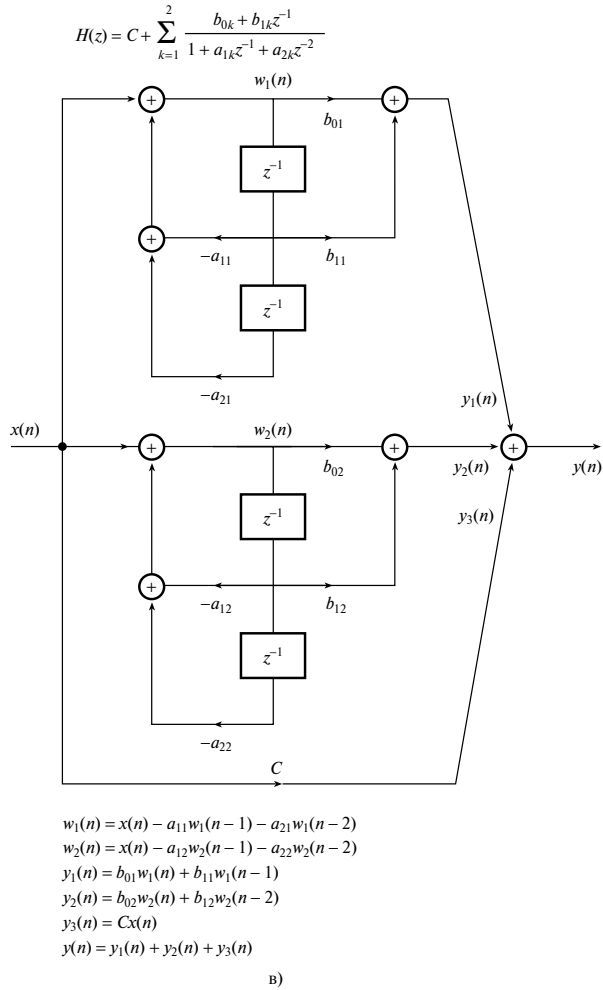
$$y_1(n) = b_{01}w_1(n) + b_{11}w_1(n-1) + b_{21}w_1(n-2)$$

$$w_2(n) = y_1(n) - a_{12}w_2(n-1) - a_{22}w_2(n-2)$$

$$y(n) = b_{02}w_2(n) + b_{12}w_2(n-1) + b_{22}w_2(n-2)$$

б)

Рис. 6.6. Прямая реализация БИХ-фильтра четвертого порядка (панель а). Каскадная реализация БИХ-фильтра четвертого порядка (панель б). Параллельная реализация БИХ-фильтра четвертого порядка (панель в)



- **Ошибки округления.** Использование для фильтрации арифметики конечной точности дает результаты, представление которых требует дополнительных битов. Если результаты квантуются до допустимой длины слова (часто для этого используется округление), возникает шум округления. В результате возможны такие нежелательные следствия, как неустойчивость БИХ-фильтров.
- **Переполнение.** Этот эффект проявляется, когда результат сложения превышает разрешенную длину слова. Это приводит к неверным выходным выборкам и возможной неустойчивости БИХ-фильтров.

Степень ухудшения фильтра зависит от 1) длины слова и типа арифметики, используемой для фильтрации, 2) метода квантования коэффициентов фильтра и переменных до выбранных размеров, 3) структуры фильтра. Зная эти факторы, разработчик может оценить влияние конечной разрядности на производительность фильтра и при необходимости принять меры.

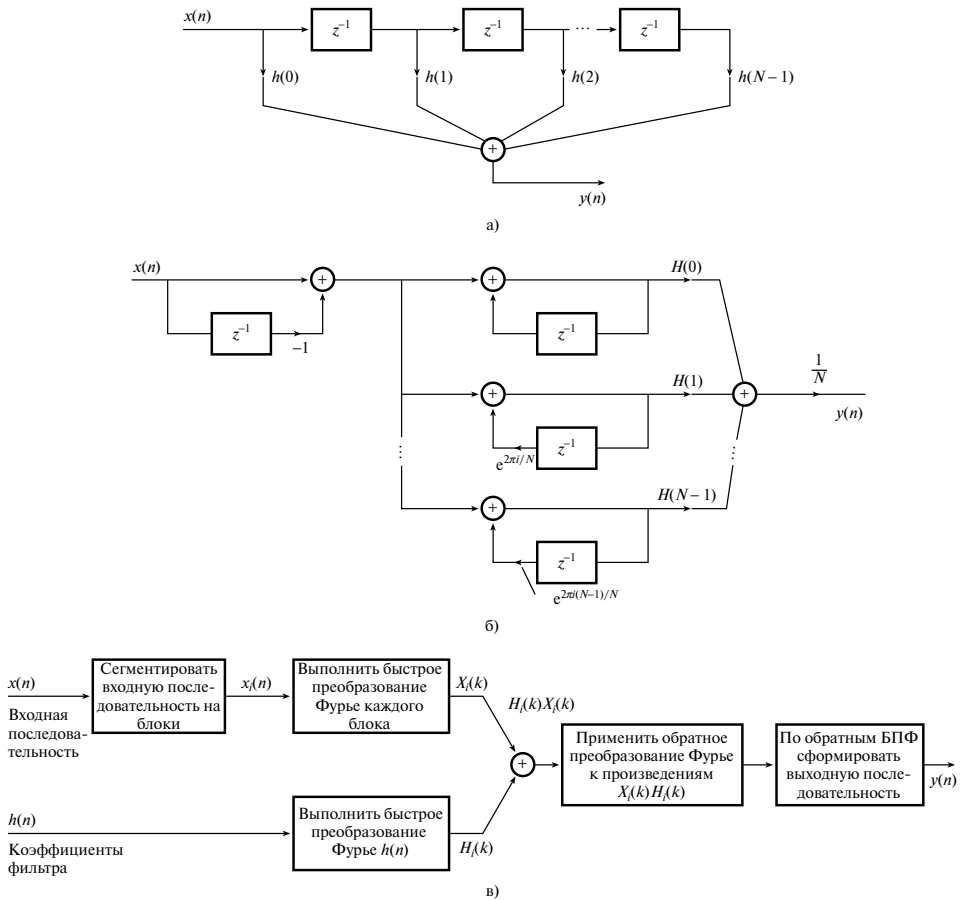


Рис. 6.7. Структуры реализации КИХ-фильтров: а) трансверсальный фильтр (прямая структура); б) реализация частотной выборки; в) схема быстрой свертки

В зависимости от того, как требуется реализовать фильтр, некоторые эффекты могут быть незначительными. Например, если фильтр реализуется как программа на языке высокого уровня, которую будут запускать на мощном компьютере, ошибки квантования коэффициентов и округления несут существенные. При обработке в реальном времени для представления входных и выходных сигналов, коэффициентов фильтра и результатов арифметических операций используются слова конечной длины (обычно 8, 12 и 16 бит). В этих случаях практически всегда необходимо проанализировать воздействие квантования на производительность фильтра.

Подробное обсуждение квантования и его влияния на производительность цифрового фильтра приводится в главе 7 для КИХ-фильтров и в главе 8 для БИХ-фильтров.

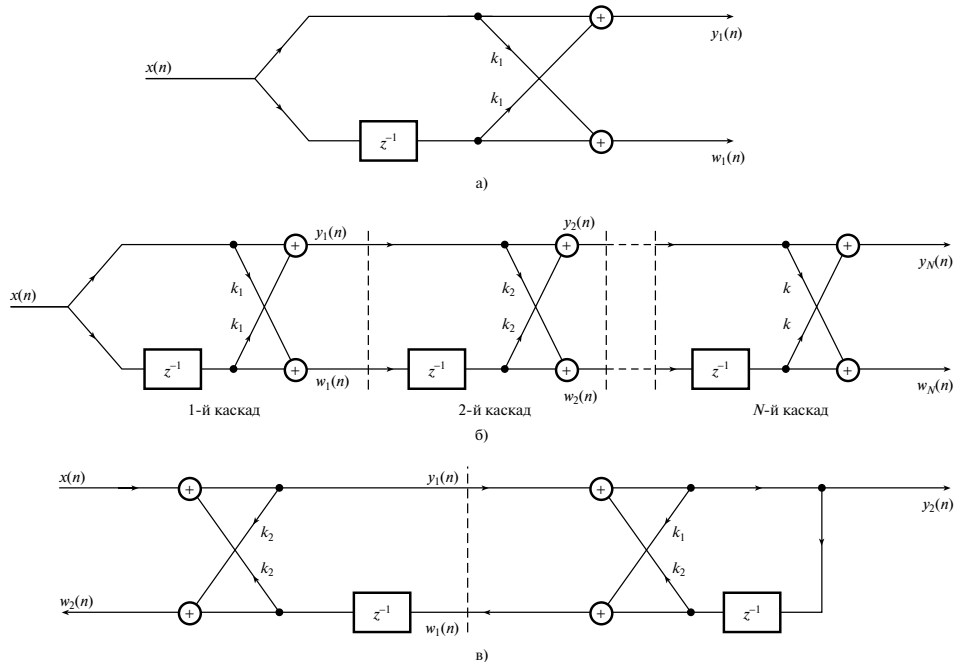


Рис. 6.8. Стандартная решетчатая структура (панель а). N -каскадный КИХ-фильтр с решетчатой структурой (панель б). Двухкаскадная решетчатая структура БИХ-фильтра с заданными всеми полюсами (панель в)

6.4.5. Реализация фильтра

Вычислив коэффициенты фильтра, выбрав подходящую структуру реализации и убедившись, что ухудшение фильтра, вызванное оцифровкой коэффициентов и переменных фильтра в слова выбранной длины, приемлемо, разностное уравнение требуется воплотить на аппаратном или программном уровне. Вне зависимости от выбранного метода, выход фильтра для каждой выборки должен вычисляться согласно разностному уравнению (предполагается реализация во временной области).

Как следует из разностных уравнений (формулы (6.2) и (6.3)), вычисление $y(n)$ (выхода фильтра) включает только умножение, сложение/вычитание и задержку. Следовательно, для реализации фильтра требуются следующие основные составляющие:

- память (например, ПЗУ) для хранения коэффициентов фильтра;
- память (такая как ОЗУ) для хранения текущих и предыдущих входов и выходов, т.е. $\{x(n), x(n-1), \dots\}$ и $\{y(n), y(n-1), \dots\}$;
- аппаратный или программный умножитель (умножители);
- сумматор или схема арифметической логики.

Разработчик обеспечивает данные базовые составляющие и гарантирует, что они соответствующим образом сконфигурированы под нужное приложение. Способ конфигурирования компонентов напрямую связан с тем, в каком времени будет проводиться

обработка (в реальном или модельном (пакетная обработка)). При обработке в модельном времени все данные уже хранятся на каком-то запоминающем устройстве. Такая ситуация, например, характерна для приложений, где нужно получить некоторые экспериментальные данные, а затем их проанализировать. В таких случаях фильтр часто реализуется на языке высокого уровня и запускается на универсальном компьютере, где все основные составляющие уже сконфигурированы. Таким образом, обработку в модельном времени можно описать как исключительно программную реализацию (хотя разработчик может пожелать ввести в процесс дополнительные аппаратные средства для увеличения скорости обработки).

При обработке в реальном времени от фильтра требуется либо 1) работать при наличии входной выборки $x(n)$ и выдавать выходную выборку $y(n)$ перед поступлением следующей входной выборки (т.е. в пределах межвыборочного интервала), либо 2) работать с входным блоком данных (используя, например, метод БПФ) для получения выходного блока в течение времени, пропорционального длине блока. Фильтрация в реальном времени может требовать быстрого и специфического аппаратного обеспечения, если частота дискретизации очень высока или если необходим фильтр высокого порядка. Для большинства приложений работы с аудиочастотами будет достаточно процессоров ЦОС, подобных DSP56000 (Motorola) или TMS320C25 (Texas Instruments), которые, кстати, допускают достаточную гибкость реализации. В данных процессорах уже вмонтированы все требуемые базовые блоки, включая аппаратные умножители. В некоторых приложениях привлекательную альтернативную реализацию можно получить на стандартных 8- или 16-битовых микропроцессорах, таких как процессоры семейств Motorola 6800 или 68000. Помимо блоков обработки сигналов разработчик может также, в зависимости от типа источника и получателя данных, вводить в фильтр подходящие интерфейсы ввода-вывода (например, при аналого-цифровом преобразовании) с цифровым аппаратным обеспечением. Подробное обсуждение реализации КИХ-фильтров представлено в главе 7, а БИХ-фильтров — в главе 8. Аппаратное обеспечение ЦОС описано в главах 12, 13.

6.5. Примеры

Пример 6.3

Рассмотрите пять основных этапов проектирования цифровых фильтров, используя для иллюстрации ответа следующую задачу.

Для подавления в реальном времени шума физиологической природы требуется цифровой фильтр, который должен удовлетворять следующим спецификациям амплитудной характеристики:

полоса пропускания	0–10 Гц;
полоса подавления	20–64 Гц;
частота дискретизации	128 Гц;
максимальное отклонение в полосе пропускания	<0,036 дБ;
затухание в полосе подавления	>30 дБ.

Другие важные требования:

- 1) гармоническая связь компонентов различных внутрисполосных сигналов искажается минимально;
- 2) фильтрация является частью большего процесса, поэтому время, доступное для проведения фильтрации, ограничено;
- 3) фильтр требуется реализовать с использованием процессора ЦОС TMS32010 (Texas Instruments), аналоговый вход оцифровывается до 12 бит.

Решение

Требуемый фильтр разработан и в настоящее время используется в определенном биомедицинском проекте, включающем обработку сигналов. Ниже приводится только краткое обсуждение процесса разработки, подробное исследование откладывается до главы 7, где полностью описаны методы проектирования КИХ-фильтров.

1. *Спецификация требований.* Как рассматривалось выше, разработчик должен указать точное предназначение и требования к рабочим характеристикам фильтра плюс все важные ограничения. В данном примере все это уже сделано.
2. *Вычисление подходящих коэффициентов.* Для того чтобы удовлетворить требованиям минимального искажения и ограниченного времени обработки, следует использовать КИХ-фильтр с линейной фазовой характеристикой, коэффициенты которого получены оптимальным методом.
3. *Выбор структуры фильтра.* Если требуется использовать процессор TMS32010, наиболее эффективной реализации можно добиться с трансверсальной структурой.
4. *Анализ влияния конечной разрядности.* Поскольку будет задействован процессор TMS32010, для эффективности следует использовать арифметику с фиксированной запятой, причем каждый коэффициент представляется 16 бит (после округления). Ухудшение КИХ-фильтра может произойти после квантования входного сигнала и коэффициентов из-за ошибок округления и переполнения, поэтому следует проверить, достаточно ли выбранной длины слова. Анализ влияния конечной длины слова в этом случае показывает, что входной шум квантования и отклонение от частотной характеристики вследствие квантования коэффициентов несущественны. Для уменьшения ошибок округления до несущественного уровня используется 2-битовый накопитель TMS32010, с помощью которого производится суммирование произведений коэффициентов и выборок, причем округляются только окончательные суммы. Чтобы избежать переполнения, каждая величина перед квантованием до 16 бит должна делиться на $\sum_{k=0}^{N-1} |h(k)|$.
5. *Реализация.* На этом этапе нужно спроектировать и сконфигурировать аппаратный модуль на базе TMS32010 (если он еще не готов) с необходимыми интерфейсами ввода-вывода. Кроме того, следует записать для TMS32010 программу обработки протокола ввода-вывода и вычислить выход фильтра $y(n) = \sum_{k=0}^{N-1} h(k)x(n-k)$ для каждого нового входа $x(n)$.

Пример 6.4

Аналоговый фильтр требуется преобразовать в эквивалентный цифровой, который будет работать с частотой 256 Гц. Передаточная функция аналогового фильтра имеет вид

$$H(s) = \frac{1}{s^3 + 2s^2 + 2s + 1}.$$

1. Вычислите подходящие коэффициенты фильтра.
2. Предполагая, что цифровой фильтр будет реализован с использованием каскадной структуры, изобразите подходящую блок-схему реализации и выведите разностные уравнения.
3. Повторите п. 2 для параллельной структуры.

Решение

1. Для сохранения амплитудной характеристики аналогового фильтра при вычислении коэффициентов использовался билинейный метод. Применяя билинейное преобразование к аналоговой передаточной функции (подробности см. в главе 8), получаем следующую передаточную функцию:

$$H(z) = \frac{0,1432(1 + 3z^{-1} + 3z^{-2} + z^{-3})}{1 - 0,1801z^{-1} + 0,3419z^{-2} - 0,0165z^{-3}}.$$

2. При каскадной реализации $H(z)$ факторизуется с использованием элементарных дробей:

$$H(z) = 0,1432 \frac{1 + 2z^{-1} + z^{-2}}{1 - 0,1307z^{-1} + 0,3355z^{-2}} \frac{1 + z^{-1}}{1 - 0,0490z^{-1}}.$$

Представление фильтра в виде блок-схемы дано на рис. 6.9, соответствующий набор разностных уравнений приводится ниже.

$$w_1(n) = 0,1432x(n) + 0,1307w_1(n-1) - 0,3355w_1(n-2),$$

$$y_1(n) = w_1(n) + 2w_1(n-1) + w_1(n-2),$$

$$w_2(n) = y_1(n) + 0,049w_2(n-1),$$

$$y_2(n) = w_2(n) + w_2(n-1).$$

3. При параллельной реализации $H(z)$ выражается через элементарные дроби (подробности см. в главах 4 и 8):

$$H(z) = \frac{1,2916 - 0,08407z^{-1}}{1 - 0,131z^{-1} + 0,3355z^{-2}} + \frac{7,5268}{1 - 0,049z^{-1}} - 8,6753.$$

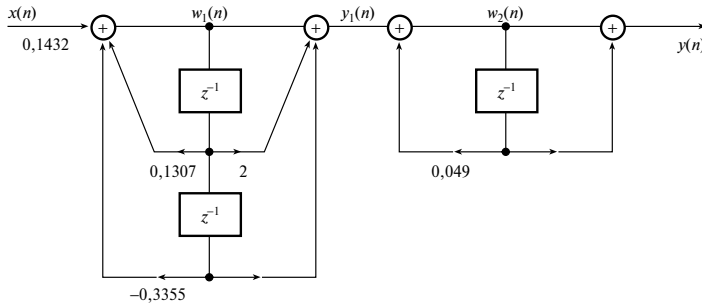


Рис. 6.9

Схема параллельной реализации показана на рис. 6.10, соответствующий набор разностных уравнений приводится ниже.

$$w_1(n) = x(n) + 0,131w_1(n-1) - 0,3355w_1(n-2),$$

$$y_1(n) = 1,2916w_1(n) - 0,08407w_1(n-1),$$

$$w_2(n) = x(n) + 0,049w_2(n-1),$$

$$y_2(n) = 7,5268w_2(n),$$

$$y_3(n) = -8,6753x(n),$$

$$y(n) = y_1(n) + y_2(n) + y_3(n).$$

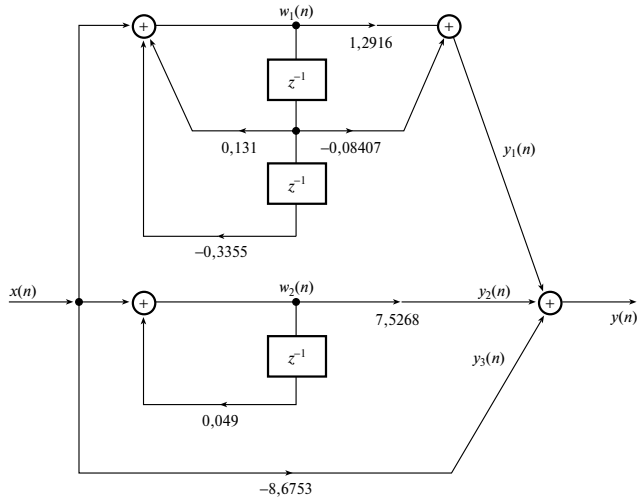


Рис. 6.10

Пример 6.5

Передаточная функция КИХ-фильтра записывается следующим образом:

$$H(z) = 1 - 1,3435z^{-1} + 0,9025z^{-2}.$$

Изобразите блок-схему реализации для каждого из следующих случаев:

- 1) трансверсальная структура;
- 2) двухкаскадная решетчатая структура.

Для решетчатой структуры вычислите значения коэффициентов.

Решение

1. С помощью передаточной функции получаем схему трансверсальной структуры, представленную на рис. 6.11. Вход и выход трансверсальной структуры связаны следующим соотношением:

$$y(n) = x(n) + h(1)x(n-1) + h(2)x(n-2). \quad (6.6)$$

2. Двухкаскадная решетчатая структура фильтра представлена на рис. 6.12. Выход структуры следующим образом связан со входом:

$$\begin{aligned} y_2(n) &= y_1(n) + k_2 w_1(n-1) = \\ &= x(n) + k_1(1 + k_2)x(n-1) + k_2 x(n-2), \end{aligned} \quad (6.7)$$

$$w_2(n) = k_2 x(n) + k_1(1 + k_2)x(n-1) + x(n-2).$$

Сравнивая формулы (6.6) и (6.7, а) и приравнявая коэффициенты, получаем

$$k_1 = \frac{h(1)}{1 + h(2)}, \quad k_2 = h(2).$$

Следовательно, $k_2 = 0,9025$ и $k_1 = -1,3435/(1 + 0,9025) = -0,7062$.

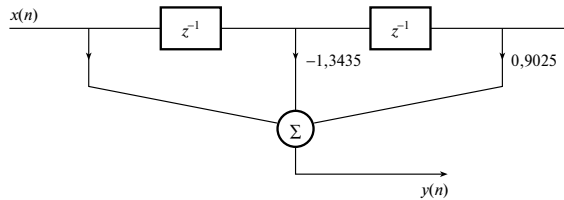


Рис. 6.11

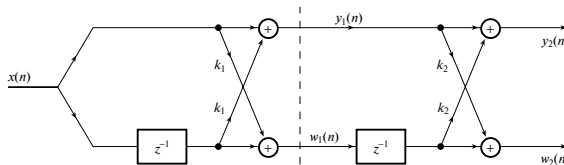


Рис. 6.12

Обратите внимание на то, что коэффициенты функций $y_2(n)$ и $w_2(n)$ (уравнения (6.7, а и б)) идентичны, за исключением того, что один записан в обратном порядке. Данная особенность характерна для КИХ-фильтров с решетчатой структурой. Подробно решетчатая структура, в том числе рекурсивные схемы преобразования коэффициентов КИХ- и БИХ-фильтров в коэффициенты эквивалентных решетчатых структур, описана в различных книгах (например, [1]).

6.6. Резюме

Термином *цифровой фильтр* называют аппаратную или программную реализацию математического алгоритма, входом которого является цифровой сигнал, а выходом — другой цифровой сигнал, форма которого и/или амплитудная и фазовая характеристики специальным образом модифицированы. Во многих приложениях цифровые фильтры предпочтительнее аналоговых, поскольку они позволяют более точно воплотить амплитудные и фазовые спецификации. Кроме того, для цифровых фильтров нехарактерно присущее аналоговым фильтрам изменение характеристик в зависимости от температуры и напряжения.

В данной главе представлена общая схема разработки КИХ- и БИХ-фильтров от спецификации до реализации. Простая пошаговая процедура разработки подобных фильтров включает пять этапов: 1) спецификация; 2) вычисление подходящих коэффициентов; 3) реализация фильтра с использованием подходящей структуры; 4) квантование коэффициентов и переменных фильтра в слова подходящего размера и анализ возникших вследствие этого ошибок; 5) реализация полученного результата посредством аппаратного или программного кодирования фильтра на процессоре, который в действительности и будет выполнять фильтрацию входных данных.

Задачи

- 6.1. Предположим, можно использовать любой из шести методов вычисления коэффициентов, описанных в разделе 6.4.2. Укажите, какой из этих методов следует использовать в каждом из перечисленных ниже случаев. Ответ аргументируйте.
 1. Выравнивание фазы (фазовой задержки) в цифровой системе связи.
 2. Моделирование аналоговой системы.
 3. Система подавления шума с высокой пропускной способностью, требующая фильтр с резкой амплитудной характеристикой.
 4. Обработка изображений.
 5. Высококачественная цифровая аудиообработка.
 6. Биомедицинская обработка сигналов в реальном времени с минимальным искажением.
- 6.2. Ниже записаны две передаточные функции, представляющие два различных фильтра, удовлетворяющие идентичным спецификациям амплитудно-частотной характеристики.

1.

$$H(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} \times \frac{b_3 + b_4 z^{-1} + b_5 z^{-2}}{1 + a_3 z^{-1} + a_4 z^{-2}},$$

где

$$b_0 = 3,136362 \times 10^{-1};$$

$$b_1 = -5,456657 \times 10^{-2};$$

$$b_2 = 4,635728 \times 10^{-1};$$

$$b_3 = -5,456657 \times 10^{-2};$$

$$b_4 = 3,136362 \times 10^{-1};$$

$$b_5 = 4,635728 \times 10^{-1};$$

$$a_1 = -8,118702 \times 10^{-1};$$

$$a_2 = 3,339288 \times 10^{-1};$$

$$a_3 = -2,794577 \times 10^{-1};$$

$$a_4 = 3,030631 \times 10^{-1}.$$

2.

$$H(z) = \sum_{k=0}^{22} h_k z^{-k},$$

где

$$h_0 = 0,39826480 \times 10^{-1} = h_{22};$$

$$h_1 = -0,16874380 \times 10^{-1} = h_{21};$$

$$h_2 = 0,34781130 \times 10^{-1} = h_{20};$$

$$h_3 = 0,12052890 \times 10^{-1} = h_{19};$$

$$h_4 = -0,44731860 \times 10^{-1} = h_{18};$$

$$h_5 = 0,27894610 \times 10^{-1} = h_{17};$$

$$h_6 = -0,87573360 \times 10^{-1} = h_{16};$$

$$h_7 = -0,90972060 \times 10^{-1} = h_{15};$$

$$h_8 = -0,15667550 \times 10^{-1} = h_{14};$$

$$h_9 = -0,28499560 \times 10^0 = h_{13};$$

$$h_{10} = 0,74035030 \times 10^{-1} = h_{12};$$

Для каждого фильтра $h_{11} = 0,62349560 \times 10^0.$

- а) определите, конечна или бесконечна его импульсная характеристика;
- б) представьте действие фильтра в форме блок-схемы и запишите разностное уравнение;
- в) определите и прокомментируйте требования к вычислениям и памяти.

- 6.3. Нужен цифровой фильтр, устраняющий сетевой компонент с электрокардиограммы (ЭКГ) плода, хранящейся в памяти мэйнфрейма. Данные оцифрованы до 12-битовой точности.

В спецификации фильтра указаны такие требования:

затухание на частоте питающей сети	> 50 дБ;
неравномерность в полосе пропускания	< 0,05 дБ;
границы полосы пропускания	0–0,09 и 0,11–0,5 (нормировано);
частота дискретизации	500 Гц.

Фильтр требуется реализовать как программу на языке высокого уровня, которую можно вызывать из основной программы анализа. Любое искажение сигнала со стороны фильтра должно минимизироваться, поскольку важные сигналы ЭКГ легко разрушаются.

Тщательно рассмотрите все вопросы, возникающие при разработке подобного фильтра, уделив внимание альтернативам, доступным разработчику, и предложите обоснованные рекомендации.

- 6.4. Требуется цифровой фильтр для предварительной обработки электрокардиограммы плода, т.е. электрической активности сердца, чтобы было легче детектировать сердцебиение плода при наличии дрейфа постоянной составляющей (baseline wander), интерференции с сигналом питающей сети, маточных сокращений и движения ребенка или матери. Колебание базовой составляющей (baseline sways) и артефакты движения обычно проявляются в диапазоне частот 0–10 Гц, тогда как интерференция с сигналом питающей сети дает сигнал, центрированный на частоте около 50 или 60 Гц (в зависимости от страны). Наибольшая энергия ЭКГ, необходимая для детектирования сердцебиения, предположительно приходится на промежуток 5–50 Гц.

Предположим, что доступны данные электрокардиограммы плода, прошедшие аналоговый полосовой фильтр (полоса — от 0,05 до 100 Гц) и оцифрованные (500 выборок/с) с разрешением 8 бит.

1. Если требуется использовать БИХ-фильтр, каким будет набор спецификаций цифрового фильтра? Ответ аргументируйте.
2. Повторите п. 2 для КИХ-фильтра.

Какой из двух фильтров лучше в данном случае? Почему?

- 6.5. Для выполнения значительной фильтрации, предотвращающей наложение в определенной системе передачи речи, требуется цифровой фильтр, который был бы лучше используемого активного фильтра. В текущий момент аналоговый входной сигнал системы дискретизируется с частотой 8 кГц после прохождения через активный полосовой фильтр со следующими спецификациями:

полоса пропускания	0–3,4 кГц;
граничная частота полосы подавления	8 кГц;
затухание в полосе подавления	30 дБ;
затухание на 4 кГц	14 дБ;
неравномерность в полосе пропускания	<0,1 дБ.

Определите с помощью блок-схемы, как можно удовлетворить спецификациям при использовании цифровой фильтрации. Задайте тип цифрового фильтра (выбор аргументируйте) и его характеристики.

- 6.6. От системы связи требуется следующее: восстановить тактовую частоту по зашумленным данным, которые получены с удаленного передатчика, используя для этого методы цифровой фильтрации, причем данные должны извлекаться достоверно. Известно, что тактовая частота удаленного передатчика равна 2,048 МГц. Обсудите характеристики подходящего цифрового фильтра и задайте его передаточную функцию.
- 6.7. Коэффициенты решетчатого КИХ-фильтра равны: $k_1 = -0,266$ и $k_2 = 0,69$. Изобразите диаграмму реализации данного решетчатого фильтра. Вычислите коэффициенты импульсной характеристики данного фильтра и изобразите на диаграмме эквивалентную трансверсальную структуру.
- 6.8. Цифровой БИХ-фильтр второго порядка характеризуется следующей передаточной функцией:

$$H(z) = \frac{1}{1 - 0,9z^{-1} + 0,81z^{-2}}.$$

Изобразите диаграмму реализации каждой из перечисленных структур:

- 1) прямая;
- 2) решетчатая.

Найдите по данной передаточной функции коэффициенты решетчатой структуры.

Литература

1. Proakis J. G. and Manolakis D. G. (1992) *Digital Signal Processing*, 2nd edn. New York: Macmillan.

Дополнительная литература

- DeFatta D. J., Lucas J. G. and Hodgkiss W. S. (1988) *Digital Signal Processing*. New York: Wiley.
- Elliott D. F. (ed.) (1987) *Handbook of Digital Signal Processing*. London: Academic Press.
- Oppenheim A. V. and Schaffer R. W. (1975) *Digital Signal Processing*. Englewood Cliffs NJ: Prentice-Hall.
- Parks T. W. and Burrus C. S. (1987) *Digital Filter Design*. New York: Wiley.
- Rabiner L. R. and Gold B. (1975) *Theory and Application of Digital Signal Processing*. Englewood Cliffs NJ: Prentice-Hall.
- Rabiner L. R., Cooley J. W., Helms H. D., Jackson L. B., Kaiser J. F., Rader C. M., Schaffer R. W., Steiglitz K. and Weinstein C. J. (1972) Terminology in digital signal processing. *IEEE Trans. Audio Electroacoustics*, **20** (December), 322–337.
- Taylor F. J. (1983) *Digital Filter Design Handbook*. New York: Dekker.

Разработка фильтров с конечной импульсной характеристикой (КИХ-фильтров)

7.1. Вступление	382
7.2. Разработка КИХ-фильтров	387
7.3. Спецификации КИХ-фильтра	388
7.4. Методы расчета коэффициентов КИХ-фильтров	389
7.5. Метод взвешивания	390
7.6. Оптимизационные методы	405
7.7. Метод частотной выборки	418
7.8. Сравнение метода вырезания, оптимального метода и метода частотной выборки	437
7.9. Разработка КИХ-фильтров — специальные вопросы	439
7.10. Структуры реализаций КИХ-фильтров	445
7.11. Влияние конечной разрядности на цифровые КИХ-фильтры	450
7.12. Пример разработки	460
7.13. Резюме	464
7.14. Примеры применения КИХ-фильтров	464
Задачи	465
Литература	480
Дополнительная литература	480
Приложения	481

В данной главе рассматривается разработка КИХ-фильтров от спецификации и вычисления коэффициентов до анализа влияния конечной разрядности и реализации. Приводится несколько подробных примеров, призванных проиллюстрировать различные этапы разработки и подчеркнуть важные концепции. Разработка фильтра рассмотрена полностью, чтобы показать, как связаны различные этапы, и помочь читателям, желающим самостоятельно разрабатывать фильтры. Упомянутые в тексте программы разработки фильтров с помощью MATLAB можно найти в Web (подробности см. в предисловии к книге), кроме того, указана литература, в которой описаны программы, с помощью которых можно повторить представленные ниже выводы или разработать фильтры, необходимые конкретному пользователю.

7.1. Вступление

Перед тем как приступить к разработке КИХ-фильтров, напомним важные их характеристики.

7.1.1. Резюме по ключевым особенностям КИХ-фильтров

1. Стандартный КИХ-фильтр характеризуется следующими уравнениями:

$$y(n) = \sum_{k=0}^{N-1} h(k)x(n-k), \quad (7.1, a)$$

$$H(z) = \sum_{k=0}^{N-1} h(k)z^{-k}, \quad (7.1, б)$$

где $h(k)$, $k = 0, 1, \dots, N-1$ — коэффициенты импульсной характеристики, $H(z)$ — передаточная функция фильтра, а N — длина фильтра, т.е. число коэффициентов фильтра. Формула (7.1, а) — это разностное уравнение КИХ-фильтра. Данное уравнение записано во временной области и представляет КИХ-фильтр в его нерекурсивной форме: текущая выходная выборка $y(n)$ — это функция только текущего и прошедших значений на входе $x(n)$. При реализации КИХ-фильтров в такой форме, т.е. через прямое представление формулы (7.1, а), фильтры всегда устойчивы. В формуле (7.1, б) записана передаточная функция фильтра, нужная для анализа фильтра, например, расчета частотной характеристики.

2. КИХ-фильтры могут иметь точную линейную фазовую характеристику (см. раздел 7.1.2).
3. КИХ-фильтры очень просто реализовать. Архитектура всех существующих процессоров ЦОС подходит для фильтрации с конечной импульсной характеристикой. Кроме того, нерекурсивные КИХ-фильтры менее подвержены эффектам конечной разрядности, чем БИХ-фильтры. Существуют также рекурсивные КИХ-фильтры, использование которых иногда вычислительно выгоднее (подробности см. в разделе 7.7).

Итак, КИХ-фильтры стоит использовать, если на передний план выходят описанные выше преимущества, в частности, возможность получения линейной фазовой характеристики. Вопросы выбора между КИХ- и БИХ-фильтрами рассмотрены в разделе 6.3.

7.1.2. Линейная фазовая характеристика и ее следствия

Одним из важнейших свойств КИХ-фильтров является возможность получения точной линейной фазовой характеристики. По этой причине остановимся на данном свойстве подробнее. При прохождении сигнала через фильтр модификации подвергается амплитуда и/или фаза данного сигнала. Природа и величина изменения сигнала зависят

от амплитудной и фазовой характеристик фильтра. Удобной мерой модификации фазовой характеристики сигнала является фазовая или групповая задержка фильтра. Если сигнал состоит из нескольких частотных компонентов (например, речевой или модулированный сигнал), фазовая задержка фильтра — это величина временной задержки, которую испытывает каждый частотный компонент сигнала при прохождении через фильтр. Групповая задержка — это средняя временная задержка составного сигнала. Математически фазовая задержка равна минус углу сдвига фазы, деленному на частоту, тогда как групповая задержка — это минус производная фазы по частоте:

$$T_p = -\theta(\omega)/\omega, \quad (7.2, a)$$

$$T_g = -d\theta(\omega)/d\omega. \quad (7.2, б)$$

Фильтр с нелинейной фазовой характеристикой будет искажать фазу сигнала, проходящего через этот фильтр. Это объясняется тем, что частотные компоненты сигнала будут задерживаться на величину, не пропорциональную частоте, нарушая таким образом их гармоническую связь. Подобное нарушение нежелательно во многих ситуациях (например, передаче данных, воспроизведении музыки и видео, биомедицине) и его можно избежать, используя фильтры с линейными фазовыми характеристиками в рабочем диапазоне частот.

Говорят, что фильтр имеет линейную фазовую характеристику, если выполняется одно из следующих соотношений:

$$\theta(\omega) = -\alpha\omega, \quad (7.3, a)$$

$$\theta(\omega) = \beta - \alpha\omega, \quad (7.3, б)$$

где α и β — константы. Если фильтр удовлетворяет условию (7.3, а), у него постоянны групповая и фазовая задержки. Можно показать, что условие (7.3, а) удовлетворяется, если импульсная характеристика фильтра обладает положительной симметрией. Фазовая характеристика в этом случае является просто функцией длины фильтра:

$$h(n) = h(N - n - 1), \quad \begin{cases} n = 0, 1, \dots, (N - 1)/2 & (N - \text{нечетное}) \\ n = 0, 1, \dots, (N/2) - 1 & (N - \text{четное}) \end{cases}$$

$$\alpha = (N - 1)/2$$

Условие (7.3, б) удовлетворяется, когда фильтр имеет только постоянную групповую задержку. В этом случае импульсная характеристика фильтра имеет отрицательную симметрию:

$$h(n) = -h(N - n - 1),$$

$$\alpha = (N - 1)/2,$$

$$\beta = \pi/2.$$

КИХ-фильтры с линейной фазовой характеристикой формируют важный класс в семействе КИХ-фильтров. Они обладают уникальным набором свойств, влияющих на разработку и реализацию данных фильтров. Продемонстрируем некоторые из этих свойств на примере.

Пример 7.1

1. Кратко опишите условия, необходимые для того, чтобы реальный цифровой фильтр имел линейную фазовую характеристику, и преимущества фильтров с такой характеристикой.
2. Цифровой КИХ-фильтр имеет фазовую характеристику $h(n)$, которая определена в интервале $0 \leq n \leq N - 1$. Покажите, что при $N = 7$ и $h(n)$, удовлетворяющем условию симметрии

$$h(n) = h(N - n - 1),$$

фильтр имеет линейную фазовую характеристику.

3. Повторите п. 2 при $N = 8$.

Решение

1. Необходимым и достаточным условием того, что фильтр имеет линейную фазовую характеристику, является симметрия его импульсной характеристики (согласно [14]):

$$h(n) = h(N - 1 - n) \text{ или } h(n) = -h(N - 1 - n).$$

Для нерекурсивных КИХ-фильтров память для хранения коэффициентов и число арифметических операций меньше примерно вдвое. Для рекурсивных КИХ-фильтров коэффициенты можно выбрать так, чтобы они были простыми целыми числами, что повышает скорость обработки. В фильтрах с линейной фазовой характеристикой все частотные компоненты при прохождении через фильтр задерживаются на одно и то же время, так что в этом случае фазовое искажение отсутствует.

2. Используя условие симметрии, находим, что при $N = 7$

$$h(0) = h(6); h(1) = h(5); h(2) = h(4).$$

Подставим $z = e^{i\omega T}$ в формулу (7.1, б) и получим частотную характеристику фильтра $H(\omega)$:

$$\begin{aligned} H(\omega) &= H(e^{i\omega T}) = \\ &= \sum_{k=0}^6 h(k)e^{-ik\omega T} = \\ &= h(0) + h(1)e^{-i\omega T} + h(2)e^{-2i\omega T} + \\ &\quad + h(3)e^{-3i\omega T} + h(4)e^{-4i\omega T} + h(5)e^{-5i\omega T} + h(6)e^{-6i\omega T} = \\ &= e^{-3i\omega T} [h(0)e^{3i\omega T} + h(1)e^{2i\omega T} + h(2)e^{i\omega T} + h(3) + h(4)e^{-i\omega T} + \\ &\quad + h(5)e^{-2i\omega T} + h(6)e^{-3i\omega T}]. \end{aligned}$$

Используя условие симметрии, можно сгруппировать члены, коэффициенты которых численно равны:

$$\begin{aligned} H(\omega) &= e^{-3i\omega T} [h(0)(e^{3i\omega T} + e^{-3i\omega T}) + h(1)(e^{2i\omega T} + e^{-2i\omega T}) + h(2)(e^{i\omega T} + e^{-i\omega T})] = \\ &= e^{-3i\omega T} [2h(0) \cos(3\omega T) + 2h(1) \cos(\omega T) + 2h(2) \cos(\omega T)]. \end{aligned}$$

Если положить $a(0) = h(3)$ и $a(k) = 2h(3 - k)$, $k = 1, 2, 3$, то $H(\omega)$ можно записать в компактной форме:

$$H(\omega) = \sum_{k=0}^3 a(k) \cos(n\omega T) e^{-3i\omega T} = \pm |H(\omega)| e^{i\theta(\omega)},$$

где

$$\pm |H(\omega)| = \sum_{k=0}^3 a(k) \cos(n\omega T); \theta(\omega) = -3\omega T.$$

Очевидно, что фазовая характеристика линейна.

3. В данном случае из условия симметрии следует

$$h(0) = h(7); h(1) = h(6); h(2) = h(5); h(3) = h(4).$$

Используя схему рассуждений, подобную приведенной выше, и условие симметрии, получаем:

$$\begin{aligned} H(\omega) &= e^{-7i\omega T/2} [h(0)(e^{7i\omega T/2} + e^{-7i\omega T/2}) + h(1)(e^{5i\omega T/2} + e^{-5i\omega T/2}) + \\ &+ h(2)(e^{3i\omega T/2} + e^{-3i\omega T/2}) + h(3)(e^{i\omega T/2} + e^{-i\omega T/2})] = \\ &= e^{-7i\omega T/2} [2h(0) \cos(7\omega T/2) + \\ &+ 2h(1) \cos(5\omega T/2) + 2h(2) \cos(3\omega T/2) + 2h(3) \cos(\omega T/2)] = \\ &= \pm |H(\omega)| e^{i\theta(\omega)}, \end{aligned}$$

где

$$\begin{aligned} \pm |H(\omega)| &= \sum_{k=1}^4 b(k) \cos[\omega T(k - 1/2)]; \quad \theta(\omega) = -(7/2)\omega T. \\ b(k) &= 2h(N/2 - k), k = 1, 2, \dots, N/2. \end{aligned}$$

Результаты, представленные выше, можно распространить на все КИХ-фильтры (табл. 7.1).

7.1.3. Типы КИХ-фильтров с линейной фазовой характеристикой

Существует четыре типа КИХ-фильтров с линейной фазовой характеристикой, отличающихся четностью N и типом симметрии $h(n)$ (положительная или отрицательная). Два из четырех типов фильтров с линейной фазовой характеристикой рассмотрены в примере выше. На рис. 7.1 показано, как отличаются импульсные характеристики

Таблица 7.1. Ключевые различия четырех типов КИХ-фильтров с линейной фазовой характеристикой

Симметрия импульсной характеристики	Число коэффициентов N	Частотная характеристика $H(\omega)$	Тип линейной фазовой характеристики
Положительная симметрия, $h(n) = h(N - 1 - n)$	нечетное	$e^{-i\omega(N-1)/2} \sum_{n=0}^{(N-1)/2} a(n) \cos(\omega n)$	1
	четное	$e^{-i\omega(N-1)/2} \sum_{n=1}^{N/2} b(n) \cos(\omega(n - \frac{1}{2}))$	2
Отрицательная симметрия, $h(n) = -h(N - 1 - n)$	нечетное	$e^{-i[\omega(N-1)/2 - \pi/2]} \sum_{n=1}^{(N-1)/2} a(n) \sin(\omega n)$	3
	четное	$e^{-i[\omega(N-1)/2 - \pi/2]} \sum_{n=1}^{N/2} a(n) \sin(\omega(n - \frac{1}{2}))$	4

$$a(0) = h[(N-1)/2]; \quad a(n) = 2h[(N-1)/2 - n]$$

$$b(n) = 2h(N/2 - n)$$

всех четырех типов фильтров с линейной фазовой характеристикой. Ключевые особенности этих типов фильтров сведены в табл. 7.1.

Частотная характеристика фильтра типа 2 (положительно-симметричные коэффициенты и четная длина) всегда равна нулю при $f = 0, 5$ (половина частоты дискретизации, поскольку все частоты нормированы на частоту дискретизации); см. задачу 7.1. Следовательно, фильтры данного типа не используются в качестве фильтров верхних частот. Фильтры типов 3 и 4 (отрицательно-симметричные коэффициенты) вводят сдвиг фазы на 90° , а частотная характеристика таких фильтров всегда равна нулю при $f = 0$, так что такие фильтры нельзя использовать как фильтры нижних частот. Кроме того, характеристика фильтров третьего типа всегда равна нулю при $f = 0, 5$, так что данный фильтр не стоит применять и как фильтр верхних частот. Фильтры первого типа наиболее универсальны. Фильтры третьего и четвертого типа часто используются при проектировании дифференциаторов и фильтров, реализующих преобразования Гильберта, поскольку фильтры этого типа могут давать сдвиг фазы на 90° .

Важно отметить, что фазовую задержку (фильтры типа 1 и 2) или групповую задержку (фильтры всех четырех типов) можно выразить через число коэффициентов фильтра, которые, соответственно, можно подобрать таким образом, чтобы фильтр давал нулевую фазовую или групповую задержку. Например, для фильтров первого и второго типов фазовая задержка записывается следующим образом:

$$T_p = \left(\frac{N-1}{2} \right), \quad (7.4, a)$$

а групповая задержка для фильтров третьего и четвертого типов выражается как

$$T_g = \left(\frac{N-1-\pi}{2} \right), \quad (7.4, б)$$

где T — период дискретизации.

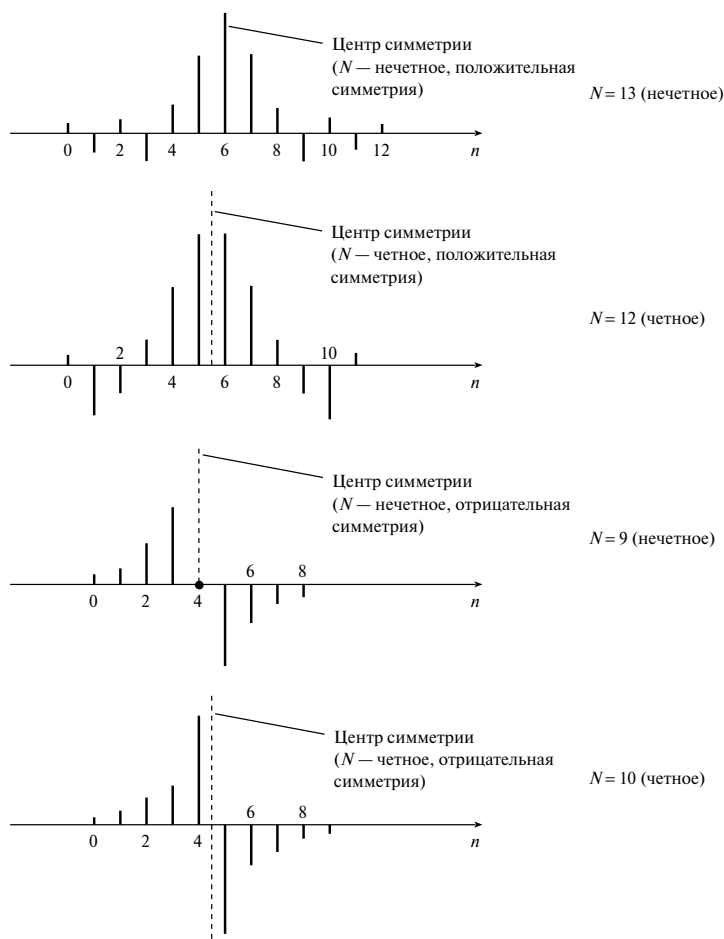


Рис. 7.1. Коэффициенты импульсной характеристики четырех типов фильтров с линейной фазовой характеристикой

7.2. Разработка КИХ-фильтров

Как говорилось в главе 6, разработка цифрового фильтра включает пять этапов.

1. *Спецификация фильтра.* На данном этапе может задаваться тип фильтра, например, фильтр нижних частот, нужная амплитудная и/или фазовая характеристика и разрешенные допуски (если есть), частота дискретизации и длина слов, которыми будут представлены входные данные.
2. *Вычисление коэффициентов.* На этом этапе определяются коэффициенты передаточной функции $H(z)$, которая удовлетворяет спецификациям, полученным на этапе 1. На выбор метода расчета коэффициентов влияет несколько факторов, важнейшими из которых являются критические требования, сформулированные на этапе 1.
3. *Выбор структуры.* Данный этап включает преобразование передаточной функции, полученной на предыдущем этапе, в подходящую фильтрующую структуру или сеть.

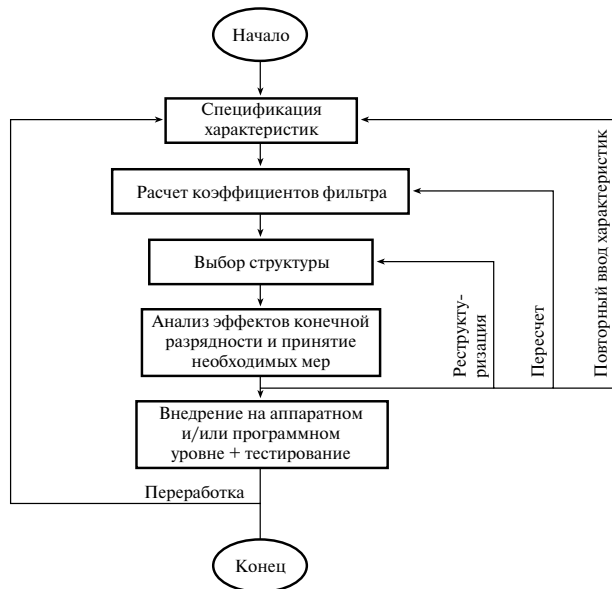


Рис. 7.2. Этапы разработки цифровых фильтров

4. *Анализ следствий конечной разрядности.* Здесь оценивается влияние квантования на коэффициенты фильтра и входные данные, а также влияние на производительность фильтра операции фильтрации со словами конечной длины.
5. *Воплощение.* На данном этапе разрабатывается программный код и/или аппаратный блок и выполняется собственно фильтрация.

Перечисленные пять этапов и их связь обозначены на рис. 7.2. Все этапы разработки КИХ-фильтра подробно рассмотрены ниже, при необходимости мы будем обращаться к примерам.

7.3. Спецификации КИХ-фильтра

Спецификации фильтров вообще подробно рассмотрены в главе 6. В данном разделе внимание акцентируется на деталях спецификаций КИХ-фильтров. Некоторые аспекты спецификаций иллюстрируются на примерах.

Итак, при рассмотрении фазовой характеристики достаточно указать, какая нужна симметрия — четная или нечетная (предполагается, что фазовая характеристика линейна). Амплитудно-частотная характеристика КИХ-фильтра часто задается в виде схемы допусков. Такая схема для фильтра нижних частот показана на рис. 7.3. Подобную схему можно легко получить и для других частотно-избирательных фильтров. Исходя из изображенного на рисунке, интерес представляют следующие параметры:

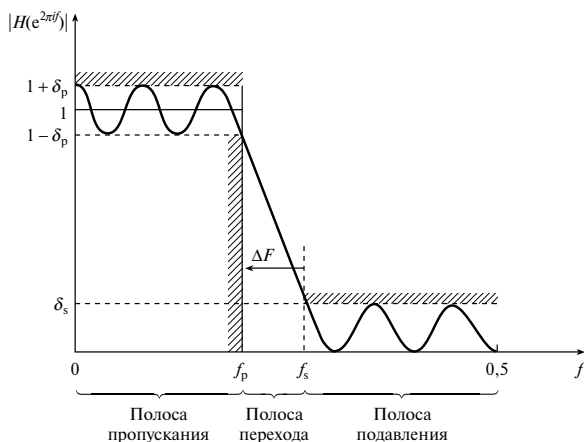


Рис. 7.3. Спецификация амплитудно-частотной характеристики фильтра нижних частот. Отклонения в полосах пропускания и подавления часто выражаются в децибелах: отклонение в полосе пропускания равно $20 \lg(1 + \delta_p)$ дБ; отклонение в полосе подавления равно $-20 \lg(\delta_s)$ дБ

- δ_p отклонение в полосе пропускания (или неравномерность);
- δ_s отклонение в полосе подавления;
- f_p граничная частота полосы пропускания;
- f_s граничная частота полосы подавления;
- F_s частота дискретизации.

На практике часто удобнее выражать δ_p и δ_s в децибелах, как сделано на приведенном рисунке. Расстояние между f_s и f_p равно ширине полосы перехода фильтра. Другой важный параметр — это длина фильтра N , которая определяет число коэффициентов фильтра. В большинстве случаев указанные параметры полностью определяют частотную характеристику КИХ-фильтра.

Существуют и другие спецификации, которые могут представлять практический интерес; это, например, спецификация максимального приемлемого числа коэффициентов фильтра (такое ограничение вводится в определенных приложениях, например, если возможная скорость обработки фиксирована). В конкретном случае может быть неясно, какой выбор набора параметров удачен, так что может использоваться метод проб и ошибок.

Пример 7.2

Пример амплитудной спецификации. Для подавления физиологического шума требуется цифровой фильтр нижних частот. Фильтр должен удовлетворять следующей спецификации:

граничная частота полосы пропускания	10 Гц,
граничная частота полосы подавления	<20 Гц,
затухание в полосе подавления	>30 дБ,
неравномерность в полосе пропускания	<0,026 дБ,
частота дискретизации	256 Гц.

Основные требования в данном случае таковы: 1) фильтр должен вводить во внутриполосные сигналы минимально возможное искажение, 2) длина фильтра должна быть минимальной и не превышать 37.

7.4. Методы расчета коэффициентов КИХ-фильтров

Напомним, что КИХ-фильтр характеризуется следующими уравнениями:

$$y(m) = \sum_{n=0}^{N-1} h(n)x(m-n),$$

$$H(z) = \sum_{n=0}^{N-1} h(n)z^{-n}.$$

Единственной целью большинства методов вычисления (или приближенного вычисления) коэффициентов КИХ-фильтров является получение значений $h(n)$, при которых фильтр удовлетворяет спецификациям, в частности, относящимся к амплитудно-частотной характеристике, и требованиям к пропускной способности. Разработано несколько методов получения $h(n)$. Наиболее широко используемыми из них являются метод вырезания, оптимальный метод и метод частотной выборки. Все три метода позволяют получать КИХ-фильтры с линейной фазовой характеристикой.

7.5. Метод взвешивания

В данном методе используется факт, что частотная характеристика фильтра $H_D(\omega)$ и соответствующая импульсная характеристика $h_D(n)$ связаны обратным преобразованием Фурье:

$$h_D(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_D(\omega) e^{i\omega n} d\omega. \quad (7.5)$$

Индекс D используется, чтобы различать идеальную и практическую импульсные характеристики. Необходимость такого разделения станет понятна несколько позже. Если $H_D(\omega)$ известна, $h_D(n)$ можно получить, применив преобразование Фурье к обеим частям уравнения (7.5). Для иллюстрации предположим, что требуется разработать фильтр нижних частот. Начать можно с идеальной фазовой характеристики, представленной на рис. 7.4, а, где ω_c — частота среза, и шкала частот нормирована ($T = 1$).

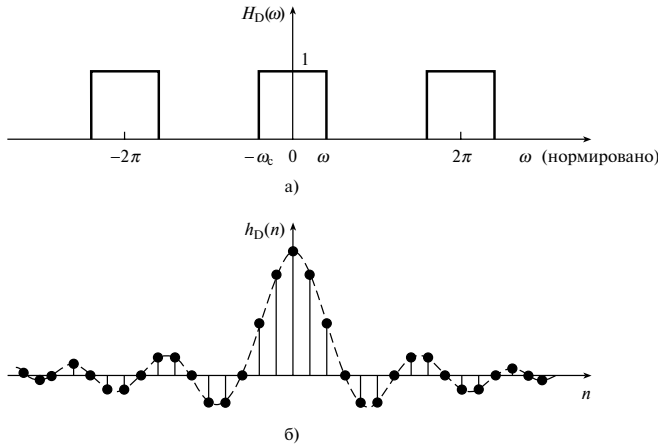


Рис. 7.4. Идеальная частотная характеристика фильтра нижних частот (панель а). Импульсная характеристика идеального фильтра нижних частот (панель б)

Допустив, что характеристика идет от $-\omega_c$ до ω_c , упрощаем интегрирование и получаем следующую импульсную характеристику:

$$\begin{aligned}
 h_D(n) &= \frac{1}{2\pi} \int_{-\pi}^{\pi} 1 \times e^{i\omega n} d\omega = \frac{1}{2\pi} \int_{-\omega_c}^{\omega_c} e^{i\omega n} d\omega = \\
 &= \frac{2f_c \sin(n\omega_c)}{n\omega_c}, \quad n \neq 0, \quad -\infty \leq n \leq \infty \\
 &= 2f_c, \quad n = 0 \text{ (используем правило Лопиталя)}.
 \end{aligned} \tag{7.6}$$

Импульсные характеристики идеальных фильтров верхних частот, полосовых фильтров и режекторных фильтров также находятся из уравнения (7.6) и все они приведены в табл. 7.2. Импульсная характеристика фильтра нижних частот изображена на рис. 7.4, б, из которого видно, что $h_D(n)$ симметрична относительно $n = 0$ (т.е. $h_D(n) = h_D(-n)$), так что фильтр будет иметь линейную (в данном случае — нулевую) фазовую характеристику. Описанный простой подход связан с некоторыми проблемами. Важнейшая из них — хотя характеристика $h_D(n)$ уменьшается при удалении от точки $n = 0$, она длится теоретически до $n = \pm\infty$. Следовательно, полученный фильтр не является КИХ-фильтром.

Очевидным является решение усечь идеальную импульсную характеристику, положив $h_D(n) = 0$ для n , больше, чем (скажем) M . При этом вводится нежелательная неравномерность и выбросы — имеет место так называемый *эффект Гиббса*. То, как отбрасывание коэффициентов сказывается на характеристике фильтра, показано на рис. 7.5. Чем больше коэффициентов осталось, тем ближе спектр фильтра к идеальной характеристике (см. рис. 7.5, б и в). Прямое усечение $h_D(n)$, как оно описано выше,

Таблица 7.2. Идеальные импульсные характеристики стандартных частотно-избирательных фильтров

Тип фильтра	Идеальная частотная характеристика, $h_D(n)$	
	$h_D(n), n \neq 0$	$h_D(0)$
Фильтр нижних частот	$2f_c \frac{\sin(n\omega_c)}{n\omega_c}$	$2f_c$
Фильтр верхних частот	$-2f_c \frac{\sin(n\omega_c)}{n\omega_c}$	$1 - 2f_c$
Полосовой фильтр	$2f_2 \frac{\sin(n\omega_2)}{n\omega_2} - 2f_1 \frac{\sin(n\omega_1)}{n\omega_1}$	$2(f_2 - f_1)$
Заграждающий фильтр	$2f_1 \frac{\sin(n\omega_1)}{n\omega_1} - 2f_2 \frac{\sin(n\omega_2)}{n\omega_2}$	$1 - 2(f_2 - f_1)$

f_c, f_1 и f_2 — нормированные частоты краев полос пропускания или подавления; N — длина фильтра

равносыльно умножению идеальной импульсной характеристики на прямоугольную весовую функцию вида

$$\begin{aligned}
 w(n) &= 1, \quad |n| = 0, 1, \dots, (M - 1)/2 \\
 &= 0.
 \end{aligned}$$

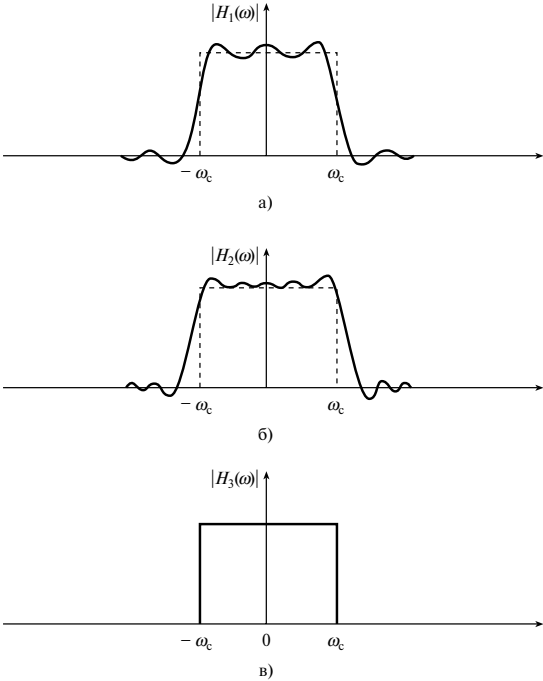


Рис. 7.5. Влияние на частотную характеристику округления идеальной импульсной характеристики до а) 13 коэффициентов; б) 25 коэффициентов; в) бесконечного числа коэффициентов

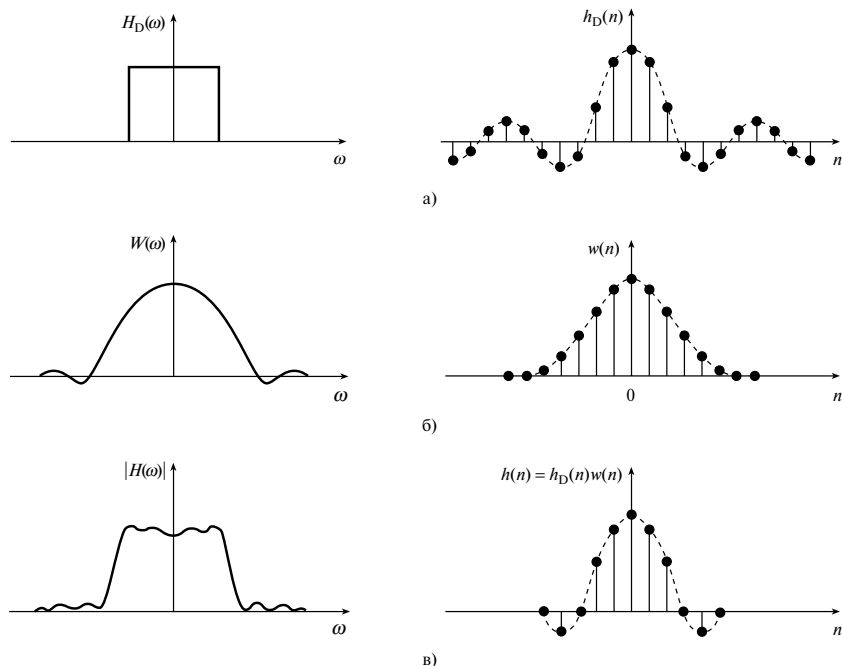


Рис. 7.6. Иллюстрация определения коэффициентов фильтра $h(n)$ с помощью метода взвешивания

В частотной области это эквивалентно свертке $H_D(\omega)$ с $W(\omega)$, где $W(\omega)$ — Фурье-образ $w(n)$. Тогда как $W(\omega)$ имеет классический вид функции $(\sin x)/x$, усечение $h_D(n)$ приводит к появлению в частотной характеристике выбросов.

На практике идеальная частотная характеристика h_D множится на подходящую весовую функцию $w(n)$ с конечной длительностью. Таким образом, получающаяся импульсная характеристика гладко затухает до нуля. Данный процесс иллюстрируется на рис. 7.6. На рис. 7.6, а показана идеальная частотная характеристика и соответствующая идеальная импульсная характеристика. На рис. 7.6, б показана весовая функция конечной длительности и ее спектр. На рис. 7.6, в показана функция $h(n)$, которая получается перемножением $h_D(n)$ и $w(n)$. Из соответствующей частотной характеристики видно, что неравномерности и выбросы, характерные для прямого усечения, в значительной степени подавлены. В то же время, ширина полосы перехода больше, чем для прямоугольной функции. Известно, что ширина полосы перехода фильтра определяется шириной основного лепестка весовой функции. Боковые лепестки приводят к появлению неравномерности в полосах пропускания и подавления.

7.5.1. Некоторые распространенные весовые функции

Было предложено несколько вырезающих функций. Одной из наиболее широко используемых является весовая функция Хэмминга, которая определяется следующим образом:

$$w(n) = 0,54 + 0,46 \cos(2\pi n/N) \begin{cases} -(N-2)/2 \leq n \leq (N-1)/2 & (N - \text{нечетное}) \\ -N/2 \leq n \leq N/2 & (N - \text{четное}) \end{cases} \quad (7.7)$$

$= 0$ в других случаях.

Характеристики функции Хэмминга и прямоугольной функции сравниваются на рис. 7.7 во временной и частотной областях. Во временной области функция Хэмминга более мягко выходит на нуль с обеих сторон. В частотной области амплитуда ее главного лепестка шире (примерно вдвое), чем амплитуда прямоугольной функции, но ее боковые лепестки меньше по сравнению с главным (примерно на 40 дБ меньше главного лепестка по сравнению с 14 дБ для прямоугольной функции). Вследствие этого функция Хэмминга даст фильтр с большей полосой перехода (поскольку у нее шире главный лепесток), но и большим затуханием в полосе подавления (поскольку уровни боковых лепестков меньше).

Связь ширины полосы перехода (от полосы пропускания к полосе подавления) фильтра, построенного на основе функции Хэмминга, с длиной фильтра выражается следующей формулой:

$$\Delta f = 3,3/N, \quad (7.8)$$

где N — длина фильтра, а Δf — нормированная ширина полосы перехода. Максимальное затухание в полосе подавления, возможное при использовании функции Хэмминга, составляет порядка 53 дБ, а минимальная амплитуда неравномерности в полосе пропускания составляет около 0,0194 дБ.

Основные характеристики самых распространенных весовых функций собраны в табл. 7.3. Отметим, что первые четыре функции имеют фиксированные характеристики, такие как ширина перехода и затухание в полосе подавления. Следовательно, их использование ограничивает свободу разработчика. Отметим также, что фильтр, построенный с помощью метода взвешивания, имеет равные неравномерности в полосе пропускания и полосе подавления, т.е. $\delta_p = \delta_s$ (см. рис. 7.3). На практике это ограничение может дать фильтр, неравномерность которого в полосе пропускания будет излишне малой.

Таблица 7.3. Важные особенности распространенных весовых функций

Функция	Ширина перехода (нормированная) (Гц)	Неравномерность в полосе пропускания (дБ)	Главный лепесток относительно бокового лепестка	Затухание в полосе подавления	Формула
Прямоугольная	$0, 9/N$	0,7416	13	21	1
Хеннинга	$3, 1/N$	0,0546	31	44	$0, 5 + 0, 5 \cos \left(\frac{2\pi n}{N} \right)$
Хэмминга	$3, 3/N$	0,0194	41	53	$0, 54 + 0, 46 \cos \left(\frac{2\pi n}{N} \right)$
Блэкмена	$5, 5/N$	0,0017	57	75	$0, 42 + 0, 5 \cos \left(\frac{2\pi n}{N-1} \right) + 0, 08 \cos \left(\frac{4\pi n}{N-1} \right)$
Кайзера	$2, 93/N (\beta = 4, 54)$	0,0274		50	$\frac{I_0(\beta) - [2n/(N-1)]^{21/2}}{I_0(\beta)}$
	$4, 32/N (\beta = 6, 76)$	0,00275		70	
	$5, 71/N (\beta = 8, 96)$	0,000275		90	

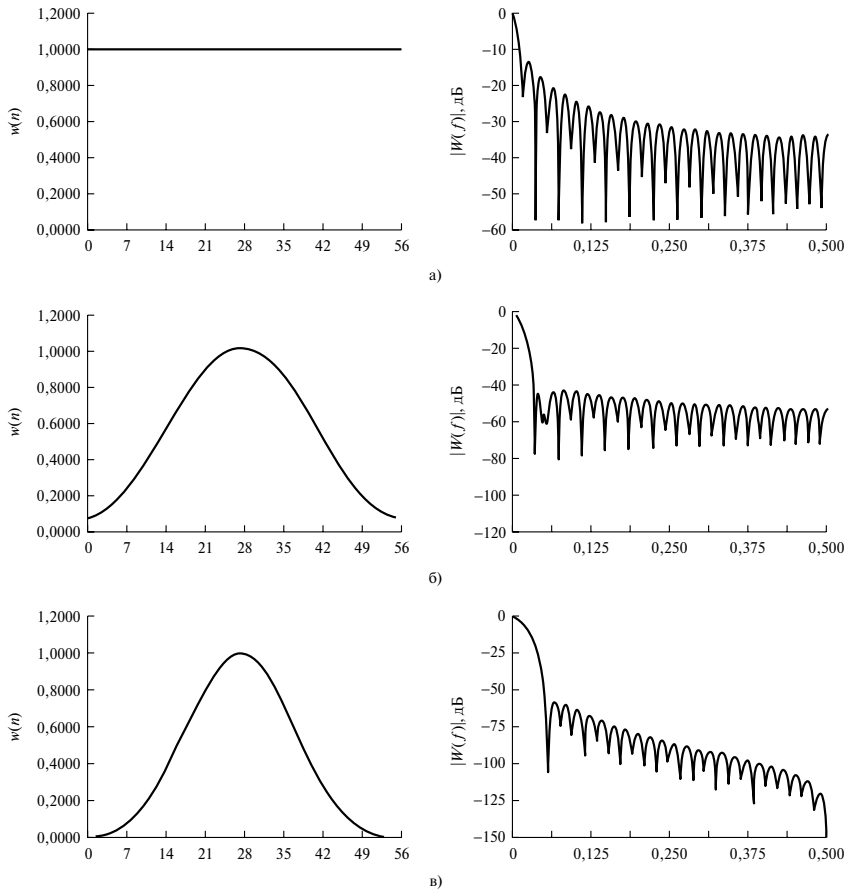


Рис. 7.7. Сравнение характеристик распространенных весовых функций во временной и частотной областях: а) прямоугольная функция; б) функция Хэмминга; в) функция Блэкмена

Окно Кайзера (Kaiser window function) несколько сглаживает очерченные выше проблемы, поскольку имеет параметр, управляющий неравномерностью, β , что позволяет разработчику играть на компромиссах между шириной перехода и неравномерностью. Функция Кайзера задается следующим образом:

$$w(n) = I_0 \left(\beta \left[1 - \left(\frac{2n}{N-1} \right)^2 \right]^{1/2} \right) / I_0(\beta) \quad - (N-1)/2 \leq n \leq (N-1)/2 \quad (7.9)$$

$$= 0 \quad \text{в других случаях,}$$

где $I_0(x)$ — модифицированная функция Бесселя первого рода нулевого порядка. Управляющий параметр β отвечает за спад вырезающей функции на краях (во временной

области). Для вычисления $I_0(x)$ обычно используется следующее разложение в степенной ряд [14]:

$$I_0(x) = 1 + \sum_{k=1}^L \left[\frac{(x/2)^k}{k!} \right]^2,$$

причем обычно $L < 25$. Эффективной реализацией указанного уравнения является алгоритм Кайзера [14].

При $\beta = 0$ функция Кайзера соответствует прямоугольной весовой функции, а при $\beta = 5,44$ функция весьма похожа на функцию Хэмминга (хотя и не идентична ей). Значение β определяется требованиями к затуханию в полосе подавления и его можно оценить с помощью одного из приведенных ниже эмпирических соотношений:

$$\beta = 0, \text{ если } A \leq 21 \text{ дБ}, \quad (7.10, a)$$

$$\beta = 0,5842(A - 21)^{0,4} + 0,07886(A - 21), \text{ если } 21 \text{ дБ} < A < 50 \text{ дБ} \quad (7.10, б)$$

$$\beta = 0,1102(A - 8,7), \text{ если } A \geq 50 \text{ дБ}, \quad (7.10, в)$$

где $A = -20 \lg(\delta)$ — затухание в полосе подавления, $\delta = \min(\delta_p, \delta_s)$, поскольку неравномерности в полосе пропускания и полосе подавления приблизительно равны, δ_p — желаемая неравномерность в полосе пропускания, а δ_s — желаемая неравномерность в полосе подавления. Число коэффициентов фильтра N подчиняется зависимости

$$N \geq \frac{A - 7,95}{14,36 \Delta f}, \quad (7.11)$$

где Δf — нормированная ширина полосы перехода. Далее полученные значения β и N используются для вычисления коэффициентов функции Кайзера $w(n)$.

7.5.2. Метод взвешивания: резюме

- *Этап 1.* Задать “идеальную” или желаемую частотную характеристику фильтра $H_D(\omega)$.
- *Этап 2.* Получить импульсную характеристику $h_D(n)$ желаемого фильтра, найдя для этого Фурье-образ частотной характеристики (формула (7.6, б)). Выражения для $h_D(n)$ стандартных частотно-избирательных фильтров приведены в табл. 7.2.
- *Этап 3.* Выбрать весовую функцию, которая удовлетворяет требованиям к полосе пропускания или затуханию, а затем определить число коэффициентов фильтра, используя подходящее выражение для связи длины фильтра с шириной перехода, Δf (записываются через частоту дискретизации).

- **Этап 4.** Получить значения выбранной весовой функции $w(n)$ и значения коэффициентов реального КИХ-фильтра $h(n)$, умножив $h_D(n)$ на $w(n)$:

$$h(n) = h_D(n)w(n). \quad (7.12)$$

Очевидно, что метод вырезания — это прямолинейный метод, включающий минимум вычислений. В действительности при таком подходе коэффициенты можно вычислить с помощью карманного калькулятора. Впрочем, существуют и компьютерные программы вычисления $h(n)$, но следует отметить, что фильтр, полученный описанным способом, не является оптимальным, т.е. во многих случаях иные методы позволяют получить аналогичный фильтр с меньшим числом коэффициентов.

Пример 7.3

С помощью метода взвешивания вычислите коэффициенты КИХ-фильтра нижних частот, который удовлетворяет следующим спецификациям:

граничные частоты полосы пропускания	1,5 кГц,
ширина полосы перехода	0,5 кГц,
затухание в полосе подавления	>50 дБ,
частота дискретизации	8 кГц.

Решение

Используя табл. 7.2, выберем $h_D(n)$ для фильтра нижних частот:

$$h_D(n) = 2f_c \frac{\sin(n\omega_c)}{n\omega_c}, \quad n \neq 0,$$

$$h_D(n) = 2f_c, \quad n = 0.$$

Из табл. 7.3 следует, что требованиям к затуханию в полосе подавления удовлетворяют функции Хэмминга, Блэкмена или Кайзера. Для простоты выберем функцию Хэмминга. Тогда $\Delta f = 0,5/8 = 0,0625$. Поскольку $N = 3,3/\Delta f = 3,3/0,0625 = 52,8$, положим $N = 53$, и коэффициенты будут равны

$$h_D(n)w(n), \quad -26 \leq n \leq 26,$$

где

$$h_D(n) = \frac{2f_c \sin(n\omega_c)}{n\omega_c}, \quad n \neq 0$$

$$h_D(n) = 2f_c, \quad n = 0,$$

$$w(n) = 0,54 + 0,46 \cos(2\pi n/53), \quad -26 \leq n \leq 26.$$

Вследствие эффекта смазывания характеристики фильтра, вводимого весовой функцией, частота среза получающегося фильтра будет отличаться от представленной в спецификации. Чтобы учесть этот эффект, используем f_c — центр полосы перехода:

$$f'_c = f_c + \Delta f/2 = (1,5 + 0,25) \text{ кГц} = 1,75 \text{ кГц} \rightarrow 1,75/8 = 0,21875.$$

Поскольку $h(n)$ — симметричная функция, вычислить требуется только значения $h(0), h(1), \dots, h(26)$, а остальные можно получить из условия симметрии.

$$n = 0 : h_D(0) = 2f_c = 2 \times 0,21875 = 0,4375$$

$$w(0) = 0,54 + 0,46 \cos(0) = 1$$

$$h(0) = h_D(0)w(0) = 0,4375$$

$$n = 1 : h_D(1) = \frac{2 \times 0,21875}{2\pi \times 0,21875} \sin(2\pi \times 0,21875)$$

$$= \frac{\sin(360^\circ \times 0,21875)}{\pi} = 0,31219$$

$$w(1) = 0,54 + 0,46 \cos(2\pi/53)$$

$$= 0,54 + 0,46 \cos(360^\circ/53) = 0,99677$$

$$h(1) = h(-1) = h_D(1)w(1) = 0,31118$$

$$n = 2 : h_D = \frac{2 \times 0,21875}{2 \times 2\pi \times 0,21875} \sin(2 \times 2\pi \times 0,21875)$$

$$= \frac{\sin(157,5^\circ)}{2\pi} = 0,06013$$

$$w(2) = 0,54 + 0,46 \cos(2\pi \times 2/53)$$

$$= 0,54 + 0,46 \cos(720^\circ/53) = 0,98713$$

$$h(2) = h(-2) = h_D(2)w(2) = 0,06012$$

$$\vdots \quad \vdots \quad \vdots \quad \vdots$$

$$n = 26 : n_D(26) = \frac{2 \times 0,21875}{26 \times 2\pi \times 0,21875} \sin(26 \times 2\pi \times 0,21875)$$

$$= -0,01131$$

$$w(26) = 0,54 + 0,46 \cos(2\pi \times 26/53) =$$

$$= 0,54 + 0,46 \cos(9360^\circ/53) = 0,08081$$

$$h(26) = h(-26) = h_D(26)w(26) = -0,000914$$

Таблица 7.4. Коэффициенты КИХ-фильтра из примера 7.3 ($N = 53$, функция Хэмминга, $f_c = 1750$ Гц)

$h[0] =$	$-9,1399895e - 04$	$= h[52]$
$h[1] =$	$2,1673690e - 04$	$= h[51]$
$h[2] =$	$1,3270280e - 03$	$= h[50]$
$h[3] =$	$3,2138355e - 04$	$= h[49]$
$h[4] =$	$-1,9238177e - 03$	$= h[48]$
$h[5] =$	$-1,4683633e - 03$	$= h[47]$
$h[6] =$	$2,3627318e - 03$	$= h[46]$
$h[7] =$	$3,4846558e - 03$	$= h[45]$
$h[8] =$	$-1,9925839e - 03$	$= h[44]$
$h[9] =$	$-6,2837232e - 03$	$= h[43]$
$h[10] =$	$4,5320247e - 09$	$= h[42]$
$h[11] =$	$9,2669460e - 03$	$= h[41]$
$h[12] =$	$4,3430586e - 03$	$= h[40]$
$h[13] =$	$-1,1271299e - 02$	$= h[39]$
$h[14] =$	$-1,1402453e - 02$	$= h[38]$
$h[15] =$	$1,0630714e - 02$	$= h[37]$
$h[16] =$	$2,0964392e - 02$	$= h[36]$
$h[17] =$	$-5,2583216e - 03$	$= h[35]$
$h[18] =$	$-3,2156086e - 02$	$= h[34]$
$h[19] =$	$-7,5449714e - 03$	$= h[33]$
$h[20] =$	$4,3546153e - 02$	$= h[32]$
$h[21] =$	$3,2593190e - 02$	$= h[31]$
$h[22] =$	$-5,3413653e - 02$	$= h[30]$
$h[23] =$	$-8,5682029e - 02$	$= h[29]$
$h[24] =$	$6,0122145e - 02$	$= h[28]$
$h[25] =$	$3,1118568e - 01$	$= h[27]$
$h[26] =$	$4,3750000e - 01$	$= h[26]$

Обратите внимание на то, что индексы коэффициентов пробегают значения от -26 до 26 . Чтобы фильтр был причинным (необходимое условие реализации), к каждому индексу было прибавлено 26 , так что новая нумерация начиналась с нуля. Коэффициенты фильтра с указанными индексами приведены в табл. 7.4. Из спектра фильтра (не показан) следует, что спецификация удовлетворяется.

Пример 7.4

Требуется, чтобы КИХ-фильтр удовлетворял следующим спецификациям:

полоса пропускания	150–250 Гц,
ширина полосы перехода	50 Гц,
неравномерность в полосе пропускания	0,1 дБ,
затухание в полосе подавления	60 дБ,
частота дискретизации	1 кГц.

Определите коэффициенты и спектр фильтра, используя метод вырезания.

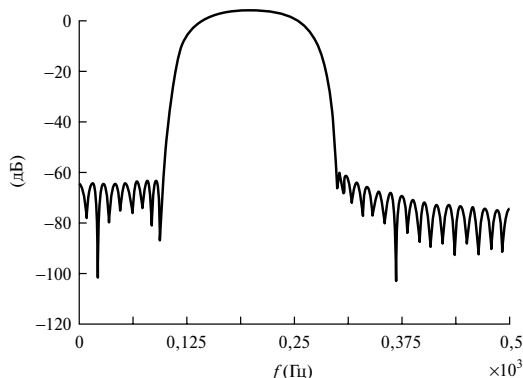


Рис. 7.8. Спектр фильтра (пример 7.4)

Решение

Из спецификации следует, что неравномерность в полосе пропускания и неравномерность в полосе подавления равны:

$$20 \lg(1 + \delta_p) = 0,1 \text{ дБ, откуда } \delta_p = 0,0115$$

$$-20 \lg(\delta_s) = 60 \text{ дБ, откуда } \delta_s = 0,001.$$

Следовательно,

$$\delta = \min(\delta_p, \delta_s) = 0,001.$$

Требования к затуханию можно удовлетворить только при использовании функции Кайзера или Блэкмена. Для функции Кайзера число коэффициентов фильтра равно

$$N \geq \frac{A - 7,95}{14,36\Delta F} = \frac{60 - 7,95}{14,36(50/1000)} = 72,49.$$

Пусть $N = 73$. Параметр неравномерности выражается следующим образом:

$$\beta = 0,1102(60 - 8,7) = 5,65.$$

Значения $N = 73$ и $\beta = 5,65$ используются как вход программы `window.c` (см. приложение), на выходе которой получаются значения $w(n)$, идеальная импульсная характеристика $h_D(n)$ и коэффициенты фильтра. Чтобы учесть эффект смазывания, при вычислении идеальной импульсной характеристики использовались частоты среза $f_{c1} - \Delta f/2$ и $f_{c2} + \Delta f/2$, т.е. $f_{c1} = 125$ Гц и $f_{c2} = 275$ Гц соответственно. Коэффициенты фильтра представлены в табл. 7.5, а спектр фильтра изображен на рис. 7.8.

При использовании функции Блэкмена число коэффициентов фильтра оценивается как

$$N = 5,5/\Delta f = 5,5/(50/1000) \approx 110.$$

Таблица 7.5. Коэффициенты фильтра Кайзера (пример 7.4)

$h[0] =$	$-1,0627330e - 04$	$= h[72]$
$h[1] =$	$-3,9118142e - 04$	$= h[71]$
$h[2] =$	$-7,5561629e - 05$	$= h[70]$
$h[3] =$	$-1,3695577e - 04$	$= h[69]$
$h[4] =$	$-6,8122013e - 04$	$= h[68]$
$h[5] =$	$5,0929290e - 04$	$= h[67]$
$h[6] =$	$2,3413494e - 03$	$= h[66]$
$h[7] =$	$8,0280013e - 04$	$= h[65]$
$h[8] =$	$-1,7031635e - 04$	$= h[64]$
$h[9] =$	$-5,5034956e - 04$	$= h[63]$
$h[10] =$	$-4,9912488e - 04$	$= h[62]$
$h[11] =$	$-4,4036355e - 03$	$= h[61]$
$h[12] =$	$-2,1639856e - 03$	$= h[60]$
$h[13] =$	$6,9094151e - 03$	$= h[59]$
$h[14] =$	$6,6067599e - 03$	$= h[58]$
$h[15] =$	$-1,6445200e - 03$	$= h[57]$
$h[16] =$	$4,5229777e - 09$	$= h[56]$
$h[17] =$	$2,1890066e - 03$	$= h[55]$
$h[18] =$	$-1,1720511e - 02$	$= h[54]$
$h[19] =$	$-1,6377726e - 02$	$= h[53]$
$h[20] =$	$6,8804519e - 03$	$= h[52]$
$h[21] =$	$1,8882837e - 02$	$= h[51]$
$h[22] =$	$2,9068601e - 03$	$= h[50]$
$h[23] =$	$4,3925286e - 03$	$= h[49]$
$h[24] =$	$1,8839744e - 02$	$= h[48]$
$h[25] =$	$-1,2481155e - 02$	$= h[47]$
$h[26] =$	$-5,2063428e - 02$	$= h[46]$
$h[27] =$	$-1,6557375e - 02$	$= h[45]$
$h[28] =$	$3,3298453e - 02$	$= h[44]$
$h[29] =$	$1,0439025e - 02$	$= h[43]$
$h[30] =$	$9,4320244e - 03$	$= h[42]$
$h[31] =$	$8,5673629e - 02$	$= h[41]$
$h[32] =$	$4,5314758e - 02$	$= h[40]$
$h[33] =$	$-1,6657147e - 01$	$= h[39]$
$h[34] =$	$-2,0669512e - 01$	$= h[38]$
$h[35] =$	$8,9135544e - 02$	$= h[37]$
$h[36] =$	$3,0000000e - 01$	$= h[36]$

Коэффициенты фильтра для функции Блэкмена не приводятся из соображений экономии места. Очевидно, впрочем, что функция Кайзера эффективнее функции Блэкмена с точки зрения числа коэффициентов, требуемых для соответствия тем же спецификациям. Вообще, по этому пункту функции Кайзера эффективнее и других весовых функций.

Пример 7.5

Используя функцию Кайзера, получите коэффициенты КИХ-фильтра с линейной фазовой характеристикой, который удовлетворяет следующим спецификациям амплитудной характеристики:

затухание в полосе подавления	40 дБ,
неравномерность в полосе пропускания	0,01 дБ,
ширина перехода	500 Гц,
частота дискретизации	10 кГц,
идеальная частота среза	1200 Гц.

Решение

Из спецификации получаем

$$20 \lg(1 + \delta_p) = 0,01 \text{ дБ, откуда } \delta_p = 0,00115$$

$$-20 \lg(\delta_s) = 40 \text{ дБ, откуда } \delta_s = 0,01.$$

Поскольку в методе вырезания неравномерность в полосе пропускания равна неравномерности в полосе подавления (так как их нельзя задать по отдельности), используем меньшую из неравномерностей:

$$\delta = \delta_s = \delta_p = 0,00115.$$

Это означает, что затухание в полосе подавления больше, чем фактически необходимо; в этом случае $-20 \lg(0,00115) = 55,8 \text{ дБ}$.

Из формулы (7.11) получаем число требуемых коэффициентов фильтра:

$$n = \frac{A - 7,95}{14,36\Delta f} = \frac{58,8 - 7,95}{14,36(500/10000)} \approx 71.$$

Если бы использовалась заданная в спецификации величина 40 дБ, N было бы равно 45. Итак, требование, чтобы δ_p было равным δ_s (условие использования метода взвешивания) привело к увеличению необходимого числа коэффициентов фильтра.

Параметры неравномерности вычисляются на основе уравнения (7.10):

$$\beta = 0,5842(58,8 - 21)^{0,4} + 0,07886(58,8 - 21) = 5,48.$$

Коэффициенты КИХ-фильтра получаются из соотношения $h(n) = h_D(n)w(n)$, где

$$h_D(n) = 2f_c \frac{\sin(n\omega_c)}{n\omega_c}, n \neq 0$$

$$h_D(n) = 2f_c, n = 0 \text{ (из табл. 7.2),}$$

а $w(n)$ представлено в уравнении (7.9).

Как объяснялось выше, частота среза, f_c , используемая для вычисления $h(n)$, отличается от той, что дана в спецификации, поскольку это позволяет учесть эффект смазывания, характерный для метода взвешивания. Выберем в качестве f_c середину полосы пропускания: $f'_c = 1200 + \Delta f/2 = 1450 \text{ Гц}$.

Далее в программе `window.c` (см. приложение) используются следующие параметры фильтра:

частота среза 1450 Гц,
 параметр неравномерности, β 5,48,
 число коэффициентов фильтра 71,
 частота дискретизации 10 кГц.

Полученные коэффициенты фильтра представлены в табл. 7.6, а спектр фильтра изображен на рис. 7.9.

Таблица 7.6. Коэффициенты фильтра, полученного с помощью функции Кайзера (пример 7.5)

$h[0] =$	$9,8470163e - 05$	$= h[70]$
$h[1] =$	$-1,3972411e - 04$	$= h[69]$
$h[2] =$	$-4,5442489e - 04$	$= h[68]$
$h[3] =$	$-4,8756977e - 04$	$= h[67]$
$h[4] =$	$2,6173965e - 05$	$= h[66]$
$h[5] =$	$8,6653647e - 04$	$= h[65]$
$h[6] =$	$1,2967984e - 03$	$= h[64]$
$h[7] =$	$6,1688894e - 04$	$= h[63]$
$h[8] =$	$-1,0445340e - 03$	$= h[62]$
$h[9] =$	$-2,4646644e - 03$	$= h[61]$
$h[10] =$	$-2,1059775e - 03$	$= h[60]$
$h[11] =$	$4,4371801e - 04$	$= h[59]$
$h[12] =$	$3,5954580e - 03$	$= h[58]$
$h[13] =$	$4,5526695e - 03$	$= h[57]$
$h[14] =$	$1,5922295e - 03$	$= h[56]$
$h[15] =$	$-3,8904820e - 03$	$= h[55]$
$h[16] =$	$-7,6398162e - 03$	$= h[54]$
$h[17] =$	$-5,6061945e - 03$	$= h[53]$
$h[18] =$	$2,2010888e - 03$	$= h[52]$
$h[19] =$	$1,0450148e - 02$	$= h[51]$
$h[20] =$	$1,1760002e - 02$	$= h[50]$
$h[21] =$	$2,8239875e - 03$	$= h[49]$
$h[22] =$	$-1,1380549e - 02$	$= h[48]$
$h[23] =$	$-1,9631856e - 02$	$= h[47]$
$h[24] =$	$-1,2665935e - 02$	$= h[46]$
$h[25] =$	$8,0061777e - 03$	$= h[45]$
$h[26] =$	$2,8182781e - 02$	$= h[44]$
$h[27] =$	$2,9474031e - 02$	$= h[43]$
$h[28] =$	$3,8724896e - 03$	$= h[42]$
$h[29] =$	$-3,5942288e - 02$	$= h[41]$
$h[30] =$	$-5,9766794e - 02$	$= h[40]$
$h[31] =$	$-3,7113570e - 02$	$= h[39]$
$h[32] =$	$4,1378026e - 02$	$= h[38]$
$h[33] =$	$1,5291289e - 01$	$= h[37]$
$h[34] =$	$2,5100632e - 01$	$= h[36]$
$h[35] =$	$2,9000000e - 01$	$= h[35]$

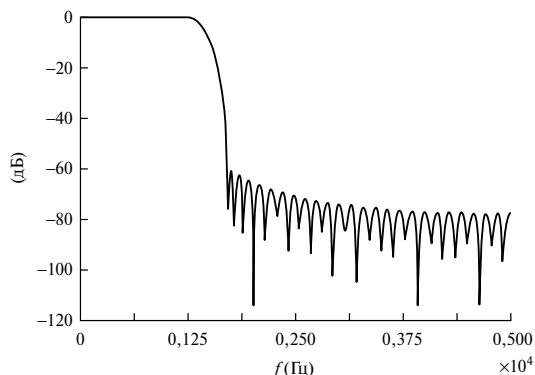


Рис. 7.9. Спектр фильтра, полученного с помощью функции Кайзера (пример 7.5)

7.5.3. Плюсы и минусы метода взвешивания

- Важным достоинством метода взвешивания является простота: его просто применить и легко понять. Этот метод включает минимальный объем вычислений даже при использовании более сложной функции Кайзера.
- Главный недостаток метода — отсутствие гибкости. Максимальная неравномерность в полосе пропускания и неравномерность в полосе подавления примерно равны, так что разработчик может получить фильтр с либо слишком маленькой неравномерностью в полосе пропускания, либо со слишком большим затуханием в полосе подавления.
- Вследствие того, что в методе фигурирует свертка спектра вырезающей функции и желаемой характеристики, невозможно точно задать граничные частоты полосы пропускания и полосы подавления.
- Для данной функции (исключая функцию Кайзера) максимальная амплитуда колебаний в характеристике фильтра фиксирована вне зависимости от того, насколько большим делать N . Следовательно, затухание в полосе подавления фиксированно для данной функции. Таким образом, для данной спецификации затухания разработчик должен найти подходящую функцию.
- В некоторых приложениях выражения формулы для $H_D(\omega)$ будут настолько сложными, что из уравнения (7.5) аналитически находить $h_D(n)$ нет смысла. В таких случаях $h_D(n)$ можно получить с помощью метода частотной выборки, а уже затем применять весовую функцию (см. раздел 7.7.1).

7.6. Оптимизационные методы

Метод вычисления оптимальных (по Чебышеву) коэффициентов КИХ-фильтра является чрезвычайно мощным, и благодаря великолепным программам разработки его весьма просто применять. По этим причинам и еще потому, что данный метод дает великолепные фильтры, он широко используется во многих сферах, где требу-

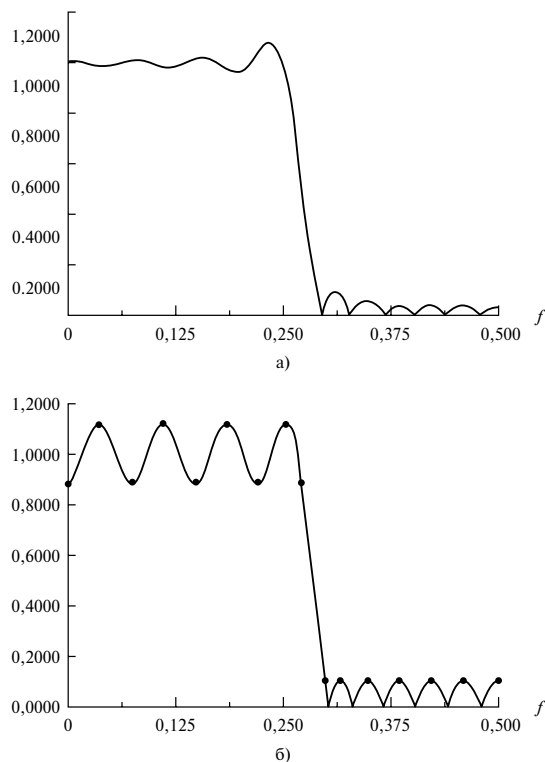


Рис. 7.10. Сравнение частотных характеристик: а) фильтра, полученного методом вырезания, б) оптимального фильтра. На панели *а* колебания характеристики больше на границе полосы; а на панели *б* колебания имеют равные амплитуды (полосы равных колебаний) в полосе пропускания и подавления

ются КИХ-фильтры. Ниже рассматриваются принципы, на основе которых построен метод, программа разработки и ее применение. Для иллюстрации метода разобрано несколько примеров.

7.6.1. Основные концепции

При вычислении коэффициентов фильтра по методу вырезания возникает проблема выбора удачной аппроксимации желаемой или идеальной частотной характеристики. Максимальные колебания характеристики фильтров, разработанных с помощью метода взвешивания, возникают возле краев полосы и уменьшаются при удалении от них (рис. 7.10, *а*). Оказывается, что если колебания распределены более равномерно по полосе пропускания и полосе подавления, как, например, на рис. 7.10, *б*, можно получить лучшую аппроксимацию желаемой частотной характеристики.

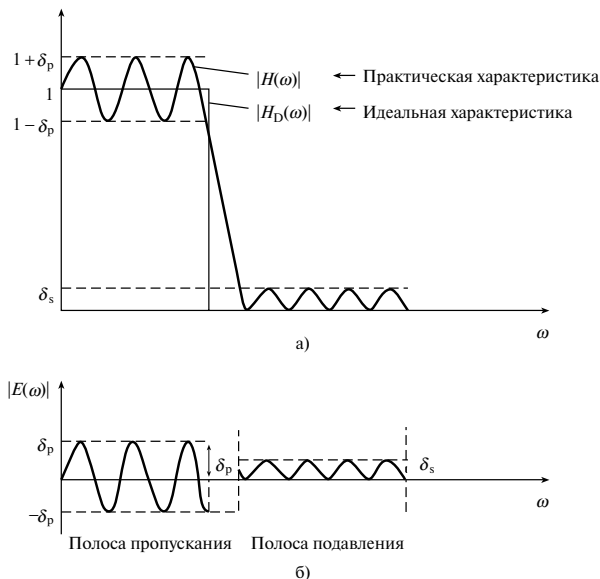


Рис. 7.11. Частотная характеристика оптимального фильтра нижних частот (панель *а*). Характеристика ошибки между идеальной и практической характеристиками ($\delta_p = 2\delta_s$) (панель *б*)

Оптимизационные методы основаны на понятии полос равных колебаний. Рассмотрим частотную характеристику фильтра нижних частот, изображенную на рис. 7.11. В полосе пропускания реальная характеристика осциллирует между значениями $1 - \delta_p$ и $1 + \delta_p$. В полосе подавления характеристика фильтра находится между 0 и δ_s . Отличие характеристик идеального и реального фильтров можно рассматривать как функцию ошибок

$$E(\omega) = W(\omega)[H_D(\omega) - H(\omega)], \quad (7.13)$$

где $H_D(\omega)$ — идеальная или желаемая характеристика, а $W(\omega)$ — весовая функция, которая позволяет определить относительную ошибку аппроксимации между различными полосами. Цель оптимального метода — определить коэффициенты фильтра $h(n)$, при которых значение максимальной взвешенной ошибки $|E(\omega)|$ минимизируется в полосе пропускания и полосе подавления. Математически это можно записать следующим образом:

$$\min[\max |E(\omega)|]$$

по полосам пропускания и полосам подавления. Было установлено (см., например, [14]), что при минимизации $\max |E(\omega)|$ характеристика фильтра будет иметь равные колебания в пределах полос пропускания и подавления, причем модуль максимального отклонения будет постоянным, и характеристика будет проходить между двумя уровнями амплитуды с чередованием знака отклонения (рис. 7.10, *б*). Далее мы зачастую не будем



Рис. 7.12. Упрощенная функциональная схема оптимального метода

разделять минимумы и максимумы характеристики, говоря об экстремумах вообще. Например, у фильтров нижних частот с линейной фазовой характеристикой имеется $r + 1$ или $r + 2$ экстремумов, где $r = (N + 1)/2$ (для фильтров типа 1) или $r = N/2$ (для фильтров типа 2). На рис. 7.10, б частоты экстремумов изображены маленькими кружочками.

При данном наборе спецификаций фильтра расположение экстремальных частот, кроме тех, что размещены на границе полос (т.е. на частотах $f = f_p$ и $f = F_s/2$), не известно *априори*. Значит, основная задача оптимального метода — это найти положения экстремальных частот. Для решения этой задачи был разработан мощный метод, в котором реализован алгоритм замены Ремеза (Remez exchange algorithm) [9, 12, 14]. Зная положения экстремумов, получить действительную частотную характеристику (а значит, импульсную характеристику) — дело техники. Для данного набора спецификаций (т.е. граничных частот полосы пропускания N отношения амплитуд колебаний характеристики в полосе пропускания и полосе подавления) оптимальный метод включает следующие ключевые этапы (см. рис. 7.12):

- использовать алгоритм замены Ремеза, чтобы найти оптимальный набор экстремальных частот;
- определить частотную характеристику, используя положения экстремумов;
- получить коэффициенты импульсной характеристики.

Сердцем оптимального метода является первый этап, где для определения экстремальных частот фильтра, амплитудно-частотная характеристика которого удовлетворяет условию оптимальности, используется итеративная процедура. Данный этап зависит от

теоремы о дизъюнкции (alternation theorem), определяющей число экстремальных частот, которые могут существовать при данном значении N .

В настоящее время доступна и широко используется программа на языке FORTRAN, в которой реализована описанная выше процедура (см. например, [9]). Программа позволяет разрабатывать различные частотно-избирательные фильтры, в том числе фильтры нижних, верхних частот, полосовые и режекторные фильтры, а также дифференциаторы и аппаратные реализации преобразования Гильберта. Кроме того, можно вычислить коэффициенты заданной пользователем произвольной частотной характеристики. Более подробно об оптимальном методе можно прочесть в названной выше книге. В приложении к данной главе рассмотрена реализация оптимального метода в форме файла MATLAB.

7.6.2. Параметры, требуемые для использования оптимальной программы

N	Число коэффициентов фильтра, т.е. длина фильтра. Данный параметр можно оценить по соотношениям, представленным в следующем разделе.
Jtype	Данный параметр задает тип фильтра. Всего возможны три типа: Jtype=1 (фильтры с несколькими полосами пропускания/подавления, в том числе фильтры нижних частот, фильтры верхних частот, полосовые и режекторные), Jtype=2 (задается дифференциатор), Jtype=3 (задается трансформатор Гильберта).
$W(\omega)$	Весовая функция. Данный параметр определяет относительную значимость каждой полосы. По сути, это позволяет использовать компромиссы между колебаниями в полосе пропускания и затуханием в полосе подавления. Весовая функция задается для каждой полосы.
Ngrid	Данный параметр задает плотность сетки. Это число частот, в которых при поиске экстремумов частотная характеристика проверяется на оптимальность (под оптимумом понимается минимум максимальной амплитуды ошибки, $ E(\omega) $ в полосе (полосах) пропускания и полосе (полосах) подавления). По умолчанию значение Ngrid равно 1. В большинстве проектных задач достаточно значения 16, 32 или 64.
Edge	Данный параметр задает граничные частоты полос (т.е. нижние и верхние края полос фильтра). Все частоты должны вводиться в нормированной форме (относительно половины частоты дискретизации). Максимальное значение поддерживаемых полос (пропускания и/или подавления) — 10.

Отметим, что в последних реализациях данного алгоритма могут использоваться символы, отличные от указанных выше. Фактически некоторые реализации имеют более дружелюбный интерфейс к программе разработке, чем исходная реализация.

7.6.3. Соотношения для оценки длины фильтра N

На практике число коэффициентов фильтра неизвестно. Его можно оценить, используя одно из соотношений, приведенных ниже.

7.6.3.1. Фильтр нижних частот (согласно [4])

$$N \approx \frac{D_{\infty}(\delta_p, \delta_s)}{\Delta F} - f(\delta_p, \delta_s)\Delta F + 1, \quad (7.14)$$

где ΔF — ширина полосы пропускания, нормированная на частоту дискретизации,

$$\begin{aligned} D_{\infty}(\delta_p, \delta_s) &= \lg \delta_s [a_1 (\lg \delta_p)^2 + a_2 \lg \delta_p + a_3] + \\ &\quad + [a_4 (\lg \delta_p)^2 + a_5 \lg \delta_p + a_6] \\ f(\delta_p, \delta_s) &= 11,01217 + 0,51244 [\lg \delta_p - \lg \delta_s] \\ a_1 &= 5,309 \times 10^{-3} & a_2 &= 7,114 \times 10^{-2} \\ a_3 &= -4,761 \times 10^{-1} & a_4 &= -2,66 \times 10^{-3} \\ a_5 &= -5,941 \times 10^{-1} & a_6 &= -4,278 \times 10^{-1} \end{aligned}$$

Здесь δ_p — неравномерность в полосе пропускания, а δ_s — неравномерность, или колебание, характеристики в полосе подавления.

7.6.3.2. Полосовой фильтр (согласно [10])

$$N \approx \frac{C_{\infty}(\delta_p, \delta_s)}{\Delta F} + g(\delta_p, \delta_s)\Delta F + 1, \quad (7.15)$$

где

$$\begin{aligned} C_{\infty}(\delta_p, \delta_s) &= \lg \delta_s [b_1 (\lg \delta_p)^2 + b_2 \lg \delta_p + b_3] + \\ &\quad + [b_4 (\lg \delta_p)^2 + b_5 \lg \delta_p + b_6] \\ g(\delta_p, \delta_s) &= -14,6 \lg \left(\frac{\delta_p}{\delta_s} \right) - 16,9 \\ b_1 &= 0,01201 & b_2 &= 0,09664 \\ b_3 &= -0,51325 & b_4 &= 0,00203 \\ b_5 &= -0,5705 & b_6 &= -0,44314 \end{aligned}$$

Здесь δF — ширина полосы перехода, нормированная на частоту дискретизации.

В приложении к данной главе приводится программа на языке С, предназначенная для вычисления значения N на основе приведенных выше выражений (7.14) или (7.15).

7.6.4. Резюме по процедуре вычисления коэффициентов фильтра с помощью оптимального метода

- *Этап 1.* Задать граничные частоты полос (т.е. полос пропускания и полос подавления), неравномерность в полосе пропускания и затухание в полосе подавления (в децибелах или обычных единицах) и частоту дискретизации.

- *Этап 2.* Нормировать каждую граничную частоту, разделив ее на частоту дискретизации, и определить нормированную ширину полосы перехода.
- *Этап 3.* Использовать неравномерность в полосе пропускания и затухание в полосе перехода (см. примечание ниже) для оценки длины фильтра N из уравнения (7.14) или (7.15). Обычно значение N , требуемое для соответствия спецификации, берется на несколько единиц (2 или 3) больше, чем значение, определенное из этих уравнений.
- *Этап 4.* Получить весовые коэффициенты для каждой полосы из отношения колебаний в полосе пропускания и подавления, выраженных в обычных единицах. Весовые коэффициенты каждой полосы удобно представить целыми числами. Например, фильтр нижних частот с колебанием характеристики в полосе пропускания и полосе подавления 0,01 и 0,03 (неравномерность в полосе пропускания и затухание в полосе подавления равны соответственно 0,09 дБ и 30,5 дБ) будет иметь весовые коэффициенты 3 для полосы пропускания и 1 для полосы подавления. Колебания характеристики полосового фильтра, равные 0,001 в полосе пропускания и 0,0105 во всех полосах подавления, будут иметь весовые коэффициенты 21 для полосы пропускания и 2 для каждой полосы подавления.
- *Этап 5.* Ввести параметры в программу оптимальной разработки и получить следующие величины: N , граничные частоты полос и весовые коэффициенты для каждой полосы, а также подходящую плотность сетки (обычно 16 или 32).
- *Этап 6.* Проверить неравномерность в полосе пропускания и затухание в полосе подавления, полученные на выходе программы, на предмет соответствия спецификациям.
- *Этап 7.* Если спецификации не удовлетворяются, увеличить значение N и повторить этапы 5 и 6, пока соответствие не будет достигнуто; далее получить и проверить частотную характеристику, чтобы убедиться, что она удовлетворяет спецификации.

Следует отметить, что на этапе аппроксимации оптимальная программа рассматривает только полосу пропускания и полосу подавления, считая область перехода несущественной. Чтобы избежать проблем со сходимостью алгоритма, лучше всего при разработке полосовых фильтров или фильтров с несколькими полосами положить области перехода равными ширине наименьшей области перехода. Если используются разные полосы перехода, следует всегда проверять частотную характеристику на предмет соответствия спецификациям, поскольку в полосах перехода могут появляться локальные максимумы и минимумы, приводящие к неожиданным характеристикам фильтра.

7.6.5. Примеры

Ниже приводятся примеры, иллюстрирующие использование программы оптимальной разработки.

Пример 7.6

Требуется полосовой фильтр с линейной фазовой характеристикой, удовлетворяющий следующим спецификациям:

полоса пропускания	900–1100 Гц,
неравномерность в полосе пропускания	<0,87 дБ,
затухание в полосе подавления	>30 дБ,
частота дискретизации	15 кГц,
частота перехода	450 Гц.

Используйте оптимальный метод для получения подходящих коэффициентов. Изобразите спектр фильтра.

Решение

Из спецификации следует, что фильтр имеет три полосы: нижнюю полосу подавления (от 0 до 450 Гц), полосу пропускания (от 900 до 1100 Гц) и верхнюю полосу подавления (от 1550 до 7500 Гц). Для использования программы оптимальной разработки частоты краев полос нужно *нормировать*, т.е. выразить как доли частоты дискретизации:

$$\begin{aligned}
 450 &\rightarrow 450/15\,000 = 0,03, \\
 900 &\rightarrow 900/15\,000 = 0,06, \\
 1100 &\rightarrow 1100/15\,000 = 0,0733, \\
 1550 &\rightarrow 1550/15\,000 = 0,1033, \\
 7500 &\rightarrow 7500/15\,000 = 0,5.
 \end{aligned}$$

Следовательно, в нормированной форме три полосы записываются как (от 0 до 0,03), (от 0,06 до 0,0733) и (от 0,1033 до 0,5).

Далее для каждой полосы требуется выбрать весовой коэффициент. Весовые коэффициенты зависят от колебаний характеристики в полосе пропускания и полосе подавления. Колебание (в обычных единицах) можно получить из данных о неравномерности в полосе пропускания и затухании в полосе подавления:

$$\begin{aligned}
 \text{неравномерность } 0,87 \text{ дБ} : 20 \lg(1 + \delta_p) &\rightarrow \delta_p = 0,10535, \\
 \text{затухание } 30 \text{ дБ} : -20 \lg(\delta_s) &\rightarrow \delta_s = 0,031623.
 \end{aligned}$$

Отношение δ_p к δ_s равно $3,33 = 10/3$:

$$\frac{\delta_p}{\delta_s} = \frac{10}{3} = \frac{\text{ширина полосы подавления}}{\text{ширина полосы пропускания}}.$$

Следовательно, можно использовать весовой коэффициент 3 для полосы пропускания и коэффициент 10 для полосы подавления (отметим, что весовые коэффициенты присваиваются противоположно тому, как они входят в отношение δ_p/δ_s). С равным успехом можно применять весовой коэффициент 1 для полосы пропускания и 3,33 для полосы подавления. Плотность решетки равна 32. Используем программу оценки N на основе приведенных выше соотношений и находим, что длина фильтра равна 40. Возьмем значение $N = 41$.

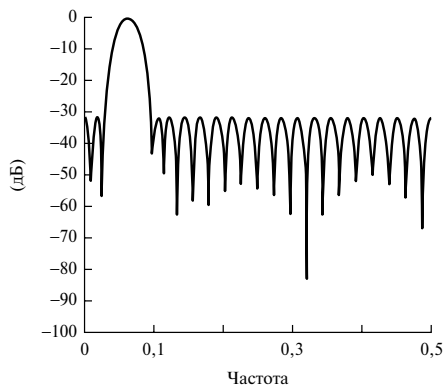


Рис. 7.13. Частотная характеристика фильтра (нормированная шкала частот)

Подытожим вход оптимальной программы:

длина фильтра, N	41,
тип фильтра, $Jtype$	1,
весовые коэффициенты, $W(\omega)$	10, 3, 10,
Ngrid	32,
граничные частоты	0; 0,03; 0,06; 0,0733; 0,1033; 0,5.

Распечатка выхода программы разработки приведена в табл. 7.7, а частотный спектр изображен на рис. 7.13. Прокомментируем результаты.

- Колебания в полосе пропускания в 3,33 раза больше, чем в полосе подавления. Это объясняется тем, что ошибки в полосе пропускания и полосах подавления учтены с весовыми коэффициентами 3 и 10 соответственно. Чем выше весовой коэффициент полосы, тем меньше колебания, или отклонения.
- Колебания характеристики в полосе пропускания и затухание в полосе подавления удовлетворяют спецификациям.
- Существует 22 экстремальные частоты, т.е. $(N + 3)/2$ максимума и минимума амплитудной характеристики. Обратите внимание на то, что края полос — это также экстремальные частоты, такими же являются частоты $f = 0$ и $f = 0,5$ Гц. Частоты краев полос всегда являются экстремальными.
- Импульсная характеристика симметрична относительно среднего коэффициента. Свойство симметрии — это необходимое условие наличия линейной фазовой характеристики. Обратите внимание на то, что для фильтра типа 1 средний коэффициент имеет наибольшее значение.

При поставленной задаче с фиксированными границами полос разработчик может нужным образом выровнять колебания в полосе пропускания и затухание в полосе подавления, используя весовые коэффициенты и N .

Таблица 7.7. Коэффициенты импульсной характеристики оптимальных фильтров (пример 7.6)

$H(1) =$	$-0,15346380E - 01$	$= H(41)$		
$H(2) =$	$-0,57805500E - 04$	$= H(40)$		
$H(3) =$	$0,50234820E - 02$	$= H(39)$		
$H(4) =$	$0,12667060E - 01$	$= H(38)$		
$H(5) =$	$0,21082060E - 01$	$= H(37)$		
$H(6) =$	$0,27764180E - 01$	$= H(36)$		
$H(7) =$	$0,30053620E - 01$	$= H(35)$		
$H(8) =$	$0,25869350E - 01$	$= H(34)$		
$H(9) =$	$0,14445660E - 01$	$= H(33)$		
$H(10) =$	$-0,31893230E - 02$	$= H(32)$		
$H(11) =$	$-0,24161370E - 01$	$= H(31)$		
$H(12) =$	$-0,44207120E - 01$	$= H(30)$		
$H(13) =$	$-0,58574530E - 01$	$= H(29)$		
$H(14) =$	$-0,63185570E - 01$	$= H(28)$		
$H(15) =$	$-0,55754610E - 01$	$= H(27)$		
$H(16) =$	$-0,36546910E - 01$	$= H(26)$		
$H(17) =$	$-0,85400990E - 02$	$= H(25)$		
$H(18) =$	$0,23083860E - 01$	$= H(24)$		
$H(19) =$	$0,52013800E - 01$	$= H(23)$		
$H(20) =$	$0,72248070E - 01$	$= H(22)$		
$H(21) =$	$0,79516810E - 01$	$= H(21)$		
	Полоса 1	Полоса 2	Полоса 3	
Нижняя граница полосы	0,000000000	0,060000000	0,103300000	
Верхняя граница полосы	0,030000000	0,073300000	0,500000000	
Искомое значение	0,000000000	1,000000000	0,000000000	
Весовой коэффициент	10,000000000	3,000000000	10,000000000	
Амплитуда колебаний	0,028891690	0,096305620	0,028891690	
Неравномерность, дБ	-30,784510000	0,798631800	-30,784510000	
Экстремальные частоты				
0,0000000	0,0208333	0,0300000	0,0600000	0,1033000
0,1122285	0,1308297	0,1538951	0,1777045	0,2015139
0,2260674	0,2506209	0,2759184	0,3004719	0,3257694
0,3503229	0,3756204	0,4001739	0,4254714	0,4500249
0,4753224	0,5000000			

Пример 7.7

Требуется цифровой узкополосный режекторный КИХ-фильтр, соответствующий таким спецификациям:

частота режекции	1,875 кГц,
подавление на частоте режекции	>60 дБ,
края полосы пропускания	1,575 и 2,175 кГц,
неравномерность в полосе пропускания	<0,01 дБ,
частота дискретизации	7,5 кГц,
число коэффициентов	61.

Для получения коэффициентов КИХ-фильтра, удовлетворяющего приведенной спецификации, используйте оптимальный метод.

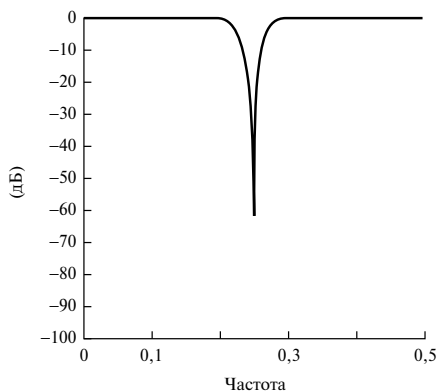


Рис. 7.14. Характеристика фильтра (нормированная шкала частот)

Решение

Фильтр имеет три полосы. Нормированные частоты трех полос и отклонения записываются следующим образом:

нижняя полоса пропускания	от 0 до 0,21,
частота режекции	0,25,
верхняя полоса пропускания	от 0,29 до 0,5,
отклонение в полосе пропускания	0,00115 (находится из $20 \lg(1 + \delta_p)$),
отклонение в полосе подавления	0,001 (находится из $-20 \lg(\delta_s)$).

Весовые коэффициенты полос равны 1; 1,1519; 1 (из отношения δ_p/δ_s). Результаты сведены в табл. 7.8 и на рис. 7.14. Обратите внимание на то, что при режекции полоса подавления состоит из единственной частоты.

Таким образом, в программу вводятся следующие граничные частоты: 0; 0,21; 0,25; 0,25; 0,29 и 0,5. Частота режекции вводится дважды, поэтому полоса подавления эффективно сжимается до одной частоты.

Пример 7.8

При разработке фильтра важно, чтобы разработчик понимал, как взаимодействуют параметры, чтобы при необходимости воспользоваться подходящими компромиссами. Данный пример помогает исследовать влияние параметров δ_p , δ_s , W и изучить различные возможности выбора.

Требуется КИХ-фильтр с линейной фазовой характеристикой, подавляющий шум физио-логической природы [2]. Фильтр предполагается использовать как часть большой ЦОС-системы реального времени, так что число коэффициентов следует максимально сократить. Характеристики фильтра должны удовлетворять следующим спецификациям:

Таблица 7.8. Коэффициенты импульсной характеристики оптимального фильтра (пример 7.7)

$H(1) =$	0, 12743640E – 02	$= H(61)$	
$H(2) =$	0, 26730640E – 05	$= H(60)$	
$H(3) =$	–0, 23681110E – 02	$= H(59)$	
$H(4) =$	–0, 17416350E – 05	$= H(58)$	
$H(5) =$	0, 43428480E – 02	$= H(57)$	
$H(6) =$	0, 53579250E – 05	$= H(56)$	
$H(7) =$	–0, 71570240E – 02	$= H(55)$	
$H(8) =$	–0, 49028620E – 05	$= H(54)$	
$H(9) =$	0, 10897540E – 01	$= H(53)$	
$H(10) =$	0, 89629280E – 05	$= H(52)$	
$H(11) =$	–0, 15605960E – 01	$= H(51)$	
$H(12) =$	–0, 85508990E – 05	$= H(50)$	
$H(13) =$	0, 21226410E – 01	$= H(49)$	
$H(14) =$	0, 12250150E – 04	$= H(48)$	
$H(15) =$	–0, 27630130E – 01	$= H(47)$	
$H(16) =$	–0, 11091200E – 04	$= H(46)$	
$H(17) =$	0, 34579770E – 01	$= H(45)$	
$H(18) =$	0, 13800660E – 04	$= H(44)$	
$H(19) =$	–0, 41774130E – 01	$= H(43)$	
$H(20) =$	–0, 11560390E – 04	$= H(42)$	
$H(21) =$	0, 48832790E – 01	$= H(41)$	
$H(22) =$	0, 12787590E – 04	$= H(40)$	
$H(23) =$	–0, 55359840E – 01	$= H(39)$	
$H(24) =$	–0, 90065860E – 05	$= H(38)$	
$H(25) =$	0, 60944450E – 01	$= H(37)$	
$H(26) =$	0, 88997300E – 05	$= H(36)$	
$H(27) =$	–0, 65232190E – 01	$= H(35)$	
$H(28) =$	–0, 38167120E – 05	$= H(34)$	
$H(29) =$	0, 67925720E – 01	$= H(33)$	
$H(30) =$	0, 27041150E – 05	$= H(32)$	
$H(31) =$	0, 93115220E + 00	$= H(31)$	
	Полоса 1	Полоса 2	Полоса 3
Нижняя граница полосы	0, 000000000	0, 250000000	0, 290000000
Верхняя граница полосы	0, 210000000	0, 250000000	0, 500000000
Искомое значение	1, 000000000	0, 000000000	1, 000000000
Весовой коэффициент	1, 000000000	1, 151900000	1, 000000000
Амплитуда колебаний	0, 000978727	0, 000849663	0, 000978727
Неравномерность, дБ	0, 008496785	–61, 414990000	0, 008496785

Экстремальные частоты

0, 0000000	0, 0161290	0, 0322580	0, 0483871	0, 0645161
0, 0806451	0, 0962701	0, 1123991	0, 1280241	0, 1431450
0, 1582660	0, 1728829	0, 1864918	0, 1980845	0, 2066530
0, 2100000	0, 2500000	0, 2900000	0, 2930243	0, 3020971
0, 3136902	0, 3272994	0, 3414128	0, 3565342	0, 3721596
0, 3877850	0, 4034105	0, 4195400	0, 4356695	0, 4517990
0, 4679285	0, 4840580			

неравномерность в полосе пропускания	<0,026 дБ,
полоса подавления	>30 дБ,
край полосы пропускания	10 Гц,
край полосы подавления	<20 Гц,
частота дискретизации	128 кГц.

Решение

Нормированные граничные частоты, неравномерность характеристики в полосе пропускания и полосе подавления записываются следующим образом:

край полосы пропускания	0,078,
край полосы подавления	<0,15625,
максимальное отклонение в полосе пропускания	<0,003,
отклонение в полосе подавления	>0,0316.

Поскольку большинство спецификаций фильтра — переменные, очевидно, будет существовать целый ряд возможных решений. Следовательно, необходимо найти наилучшее решение.

Подставляя приведенные выше значения в уравнение (7.14), получаем, что $N > 25,6$ (наименьшее возможное значение N). Затем для каждого значения N в диапазоне 25–37 вычисляется удовлетворяющая спецификациям граничная частота полосы подавления f_s :

$$f_s = f_p + \Delta f,$$

где f_s и f_p — граничные частоты полосы подавления и полосы пропускания, а Δf — ширина полосы перехода, которая определяется следующим образом ($\Delta f_{\max} = 20 - 10 \text{ Гц} = 10 \text{ Гц}$):

$$\Delta f = \frac{N - 1}{2f(\delta_p, \delta_s)} \left[1 + \frac{4f(\delta_p, \delta_s)D_{\infty}(\delta_p, \delta_s) - 1}{(N - 1)^2} \right]^{1/2}.$$

На рис. 7.15 представлено пространство решений (над кривой), которое ограничено частотой полосы подавления в 20 Гц и $N = 26$ и 37. В качестве подходящего решения выбрано значение 27. Нечетное значение N предпочтительнее, поскольку так время задержки фильтра, измеренное в количестве выборок, будет целым. Используются следующие параметры: полоса пропускания — от 0 до 0,078, полоса подавления — от 0,1523885 до 0,5 (т.е. от 19 до 64 Гц). Для полос пропускания и подавления выбраны соответственно весовые коэффициенты 10,5 и 1. Полученные в результате коэффициенты фильтра и параметры приведены в табл. 7.9. Из параметров фильтра и спектра (не приводится) видно, что спецификация удовлетворяется.

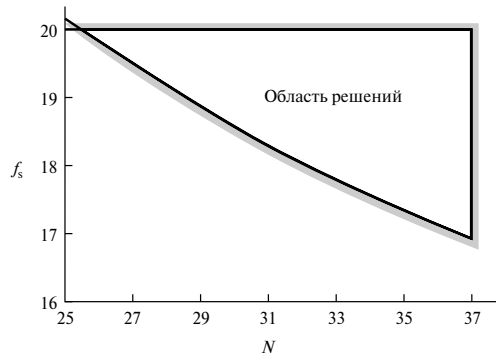


Рис. 7.15. Зависимость положения края полосы подавления от длины фильтра с указанным диапазоном возможных решений

7.7. Метод частотной выборки

Метод частотной выборки позволяет разрабатывать нерекурсивные КИХ-фильтры, в число которых входят как обычные частотно-избирательные фильтры (фильтры нижних частот, верхних частот, полосовые), так и фильтры с произвольной частотной характеристикой. Уникальное достоинство метода частотной выборки заключается в том, что он допускает рекурсивные реализации КИХ-фильтров, что позволяет получать вычислительно эффективные фильтры. При некоторых условиях можно даже разработать рекурсивные КИХ-фильтры, коэффициенты которых — целые числа, что удобно, если допустимы только примитивные арифметические операции (это справедливо, например, для систем, реализованных на стандартных микропроцессорах).

7.7.1. Нерекурсивные фильтры частотной выборки

Предположим, что требуется получить коэффициенты КИХ-фильтра, частотная характеристика которого изображена на рис. 7.16, *a*. Для начала можно взять N выборок частотной характеристики в точках kF_s/n , $k = 0, 1, \dots, N - 1$. Коэффициенты фильтра $h(n)$ можно получить, применив обратное ДПФ к частотным выборкам:

$$h(n) = \frac{1}{N} \sum_{k=0}^{N-1} H(k) e^{i(2\pi/N)nk}, \quad (7.16)$$

где $H(k)$, $k = 0, 1, \dots, N - 1$ — выборки идеальной или целевой частотной характеристики.

Таблица 7.9. Коэффициенты импульсной характеристики оптимального фильтра (пример 7.8)

$H(1) =$	$-0,13614960E - 01$	$= H(27)$
$H(2) =$	$0,34793330E - 02$	$= H(26)$
$H(3) =$	$0,11140420E - 01$	$= H(25)$
$H(4) =$	$0,16664540E - 01$	$= H(24)$
$H(5) =$	$0,12807340E - 01$	$= H(23)$
$H(6) =$	$-0,33202110E - 02$	$= H(22)$
$H(7) =$	$-0,26167170E - 01$	$= H(21)$
$H(8) =$	$-0,42047790E - 01$	$= H(20)$
$H(9) =$	$-0,34767040E - 01$	$= H(19)$
$H(10) =$	$0,55338630E - 02$	$= H(18)$
$H(11) =$	$0,75072090E - 01$	$= H(17)$
$H(12) =$	$0,15527810E + 00$	$= H(16)$
$H(13) =$	$0,21933680E + 00$	$= H(15)$
$H(14) =$	$0,24378330E + 00$	$= H(14)$
	Полоса 1	Полоса 2
Нижняя граница полосы	0,000000000	0,152388500
Верхняя граница полосы	0,078000000	0,500000000
Искомое значение	1,000000000	0,000000000
Весовой коэффициент	10,500000000	1,000000000
Амплитуда колебаний	0,002604177	0,027343860
Неравномерность, дБ	0,022589730	-31,262770000

Экстремальные частоты

0,0089286	0,0468750	0,0691964	0,0780000	0,1523885
0,1668974	0,1981473	0,2338614	0,2706916	0,3086379
0,3465842	0,3845305	0,4235928	0,4615391	0,5000000

Можно показать (см. пример 7.9), что для фильтров с линейной фазовой характеристикой и четно-симметричной импульсной характеристикой можно записать (если N — четное)

$$h(n) = \frac{1}{N} \left[\sum_{k=1}^{N/2-1} 2H(k) \cos[2\pi k(n - \alpha)/N] + H(0) \right], \quad (7.17)$$

где $\alpha = (N - 1)/2$. Если N — нечетное, верхним пределом суммы является $(N - 1)/2$. Получающийся фильтр будет иметь частотную характеристику, которая в точности совпадает с исходной характеристикой в моменты выборки. В то же время, для разных моментов выборки характеристики могут сильно отличаться (рис. 7.16, в). Для получения хорошей аппроксимации частотной характеристики нужно взять достаточное число частотных выборок.

Альтернативный фильтр (фильтр типа 2), построенный по принципу частотной выборки, получается, если выборки брать в точках

$$f_k = (k + 1/2)F_s/N, \quad k = 0, 1, \dots, N - 1. \quad (7.18)$$

Две схемы частотной выборки сравниваются на рис. 7.17. При данной спецификации фильтра оба метода дадут несколько отличные частотные характеристики. Задача разработчика — определить, какой фильтр лучше подходит для поставленной задачи.

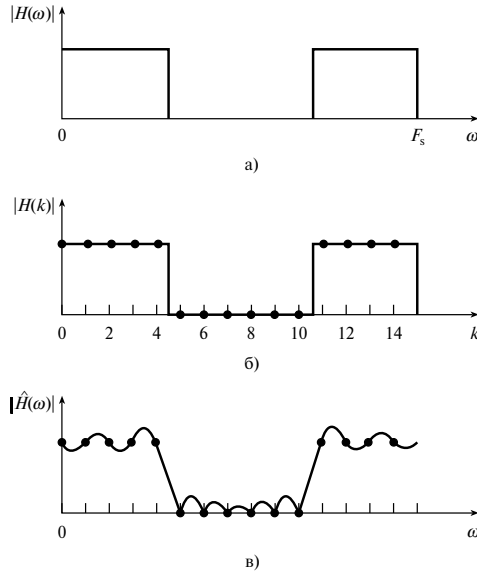


Рис. 7.16. Понятие частотной выборки: а) частотная характеристика идеального фильтра нижних частот; б) выборки идеального фильтра нижних частот; в) частотная характеристика фильтра нижних частот, выведенная из частотных выборок панели б

Пример 7.9

1. Покажите, что коэффициенты импульсной характеристики КИХ-фильтра с линейной фазовой характеристикой, четно-симметричными коэффициентами при N — четном можно выразить следующим образом:

$$h(n) = \frac{1}{N} \left[\sum_{k=1}^{N/2-1} 2H(k) \cos[2\pi k(n - \alpha)/N] + H(0) \right],$$

где $\alpha = (N - 1)/2$ и $N(k)$ — выборки частотной характеристики фильтра, взятые в точках kF_s/N .

2. Требуется, чтобы КИХ-фильтр нижних частот удовлетворял следующим спецификациям:

полоса пропускания 0–5 кГц,

частота дискретизации 18 кГц,

длина фильтра 9.

Получите коэффициенты фильтра, используя метод частотной выборки.

Решение

1.

$$h(n) = \frac{1}{N} \sum_{k=0}^{N-1} H(k) e^{i(2\pi/N)nk} = \quad (7.19)$$

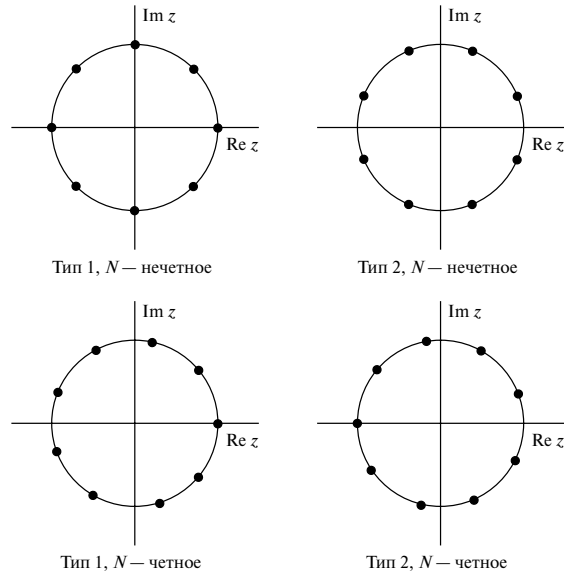


Рис. 7.17. Четыре возможные структуры выборки для двух типов фильтров (изображены на комплексной плоскости)

$$\begin{aligned}
 &= \frac{1}{N} \sum_{k=0}^{N-1} |H(k)| e^{-i2\pi\alpha k/N} e^{i2\pi kn/N} = \\
 &= \frac{1}{N} \sum_{k=0}^{N-1} |H(k)| e^{i2\pi(n-\alpha)/N} = \\
 &= \frac{1}{N} \sum_{k=0}^{N-1} |H(k)| \cos[2\pi(n-\alpha)/N] + i \sin[2\pi(n-\alpha)/N] = \\
 &= \frac{1}{N} \sum_{k=0}^{N-1} |H(k)| \cos[2\pi(n-\alpha)/N], \tag{7.20}
 \end{aligned}$$

поскольку $h(n)$ — целиком действительная функция. В важном частном случае линейной фазовой характеристики $h(n)$ будет симметричной, так что можно записать

$$h(n) = \frac{1}{N} \left[\sum_{k=0}^{N/2-1} 2|H(k)| \cos[2\pi(n-\alpha)/N + H(0)] \right]. \tag{7.21}$$

Если N — нечетное, верхняя граница суммы — $(N-1)/2$.

- Идеальная частотная характеристика изображена на рис. 7.18, а. Частотные выборки берутся в точках kF_s/N , т.е. с интервалом $18/9 = 2$ кГц. Итак, частотные выборки

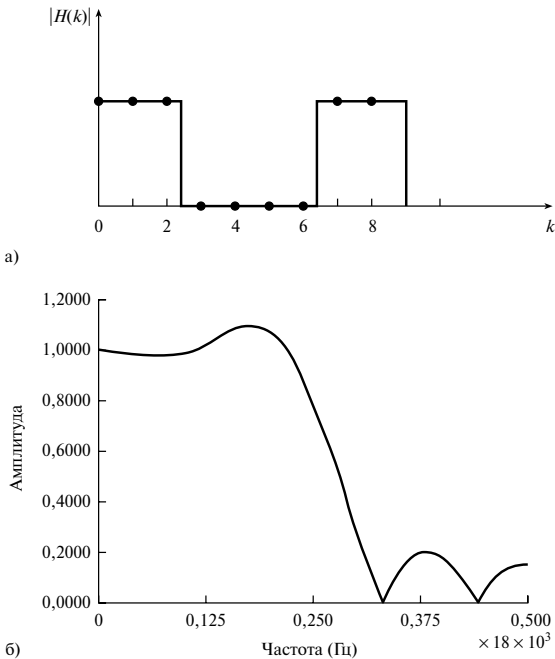


Рис. 7.18. Идеальная частотная характеристика, на которой показаны точки выборки (панель а). Частотная характеристика фильтра, построенного по принципу частотной выборки (панель б)

Таблица 7.10. Нерекурсивные коэффициенты КИХ-фильтра из примера 7.9

$h[0] =$	$7,2522627e - 02$	$= h[8]$
$h[1] =$	$-1,1111111e - 01$	$= h[7]$
$h[2] =$	$-5,9120987e - 02$	$= h[6]$
$h[3] =$	$3,1993169e - 01$	$= h[5]$
$h[4] =$	$5,5555556e - 01$	$= h[4]$

представлены следующим образом:

$$\begin{aligned} |H(k)| &= 1 & k &= 0, 1, 2 \\ &0 & k &= 3, 4 \end{aligned}$$

Используя уравнение (7.21) с верхним пределом $(N - 1)/2$ и частотные выборки, получаем коэффициенты импульсной характеристики (см. табл. 7.10). С помощью программы вычисления коэффициентов КИХ-фильтра по данным значениям частотных выборок ([Ifeachor, 2001]) получаем частотную характеристику фильтра, приведенную на рис. 7.18, б. Видно, что фильтр имеет плохую амплитудную характеристику, что вызвано резким переходом полосы пропускания (где $|H(k)| = 1$) в полосу подавления (где $|H(k)| = 0$).

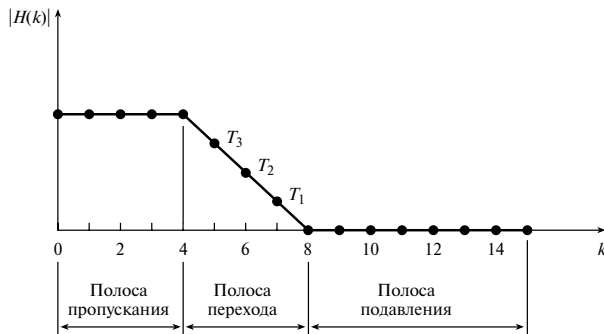


Рис. 7.19. Частотные выборки фильтра нижних частот, включая три выборки в полосе перехода. *Примечание:* поскольку амплитудная характеристика симметрична, показана только половина характеристики фильтра

7.7.1.1. Оптимизация амплитудной характеристики

Описанная выше задача сродни задаче с прямоугольной весовой функцией. Напомним, что в методе вырезания можно пожертвовать более широкой полосой передачи для улучшения амплитудной характеристики. Чтобы улучшить амплитудную характеристику фильтров, полученных по принципу частотной выборки, можно отказаться от более широкой полосы перехода и ввести в полосу перехода дополнительные частотные выборки. На рис. 7.19 приведена типичная спецификация фильтра нижних частот с тремя частотными выборками в полосе перехода. Для фильтра нижних частот затухание в полосе подавления увеличивается приблизительно на 20 дБ для каждой частотной выборки в полосе перехода [15] с соответствующим увеличением ширины перехода:

приблизительное затухание в полосе подавления $(25 + 20M)$ дБ,

приблизительная ширина перехода $(M + 1)F_s/N$.

Здесь M — число выборок в полосе перехода, N — длина фильтра.

Значения выборок в полосе перехода, которые дают оптимальное затухание в полосе подавления, определяются через процесс оптимизации [15]. Полезной целью оптимизации является поиск значений выборок в полосе перехода, $T_1, T_2, \dots, T_M$, которые минимизируют максимальную неравномерность в полосе перехода (т.е. максимизируют затухание в полосе подавления). Математически это можно сформулировать так:

$$\text{минимизировать } T_1, T_2, \dots, T_m \left[\begin{array}{c} \max |W[H_d(\omega) - H(\omega)]| \\ \{\omega \text{ принадлежит полосе подавления}\} \end{array} \right], \quad (7.22)$$

где $H_D(\omega)$ и $H(\omega)$ — идеальная и реальная частотные характеристики фильтра соответственно; W — весовой коэффициент.

В работе [15] приводится таблица оптимальных (согласно уравнению (7.22)) значений частотных выборок в полосе перехода, причем эти значения довольно широко используются на практике. Пример оптимальных значений выборок в полосе перехода приведен в табл. 7.11 для $N = 15$. В этой таблице *шириной полосы* называется число частотных выборок в полосе пропускания фильтра.

Таблица 7.11. Оптимальные частотные выборки в полосе перехода для фильтров нижних частот типа 1, полученных методом частотной выборки для $N = 15$ (адаптировано из [15])

ШП	Затухание в полосе подавления (дБ)	T_1	T_2	T_3
Одна выборка в полосе перехода, $N = 15$				
1	42,309 322 83	0,433 782 96		
2	41,262 992 86	0,417 938 23		
3	41,253 337 86	0,410 473 63		
4	41,949 077 13	0,404 058 84		
5	44,371 245 38	0,392 681 89		
6	56,014 165 88	0,357 665 25		
Две выборки в полосе перехода, $N = 15$				
1	70,605 405 85	0,095 001 22	0,589 954 18	
2	69,261 681 56	0,103 198 24	0,593 571 18	
3	69,919 734 95	0,100 836 18	0,589 432 70	
4	75,511 722 56	0,084 074 93	0,557 153 12	
5	103,460 783 00	0,051 802 06	0,499 174 24	
Три выборки в полосе перехода, $N = 15$				
1	94,611 661 91	0,014 550 78	0,184 578 82	0,668 976 13
2	104,998 130 80	0,010 009 77	0,173 607 13	0,659 515 26
3	114,907 193 18	0,008 734 13	0,163 973 10	0,647 112 64
4	157,292 575 84	0,003 787 99	0,123 939 63	0,601 811 54

В данном контексте ширина полосы (ШП) — это число частотных выборок в полосе пропускания.

В большинстве случаев значения частотных выборок в полосе перехода обычно принадлежат следующим диапазонам: для одной частотной выборки в полосе перехода

$$0,250 < T_1 < 0,450;$$

для двух частотных выборок в полосе перехода

$$0,040 < T_1 < 0,150,$$

$$0,450 < T_2 < 0,650;$$

для трех частотных выборок в полосе перехода

$$0,003 < T_1 < 0,035,$$

$$0,100 < T_2 < 0,300,$$

$$0,550 < T_3 < 0,750.$$

Более низкие значения используются в фильтрах с более широкими полосами и дают большее затухание в полосе подавления.

Пример 7.10

1. Пятнадцатиточечный КИХ-фильтр с линейной фазовой характеристикой представлен следующими частотными выборками:

$$|H(k)| = 1 \quad k = 0, 1, 2, 3, \\ 0 \quad k = 4, 5, 6, 7.$$

Предполагая, что частота дискретизации равна 2 кГц, определите частотную характеристику.

2. Сравните частотную характеристику фильтра, если а) используется одна частотная выборка в полосе перехода; б) используются две частотные выборки в полосе перехода; в) используются три частотные выборки в полосе перехода.

Решение

1. Вводя частотные выборки в программу разработки `fresamp.c` (см. приложение), получаем коэффициенты фильтра, которые приведены в столбце 2 табл. 7.12. Соответствующая частотная характеристика приведена на рис. 7.20, а.
2. В ситуации а значение частотной выборки в полосе перехода находим из табл. 7.11 — оно равно 0,4041. Следовательно, получаем такие выборки:

$$|H(k)| = 1 \quad k = 0, 1, 2, 3, \\ 0,40406 \quad k = 4, \\ 0 \quad k = 5, 6, 7.$$

Вводя эти частотные выборки в программу, вычисляем коэффициенты фильтра (сведены в табл. 7.12). Соответствующая частотная характеристика приведена на рис. 7.20, б.

В ситуации б и в частотные выборки определяются аналогичным образом:

$$|H(k)| = 1 \quad k = 0, 1, 2, 3, \\ 0,5571 \quad k = 4, \\ 0,0841 \quad k = 5, \\ 0 \quad k = 6, 7, \\ |H(k)| = 1 \quad k = 0, 1, 2, 3, \\ 0,6018 \quad k = 4, \\ 0,1239 \quad k = 5, \\ 0,0038 \quad k = 6, \\ 0 \quad k = 7.$$

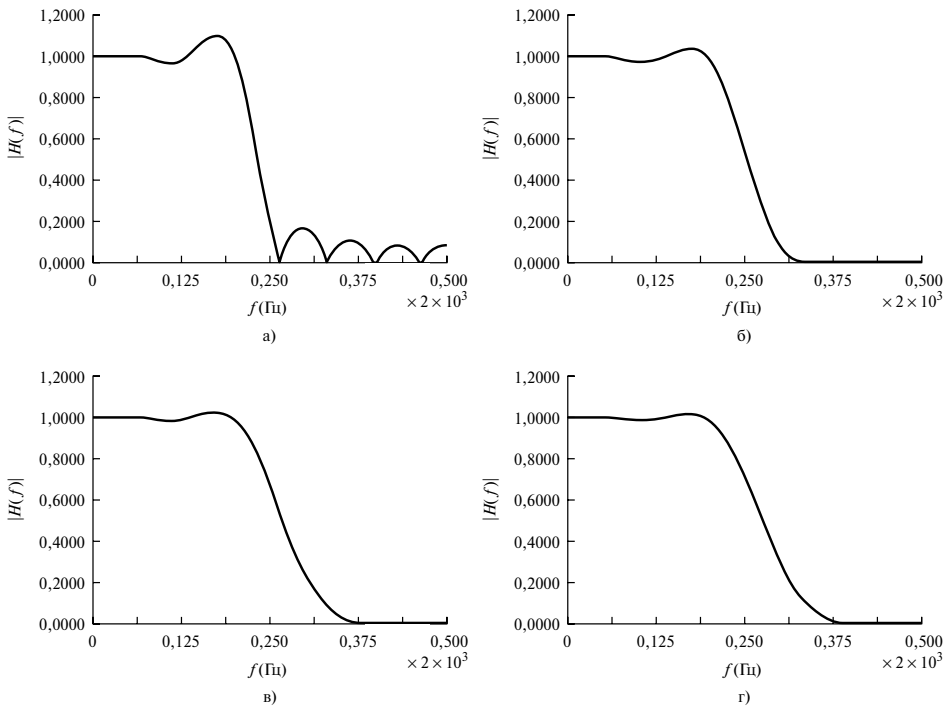


Рис. 7.20. Частотная характеристика фильтра частотной выборки при следующих условиях: а) в полосе перехода нет частотных выборок; б) одна выборка в полосе перехода; в) две выборки в полосе перехода; г) три выборки в полосе перехода

Коэффициенты для этих задач приведены в четвертом и пятом столбцах табл. 7.12. Соответствующие частотные характеристики изображены на рис. 7.20, в и г. Видно, что по мере увеличения числа выборок в полосе перехода амплитудная характеристика улучшается (с позиции неравномерности в полосе пропускания и полосе подавления) за счет увеличения ширины перехода или сглаживания характеристики.

Альтернативный подход, который можно использовать для улучшения амплитудной характеристики, — получить большое число частотных выборок, проводя дискретизацию через меньшие интервалы, вычислив импульсную характеристику с помощью формулы (7.21), а затем применив одну из рассмотренных ранее весовых функций, чтобы сократить фильтр до нужной длины.

7.7.1.2. Автоматическая разработка фильтров частотной выборки

Как отмечалось ранее, существуют таблицы оптимальных значений частотных выборок в полосе перехода (см., например, [15]), и эти таблицы широко используются для разработки фильтров частотной выборки. Если разработчик хочет создать фильтр, данных для которого нет в таблице, то с помощью линейной интерполяции можно получить приблизительные значения частотных выборок в полосе перехода. Однако это

Таблица 7.12. Коэффициенты нерекурсивного фильтра при различном числе выборок в полосе перехода

	Нет выборок в полосе перехода	Одна выборка в полосе перехода	Две выборки в полосе перехода	Три выборки в полосе перехода
$h[0] =$	$-4,9815884e-02$	$-1,3766696e-02$	$-5,7195305e-03$	$-4,2282741e-03$
$h[1] =$	$4,1202267e-02$	$-2,3832554e-03$	$-7,6781827e-03$	$-7,6031627e-03$
$h[2] =$	$6,6666666e-02$	$3,9729333e-02$	$2,3920000e-02$	$1,8793332e-02$
$h[3] =$	$-3,6487877e-02$	$1,2729081e-02$	$2,5763613e-02$	$2,8145113e-02$
$h[4] =$	$-1,0786893e-01$	$-9,1220745e-02$	$-7,3701817e-02$	$-6,6396840e-02$
$h[5] =$	$3,4078020e-02$	$-1,8619356e-02$	$-4,4185450e-02$	$-5,2511978e-02$
$h[6] =$	$3,1889241e-01$	$3,1326097e-01$	$3,0552137e-01$	$3,0183514e-01$
$h[7] =$	$4,6666667e-01$	$5,2054133e-01$	$5,5216000e-01$	$5,6393334e-01$

Поскольку коэффициенты симметричны, представлена только первая их половина.

возможно не всегда, особенно если требуется большое число выборок в полосе перехода. Более того, информация в таблицах приводится не в том виде, к которому привыкли разработчики фильтров; например, не указываются края полос и неравномерность в полосе пропускания. Не так давно была разработана универсальная компьютерная программа, автоматизирующая многие аспекты разработки нерекурсивных и рекурсивных фильтров частотной выборки [3, 15]. По сути, в этой программе значения выборок в полосе перехода оптимизируются с помощью смешанного генетического алгоритма. Цель оптимизации — получить максимальное затухание в полосе подавления для данного набора спецификаций фильтра. Использованный подход тестировался на предмет соответствия табличным результатам, приведенным в литературе, и в каждом случае признавался равносильным или более удачным. Кроме того, данная программа позволяет разрабатывать непротабулированные фильтры.

Пример 7.11

Найдите оптимальные выборки в полосе перехода и соответствующие коэффициенты для фильтра нижних частот, согласно таким спецификациям:

край полосы пропускания 0,143 (нормировано),
 край полосы подавления 0,245 (нормировано),
 число коэффициентов фильтра 49.

Решение

Из спецификации следует, что число частотных выборок $N = 49$. Номера выборок, соответствующих краям полосы пропускания и полосы подавления, равны соответственно 6 и 12. Число выборок в полосе перехода $M = 5$. Следовательно, частотные выборки идеальной амплитудно-частотной характеристики имеют вид

$$\begin{aligned}
 |H(k)| &= 1, & k &= 0, 1, \dots, 6, \\
 T_{k-6}, & & k &= 7, \dots, 11, \\
 0, & & k &= 12, \dots, 24.
 \end{aligned}$$

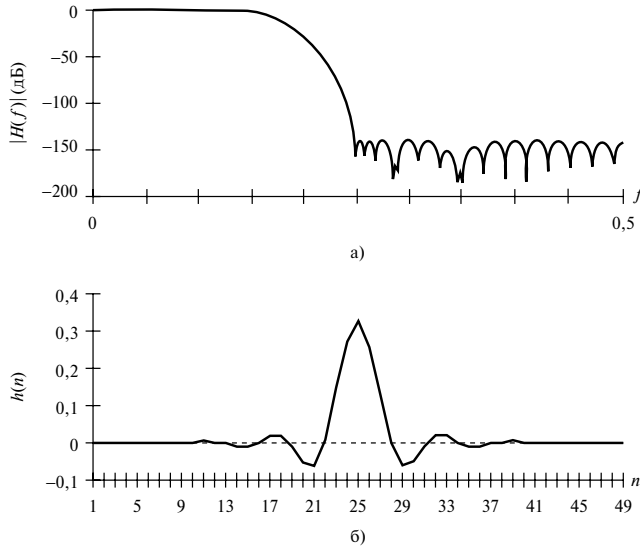


Рис. 7.21. Интерполированная частотная характеристика (панель а); коэффициенты фильтра (панель б). Неравномерность в полосе пропускания — 0,046 дБ; затухание в полосе подавления — 139,64 дБ; ширина полосы пропускания — 0,15; пять выборок в полосе перехода; 49 коэффициентов фильтра. Значения выборок в полосе перехода — 0,855456; 0,485507; 0,148961; 0,019693; 0,000644

Значения от T_1 до T_5 не заданы и находятся в процессе оптимизации с помощью программы смешанного генетического алгоритма. Результаты процесса оптимизации сведены на рис. 7.21.

Хотя смешанный генетический алгоритм дает результаты, которые несколько лучше, чем приведенные в литературе, его основное достоинство — возможность быстро получить коэффициенты непротабулированных фильтров. Причем в этом случае коэффициенты гораздо лучше тех, что получаются по схеме интерполяции. Кроме того, так можно разрабатывать фильтры с большим числом выборок в полосе перехода.

7.7.2. Рекурсивные фильтры частотной выборки

Фильтры частотной выборки в рекурсивной форме значительно выгоднее вычислительно, чем фильтры в нерекурсивной форме, если значительное число частотных выборок имеет нулевые значения. Можно показать (см. пример 7.12), что передаточную функцию КИХ-фильтра $H(z)$ можно следующим образом записать в рекурсивном виде:

$$H(z) = \frac{1 - z^{-N}}{N} \sum_{k=0}^{N-1} \frac{H(k)}{1 - e^{2\pi i k/N} z^{-1}} = H_1(z) H_2(z), \quad (7.23)$$

где

$$H_1(z) = \frac{1 - z^{-N}}{N}$$

$$H_2(z) = \sum_{k=0}^{N-1} \frac{H(k)}{1 - e^{2\pi i k/N} z^{-1}}.$$

Таким образом, очевидно, что в рекурсивной форме $H(z)$ можно рассматривать как каскад из двух фильтров: гребенчатого фильтра $H_1(z)$, который имеет N нулей, равномерно распределенных на единичной окружности, и суммы N фильтров с одним полюсом $H_2(z)$. Нули гребенчатого фильтра и полюса однополюсных фильтров совпадают на единичной окружности в точках $z_k = e^{1\pi k/N}$. Следовательно, нули компенсируют полюса, и поскольку $H(z)$ не имеет полюсов, то это — конечная импульсная характеристика (КИХ).

На практике конечная длина слова приводит к тому, что полюса $H_2(z)$ располагаются не точно на единичной окружности, так что они уже не уравниваются нулями и $H(z)$ становится потенциально неустойчивой бесконечной импульсной характеристикой (БИХ). Проблем устойчивости можно избежать, дискретизируя $H(z)$ на окружности радиуса r , который незначительно меньше единицы. В этом случае передаточная функция становится такой:

$$H(z) = \frac{1 - r^N z^{-N}}{N} \sum_{k=0}^{N-1} \frac{H(k)}{1 - r e^{2\pi i k/N} z^{-1}}. \quad (7.24)$$

Вообще, частотные выборки $H(k)$ — это комплексные величины. Следовательно, непосредственная реализация уравнения (7.23) или (7.24) потребует комплексной арифметики. Чтобы избежать этого усложнения, воспользуемся симметрией, присущей частотной характеристике любого КИХ-фильтра с действительной импульсной характеристикой $h(n)$. Можно показать (см. пример 7.12), что для обычного частотно-избирательного фильтра с линейной фазовой характеристикой (четно-симметричная импульсная характеристика) уравнение (7.24) можно представить в виде

$$H(z) = \frac{1 - r^N z^{-N}}{N} \times$$

$$\times \left[\sum_{k=1}^M \frac{|H(k)| 2 \cos(2\pi k \alpha / N) - 2r \cos[2\pi k(1 + \alpha)/N] z^{-1}}{1 - 2r \cos(2\pi k/N) z^{-1} + r^2 z^{-2}} + \frac{H(0)}{1 - z^{-1}} \right], \quad (7.25)$$

где $\alpha = (N - 1)/2$. При нечетном N $M = (N - 1)/2$, при четном N $M = N/2 - 1$. Диаграмма реализации уравнения (7.25) приведена на рис. 7.22.

7.7.3. Фильтры частотной выборки с простыми коэффициентами

Рекурсивная реализация КИХ-фильтров значительно уменьшает число арифметических операций, производимых в цифровых фильтрах. Если фильтр к тому же имеет коэффициенты, которые являются просто целыми числами (или степенями двойки),

Пример 7.12

1. Передаточная функция КИХ-фильтра определена следующим образом:

$$H(z) = \sum_{n=0}^{N-1} h(n)z^{-n}. \quad (7.26, a)$$

Начав с приведенного выражения, покажите, что $H(z)$ для КИХ-фильтра с линейной фазовой характеристикой и четно-симметричной импульсной характеристикой можно следующим образом выразить в рекурсивной форме:

$$H(z) = \frac{1 - r^N z^{-N}}{N} \times \left[\sum_{k=1}^M \frac{|H(k)| 2 \cos(2\pi k \alpha / N) - 2r \cos[2\pi k(1 + \alpha)/N] z^{-1}}{1 - 2r \cos(2\pi k / N) z^{-1} + r^2 z^{-2}} + \frac{H(0)}{1 - z^{-1}} \right],$$

где $\alpha = (N - 1)/2$, а $H(k)$ — выборки частотной характеристики фильтра, взятые в точках kF_s/N .

2. Требуется, чтобы фильтр нижних частот удовлетворял следующим спецификациям:
- | | |
|-----------------------|----------|
| полоса пропускания | 0–4 кГц, |
| частота дискретизации | 18 кГц, |
| длина фильтра | 9. |

Найдите передаточную функцию фильтра в рекурсивной форме, используя метод частотной выборки и учитывая, что радиус равен $r = 1$. Изобразите диаграмму реализации и сравните ее вычислительную сложность со сложностью реализации КИХ в прямой форме.

Решение

1. Импульсную характеристику фильтра можно определить через его частотные выборки:

$$h(n) = \frac{1}{N} \sum_{k=0}^{N-1} H(k) r^n e^{2\pi i n k / N} \quad k = 0, 1, \dots, N - 1, r \leq 1. \quad (7.26, б)$$

Используя уравнения (7.26, а и б), получаем следующую передаточную функцию $H(z)$:

$$H(z) = \sum_{n=0}^{N-1} h(n) z^{-n} = \sum_{n=0}^{N-1} \left[\frac{1}{N} \sum_{k=0}^{N-1} H(k) r^n e^{2\pi i n k / N} \right] z^{-n}.$$

Меняя порядок суммирования, получаем

$$H(z) = \frac{1}{N} \sum_{k=0}^{N-1} H(k) \left\{ \sum_{n=0}^{N-1} [r e^{2\pi i n k / N} z^{-1}]^n \right\}. \quad (7.27)$$

Далее сумму геометрической прогрессии можно выразить как

$$S_N = \sum_{n=0}^{N-1} \delta^n = \frac{1 - \delta^N}{1 - \delta} \quad \delta \neq 1.$$

В нашем случае при $\delta = re^{2\pi ik/N} z^{-1}$ можно записать

$$\begin{aligned} \sum_{n=0}^{N-1} [re^{2\pi ik/N} z^{-1}]^n &= \frac{1 - (re^{2\pi ik/N} z^{-1})^N}{1 - re^{2\pi ik/N} z^{-1}} = \frac{1 - r^N e^{2\pi ik} z^{-N}}{1 - re^{2\pi ik/N} z^{-1}} = \\ &= \frac{1 - r^N z^{-N}}{1 - re^{2\pi ik/N} z^{-1}}, \end{aligned}$$

поскольку $e^{2\pi ik} = \cos(2\pi k) = 1, k = 0, 1, \dots$

Следовательно, формулу (7.27) можно переписать так:

$$H(z) = \frac{1 - r^N z^{-N}}{N} \sum_{k=0}^{N-1} \frac{H(k)}{1 - re^{2\pi ik/N} z^{-1}} = H_1(z) H_2(z), \quad (7.28)$$

где

$$\begin{aligned} H_1(z) &= \frac{1 - r^N z^{-N}}{N}, \\ H_2(z) &= \sum_{k=0}^{N-1} \frac{H(k)}{1 - re^{2\pi ik/N} z^{-1}}. \end{aligned}$$

Далее, разложение $H_2(z)$ имеет вид

$$\begin{aligned} H_2(z) &= \frac{H(0)}{1 - rz^{-1}} + \frac{H(1)}{1 - re^{2\pi i/N} z^{-1}} + \frac{H(2)}{1 - re^{2\pi i2/N} z^{-1}} + \\ &+ \dots + \frac{H(N-2)}{1 - re^{2\pi i(N-2)/N} z^{-1}} + \frac{H(N-1)}{1 - re^{2\pi i(N-1)/N} z^{-1}}. \end{aligned}$$

Для фильтра с действительными коэффициентами справедливо следующее условие симметрии:

$$H(k) = H^*(N-k), e^{2\pi i(N-k)/N} - e^{-2\pi ik/N}.$$

Таким образом, можно записать $H_2(z)$

$$\begin{aligned} H_2(z) &= \frac{H(0)}{1 - rz^{-1}} + \frac{H(1)}{1 - re^{2\pi i/N} z^{-1}} + \frac{H(2)}{1 - re^{2\pi i2/N} z^{-1}} + \\ &+ \dots + \frac{H^*(2)}{1 - re^{-2\pi i2/N} z^{-1}} + \frac{H^*(1)}{1 - re^{-2\pi i/N} z^{-1}}. \end{aligned}$$

Итак, полюсы появляются комплексно-сопряженными парами (исключая одно значение при $k = 0$, если N — нечетное, и два при $k = 0$ и $k = N/2$, если N — четное).

Для фильтров с линейной фазовой характеристикой и четной длиной $H(N/2) = 0$. Объединяя для k -го полюса комплексно-сопряженные фрагменты, получим

$$\begin{aligned} H_2(z) &= \frac{H(k)}{1 - re^{2\pi ik/N} z^{-1}} + \frac{H^*(k)}{1 - re^{-2\pi ik/N} z^{-1}} = \\ &= \frac{H(k)(1 - re^{-2\pi ik/N} z^{-1}) + H^*(k)(1 - re^{2\pi ik/N} z^{-1})}{(1 - re^{2\pi ik/N} z^{-1})(1 - re^{-2\pi ik/N} z^{-1})}. \end{aligned} \quad (7.29)$$

Знаменатель упрощается до вида

$$(1 - re^{2\pi ik/N} z^{-1})(1 - re^{-2\pi ik/N} z^{-1}) = 1 - 2r \cos(2\pi k/N) z^{-1} + r^2 z^{-2}. \quad (7.30)$$

Для фильтра с линейной фазовой характеристикой и четно-симметричной импульсной характеристикой $H(k)$ выражается следующим образом:

$$H(k) = |H(k)|e^{-2\pi i k \alpha / N},$$

где $\alpha = (N - 1)/2$. Таким образом, числитель можно упростить до такого вида:

$$\begin{aligned} &|H(k)|e^{-2\pi i k \alpha / N} (1 - re^{-2\pi ik/N} z^{-1}) + |H(k)|e^{2\pi i k \alpha / N} (1 - re^{2\pi ik/N} z^{-1}) = \\ &= |H(k)|[e^{-2\pi i k \alpha / N} (1 - re^{-2\pi ik/N} z^{-1}) + e^{2\pi i k \alpha / N} (1 - re^{2\pi ik/N} z^{-1})] = \\ &= |H(k)|[e^{-2\pi i k \alpha / N} - re^{-2\pi ik/N} e^{-2\pi i k \alpha / N} z^{-1} + e^{2\pi i k \alpha / N} - re^{2\pi ik/N} e^{2\pi i k \alpha / N} z^{-1}] = \\ &= |H(k)| \{ 2 \cos(2\pi k \alpha / N) - [re^{-2\pi i k (1+\alpha)/N} z^{-1} + re^{2\pi i k (1+\alpha)/N} z^{-1}] \} = \\ &= |H(k)| \{ 2 \cos(2\pi k \alpha / N) - 2r \cos[2\pi k (1 + \alpha)/N] z^{-1} \}. \end{aligned} \quad (7.31)$$

Объединяя формулы (7.30) и (7.31), $H(z)$ можно записать в виде

$$\begin{aligned} H(z) &= \\ &= \frac{1 - r^N z^{-N}}{N} \left[\sum_{k=1}^M \frac{|H(k)| [2 \cos(2\pi k \alpha / N) - 2r \cos[2\pi k (1 + \alpha)/N] z^{-1}]}{1 - 2r \cos(2\pi k/N) z^{-1} + r^2 z^{-2}} + \right. \\ &\quad \left. + \frac{H(0)}{1 - r z^{-1}} \right] \end{aligned} \quad (7.32, a)$$

Если N нечетное, $M = (N - 1)/2$, если же четное, $M = (N/2) - 1$.

- При $N = 9$ характеристика выбирается через интервал $18/9 = 2$ кГц. Следовательно, частотные выборки определяются как

$$\begin{aligned} |H(k)| &= 1, & k &= 0, 1, 2, \\ &0, & k &= 3, 4. \end{aligned}$$

В этом случае $\alpha = (N - 1)/2 = (9 - 1)/2 = 4$ и $r = 1$.

Из уравнения (7.32, а), используя приведенные выше значения частотных выборок, получаем следующую функцию $H(z)$:

$$H(z) = \frac{1 - z^{-9}}{9} \left[\frac{2|H(1)|\{\cos(2\pi 4/9) - \cos[2\pi 5/9]z^{-1}\}}{1 - 2\cos(2\pi/9)z^{-1} + z^{-2}} + \frac{2|H(2)|\{\cos(2\pi \times 2 \times 4/9) - \cos[2\pi \times 2 \times 5/9]z^{-1}\}}{1 - 2\cos(4\pi/9)z^{-1} + z^{-2}} + \frac{1}{1 - z^{-1}} \right].$$

Далее находим значения $\cos(8\pi/9) = -0,9397$, $\cos(10\pi/9) = -0,9397$, $\cos(2\pi/9) = 0,7660$, $\cos(16\pi/9) = 0,7660$, $\cos(20\pi/9) = 0,7660$ и $\cos(4\pi/9) = 0,1736$. Подставляя эти значения в уравнение выше, получаем:

$$\begin{aligned} H(z) &= \frac{1 - z^{-9}}{9} \left[\frac{2\{-0,9397 + 0,9397z^{-1}\}}{1 - 2 \times 0,7660z^{-1} + z^{-2}} + \frac{2\{0,7660 - 0,7660z^{-1}\}}{1 - 2 \times 0,1736z^{-1} + z^{-2}} + \frac{1}{1 - z^{-1}} \right] = \\ &= \frac{1 - z^{-9}}{9} \left[\frac{-1,8794(1 - z^{-1})}{1 - 1,5320z^{-1} + z^{-2}} + \frac{1,5320(1 - z^{-1})}{1 - 0,3472z^{-1} + z^{-2}} + \frac{1}{1 - z^{-1}} \right]. \end{aligned}$$

Диаграмма реализации данного уравнения приведена на рис. 7.23. Вычислительная сложность непосредственной реализации фильтра и реализации через частотные выборы приводится ниже.

	Число операций сложения	Число операций умножения	Память
прямая реализация	8	9	18
реализация через выборы	10	7	25

На основе данных рис. 7.23 получаем следующие разностные уравнения:

$$\begin{aligned} x'(n) &= (1/9)[x(n) - x(n - 9)], \\ y_1(n) &= x'(n) + y(n - 1), \\ w_2(n) &= 1,5320w_2(n - 1) - w_2(n - 2) + x'(n), \\ y_2(n) &= -1,8794w_2(n) + 1,8794w_2(n - 1), \\ w_3(n) &= 0,3472w_3(n - 1) - w_3(n - 2) + x'(n), \\ y_3(n) &= 1,5320w_3(n) - 1,5320w_3(n - 1), \\ y(n) &= y_1(n) + y_2(n) + y_3(n) \end{aligned} \quad (7.32, б)$$

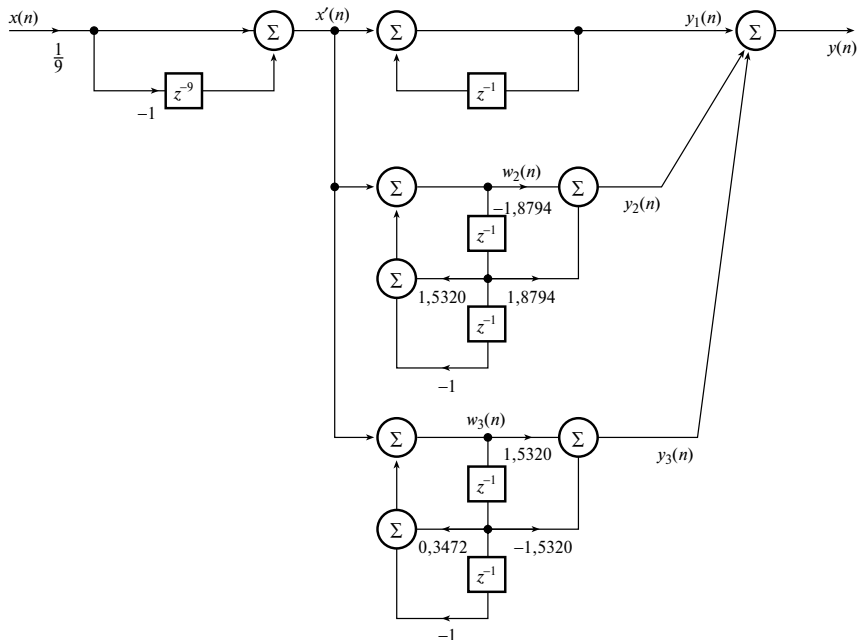


Рис. 7.23. Диаграмма реализации для фильтра частотной выборки из примера 7.12

Пример 7.13

Получите передаточную функцию и разностное уравнение для следующих фильтров.

1. Рекурсивный КИХ-фильтр нижних частот с целыми коэффициентами, удовлетворяющий следующим спецификациям:
 центральная частота 0 Гц,
 частота дискретизации 18 кГц.
2. Рекурсивный полосовой КИХ-фильтр с целыми коэффициентами, удовлетворяющий следующим спецификациям:
 центральная частота 3 Гц,
 частота дискретизации 12 кГц.

Решение

1. Если $N = 9$, интервал между частотными выборками будет равен $18/9 = 2$ кГц. Диаграмма нулей и полюсов для такого выбора изображена на рис. 7.24, а, а соответствующая амплитудная характеристика — на рис. 7.24, б. Из рис. 7.24, а следует, что передаточная функция имеет вид

$$H(z) = \frac{1 - z^{-9}}{9} \frac{1}{1 - z^{-1}}.$$

Соответствующее разностное уравнение имеет вид

$$y(n) = y(n-1) + (1/9)[x(n) - x(n-9)].$$

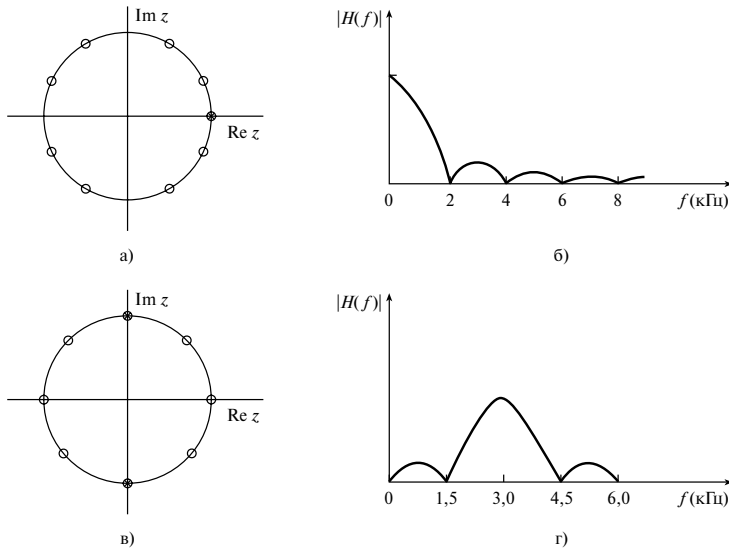


Рис. 7.24. Диаграмма нулей и полюсов (панель *а*). Схематическое изображение амплитудной характеристики рекурсивного КИХ-фильтра с целыми коэффициентами (панель *б*). Диаграмма нулей и полюсов для простого полосового фильтра с целыми коэффициентами (панель *в*). Соответствующая амплитудная характеристика (панель *г*)

2. Поскольку полоса пропускания центрирована на 3 кГц, нужно аккуратно выбрать моменты выборки. Положив $N = 8$, получаем один из вариантов диаграммы нулей и полюсов и соответствующей амплитудной характеристики (изображены на рис. 7.24, *в* и *г* соответственно).

При $N = 8$ интервал дискретизации составляет $12/8 = 1,5$ кГц. Передаточная функция имеет вид

$$H(z) = \frac{1 - z^{-8}}{8} \frac{1}{1 - z^{-2}}.$$

Соответствующее разностное уравнение записывается так:

$$y(n) = -y(n - 2) + (1/8)[x(n) - x(n - 8)].$$

Очевидно, что определение передаточной функции при частотной выборке с целыми коэффициентами — это очень простой процесс. В то же время, амплитудная характеристика подобных фильтров часто плоха, а свобода разработчика состоит только в выборе местоположения полосы пропускания. Для улучшения характеристик, относящихся к затуханию и частотам среза таких фильтров, передаточную функцию можно увеличить в целое число раз [7, 8].

7.7.4. Метод частотной выборки: резюме

- *Этап 1.* Задать идеальную или желательную частотную характеристику, затухание в полосе подавления и границы полос целевого фильтра.

- *Этап 2.* Исходя из спецификации выбрать фильтр частотной выборки первого (выборки берутся с интервалом kF_s/N) или второго типа (выборки берутся с интервалом $(K + 1/2)F_s/N$).
 - *Этап 3.* Использовать спецификацию (этап 1) и таблицы разработки [15] для определения N , числа частотных выборок идеальной частотной характеристики, M , числа выборок в полосе перехода, ширины полосы перехода, числа частотных выборок в полосе пропускания и T_i , значений выборок в полосе перехода ($i = 1, 2, \dots, M$).
 - *Этап 4.* Использовать подходящее уравнение для расчета коэффициентов фильтра.
- Вместо этапов 2 и 4 можно использовать компьютерную программу с реализацией генетического алгоритма [3].

7.8. Сравнение метода взвешивания, оптимального метода и метода частотной выборки

Оптимизационные методы — это простой и эффективный путь вычисления коэффициентов КИХ-фильтра. Хотя метод позволяет полностью контролировать спецификацию фильтра, необходимым требованием является наличие программной поддержки разработки фильтра. Для большинства приложений оптимальный метод дает фильтры с хорошими амплитудными характеристиками при разумных значениях N . Метод особенно хорош при разработке трансформаторов и дифференциаторов Гильберта. При использовании для разработки названных устройств других методов будут получены большие ошибки аппроксимации.

Если программная поддержка оптимального метода отсутствует или амплитуды колебаний характеристики в полосе пропускания и полосе подавления равны, можно воспользоваться методом вырезания. Этот метод весьма прост в применении и интуитивно понятен. В то же время, оптимальный метод часто предлагает более экономное решение с точки зрения числа коэффициентов фильтра. Кроме того, метод вырезания не позволяет разработчику строго управлять частотами среза или неравномерностью в полосе пропускания или подавления.

Только метод частотной выборки который допускает и рекурсивную, и нерекурсивную реализацию КИХ-фильтра, и этот метод стоит использовать, когда предвидится, что такие реализации, как рекурсивный подход, будут вычислительно экономными. Фильтр с целыми коэффициентами следует применять, только если обязательна примитивная арифметика и простота программирования (например, программирование на ассемблере на стандартном микропроцессоре). В таких случаях всегда следует проверять, приемлема ли получающаяся в результате плохая амплитудная характеристика. Метод частотной выборки позволяет легко разрабатывать фильтры с произвольными амплитудно-частотными характеристиками. В то же время этот метод не позволяет строго контролировать положение граничных частот или неравномерность в полосе пропускания и зависит от наличия таблиц разработки, подобных приведенным в работе [15], хотя еще можно использовать компьютерную программу поддержки разработки [3].

Пример 7.14

Требуется два полосовых КИХ-фильтра с линейной фазовой характеристикой, удовлетворяющих следующим спецификациям: для фильтра 1 —

полоса пропускания	8–12 кГц,
неравномерность в полосе подавления	0,001,
максимальная неравномерность в полосе пропускания	0,001,
частота дискретизации	44,14 кГц,
ширина полосы перехода	3 кГц.

для фильтра 2 —

полоса пропускания	8–12 кГц,
неравномерность в полосе подавления	0,001,
максимальная неравномерность в полосе пропускания	0,01,
частота дискретизации	44,14 кГц,
ширина полосы перехода	3 кГц.

Получите и сравните частотные характеристики для каждого фильтра, используя

- 1) метод вырезания;
- 2) метод частной выборки;
- 3) оптимальный метод.

Решение

1. *Метод взвешивания.* Согласно спецификации фильтра 1 неравномерность в полосе пропускания равна $20 \lg(1 + 0,001) = 0,00868$ дБ, а затухание в полосе подавления — $-20 \lg(0,001) = 60$ дБ. Из формул (7.10) и (7.11) находим параметры функции Кайзера:

частоты среза	6,5 и 13,5 кГц,
параметр неравномерности, β	5,653,
число коэффициентов фильтра	53,
частота дискретизации	44,14 кГц.

Для фильтра 2 получаем те же результаты, поскольку в методе вырезания неравномерность в полосе пропускания всегда приблизительно равна неравномерности в полосе подавления.

Спектр итогового фильтра представлен на рис. 7.25, а.

2. *Метод частной выборки.* Для спецификации 1 предполагается использовать фильтр частотной выборки первого типа и фильтр длиной 53, как и для метода взвешивания. Из таблиц разработки [15] находим, что для получения желаемого затухания в полосе подавления 60 дБ при $F_s = 44,14$ кГц, $M = 2$, $N = 53$ нужны две частотные выборки в полосе перехода. Дискретизация идеальной частотной характеристики при $N = 53$ дает

$$\begin{array}{ll}
|H(k)| = 0 & k = 0, 1, \dots, 7, \\
0,10689 & k = 8, \\
0,59253 & k = 9, \\
1 & k = 10-14, \\
0,59253 & k = 15, \\
0,10689 & k = 16, \\
0 & k = 17-26.
\end{array}$$

С помощью программы `fresamp.c` (см. приложение) был получен фильтр, частотная характеристика которого изображена на рис. 7.25, б.

Поскольку затухание в полосе подавления одинаково для обоих фильтров, фильтр 2 совпадает с фильтром 1.

3. *Оптимизационные методы.* Для фильтра 1 из спецификации находим нормированные граничные частоты: 0; 5/44,14; 8/44,14; 12/44,14; 15/44,14 и 22,07/44,14, т.е. 0; 0,11328; 0,18124; 0,27186; 0,33983 и 0,5. Используя программу, приведенную в приложении, находим, что $N = 49,6$. Поскольку амплитуды колебаний характеристики в полосе пропускания и полосе подавления равны, весовые коэффициенты трех полос одинаковы. В программе оптимальной разработки используются следующие входные параметры:

число коэффициентов фильтра 49,
 граничные частоты 0; 0,11328; 0,18124; 0,27186; 0,33983; 0,5,
 весовые коэффициенты 5; 5; 5.

Входные параметры для фильтра 2:

число коэффициентов фильтра 39 (39,45),
 граничные частоты 0; 0,11328; 0,18124; 0,27186; 0,33983; 0,5,
 весовые коэффициенты 10; 1; 10.

Получающиеся частотные характеристики оптимальных фильтров приведены на рис. 7.25, в и г.

7.9. Разработка КИХ-фильтров — специальные вопросы

7.9.1. Полуполосные КИХ-фильтры

Полуполосные фильтры — это частный случай КИХ-фильтров. Основное преимущество таких фильтров заключается в том, что примерно половина их коэффициентов равны нулю, т.е. объем вычислений сокращается вдвое. Данная особенность делает полуполосные фильтры привлекательными в таких приложениях, как обработка при нескольких скоростях, где требуется эффективная фильтрация с целью защиты от наложения спектров и/или фильтрация для подавления зеркальных частот при изменении частоты дискретизации данных (подробности см. в главе 9).

Причинные полуполосные фильтры характеризуются следующими особенностями.

1. Амплитуды колебаний характеристики в полосе пропускания и подавления равны, т.е.

$$\delta_p = \delta_s = \delta. \quad (7.33)$$

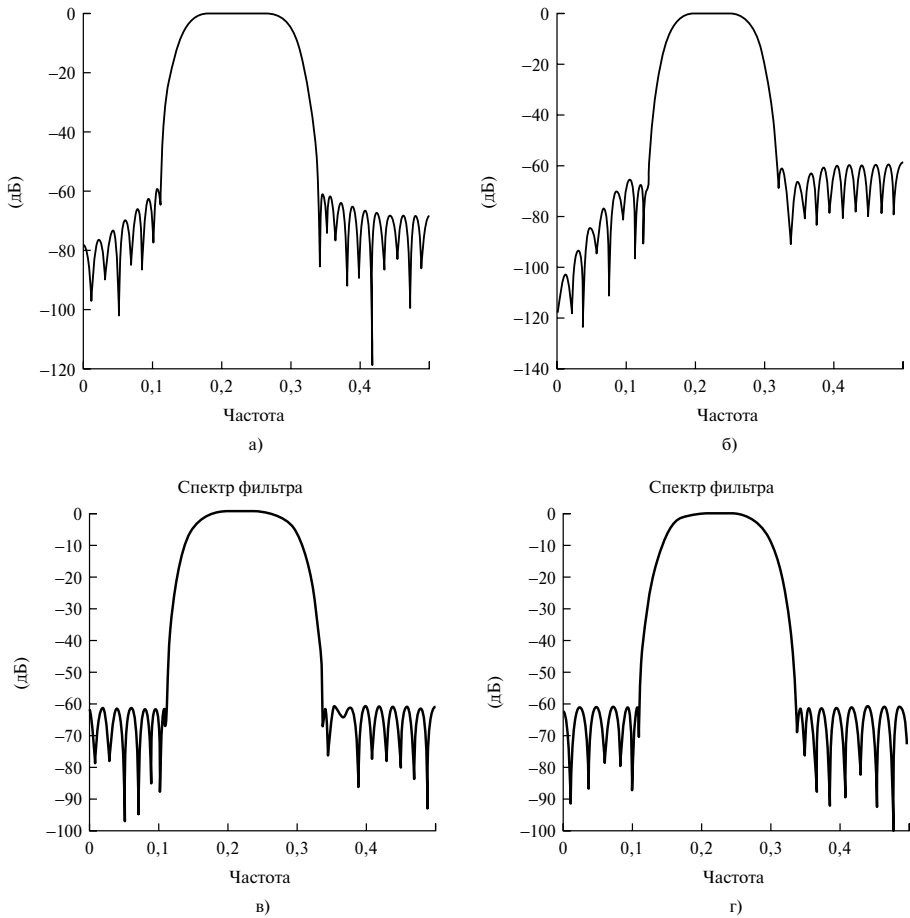


Рис. 7.25. Сравнение частотных характеристик фильтров, полученных с использованием метода взвешивания, метода частотной выборки и оптимального метода: а) характеристика фильтра, полученного с помощью функции Кайзера (фильтры 1 и 2); б) характеристика фильтра, полученного с помощью частотной выборки (фильтры 1 и 2); в) характеристика фильтра, полученного с помощью оптимального метода (фильтр 1); г) характеристика фильтра, полученного с помощью оптимального метода (фильтр 2)

2. Граничные частоты полос пропускания и подавления связаны следующим образом:

$$f_s = \frac{F_s}{2} - f_p. \quad (7.34)$$

3. Частотная характеристика симметрична относительно четверти частоты дискретизации. Следовательно, на частоте $f = F_s/4$

$$H\left(\frac{F_s}{4} + f\right) = 1 - H\left(\frac{F_s}{4} - f\right). \quad (7.35)$$

Кроме того, на этой частоте нормированная частотная характеристика уменьшается вдвое, т.е.

$$|H(f)| = 0,5 \left(\text{на частоте } f = \frac{F_s}{4} \right).$$

4. У единичной импульсной характеристики при нечетном N нулю равны все коэффициенты, кроме

$$\begin{aligned} h(2n) &= 0 & n &= 0, 1, \dots, (N-1)/4 \\ 0,5 & & n &= (N-1)/2. \end{aligned} \quad (7.36)$$

Для расчета коэффициентов полуполосных фильтров можно использовать любой из описанных выше методов, подходящих для КИХ-фильтров, например, метод весовых функций или оптимальный метод. При реализации этих методов следует использовать условия, приведенные в выражениях (7.33) и (7.34).

Пример 7.15

С помощью метода весовых функций определите коэффициенты КИХ-фильтра нижних частот по следующим спецификациям:

край полосы пропускания	2 кГц;
ширина полосы перехода	0,5 кГц;
затухание в полосе подавления	>50 дБ;
частота дискретизации	8 кГц.

Решение

Коэффициенты фильтра перечислены в табл. 7.13, а спектр фильтра показан на рис. 7.26, а. Из табл. 7.13 очевидно, что начиная с $h(0)$ коэффициенты через один равны нулю (с точностью до ошибки при численном расчете $h(n)$). Вследствие этого при фильтрации каждую вторую входную выборку можно не учитывать (частота дискретизации эффективно уменьшается вдвое).

Таблица 7.13. Коэффициенты полуполосного фильтра нижних частот (функция Хэмминга, $N = 53$, $f_c = 2000$ Гц)

$h[0] =$	$-1,1243421e - 09$	$= h[52]$
$h[1] =$	$1,1109516e - 03$	$= h[51]$
$h[2] =$	$1,3921496e - 09$	$= h[50]$
$h[3] =$	$-1,6473646e - 03$	$= h[49]$
$h[4] =$	$-2,0024685e - 09$	$= h[48]$
$h[5] =$	$2,6429869e - 03$	$= h[47]$
$h[6] =$	$2,9211490e - 09$	$= h[46]$
$h[7] =$	$-4,1909615e - 03$	$= h[45]$
$h[8] =$	$-4,0967870e - 09$	$= h[44]$
$h[9] =$	$6,4068290e - 03$	$= h[43]$
$h[10] =$	$5,4636006e - 09$	$= h[42]$
$h[11] =$	$-9,4484947e - 03$	$= h[41]$
$h[12] =$	$-6,9451110e - 09$	$= h[40]$
$h[13] =$	$1,3555871e - 02$	$= h[39]$
$h[14] =$	$8,4584215e - 09$	$= h[38]$
$h[15] =$	$-1,9134767e - 02$	$= h[37]$
$h[16] =$	$-9,9188559e - 09$	$= h[36]$
$h[17] =$	$2,6953222e - 02$	$= h[35]$
$h[18] =$	$1,1244697e - 08$	$= h[34]$
$h[19] =$	$-3,8674295e - 02$	$= h[33]$
$h[20] =$	$-1,2361758e - 08$	$= h[32]$
$h[21] =$	$5,8666205e - 02$	$= h[31]$
$h[22] =$	$1,3207536e - 08$	$= h[30]$
$h[23] =$	$-1,0304890e - 01$	$= h[29]$
$h[24] =$	$-1,3734705e - 08$	$= h[28]$
$h[25] =$	$3,1728215e - 01$	$= h[27]$
$h[26] =$	$5,0000000e - 01$	$= h[26]$

Обратите внимание на то, что каждый второй коэффициент равен нулю (не совсем точно из-за ошибок численного счета)

7.9.2. Преобразование частоты

В некоторых ситуациях может потребоваться преобразовать в реальном времени характеристику фильтра нижних частот в характеристику эквивалентного фильтра верхних частот. Такое изменение возможно благодаря простой связи, существующей между названными фильтрами. Коэффициенты КИХ-фильтра верхних частот можно тривиальным образом получить из характеристик эквивалентного фильтра нижних частот, следующим образом изменив знаки коэффициентов:

$$h_{\text{вч}} = (-1)^n h_{\text{нч}}(n). \quad (7.37)$$

Данное соотношение основано на следующем факте: частотная характеристика фильтра верхних частот аналогична характеристике фильтра нижних частот, транслированной на половину частоты дискретизации (см. рис. 7.26, б). Следовательно, частотную харак-

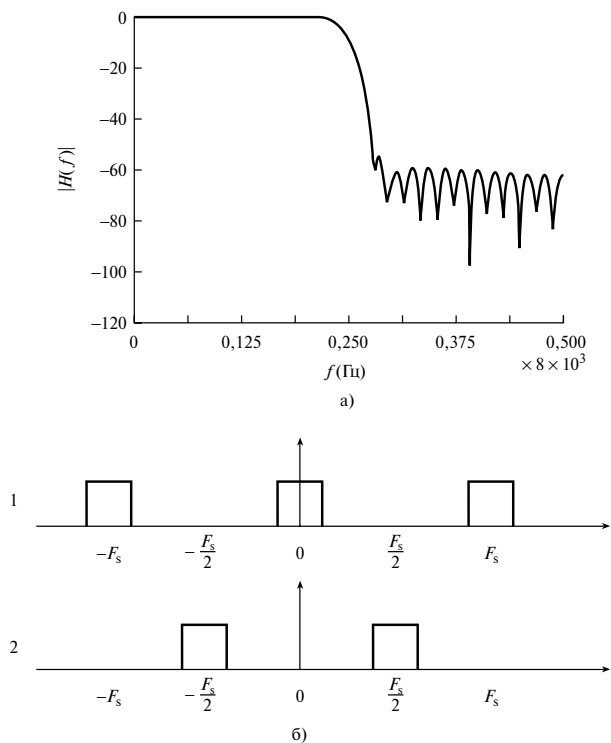


Рис. 7.26. Частотная характеристика полуполосного фильтра нижних частот (панель а). Частотная характеристика 1) идеального фильтра нижних частот; 2) эквивалентного идеального фильтра верхних частот (панель б)

теристику фильтра верхних частот можно получить из характеристики фильтра нижних частот, заменив f на $F_s/2 - f$:

$$H_{\text{вч}} = H_{\text{нч}} \left(\frac{F_s}{2} - f \right). \quad (7.38)$$

Пример 7.16

Фильтр нижних частот имеет следующие характеристики:

край полосы пропускания 1,5 кГц,
 частота дискретизации 10 кГц,
 число коэффициентов 15.

1. Получите коэффициенты фильтра нижних частот, используя весовую функцию Хэмминга.
2. Запишите спецификации эквивалентного фильтра верхних частот и с их помощью получите коэффициенты этого фильтра.
3. Вычислите коэффициенты эквивалентного фильтра верхних частот, используя приведенное выше преобразование.

Таблица 7.14. Коэффициенты фильтра нижних частот и эквивалентного фильтра верхних частот

	<i>Фильтр нижних частот</i>	<i>Фильтр верхних частот</i>
$h(0)$	$1,2654 \times 10^{-3}$	$1,2654 \times 10^{-3}$
$h(1)$	$-5,2341 \times 10^{-3}$	$5,2341 \times 10^{-3}$
$h(2)$	$-1,9735 \times 10^{-2}$	$-1,9735 \times 10^{-3}$
$h(3)$	$-2,3009 \times 10^{-2}$	$2,3009 \times 10^{-3}$
$h(4)$	$2,2366 \times 10^{-2}$	$2,2366 \times 10^{-2}$
$h(5)$	$1,2833 \times 10^{-1}$	$-1,2833 \times 10^{-1}$
$h(6)$	$2,4728 \times 10^{-1}$	$2,4728 \times 10^{-1}$
$h(7)$	$3,0000 \times 10^{-1}$	$-3,0000 \times 10^{-1}$

Решение

1. Используя приведенные параметры как вход программы `window.c`, получаем коэффициенты фильтра нижних частот (см. табл. 7.14).
2. Спецификации эквивалентного фильтра верхних частот записываются так:
 край полосы пропускания $F_s/2 - f_c = 5000 - 1500$ кГц = 3500 кГц,
 частота дискретизации 10 кГц,
 число коэффициентов 15.

Используя эти параметры как вход программы `window.c`, получаем коэффициенты фильтра верхних частот, которые перечислены в табл. 7.14.

3. Применяя приведенное выше простое преобразование, получаем коэффициенты фильтра верхних частот, которые аналогичны полученным в п. 2.

7.9.3. Вычислительно эффективные КИХ-фильтры

В некоторых ситуациях методы, описанные в данной главе, могут не подходить. Например, в определенных приложениях фазовая задержка, вводимая КИХ-фильтром с линейной фазовой характеристикой, может быть неприемлемо длительной (например, фазовая задержка для КИХ-фильтров первого типа равна $(N - 1)T/2$, а при больших N эта величина значительна). В системе контроля, например, использование подобных фильтров внутри контура обратной связи может привести к неустойчивости. В таких случаях лучше применять фильтры с минимальной фазовой задержкой [13].

То, что фильтры, полученные с помощью оптимального метода, имеют равные амплитуды колебаний характеристики в полосе пропускания и подавления, может привести к появлению эхо-сигналов в импульсной характеристике фильтра, что на практике зачастую нежелательно. При гладкой частотной характеристике эхо-сигналы в хвостах импульсной характеристики уменьшаются.

В таких приложениях, как обработка изображений, число арифметических операций при использовании стандартных КИХ-фильтров может быть слишком большим. К сожалению, фильтры с целыми коэффициентами не подходят для подобных приложений из-за их плохой амплитудной характеристики. В этом случае следует использовать КИХ-фильтры, которые требуют только очень простых арифметических операций, но амплитудная характеристика которых подобна характеристике стандартных КИХ-фильтров (см., например, [11] или [17]).

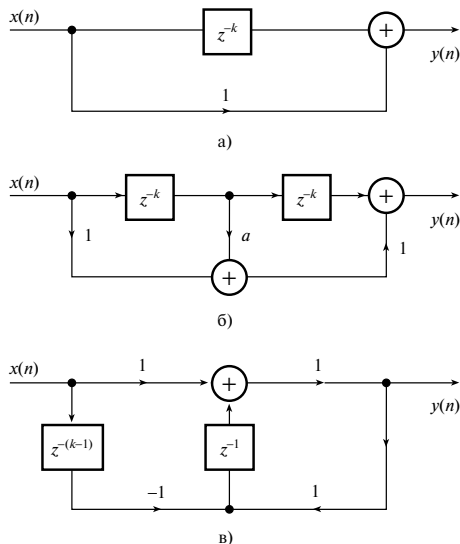


Рис. 7.27. Примеры примитивных блоков КИХ-фильтров: а) схема суммирования ($H(z) = 1 + z^{-k}$); б) схема двойного суммирования ($H(z) = az^{-k} + z^{-2k}$); чтобы избежать прямого умножения $a = \pm 2^n$, n — целое; умножение на целое число можно реализовать с помощью сдвигов); в) блок рекурсивной текущей суммы ($H(z) = (1 - z^{-k})/(1 - z^{-1}) = 1 + z^{-1} + z^{-2} + \dots + z^{-(k-1)}$).

Типичный фильтр состоит из 4–7 таких блоков, соединенных каскадом

Для решения указанных проблем предлагается использовать каскадную систему, состоящую из нескольких элементарных блоков, подобных изображенным на рис. 7.27. В каждом элементарном блоке можно реализовать почти любое возможное умножение. Основные проблемы такого подхода заключаются в трудности эффективного выбора элементарных фильтрующих блоков каскада и в том, что эффективно разработать удастся только фильтры небольших порядков. Для дальнейшей борьбы с возникающими проблемами можно использовать генетические алгоритмы [16].

7.10. Структуры реализаций КИХ-фильтров

КИХ-фильтр характеризуется следующей передаточной функцией $H(z)$:

$$H(z) = \sum_{n=0}^{N-1} h(n)z^{-n}.$$

Структура реализации фильтра — это, по сути, представление в виде блок-схемы (или функциональной схемы) различных теоретически эквивалентных способов записи передаточной функции. В большинстве случаев такие структуры состоят из соединения умножителей, сумматоров и элементов задержки. Существует множество структур реализации КИХ-фильтров, но в данной книге рассматриваются только самые распространенные.

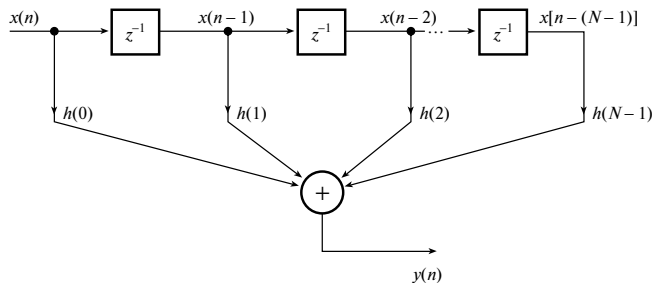


Рис. 7.28. Трансверсальная структура фильтра

7.10.1. Трансверсальная структура

Трансверсальная структура (или схема задержки с отводами) изображена на рис. 7.28. Вход $x(n)$ и выход $y(n)$ фильтра, представленного с помощью данной структуры, связаны простым соотношением

$$y(n) = \sum_{m=0}^{N-1} h(m)x(n-m). \quad (7.39)$$

На этом рисунке символом z^{-1} представлена задержка в одну выборку или в одну единицу времени. Таким образом, $x(n-1)$ — это $x(n)$, задержанное на время одной выборки. В цифровых реализациях блоки, помеченные символом z^{-1} , могут представлять регистры сдвига или более распространенные ячейки памяти в ОЗУ. Описанная трансверсальная структура является наиболее популярным представлением КИХ-фильтров.

Выходная выборка $y(n)$ представляет собой взвешенную сумму текущего входа $x(n)$ и $N-1$ предыдущей входной выборки, т.е. выборок с $x(n-1)$ по $x(n-N)$. При выборе трансверсальной структуры вычисление каждой выходной выборки $y(n)$ требует

- $N-1$ ячеек памяти для хранения $N-1$ входной выборки;
- N ячеек памяти для хранения N коэффициентов;
- N операций умножения;
- $N-1$ операции сложения.

7.10.2. Структура с линейной фазовой характеристикой

Разновидностью трансверсальной структуры является структура с линейной фазовой характеристикой, в которой для снижения вычислительной сложности реализации фильтра используется симметрия коэффициентов импульсной характеристики КИХ-фильтров с линейной фазовой характеристикой.

В фильтрах с линейной фазовой характеристикой коэффициенты симметричны, т.е. $h(n) = \pm h(N-n-1)$. Следовательно, уравнение фильтра можно переписать с учетом симметрии, существенно снизив число сложений и умножений. Для фильтров первого

и второго типа с линейной фазовой характеристикой передаточную функцию можно записать как

$$H(z) = \sum_{n=0}^{(N-1)/2-1} h(n)[z^{-n} + z^{-(N-1-n)}] + h\left(\frac{N-1}{2}\right) z^{-(N-1)/2}, \quad (7.40, a)$$

N — нечетное;

$$H(z) = \sum_{n=0}^{N/2-1} h(n)[z^{-n} + z^{-(N-1-n)}], \quad N \text{ — четное.} \quad (7.40, б)$$

Соответствующие разностные уравнения выглядят так:

$$y(n) = \sum_{k=0}^{(N-1)/2-1} h(k) \{x(n-k) + x[n - (N-1-k)]\} + h[(N-1)/2]x[n - (N-1)/2], \quad (7.41, a)$$

$$y(n) = \sum_{k=0}^{(N-1)/2-1} h(k) \{x(n-k) + x[n - (N-1-k)]\}. \quad (7.41, б)$$

Сравнивая выражения (7.39) и (7.41), видим, что структура с линейной фазовой характеристикой вычислительно эффективнее и требует приблизительно вдвое меньшего числа операций умножения и сложения. Впрочем, в большинстве ЦОС-процессоров реализация по формуле (7.39) получается более эффективной, поскольку вычислительное преимущество формулы (7.41) теряется на фоне более сложной системы индексации данных.

Пример 7.17

КИХ-фильтр с линейной фазовой характеристикой имеет семь коэффициентов, которые перечислены ниже. Изобразите диаграмму реализации данного фильтра, используя а) прямую (трансверсальную) структуру и б) структуру с линейной фазовой характеристикой. Сравните вычислительную сложность обеих реализаций.

$$h(0) = h(6) = -0,032,$$

$$h(1) = h(5) = 0,038,$$

$$h(2) = h(4) = 0,048,$$

$$h(3) = -0,048.$$

Решение

Диаграммы реализации изображены на рис. 7.29.

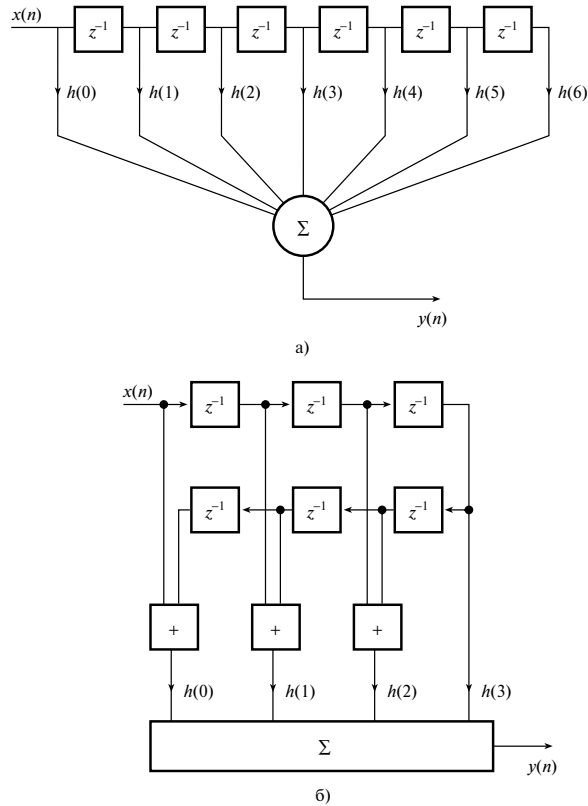


Рис. 7.29. Трансверсальная структура (панель *а*); структура с линейной фазовой характеристикой фильтра из примера 7.17 (панель *б*)

7.10.3. Другие структуры

7.10.3.1. Быстрая свертка

Метод быстрой свертки включает выполнение в частотной области операции свертки, представленной в уравнении (7.39). Как говорилось в главе 5, свертка во временной области эквивалентна умножению в частотной. Проще говоря, при названном подходе для выполнения фильтрации вначале вычисляется ДПФ $x(n)$ и $h(n)$ (обе функции соответствующим образом дополняются нулями), оба образа перемножаются, а затем находится обратное результата. Данная концепция иллюстрируется на рис. 7.30. На практике при фильтрации в реальном времени используются методы, именуемые *наложением-сложением* (overlap-add) и *наложением-хранением* (overlap-save). Данные схемы рассмотрены в главе 5.

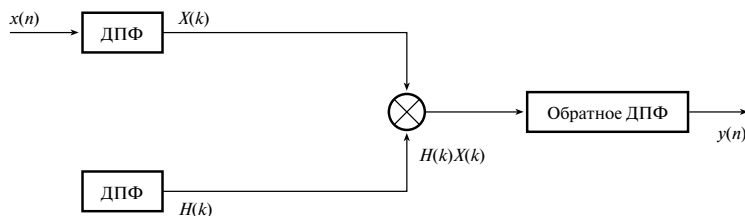


Рис. 7.30. Иллюстрация быстрой свертки

7.10.3.2. Структура частотной выборки

В структуре, реализованной по принципу частотной выборки, фильтры характеризуются выборками нужной частотной характеристики $H(k)$, а не коэффициентами импульсной характеристики. Данная ситуация подробно обсуждалась выше. Для узкополосных фильтров большинство частотных выборок будет равно нулю, так что получающийся фильтр потребует меньшего числа коэффициентов, а следовательно, операций умножения и сложения, чем эквивалентная трансверсальная структура. Типичная диаграмма реализации данного фильтра изображена на рис. 7.22.

7.10.3.3. Транспонированная и каскадная структуры

Транспонированная структура подобна прямой, только частичные суммы переносятся в последующие этапы. Данный метод более чувствителен к шуму округления, чем прямой метод. При каскадной реализации передаточная функция $H(z)$ выражается как произведение блоков второго и первого порядков. В современных цифровых реализациях транспонированная и каскадная структуры используются редко.

7.10.4. Выбор структуры

Выбор структуры зависит от многих факторов и компромиссов между легкостью реализации (сложность требуемого программного или аппаратного обеспечения), сложностью получения коэффициентов импульсной характеристики или передаточной функции и их относительной чувствительности к квантованию коэффициентов. На практике точность представления каждого коэффициента ограничивается длиной слова используемого процессора. Использование только конечного числа битов для представления каждого коэффициента приводит к смещению нулей с нужных положений, т.е. искажению частотной характеристики. Степень искажения характеристики зависит от используемой структуры и числа битов.

Прямая структура легко поддается программированию и эффективной реализации на большинстве чипов ЦОС, если в них внедрены команды, подходящие для трансверсальной КИХ-фильтрации. Данная структура наиболее широко используется при реализации нерекурсивных фильтров, и главной ее привлекательной особенностью является простота, поскольку требуется минимум компонентов и несложный доступ к памяти. Каскадная структура менее чувствительна к ошибкам коэффициентов и шуму

квантования, но для получения коэффициентов требуется проделать больше работы, а программирование невозможно выполнить на архитектуре чипов ЦОС. Структура с использованием быстрой свертки значительно выгоднее других структур с точки зрения объема вычислений, но зависит от доступности реализации БПФ.

Структура частотной выборки в узкополосных частотно-избирательных фильтрах вычислительно эффективнее, чем эквивалентная трансверсальная структура. В таких фильтрах и при такой структуре лишь относительно небольшое число частотных выборок не равно нулю, так что на одну выходную выборку приходится лишь несколько операций умножения. В то же время структура, реализованная по принципу частотной выборки, может требовать более сложного программирования из-за более сложной системы индексирования в разностном уравнении (сравните, например, уравнения (7.39) и (7.32, б)). Чтобы избежать проблем с устойчивостью при использовании структуры частотной выборки, полюса и нули должны располагаться внутри единичной окружности близко к ней, например, на окружности радиуса $r = 0,99$. Такая структура — естественный выбор, если рекурсивная реализация КИХ-фильтров обязательна. Эта высококомбинаторная структура естественным образом подходит для параллельной обработки.

Вообще, трансверсальную структуру стоит использовать, если из спецификации не следует обязательность структуры частотной выборки, или если не нужно знать спектр данных (в этом случае рекомендуется техника быстрой свертки).

7.11. Влияние конечной разрядности на цифровые КИХ-фильтры

На практике цифровые КИХ-фильтры часто реализуются на основе процессоров ЦОС (например, TMS320C50 от Texas Instruments), алгоритмических чипов ЦОС, разработанных для КИХ-фильтрации (таких как A100 от INMOS). Если же требуется высокая скорость, используются стандартные блоки умножителей, элементов памяти, сумматоров и контроллеров (например семейство PDSP1600 от Plessey). В этих случаях число битов, используемых для представления входных данных фильтра, коэффициентов фильтра и выполнения арифметических операций, должно быть минимальным для эффективного использования и снижения стоимости цифрового фильтра. Проблемы, вызванные использованием конечного числа битов, называются *эффектами конечной разрядности* и в общем случае приводят к снижению производительности фильтра.

В данном разделе рассмотрено влияние конечной разрядности на производительность цифровых КИХ-фильтров и предлагаются способы минимизировать это влияние. Для иллюстрации используется реализация КИХ-фильтра в прямой форме как наиболее привлекательная в современной сфере обработки сигналов, а для квантования применяется округление — простейший и наиболее распространенный метод.

Существует четыре механизма влияния конечной разрядности на производительность цифровых КИХ-фильтров.

1. *Шум АЦП*. Это привычный шум квантования АЦП, который присутствует, когда на вход фильтра подается аналоговый сигнал. Шум АЦП ограничивает максимально

достижимое отношение. Влияние этого фактора можно снизить, используя дополнительные биты, согласующиеся с собственным шумом сигнала (см. главу 13), и/или применив метод обработки при нескольких скоростях для улучшения отношения сигнал-шум (см. главу 9).

2. *Ошибки квантования коэффициентов.* Проистекают от представления коэффициентов фильтра конечным числом битов. Следствием этого является модификация желаемой частотной характеристики. В полосе подавления фильтра, например, этот фактор ограничивает максимальное возможное затухание, т.е. допускает передачу дополнительных сигналов. Прямое решение данной проблемы — использовать достаточное число битов для представления коэффициентов фильтра. В то же время, для минимизации длины слова, используемой в коэффициентах, можно применить методы оптимизации, направленные на эффективный выбор коэффициентов.
3. *Ошибки округления при квантовании результатов арифметических операций.* Эти ошибки могут появляться, например, при отбрасывании младших разрядов перед запоминанием результата умножения (что автоматически приходится делать из-за конечной разрядности используемого процессора). Названные ошибки снижают отношение сигнал-шум. Чтобы уменьшить их, округление можно использовать после суммирования произведений, которое выполняется с двойной длиной. Степень ошибки зависит от типа используемой арифметики и структуры фильтра.
4. *Арифметическое переполнение.* Возникает, когда частичные суммы или выход фильтра превышают разрешенную длину слова системы. По сути, при появлении переполнения выходная выборка будет неверной (обычно это изменение знака). Чтобы уменьшить переполнение или избежать его, можно масштабировать коэффициенты фильтра, разделив каждый коэффициент на множитель, чтобы выходная выборка фильтра никогда не превышала разрешенную длину слова. Очевидно, что такое решение возможно за счет снижения отношения сигнал-шум.

В следующих разделах подробно рассматриваются факторы 2 и 4.

7.11.1. Ошибки квантования коэффициентов

Коэффициенты фильтра, полученные любым методом аппроксимации, например, методом вырезания или оптимальным (замена Ремеза (Remez exchange)), обычно весьма точны до определенных знаков в десятичной записи. Для реализации фильтра коэффициенты нужно представить фиксированным числом битов, и довольно часто это число определяется длиной слова используемого процессора. Например, если для реализации фильтра используется один из 16-битовых процессоров ЦОС, то логично представить каждый коэффициент фильтра 16 бит. В то же время, если сделать так, автоматически вводится ошибка, которая приводит к отклонению частотной характеристики фильтра с конечной длиной слова от желаемой частотной характеристики. В некоторых случаях это отклонение означает, что фильтр уже не удовлетворяет исходной спецификации.

Пример 7.18

Определите влияние квантования через округление до 8 бит на коэффициенты следующего фильтра:

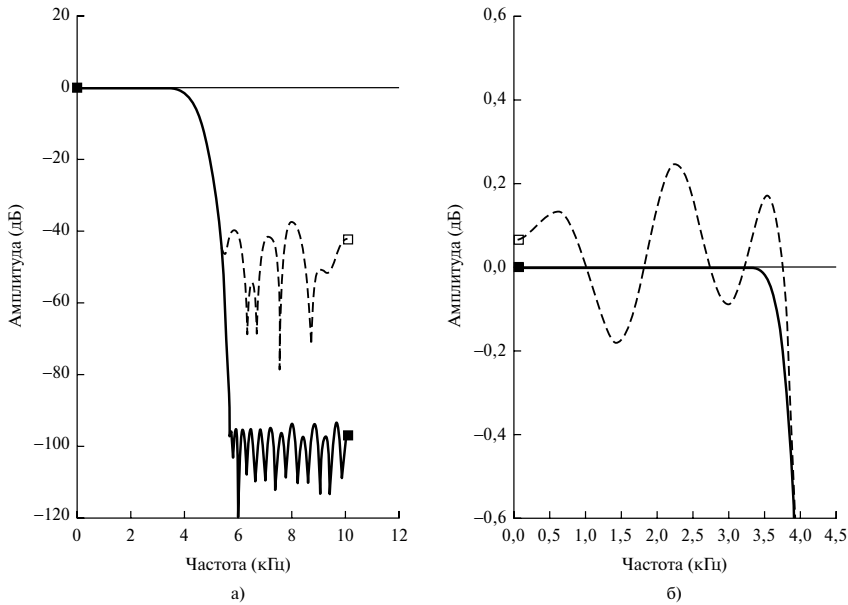


Рис. 7.31. Влияние квантования коэффициентов (пример 7.18)
(панель а). Полоса пропускания (■ — неквантованная характеристика, □ — квантованная) (панель б)

затухание в полосе подавления	> 90 дБ,
неравномерность в полосе пропускания	$< 0,002$ дБ,
край полосы пропускания	$3,375$ кГц,
край полосы подавления	$5,625$ кГц,
частота дискретизации	20 кГц,
число коэффициентов	45 .

Решение

Используем программу `optimal.c` со следующим входом:

число коэффициентов фильтра	45 ,
краевые частоты	$0; 0,16875; 0,28125; 0,5$,
весовые коэффициенты	$1; 7,28$.

Коэффициенты фильтра до и после округления до 8 бит перечислены в табл. 7.15. Соответствующие частотные характеристики изображены на рис. 7.31. Видно, что после квантования минимальное затухание в полосе подавления равно 36 дБ, ухудшение наблюдается более чем на 58 дБ. Очевидно, что в данном примере для представления коэффициентов требуются слова, которые больше 8 бит.

Влияние ошибок коэффициентов проявляется в отклонении частотной характеристики от желаемой формы. Данное отклонение в предельном случае может означать,

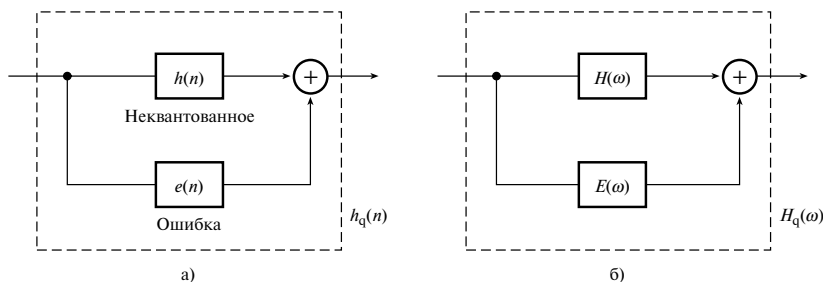


Рис. 7.32. Квантование коэффициентов: а) влияние на коэффициенты импульсной характеристики; б) влияние на частотную характеристику фильтра

что фильтр не удовлетворяет спецификации. В конкретной задаче разработки фильтра подходящую длину слова можно определить, получив частотные характеристики для нескольких различных размеров коэффициентов. По этим данным определяется минимальное число битов, требуемое для удовлетворения спецификации. Кроме того, существенную информацию о разработке фильтров с конечной длиной слова можно получить, анализируя ошибки, которые вводятся при квантовании коэффициентов.

Квантованный и неквантованный коэффициенты, $h_q(n)$ и $h(n)$, соответственно связаны соотношением

$$h_q(n) = h(n) + e(n), \quad n = 0, 1, \dots, N-1, \quad (7.42)$$

где $e(n)$ — ошибка между квантованным и неквантованным коэффициентами. В частотной области формулу (7.42) можно переписать следующим образом:

$$H_q(\omega) = H(\omega) + E(\omega), \quad (7.43)$$

где $E(\omega)$, ошибка желательной частотной характеристики, выражается следующим образом:

$$E(\omega) = \sum_{m=0}^{N-1} e(m) \exp(-i\omega m),$$

а $H_q(\omega)$ и $H(\omega)$ — частотные характеристики фильтров с квантованными и неквантованными коэффициентами соответственно. Схематическое представление уравнений (7.42) и (7.43) дано соответственно на рис. 7.32, а и б. Видим, что физически $e(n)$ можно рассматривать как импульсную характеристику другого фильтра, соединенного параллельно с нужным фильтром [14]. В частотной области влияние ошибки коэффициентов представляется паразитной передаточной функцией, подающейся параллельно с передаточной функцией очень точного фильтра. Целью разработчика является ограничение амплитуды ошибки $E(\omega)$, чтобы частотная характеристика реального фильтра удовлетворяла спецификации.

Для частотно-избирательных фильтров (фильтры нижних частот, верхних частот, полосовые, режекторные) были найдены выражения для пределов ошибок частотной характеристики. Данные пределы можно использовать при определении подходящей длины коэффициентов данного фильтра. Кроме того, пределы полезны при оценке длин

Таблица 7.15. Коэффициенты фильтра до и после квантования до 8 бит

$h(n)$	$h_q(n)$
$-1,05023e-04$	$0,00000e+00$
$-1,25856e-04$	$0,00000e+00$
$3,07141e-04$	$0,00000e+00$
$6,79484e-04$	$0,00000e+00$
$-2,89029e-04$	$0,00000e+00$
$-1,77474e-03$	$0,00000e+00$
$4,08318e-04$	$0,00000e+00$
$3,43482e-03$	$0,00000e+00$
$2,66515e-03$	$0,00000e+00$
$-5,00314e-03$	$-7,81250e-03$
$-7,30591e-03$	$-7,81250e-03$
$5,09712e-03$	$7,81250e-03$
$1,48422e-02$	$1,56250e-02$
$-1,40255e-03$	$0,00000e+00$
$-2,49785e-02$	$-2,34375e-02$
$-9,39383e-03$	$-7,81250e-03$
$3,64568e-02$	$3,90625e-02$
$3,28505e-02$	$3,12500e-02$
$-4,72008e-02$	$-4,68750e-02$
$-8,52427e-02$	$-8,59375e-02$
$5,48855e-02$	$5,46875e-02$
$3,10921e-01$	$3,12500e-01$
$4,42322e-01$	$4,45212e-01$
$3,10921e-01$	$3,12500e-01$
$5,48855e-02$	$5,46875e-02$
$-8,52427e-02$	$-8,59375e-02$
$-4,72008e-02$	$-4,68750e-02$
$3,28505e-02$	$3,12500e-02$
$3,64568e-02$	$3,90625e-02$
$-9,39383e-03$	$-7,81250e-03$
$-2,49785e-02$	$-2,34375e-02$
$-1,40255e-03$	$0,00000e+00$
$1,48422e-02$	$1,56250e-02$
$5,09712e-03$	$7,81250e-03$
$-7,30591e-03$	$-7,81250e-03$
$-5,00314e-03$	$-7,81250e-03$
$2,66515e-03$	$0,00000e+00$
$3,43482e-03$	$0,00000e+00$
$4,08318e-04$	$0,00000e+00$
$-1,77474e-03$	$0,00000e+00$
$-2,89029e-04$	$0,00000e+00$
$6,79484e-04$	$0,00000e+00$
$3,07141e-04$	$0,00000e+00$
$-1,25856e-04$	$0,00000e+00$
$-1,05023e-04$	$0,00000e+00$

коэффициентов адаптивных КИХ-фильтров, поскольку *априори* точные характеристик таких фильтров неизвестны (см. главу 10).

Для КИХ-фильтра, реализованного в прямой форме, с квантованием через округление наиболее часто используются следующие пределы:

$$|E(\omega)| = N2^{-B}, \quad (7.44, a)$$

$$|E(\omega)| = 2^{-B}(N/3)^{1/2}, \quad (7.44, б)$$

$$|E(\omega)| = 2^{-B}[(N \ln N)/3]^{1/2}, \quad (7.44, в)$$

где B — число битов, представляющих каждый коэффициент, а N — длина фильтра. Предел (7.44, а) — это абсолютный верхний предел, выведенный в предположении наихудшей ситуации (см. пример 7.19), так что он чрезмерно пессимистичен. Пределы (7.44, б) и (7.44, в) представляют собой статистические пределы и могут дать более точную оценку ошибок частотной характеристики и длины коэффициентов. При выводе статистических пределов предполагалось, что ошибки квантования $e(n)$ равномерно распределены и имеют нулевое среднее.

Пример 7.19

1. Указав сделанные предположения, покажите, что максимально возможное затухание в полосе подавления $A_{\max}$ для КИХ-фильтра нижних частот, реализованного в прямой форме, с квантованием коэффициентов через округление ограничено величиной

$$A_{\max} \leq 20 \lg(2^{-B} N). \quad (7.45)$$

2. КИХ-фильтр нижних частот удовлетворяет следующим спецификациям:

отклонение в полосе пропускания	0,05 дБ,
частота дискретизации	10 кГц,
край полосы пропускания	1,8 кГц,
ширина перехода	500 Гц,
число коэффициентов	65.
- 2.1. Оцените число битов, требуемых для представления каждого коэффициента, чтобы фильтр имел в полосе подавления затухание не меньше 60 дБ.
- 2.2. Используя длины коэффициентов, найденные в п. 2.1, оцените ожидаемое увеличение неравномерности в полосе пропускания и снижение затухания в полосе подавления в децибелах.
- 2.3. Сравните реальное затухание в полосе подавления и неравномерность в полосе пропускания фильтра, используя длины коэффициентов, найденные в п. 2.1.

Решение

1. Определим характеристику $E(\omega)$, соответствующую ошибкам квантования коэффициентов $e(m)$, как

$$E(\omega) = \sum_{m=0}^{N-1} e(m) \exp(-i\omega m),$$

где N — длина фильтра. При округлении ошибка квантования в наиболее неблагоприятном случае составляет $|e(m)| = 2^{-(B-1)}/2 = 2^{-B}$, где B — длина коэффициентов (предполагается представление чисел в форме поразрядного дополнения до двух). Если предположить наибольшую ошибку для всех коэффициентов, получим

$$\begin{aligned} |E(\omega)| &= \sum_{m=0}^{N-1} |e(m)| \exp(-i\omega m) = \sum_{m=0}^{N-1} 2^{-B} \exp(-i\omega m) = \\ &= 2^{-B} \sum_{m=0}^{N-1} \exp(-i\omega m) = 2^{-B} N. \end{aligned}$$

Если рассматривать $e(m)$ как импульсную характеристику другого фильтра, соединенного параллельно с необходимым, то граничное отклонение в полосе пропускания (или подавления) равно $2^{-B}N$, так что

$$A_{\max} < 20 \lg(2^{-B}N) \text{ дБ.}$$

Очевидно, данная оценка чрезмерно консервативна. На практике обычно достаточно меньшего числа битов, чем из нее следует. Впрочем, данный предел — это простой показатель, который легко использовать.

- 2.1. Положив в приведенном выше пределе $A_{\max} = 60$ дБ, $N = 65$, находим, что $B = 15,988$ бит. Следовательно, требуемая длина коэффициентов составляет $B = 16$ бит.
- 2.2. После квантования максимальная неравномерность в полосе пропускания в наихудшем случае равна $R_{\max}$, а затухание в полосе подавления — $A_{\max}$. Данные величины можно выразить следующим образом:

$$\begin{aligned} R_{\max} &= 20 \lg(1 + \delta_p + |E(\omega)|) = 20 \lg(1 + 0,005773 + 0,001) = \\ &= 0,0586 \text{ дБ,} \end{aligned}$$

т.е. наблюдаем увеличение на 0,0086 дБ.

$$A_{\max} = -20 \lg(\delta_s + |E(\omega)|) = -20 \lg(0,001 + 0,001) = 54 \text{ дБ,}$$

т.е. данный параметр уменьшился на 6 дБ (δ_p и δ_s — отклонение в полосе пропускания и подавления для неквантованного фильтра).

- 2.3. Используем программу разработки оптимального фильтра со следующими параметрами:

число коэффициентов	65,
краевые частоты	0; 0,18; 0,23; 0,5,
весовые коэффициенты полос пропускания/подавления	1; 5,773.

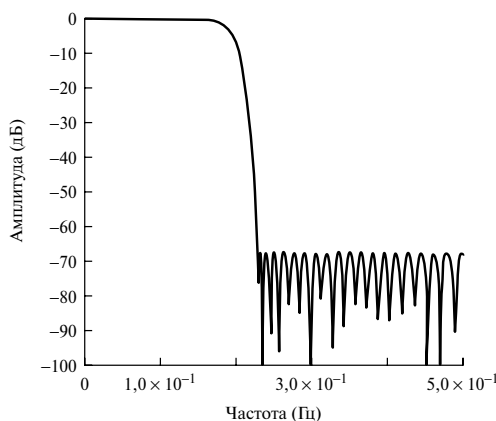


Рис. 7.33. Спектр фильтра до квантования (пример 7.19)

Спектр фильтра до квантования показан на рис. 7.33. Существенной разницы между частотной характеристикой до квантования и после (16-битовые слова) не наблюдается. Неравномерность в полосе пропускания и затухание в полосе подавления после квантования равны 0,0227 и 64,15 дБ по сравнению с 0,0224 и 66,96 дБ до квантования.

Очевидно, что основным следствием квантования коэффициентов является возможное увеличение максимальной неравномерности в полосе пропускания и снижение максимального затухания в полосе подавления. Существуют практические процедуры, позволяющие учесть данные эффекты при вычислении коэффициентов фильтров. По сути, при этом спецификации неквантованных фильтров отображаются в новый набор спецификаций, который затем используется для получения коэффициентов, причем после квантования коэффициентов удовлетворяются исходные спецификации.

Фильтр, получающийся в результате такого преобразования, может не быть оптимальным. Этот факт инициировал разработку методов оптимизации для получения коэффициентов КИХ-фильтров с конечной длиной слова, таких как алгоритмы смешанного целочисленного программирования [6]. Новые подходы приводят к значительному уменьшению длин слов, используемых для представления коэффициентов, по сравнению с простым округлением, но поиск подходящих коэффициентов часто включает значительные вычислительные издержки уже при N умеренно больших. Практичнее использовать один из пределов (7.44) для оценки числа битов, необходимых для представления коэффициентов. Требуемая длина слова обычно на 1–4 бит больше данного значения, а точная величина определяется при изучении частотной характеристики, отвечающей длинам слов в данном диапазоне.

7.11.2. Ошибки округления

Напомним, что разностное уравнение КИХ-фильтра выглядит так:

$$y(n) = \sum_{m=0}^{N-1} h(m)x(n-m), \quad (7.46)$$

где каждая переменная представляется конечным числом битов. Как правило, входные и выходные выборки $x(n-m)$ и $y(n)$ представляются 12 бит каждая, а коэффициенты представляются 16 бит в формате дополнения до двух.

Из уравнения (7.46) видно, что выход фильтра — это сумма произведений $h(m)$ и $x(n-m)$. После каждого умножения произведение содержит больше битов, чем $h(m)$ или $x(n-m)$ отдельно. Например, если 12-битовый вход множится на 16-битовый коэффициент, результат размером 28 бит требуется квантовать до прежних 16 бит, если, например, далее его нужно занести в память, или до 12 бит, если затем его нужно подать на выход ЦАП. Квантование приводит к ошибкам, влияние которых сходно с влиянием шума АЦП, но иногда проявляется более значительно. Обычно в таких случаях для квантования результата арифметической операции либо а) результат усекается, т.е. оставляются только старшие биты и отбрасываются младшие, либо б) результаты округляются, т.е. выбираются старшие биты, наиболее близкие к неокругленному результату. Для реализации этого к результату прибавляется половина младшего разряда.

Ошибки округления можно минимизировать, если точно представить все произведения в регистрах двойной длины, а результаты округлять после получения финальной суммы, т.е. после вычисления $y(n)$. Данный подход приводит к меньшей ошибке, чем альтернативный метод округления каждого произведения отдельно до суммирования.

7.11.3. Ошибки переполнения

Переполнение происходит, когда сумма двух чисел, обычно больших чисел одного знака, превышает разрешенную длину слова. Значит, переполнение в контексте формулы (7.46) может произойти при сложении двух произведений, $h(0)x(n)$ и $h(1)x(n-1)$.

Если окончательный выход $y(n)$ согласуется по размеру с данной длиной слова, переполнение в частичных суммах не существенно. Данное свойство является приятной особенностью арифметики с дополнением до двух. В то же время, если выход $y(n)$ не вмещается в разрешенные границы, очевидно, что значение выходной выборки, поданной, например, в ЦАП, будет неверным, и эту ситуацию следует предотвратить. Можно детектировать и корректировать переполнение, но издержки в данном случае могут быть большими. Другая альтернатива — масштабировать коэффициенты и/или входные данные, чтобы избежать переполнения или держать его в определенных рамках. Для масштабирования коэффициентов можно использовать один из следующих подходов:

$$h(m) = \frac{h(m)}{\sum_{k=0}^{N-1} |h(k)|}, \quad (7.47)$$

$$h(m) = \frac{h(m)}{\left[\sum_{k=0}^{N-1} h^2(k) \right]^{1/2}}. \quad (7.48)$$

Если использовать преобразование (7.47, а), переполнение не произойдет никогда, но масштабирование в таком виде часто излишне, поскольку рассчитано на наихудший вариант переполнения, что на практике почти нереально. Кроме того, такой подход вводит больший шум квантования коэффициентов, чем метод с использованием преобразования (7.47, б), в котором предполагается, что переполнение происходит время от времени.

Подобно коэффициентам можно масштабировать входные данные, что часто приводит к улучшению отношения сигнал-шум. Третий подход — это масштабировать вход и выход так, чтобы получить наилучшее возможное отношение сигнал-шум. Эффективным является масштабирование с масштабом, представляющим собой степень двойки.

7.12. Методы реализации КИХ-фильтров

Разностное уравнение для цифрового КИХ-фильтра имеет вид

$$y(n) = \sum_{k=0}^{N-1} h(k)x(n-k). \quad (7.49)$$

Итак, на этапе аппроксимации получены коэффициенты $h(k)$, выбрана подходящая структура, и анализ показал, что числа битов, выделенного для представления переменных и выполнения арифметических операций, достаточно. Финальный этап — реализовать фильтр, а ключевой вопрос здесь — произвести программный код и/или аппаратную реализацию выбранной структуры фильтра. Предположим, что используется трансверсальная структура (как наиболее популярная), которая характеризуется уравнением (7.48).

Изучая уравнение, видим, что для вычисления $y(n)$ требуются только операции умножения, сложения/вычитания и задержки. Следовательно, для реализации фильтра нужны также базовые компоненты:

- память (ОЗУ) для хранения текущей и предыдущих входных выборок, $x(n)$ и $x(n-k)$;
- память (ОЗУ или ПЗУ) для хранения коэффициентов фильтра, $h(k)$;
- умножитель (аппаратный или программный);
- сумматоры или арифметико-логические устройства (АЛУ).

Данные компоненты плюс средства управления ими и образуют цифровой фильтр. Если источник входных данных аналоговый, дополнительно потребуется АЦП. Аналогично, если выход должен поступать на аналоговое устройство, потребуется ЦАП. Следовательно, фильтр реального времени имеет структуру, изображенную на рис. 7.34. Реализация фильтра традиционно разделяется на две части: аппаратную и программную. Впрочем, данное разделение уже становится артефактом в современной ЦОС, поскольку на настоящий момент практически не осталось истинно аппаратных решений,



Рис. 7.34. Упрощенная блок-схема цифрового фильтра реального времени с аналоговыми входным/выходным сигналами

так как большинство устройств, используемых в фильтрации, теперь программируемые. В данной книге любую реализацию на больших системах, таких как мэйнфреймы и персональные компьютеры, будем рассматривать как программную. В этих случаях для кодирования уравнения фильтра и действий, которые производились бы в автономном режиме, будем использовать язык высокого уровня. Реализации с использованием устройств ЦОС и специального аппаратного обеспечения, включая стандартные микропроцессоры, будут рассматриваться как аппаратные. В таких случаях уравнение фильтрации для определенного устройства может представляться аппаратно-реализованным программным обеспечением или кодом на языке ассемблера.

В большинстве приложений основной целью часто является работа в реальном времени. Наилучший выбор в этих ситуациях — аппаратная реализация. Аппаратные реализации предлагают большую скорость, но они менее гибки. В настоящее время при выборе аппаратной реализации используется три подхода: стандартные микропроцессоры (например, Motorola 68000) и процессоры ЦОС (такие как Texas Instruments TMS320), стандартные аппаратные блоки и арифметико-логические устройства. При использовании в качестве основы алгоритмов и процессоров ЦОС различные устройства, требуемые для фильтрации (умножители, сумматоры и т.д.), реализуются на аппаратном уровне и вводятся в одну ИС с использованием технологии СБИС. В то же время, алгоритмические процессоры уже предназначены для выполнения КИХ-фильтрации. Разработчику достаточно предоставить коэффициенты фильтра и необходимую связующую логику для сопряжения процессора и внешнего мира (примеры подобных процессоров — Motorola DSP56200 и INMOS A100). Архитектуры и системы команд процессоров ЦОС, оптимизированные под КИХ-фильтрацию, гибче алгоритмических процессоров, но медленнее.

Разработка систем с использованием программных или аппаратных подходов рассмотрена в главах 12 и 13.

На рис. 7.35 изображена функциональная схема КИХ-фильтрации, из которой видно, что в каждый момент выборки данные вначале смещаются на одну позицию, считываются и сохраняется последняя входная выборка $x(n)$, и с помощью разностного уравнения вычисляется текущая выходная выборка.

7.13. Пример разработки

Пример 7.20

Разработайте и реализуйте полосовой фильтр с линейной фазовой характеристикой согласно следующей спецификации:

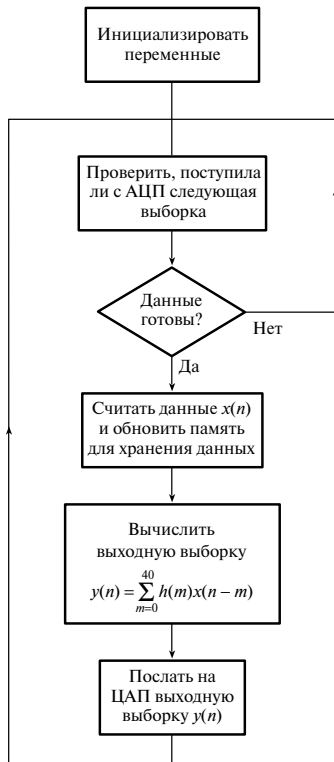


Рис. 7.35. Упрощенная функциональная схема трансверсального КИХ-фильтра реального времени

полоса пропускания	900–1100 Гц,
неравномерность в полосе пропускания	0,87 дБ,
затухание в полосе подавления	>30 дБ,
частота дискретизации	15 кГц,
число коэффициентов	41.

Для реализации фильтра следует использовать целевую плату TMS32010 (см. главу 13).

Решение

Напомним, что разработка КИХ-фильтра состоит из пяти этапов.

- *Этап 1: спецификация.* Спецификация уже дана.
- *Этап 2: расчет коэффициентов.* Используем для вычисления коэффициентов оптимальный метод, дающий наименьшее число коэффициентов фильтра (для нерекурсивных КИХ-фильтров), поскольку он доступен. Коэффициенты данного фильтра уже вычислены в предыдущем примере; см. табл. 7.7. Соответствующая частотная характеристика приведена на рис. 7.13.

- *Этап 3: выбор структуры.* Выбрана трансверсальная структура (см. рис. 7.29, а), поскольку она допускает более эффективную реализацию при использовании процессора TMS32010. Разностное уравнение для данной структуры имеет вид

$$y(n) = \sum_{m=0}^{40} h(m)x(n-m).$$

- *Этап 4: квантование и анализ ошибок.* Поскольку используется процессор TMS32010, для эффективной работы каждый коэффициент следует квантовать до 16 бит. Для этого множим каждый коэффициент на 2^{15} , а затем округляем результат до ближайшего целого числа. Покажем для примера как квантуются первые два коэффициента:

$$h(0) = -0,01534638 \times 2^{15} = -502,87 \approx -503,$$

$$h(1) = -0,0000578055 \times 2^{15} = -1,89 \approx -2.$$

Квантованные и неквантованные коэффициенты перечислены в табл. 7.16. Частотную характеристику квантованного фильтра следует проверить на предмет соответствия спецификациям, особенно в полосе подавления. Находим, что после квантования до 16 бит наблюдается незначительное отличие характеристик квантованного и неквантованного фильтров.

При использовании TMS32010 частичные суммы, следующие из разностного уравнения, выносятся в 32-битовый накапливающий сумматор. Используется довольно большой (32 бит) регистр произведения. Следовательно, влияние ошибок округления при $N = 41$ будет небольшим. Возможность переполнения в данном примере не рассматривается. Если бы это было нужным, для подавления переполнения можно было бы умножить каждый коэффициент, полученный на этапе 2, на подходящий масштабный множитель (ММ). Например,

$$\text{ММ} = \sum_{m=0}^{40} |h(m)|.$$

Целевая плата имеет только 8-битовый АЦП. Это ограничит динамический диапазон сигнала, который можно обработать, всего до 48 дБ. В аудиосистеме высокого качества, например, уровень шума квантования был бы неприемлемым, поэтому в таких случаях следует увеличивать разрешение АЦП.

- *Этап 5: реализация.* Функциональная схема операции КИХ-фильтрации изображена на рис. 7.35. Далее эта схема транслируется в код ассемблера TMS32010 и сохраняется в памяти программы (в главе 12 приводятся детали разработки и кодирования операций КИХ-фильтрации)

Таблица 7.16. Неквантованные ($h(m)$) и квантованные ($h_q(m)$) коэффициенты фильтра из примера 7.20

m	Неквантованные коэффициенты, $h(m)$	Квантованные коэффициенты, $h_q(m)$
0	$-1,534638e-02$	-503
1	$-5,780550e-05$	-2
2	$5,023483e-03$	165
3	$1,266706e-02$	415
4	$2,108206e-02$	691
5	$2,776418e-02$	910
6	$3,005362e-02$	985
7	$2,586935e-02$	848
8	$1,444566e-02$	473
9	$-3,189323e-03$	-105
10	$-2,416137e-02$	-792
11	$-4,420712e-02$	-1449
12	$-5,857453e-02$	-1919
13	$-6,318557e-02$	-2070
14	$-5,575461e-02$	-1827
15	$-3,654699e-02$	-1198
16	$-8,540099e-03$	-280
17	$2,308386e-02$	756
18	$5,201380e-02$	1704
19	$7,224807e-02$	2367
20	$7,951681e-02$	2606
21	$7,224807e-02$	2367
22	$5,201380e-02$	1704
23	$2,308386e-02$	756
24	$-8,540099e-03$	-280
25	$-3,654699e-02$	-1198
26	$-5,575461e-02$	-1827
27	$-6,318557e-02$	-2070
28	$-5,857453e-02$	-1919
29	$-4,420712e-02$	-1449
30	$-2,416137e-02$	-792
31	$-3,189323e-03$	-105
32	$1,444566e-02$	473
33	$2,586935e-02$	848
34	$3,005362e-02$	985
35	$2,776418e-02$	910
36	$2,108206e-02$	691
37	$1,266706e-02$	415
38	$5,023482e-03$	165
39	$-5,780550e-05$	-2
40	$-1,534638e-02$	-503

7.14. Резюме

Разработку цифровых фильтров можно разделить на пять взаимозависимых этапов: спецификация фильтра, расчет коэффициентов, реализация, анализ ошибок и реализация фильтра в аппаратной и/или программной форме.

Спецификация фильтра зависит от приложения, но должна включать спецификацию амплитудной и/или фазовой характеристики.

Расчет коэффициентов, по сути, включает поиск значений $h(m)$, соответствующих спецификациям. Наиболее распространенные методы вычисления коэффициентов КИХ-фильтра: 1) вырезание; 2) частотная выборка; 3) оптимальный метод. Метод взвешивания прост, но ему недостает гибкости, особенно при наличии разных амплитуд колебаний характеристики в полосе пропускания и подавления. Метод частотной выборки великолепно подходит для рекурсивной реализации КИХ-фильтров и в тех случаях, когда требуются фильтры, отличные от стандартных частотно-избирательных фильтров (нижних частот, верхних частот, полосовых и режекторных). Оптимизационный метод наиболее мощный и гибкий. Все три метода подробно рассмотрены в данной главе.

Тремя наиболее распространенными структурами реализации КИХ-фильтров являются трансверсальная, которая включает прямую свертку с использованием коэффициентов фильтра, структура частотной выборки, которая непосредственно связана с вычислением коэффициентов методом частотной выборки, и структура быстрой свертки. Выбор структуры зависит от предполагаемого приложения.

На производительность КИХ-фильтров большой длины или с сильным затуханием в полосе подавления может влиять конечная длина слова, например, частотные характеристики этих фильтров могут меняться после квантования коэффициентов. Следовательно, с помощью характеристик таких фильтров необходимо проверять, достаточно ли длинные слова используются, особенно если длину слова предполагается выбрать меньшей, чем 12 бит.

К реализации фильтра обычно приступают после успешного завершения первых четырех этапов, и этот последний этап включает программное кодирование или аппаратную реализацию выбранной структуры.

7.15. Примеры применения КИХ-фильтров

Существует множество областей, где применяются КИХ-фильтры, в том числе обработка при нескольких скоростях [1], подавление шума [2], согласованная фильтрация (см. главу 13) и обработка изображений [17].

В системах с обработкой при нескольких скоростях КИХ-фильтры успешно используются для эффективной цифровой фильтрации с целью подавления наложения и зеркальных частот в таких областях, как сбор высококачественных данных и воспроизведение компакт-дисков (см. главу 9).

Задачи

Концепция КИХ-фильтра

- 7.1. Частотную характеристику $H(\omega)$ КИХ-фильтра второго типа с линейной фазовой характеристикой можно выразить так (см. табл. 7.1):

$$H(\omega) = \exp^{-i\omega(N-1)/2} \sum_{n=1}^{N/2} b(n) \cos[\omega(n - \frac{1}{2})],$$

где $b(n)$ зависит от коэффициентов фильтра. Объясните, почему фильтры с приведенной выше характеристикой неустойчивы при использовании в качестве фильтров верхних частот. Для иллюстрации ответа используйте простую ситуацию (например, $N = 4$).

- 7.2. КИХ-фильтр имеет импульсную характеристику $h(n)$, которая определена в интервале $0 \leq n \leq N-1$. Покажите, что при N четном и $h(n)$ четно-симметричном (т.е. $h(n) = h(N - n - 1)$) фильтр имеет линейную фазовую характеристику. Получите выражения для амплитудной и фазовой характеристики фильтра.

Метод взвешивания

- 7.3. Покажите, что импульсная характеристика идеального полосового фильтра (см. табл. 7.2) записывается как

$$\begin{aligned} h_D(n) &= 2f_2 \frac{\sin n\omega_2}{n\omega_2} - 2f_1 \frac{\sin n\omega_1}{n\omega_1} \quad n \neq 0 \\ &= 2(f_2 - f_1) \quad n = 0, \end{aligned}$$

где f_1 и f_2 — нижняя и верхняя частоты полосы пропускания соответственно.

- 7.4. 1. Используя метод вырезания, получите коэффициенты цифрового КИХ-фильтра нижних частот согласно такой спецификации:

затухание в полосе подавления	50 дБ,
край полосы пропускания	3,4 кГц,
ширина перехода	0,6 кГц,
частота дискретизации	8 кГц.

В ответе укажите использованную весовую функцию и аргументы в пользу такого выбора.

2. Предполагая, что коэффициенты фильтра хранятся в смежных ячейках памяти микрокомпьютера, перечислите значения коэффициентов в порядке их хранения.
3. Изобразите и кратко прокомментируйте функциональную схему прямой программной реализации фильтра в реальном времени и предложите два пути улучшения эффективности программной реализации.

Примечание. Можно использовать информацию, предоставленную в табл. 7.2.

Оптимизационные методы (Паркса–Мак-Клиллана)

- 7.5. 1. КИХ-фильтр с линейной фазовой характеристикой имеет импульсную характеристику, удовлетворяющую следующему условию симметрии:

$$h(n) = h(N - n - 1), \quad n = 0, 1, \dots, (N - 1)/2,$$

где N — число коэффициентов фильтра. Предполагая, что N — нечетное, определите амплитудную и фазовую характеристику фильтра и покажите, что фильтр имеет постоянную фазовую и групповую задержки. Прокомментируйте практическое значение линейной фазовой характеристики цифрового фильтра.

2. Требуется создать цифровой полосовой фильтр с линейной фазовой характеристикой для извлечения характерных элементов в некотором анализаторе сигналов. Нужно, чтобы фильтр удовлетворял следующим спецификациям:

полоса пропускания	12–16 кГц,
ширина перехода	3 кГц,
частота дискретизации	96 кГц,
неравномерность в полосе пропускания	0,01 дБ,
затухание в полосе подавления	80 дБ.

Предположим, что коэффициенты фильтра нужно вычислить с использованием оптимального метода (замена Ремеза). Определите следующие параметры фильтра:

- число коэффициентов фильтра, N ;
- подходящие весовые коэффициенты для полос фильтра;
- частоты краев полос в форме, подходящей для оптимального метода. Кратко объясните роли весовых коэффициентов и частот сетки в оптимальном методе.

Предложите подходящую для поставленной задачи плотность решетки.

Можно использовать информацию, предоставленную в табл. 7.17.

- 7.6. Цифровой КИХ-фильтр нижних частот должен удовлетворять следующим спецификациям:

затухание в полосе подавления	>40 дБ,
край полосы пропускания	100 Гц,
неравномерность в полосе пропускания	<0,05 дБ,
ширина перехода	10 Гц,
частота дискретизации	1024 Гц.

- Вычислите и приведите коэффициенты фильтра, указав, какой метод выбран и почему.
- Требуется с помощью метода быстрой свертки создать фильтр для работы в реальном времени. Покажите, как реализовать фильтр с преобразованием Фурье, используя метод наложения-записи. Явно укажите такие параметры, как число выборок, на которое перекрываются входные блоки, размер используемого преобразования и получение выборок из Фурье-образов.

Таблица 7.17. Соотношения для оценки длины N полосового фильтра

$$N \approx \frac{C_{\infty}(\delta_s, \delta_s)}{\Delta F} + g(\delta_p, \delta_s) \Delta F + 1,$$

где

$$C_{\infty}(\delta_p, \delta_s) = [\lg \delta_s] [b_1 (\lg \delta_p)^2 + b_2 \lg \delta_p + b_3] + \\ + [b_4 (\lg \delta_p)^2 + b_5 \lg \delta_p + b_6]$$

$$g(\delta_p, \delta_s) = -14,6 \lg \left(\frac{\delta_p}{\delta_s} \right) - 16,9$$

$$\begin{aligned} b_1 &= 0,01202 & b_2 &= 0,09664 \\ b_3 &= -0,51325 & b_4 &= 0,00203 \\ b_5 &= -0,5705 & b_6 &= -0,44314 \end{aligned}$$

ΔF — ширина перехода, нормированная на частоту дискретизации,

δ_p — неравномерность (или колебание характеристики) в полосе пропускания,

δ_s — неравномерность (или колебание характеристики) в полосе подавления

- 7.7. Требуется 41-точечный КИХ-дифференциатор с линейной фазовой характеристикой, который должен удовлетворять следующим спецификациям:

край полосы пропускания	1 кГц,
край полосы подавления	1,5 кГц,
частота дискретизации	10 кГц,
отклонение в полосе пропускания	0,01,
отклонение в полосе подавления	0,01.

Вычислите коэффициенты дифференциатора, используя оптимальный метод (алгоритм Паркса–Мак-Клиллана/замена Ремеза). Изобразите его амплитудно-частотную характеристику.

- 7.8. Требуется разработать 43-точечный КИХ-фильтр, реализующий преобразование Гильберта, с линейной фазовой характеристикой, который удовлетворяет следующим спецификациям:

верхняя краевая частота	1 кГц,
нижняя краевая частота	4,5 кГц,
частота дискретизации	10 кГц,
отклонение в полосе пропускания	0,01.

Вычислите коэффициенты фильтра, используя оптимальный метод. Изобразите его амплитудно-частотную характеристику в децибелах.

Фильтры частотной выборки

- 7.9. Цифровой 4-точечный КИХ-фильтр с линейной фазовой характеристикой описывается следующими частотными выборками:

$$\begin{aligned} |H(k)| &= 1, & k &= 0 \\ &\frac{1}{2}, & k &= 1, 3 \\ &0, & k &= 2 \end{aligned}$$

- а) начав с общего выражения передаточной функции (7.24), покажите, что передаточная функция описанного выше фильтра содержит четыре нуля и три полюса;
 - б) изобразите диаграмму нулей и полюсов данного фильтра;
 - в) изобразите частотную характеристику фильтра;
 - г) разработайте и изобразите диаграмму реализации фильтра с объединенными комплексно-сопряженными полюсами, используя для этой цели структуру частотной выборки;
 - д) определите четыре коэффициента фильтра. Коэффициенты должны быть действительными.
- 7.10. Требуется 4-точечный КИХ-фильтр с линейной фазовой характеристикой, реализованный по принципу частотной выборки, который характеризуется следующими выборками:
- $$|H(k)| = 1, \quad k = 0$$
- $$0, \quad k = 1, 2, 3$$
- а) начав с общего выражения передаточной функции (7.24), определите число полюсов и нулей передаточной функции фильтра;
 - б) изобразите диаграмму нулей и полюсов фильтра;
 - в) разработайте и изобразите диаграмму реализации фильтра с объединенными комплексно-сопряженными полюсами, используя для этой цели структуру частотной выборки;
 - г) определите четыре коэффициента фильтра. Коэффициенты должны быть действительными.
- 7.11. Фильтры, реализованные по принципу выборки частоты, имеют определенные особенности, общие с КИХ- и БИХ-фильтрами. Некоторые из этих особенностей рассматриваются в данной задаче.
1. В чем заключается основное преимущество рекурсивных фильтров частотной выборки по сравнению с нерекурсивными эквивалентами?
 2. Опишите проблему влияния конечной длины слова, связанную с рекурсивными фильтрами частотной выборки, и предложите, как ее можно преодолеть.
 3. На рис. 7.36 изображена диаграмма нулей и полюсов полосового фильтра частотной выборки.
 - А. Изучив диаграмму нулей и полюсов, запишите значения частотных выборок $H(k)$ полосового фильтра на частотах

$$\omega_k = \frac{2\pi k}{N}, \quad k = 0, 1, \dots, 7.$$

Можно использовать любые разумные допущения.

- Б. Изобразите амплитудно-частотную характеристику фильтра, на которой отметьте моменты выборки.
- В. Вычислите коэффициенты фильтра, определив таким образом передаточную функцию $H(z)$ в рекурсивной форме.

Таблица 7.18. Оптимальные частотные выборки в полосе перехода для фильтров нижних частот типа 1, полученных методом частотной выборки для $N = 15$ (адаптировано из [15])

ШП	Затухание в полосе подавления (дБ)	T_1	T_2	T_3
Одна выборка в полосе перехода, $N = 15$				
1	42,309 322 83	0,433 782 96		
2	41,262 992 86	0,417 938 23		
3	41,253 337 86	0,410 473 63		
4	41,949 077 13	0,404 058 84		
5	44,371 245 38	0,392 681 89		
6	56,014 165 88	0,357 665 25		
Две выборки в полосе перехода, $N = 15$				
1	70,605 405 85	0,095 001 22	0,589 954 18	
2	69,261 681 56	0,103 198 24	0,593 571 18	
3	69,919 734 95	0,100 836 18	0,589 432 70	
4	75,511 722 56	0,084 074 93	0,557 153 12	
5	103,460 783 00	0,051 802 06	0,499 174 24	
Три выборки в полосе перехода, $N = 15$				
1	94,611 661 91	0,014 550 78	0,184 578 82	0,668 976 13
2	104,998 130 80	0,010 009 77	0,173 607 13	0,659 515 26
3	114,907 193 18	0,008 734 13	0,163 973 10	0,647 112 64
4	157,292 575 84	0,003 787 99	0,123 939 63	0,601 811 54

Ширина полосы (ШП) в данном контексте — число частотных выборок в полосе пропускания.

4. Объясните, как потоковый граф БПФ в двоичной системе (рис. 7.36, б) можно использовать для вычисления импульсной характеристики фильтра по полученным выше частотным выборкам.
- 7.12. 1. С помощью схем объясните основные концепции разработки фильтров методом частотной выборки.
2. Требуется, чтобы цифровой фильтр нижних частот удовлетворял следующим спецификациям:

полоса пропускания	0–20 Гц,
частота дискретизации	300 Гц,
затухание в полосе подавления	>50 дБ,
длина фильтра	15.
3. Найдите коэффициенты передаточной функции цифрового фильтра в рекурсивной форме, используя метод частотной выборки и информацию из табл. 7.18.
 - А. Разработайте и изобразите диаграмму реализации фильтра и сравните требования к памяти и вычислениям рекурсивной реализации с реализацией КИХ-фильтра в прямой форме.
 - Б. Объясните, почему фильтр, представленный указанной выше передаточной функцией, — это все еще КИХ-фильтр, несмотря на то, что из его передаточной функции следует, что это рекурсивный фильтр. Прокомментируйте трудности, которые могут встретиться на практике при использовании рекурсивных фильтров частотной выборки, и покажите, как их можно преодолеть.

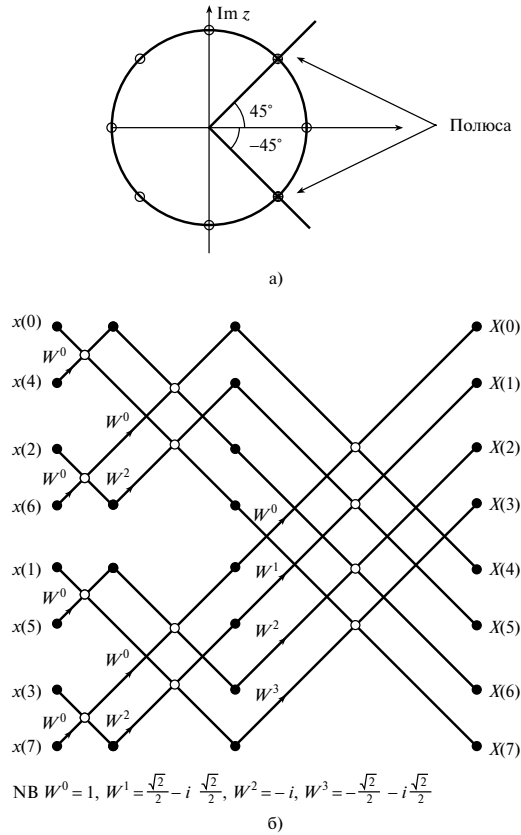


Рис. 7.36

- 7.13. На рис. 7.37 показана диаграмма нулей и полюсов простого полосового фильтра частотной выборки.
1. Изобразите амплитудно-частотную характеристику фильтра, записав таким образом значения амплитудно-частотной характеристики в точках выборки.
 2. Получите передаточную функцию фильтра частотной выборки, начав с выражения общей передаточной функции (7.24). Ответ прокомментируйте.
 3. Изобразите диаграмму реализации фильтра и запишите разностное уравнение.
 4. Сравните реализацию в форме частотной выборки и прямой форме с точки зрения их требований к вычислениям и памяти.
- 7.14. На рис. 7.38 показана диаграмма нулей и полюсов простого полосового фильтра частотной выборки.
1. Изобразите амплитудно-частотную характеристику фильтра с одними нулями.
 2. Изобразите амплитудно-частотную характеристику фильтра, записав таким образом значения амплитудно-частотной характеристики в точках выборки.

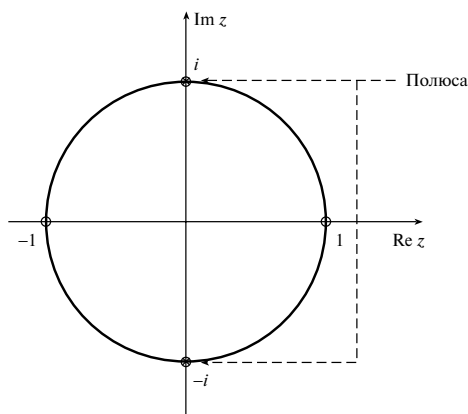


Рис. 7.37

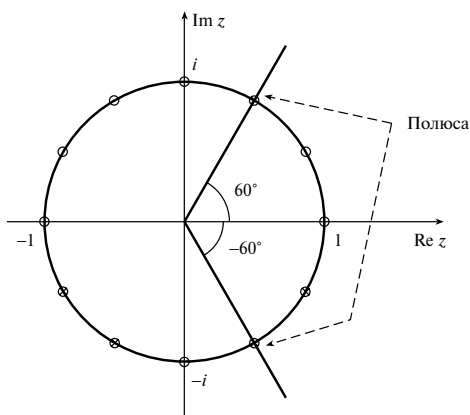


Рис. 7.38

3. Запишите передаточную функцию фильтра в рекурсивной форме.
 4. Получите передаточную функцию фильтра частотной выборки, начав с выражения общей передаточной функции (7.24). Ответ прокомментируйте.
 5. Изобразите диаграмму реализации фильтра и запишите разностное уравнение.
- 7.15.
1. Кратко обсудите условия, необходимые, чтобы реализуемый цифровой фильтр имел линейную фазовую характеристику, и достоинства фильтров с такой характеристикой.
 2. В определенной задаче обработки сигналов выходной сигнал (значимые частотные компоненты) принадлежит диапазону $0 \leq f \leq 10$ Гц и загрязнен помехами со стороны питающей сети (частота 50 Гц). Было решено избавиться от интерференции, используя цифровой фильтр с линейной фазовой

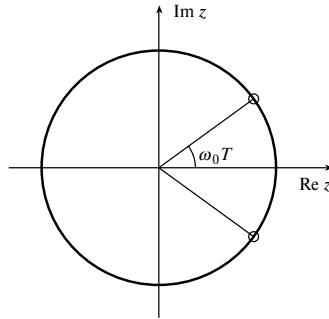


Рис. 7.39. Диаграмма нулей и полюсов для задачи 7.15; $\omega_0 T = \pi/5$ радиан

характеристикой после оцифровки составного сигнала с частотой 500 выборок/с. На первом этапе разработки фильтра была получена диаграмма нулей и полюсов, приведенная на рис. 7.39. Получите передаточную функцию $H(z)$ фильтра и разностное уравнение.

3. Фильтр, полученный в п. 2, нужно реализовать на микрокомпьютере с простой арифметикой, ограниченной только сложением/вычитанием и сдвигами. Переделайте фильтр так, чтобы коэффициенты были целыми числами. При этом число коэффициентов фильтра или частота дискретизации увеличиваться не должны.
4. Покажите, что фазовая характеристика $\theta(\omega)$ фильтра, полученного в п. 3, записывается так:

$$\theta(\omega) = -\omega T.$$

- 7.16. 1. Требуется узкополосный цифровой фильтр реального времени с линейной фазовой характеристикой для использования в определенной биомедицинской системе. Обоснуйте применение в такой системе фильтра частотной выборки.

Предположим, что передаточная функция N -точечного фильтра частотной выборки записывается так:

$$H(z) = \frac{1 - r^N z^{-N}}{N} \left(\sum_{k=1}^M \frac{|H(k)| [2 \cos(2\pi k \alpha / N) = 2r \cos[2\pi k(1 + \alpha)/N]]}{1 - 2r \cos(2\pi k / N) z^{-1} + r^2 z^{-2}} + \frac{H(0)}{1 - z^{-1}} \right),$$

где $H(k)$ — выборки желаемой частотной характеристики, взятые в точках F_s/N , $\alpha = (N - 1)/2$.

2. Нужный фильтр описывается такой спецификацией:

полоса пропускания	48–52 Гц,
ширина перехода	2 Гц,
частота дискретизации	500 Гц,
затухание в полосе подавления	>60 дБ.

Задайте подходящие частотные выборки $|H(k)|$. Разработайте и изобразите диаграмму реализации фильтра. Сравните полученный фильтр с эквивалентной трансверсальной структурой с точки зрения требований к вычислениям и памяти.

3. Прокомментируйте полученную выше функцию $H(z)$ и сложности, которые могут возникнуть при использовании рекурсивных фильтров частотной выборки. Укажите, как эти трудности можно преодолеть.

Объясните, почему $H(z)$ описывает рекурсивный фильтр и получите импульсную характеристику $h(n)$ этого фильтра, имеющую конечную длительность.

- 7.17. Требуется N -точечный КИХ-фильтр со следующей частотной характеристикой:

$$H(e^{i\omega}) = |H(e^{i\omega})|e^{-i\omega\alpha},$$

где $\alpha = (N - 1)/2$. Предполагается, что N выборок $H(e^{i\omega})$ берутся в точках $f_k = (k + 1/2)F_s/N$, $k = 0, 1, \dots, N - 1$.

1. Покажите, что для N четного импульсная характеристика имеет вид

$$h(n) = \frac{1}{N} \left\{ \sum_{k=0}^{N/2-1} 2|H(k)| \cos[2\pi(n - \alpha)(k + 1/2)/N] \right\}.$$

2. Покажите, что при N нечетном импульсная характеристика имеет вид

$$h(n) = \frac{1}{N} \left\{ \sum_{k=0}^{(N-3)/2} 2|H(k)| \cos[2\pi(n - \alpha)(k + 1/2)/N] + |H[(N-1)/2]| \cos[\pi(n - \alpha)] \right\}.$$

3. Получите выражение для передаточной функции $H(z)$ в рекурсивной форме для п. 1 и 2.

Особые КИХ-фильтры

- 7.18. КИХ-фильтр верхних частот характеризуется следующими коэффициентами импульсной характеристики:

$$h(n) = 0, 127; -0, 026; -0, 237; 0, 017; 0, 434.$$

Запишите коэффициенты эквивалентного фильтра нижних частот с помощью преобразования частоты, описанного в разделе 7.9.2.

- 7.19. Вычислите коэффициенты полуполосного КИХ-фильтра, используя вырезающую функцию Кайзера. Фильтр должен удовлетворять следующим спецификациям:

неравномерность в полосе пропускания	0,5 дБ,
затухание в полосе подавления	45 дБ,
край полосы пропускания	2 кГц,
частота дискретизации	10 кГц.

- 7.20. Повторите задачу 7.19, используя оптимальный метод.

Реализация КИХ-фильтра

- 7.21. Аналоговый сигнал засорен компонентом с частотой 50 Гц и его гармониками на частотах 100, 150, 200, 250 и 300 Гц. Предположите, что паразитный сигнал выбирается и дискретизируется с частотой 1 кГц.

Найдите передаточную функцию простого цифрового фильтра, удаляющего помеху и ее гармоники. Изобразите диаграмму реализации цифрового фильтра. Сравните и противопоставьте влияние конечной разрядности на производительность цифровых фильтров и влияние допусков на элементы схемы на производительность аналоговых фильтров. Для иллюстрации ответа используйте узкополосный режекторный фильтр.

- 7.22. 1. Оцените влияние конечной разрядности на производительность цифровых КИХ-фильтров реального времени и предложите, как это влияние минимизировать.
2. В определенной системе цифровой обработки сигналов реального времени каждый коэффициент N -точечного КИХ-фильтра представлен n -битовым числом в формате дополнения до двух. Покажите, что максимальное затухание в полосе подавления $A_{\max}$ ограничено величиной

$$A_{\max} < 20 \lg N 2^{-B},$$

где B — длина коэффициента, а N — длина фильтра.

Используйте любые разумные допущения и прокомментируйте приведенную выше границу.

3. Ниже перечислены коэффициенты 7-точечного КИХ-фильтра. Изобразите такую диаграмму реализации фильтра, чтобы для вычисления выхода требовалось минимальное число операций умножения.

$$h(0) = -0,3$$

$$h(1) = 0,4$$

$$h(2) = 0,2$$

$$h(3) = 0,5$$

$$h(4) = 0,2$$

$$h(5) = 0,4$$

$$h(6) = -0,3$$

- 7.23. При реализации цифрового КИХ-фильтра с коэффициентами, представленными в формате с неподвижной точкой, используется дробная арифметика с дополнением до двух, коэффициенты представлены тремя битами (включая знак).

1. Определите и перечислите все возможные десятичные числа, которые можно представить. Укажите наибольшее и наименьшее десятичные числа, которые можно перечислить.

2. Ниже перечислены неквантованные коэффициенты КИХ-фильтра. Предполагается, что коэффициенты квантуются до трех бит после выделения целой части (знак включен). Перечислите квантованные коэффициенты вместе с их ошибками квантования.

n	$h(n)$
0	-0,14975
1	0,256872
2	0,69940
3	0,256872
4	-0,149725

3. Повторите п. 2, если коэффициенты округляются.

7.24. Коэффициенты КИХ-фильтра имеют вид $\{h(n) = \{-1; 0,5; 0,75\}\}$.

1. Изобразите структуру фильтра, предполагая трансверсальную реализацию.
2. Предполагая, что коэффициенты и выборки входных данных представляются 3 бит (включая бит знака), после выделения целой части определите и занесите в таблицу значения квантованных коэффициентов в двоичной и десятичной записи.
3. Покажите, что если на вход подаются данные $\{x(n) = \{0,5; -1; -0,5\}\}$, выход $y(n)$ по-прежнему будет верным, несмотря на переполнение на стадии промежуточных результатов (предполагается использование накапливающего сумматора двойной длины).
4. Покажите, что вход $\{x(n)\} = \{-1; -0,75; 0,5\}$ приведет к неверным выходным значениям из-за переполнения. Как предотвратить переполнение?

7.25. Разработайте цифровой фильтр нижних частот реального времени для подавления физиологического шума. Фильтр будет использоваться как часть большей системы ЦОС, так что число арифметических операций в фильтре должно быть минимальным.

Фильтр должен удовлетворять следующим спецификациям:

полоса пропускания	8–12 Гц,
неравномерность в полосе пропускания	0,1 дБ,
ширина перехода	2 Гц,
затухание в полосе подавления	30 дБ,
частота дискретизации	100 Гц.

Дополнительные требования:

- а) минимальное искажение гармонической связи между компонентами внутриполосных сигналов;
- б) фильтр нужно реализовать с использованием процессора TMS320C25 с аналоговым входом, оцифрованным до 12 бит.

7.26. Разработайте многополосный цифровой КИХ-фильтр, удовлетворяющий следующим спецификациям:

полоса 1	0–0,5 кГц	
	затухание в полосе подавления	49 дБ
полоса 2	1–1,5 кГц	
	неравномерность в полосе пропускания	0,3 дБ
полоса 3	1,8–2,5 кГц	
	затухание в полосе подавления	38 дБ
полоса 4	3–3,6 кГц	
	неравномерность в полосе пропускания	0,3 дБ
полоса 5	4,1–5 кГц	
	затухание в полосе подавления	55 дБ

Для реализации фильтра должна использоваться система, включающая процессор TMS320C25, 12-битовый АЦП и 12-битовый ЦАП с частотой дискретизации 10 кГц.

- 7.27. Обсудите пять основных этапов разработки цифрового фильтра, используя для иллюстрации ответа приведенную ниже задачу.

Требуется цифровой фильтр для подавления в реальном времени физиологического шума. Фильтр должен удовлетворять следующим спецификациям:

полоса пропускания	0–10 Гц,
полоса подавления	20–64 Гц,
частота дискретизации	256 Гц,
максимальная неравномерность в полосе пропускания	0,026 дБ,
затухание в полосе подавления	30 дБ.

Дополнительные требования:

- фильтр должен иметь линейную фазовую характеристику, чтобы минимально исказить внутрисполосные компоненты сигнала;
- время, доступное для проведения фильтрации, ограничено, фильтр — это часть большого процесса;
- фильтр нужно реализовать с использованием процессора TMS320C25 с аналоговым входом, оцифрованным до 12 бит.

Задачи для решения с помощью MATLAB

- 7.28. Используйте MATLAB для вычисления коэффициентов, изобразите амплитудно-частотную характеристику в децибелах и определите положения нулей каждого из следующих фильтров (получены с помощью весовой функции Хэмминга; частота дискретизации 2 кГц):
- 7-точечный полосовой КИХ-фильтр, край полосы пропускания — 200 Гц, край полосы подавления — 500 Гц;
 - 8-точечный полосовой КИХ-фильтр, край полосы пропускания — 200 Гц, край полосы подавления — 500 Гц;

- в) 7-точечный КИХ-дифференциатор, край полосы пропускания — 200 Гц, край полосы подавления — 500 Гц;
- г) 8-точечный КИХ-фильтр, реализующий преобразование Гильберта, край полосы пропускания — 200 Гц, край полосы подавления — 500 Гц.

Прокомментируйте различие и/или сходство в положениях нулей.

- 7.29. С помощью метода взвешивания требуется разработать 41-точечный полосовой КИХ-фильтр, аппроксимирующий следующую идеальную амплитудную характеристику:

$$H(f) = 1, \quad 2 \text{ кГц} \leq f \leq 4 \text{ кГц}$$

0 — в противном случае

Определите коэффициенты частотной характеристики фильтра и изобразите его амплитудно- и фазово-частотную характеристики с помощью MATLAB для следующих случаев:

- 1) прямоугольная весовая функция;
- 2) весовая функция Хэмминга.

- 7.30. Требуется, чтобы оптимальный КИХ-фильтр нижних частот с линейной фазовой характеристикой удовлетворял следующим спецификациям:

длина фильтра	21,
край полосы пропускания	2 кГц,
край полосы подавления	3 кГц,
частота дискретизации	10 кГц.

- 1. С помощью MATLAB вычислите коэффициенты фильтра и изобразите его амплитудную характеристику в децибелах и фазовую характеристику в градусах.
- 2. Вычислите и изобразите характеристики фазовой и групповой задержки фильтра.
- 3. Изучив амплитудную и фазовую характеристики, определите положение нулей.
- 4. Объясните, почему фазовая характеристика имеет разрывы. Как можно их скорректировать?

- 7.31. Требуется, чтобы цифровой КИХ-фильтр удовлетворял следующим спецификациям:

полоса пропускания	150–250 Гц,
ширина перехода	50 Гц,
неравномерность в полосе пропускания	0,1 дБ,
затухание в полосе подавления	60 дБ,
частота дискретизации	1 кГц.

Используйте функцию Хэмминга и MATLAB для расчета коэффициентов фильтра.

- 7.32. Необходим полосовой КИХ-фильтр с линейной фазовой характеристикой, удовлетворяющий следующим спецификациям:

полоса пропускания	8–12 Гц,
неравномерность в полосе подавления	0,001,
неравномерность в полосе пропускания	0,01,
частота дискретизации	48 кГц,
ширина перехода	3 кГц.

Используйте MATLAB, чтобы получить для каждого из указанных случаев амплитудно-частотную характеристику фильтра.

1. Используется весовая функция Хэмминга.
2. Используется окно Кайзера.
3. Используется оптимальный метод.
4. Используется метод частотной выборки.

Сравните четыре варианта.

- 7.33. Для выделения характерных элементов в определенном анализаторе сигналов требуется цифровой полосовой фильтр с линейной фазовой характеристикой. Фильтр должен удовлетворять следующим спецификациям:

полоса пропускания	12–16 Гц,
ширина перехода	3 кГц,
частота дискретизации	96 кГц,
неравномерность в полосе пропускания	0,01 дБ,
затухание в полосе подавления	80 дБ.

Для вычисления коэффициентов используйте оптимальный метод. Определите с помощью MATLAB следующие величины:

- 1) число коэффициентов фильтра, N ;
- 2) коэффициенты фильтра.

Изобразите амплитудно-частотную характеристику.

- 7.34. Требуется многополосный цифровой КИХ-фильтр, удовлетворяющий следующим спецификациям:

полоса 1:	0–0,5 кГц, затухание в полосе подавления ≥ 49 дБ,
полоса 2:	1–1,5 кГц, неравномерность в полосе пропускания 0,3 дБ,
полоса 3:	1,8–2,5 кГц, затухание в полосе подавления 38 дБ,
полоса 4:	3–3,6 кГц, неравномерность в полосе пропускания 0,3 дБ,
полоса 5:	4,1–5 кГц, затухание в полосе подавления 55 дБ.

Используйте оптимальный метод и MATLAB для расчета коэффициентов фильтра и изобразите его амплитудно-частотную характеристику. Частоту дискретизации считайте равной 10 кГц, а ширину полосы перехода — 100 Гц.

- 7.35. Требуется оптимальный фильтр нижних частот, удовлетворяющий следующим спецификациям:

полоса пропускания	0–6 кГц,
ширина полосы перехода	1 кГц,
неравномерность в полосе пропускания	0,1 дБ,
затухание в полосе подавления	50 дБ,
частота дискретизации	16 кГц.

Определите длину фильтра и коэффициенты с помощью команд MATLAB `remezord` и `remez`.

Изобразите амплитудно-частотную характеристику фильтра.

- 7.36. Вычислите коэффициенты полуполосного КИХ-фильтра, используя весовую функцию Кайзера и MATLAB. Фильтр должен удовлетворять следующим спецификациям:

неравномерность в полосе пропускания	0,5 дБ,
затухание в полосе подавления	45 дБ,
край полосы пропускания	2 кГц,
частота дискретизации	10 кГц.

Повторите задачу 7.19, используя оптимальный метод и команды MATLAB `remezord` и `remez`.

- 7.37. Требуется разработать 41-точечный КИХ-дифференциатор с линейной фазовой характеристикой, удовлетворяющий следующим спецификациям:

край полосы пропускания	1 кГц,
край полосы подавления	1,5 кГц,
частота дискретизации	10 кГц,
неравномерность в полосе пропускания	0,01,
неравномерность в полосе подавления	0,01.

Рассчитайте коэффициенты дифференциатора, используя оптимальный метод (алгоритм Паркса–Мак-Клиллана) и MATLAB. Изобразите его амплитудно-частотную характеристику.

- 7.38. Требуется разработать 43-точечный КИХ-фильтр, реализующий преобразование Гильберта, с линейной фазовой характеристикой, удовлетворяющий следующим спецификациям:

нижняя граничная частота	1 кГц,
верхняя граничная частота	4,5 кГц,
частота дискретизации	10 кГц,
колебание характеристики в полосе пропускания	0,01.

Вычислите коэффициенты фильтра, используя оптимальный метод (алгоритм Паркса–Мак-Клиллана) и MATLAB. Изобразите его амплитудно-частотную характеристику в децибелах.

Литература

1. Crochiere R. E. and Rabiner L. R. (1981) Interpolation and decimation of digital signals — a tutorial review. *Proc. IEEE*, **69**(3), 300–331.
2. Hamer C. F., Ifeachor E. C. and Jervis B. W. (1985) Digital filtering of physiological signals with minimal distortion. *Medical and Biol. Eng. and Computing*, **23**, 274–278.
3. Harris S. P. and Ifeachor E. C. (1998) Automatic design of frequency sampling filters by hybrid Genetic Algorithm Techniques. *IEEE Transactions on Signal Processing*, **46**(12), December, 3304–3314.
4. Herrman O., Rabiner R. L. and Chan D. S. K. (1973) Practical design rules for optimum finite impulse response digital filters. *Bell System Technical J.*, **52**, 769–799.
5. Ifeachor E. C. and Harris S. P. (1993) A new approach to frequency sampling filter design, in *Proc. IEEE/IEEE Workshop Natural Algorithms in Signal Processing*, 5/1–8.
6. Lawrence V. B. and Salazar A. C. (1980) Finite precision design of linear-phase FIR filters. *Bell System Technical J.*, **59**(9), 1575–1598.
7. Lynn P. A. (1973) Recursive digital filters with linear phase characteristics. *Computer J.*, **15**, 337.
8. Lynn P. A. (1975) Frequency sampling filters with integer multipliers. In *Introduction to Digital Filtering*, Bogner R. E. and Constantinides A. G. (eds). New York: Wiley.
9. McClellan J. H., Parks T. W. and Rabiner L. R. (1973) A computer program for designing optimum FIR linear phase digital filters. *IEEE Trans. Audio Electroacoustics*, **21**, 506–526.
10. Mintzer F. and Liu B. (1979) Practical design rules for optimum FIR bandpass digital filters. *IEEE Trans. Acoustics, Speech Signal Processing*, **27**(2), 204–206.
11. Mitra S. K. and Kaiser J. F. (1993) *Handbook for Digital Signal Processing*. New York: Wiley.
12. Oppenheim A. V. and Schaffer R. W. (1975) *Digital Signal Processing*. Englewood Cliffs NJ: Prentice-Hall.
13. Parks T. W. and Burrus C. S. (1987) *Digital Filter Design*. New York: Wiley.
14. Rabiner L. R. and Gold B. (1975) *Theory and Applications of Digital Signal Processing*. Englewood Cliffs NJ: Prentice-Hall.
15. Rabiner L. R., Gold B. and McGonegal C. A. (1970) An approach to the approximation problem for nonrecursive digital filters. *IEEE Trans. Audio Electroacoustics*, **18**, 83–106.
16. Suckley D. (1990) Genetic algorithm in the design of FIR filters. *IEEE Proc. Part G*, **138**(2), 234–238.
17. Wade G., van Eetvelt P. and Darwen H. (1990) Synthesis of efficient low-order FIR filters from primitive sections. *IEEE Proc. Part G*, **137**(5), 367–372.

Дополнительная литература

- Bateman A. and Yates W. (1988) *Digital Signal Processing Design*. London: Pitman.
- Chan D. S. K. and Rabiner L. R. (1973) Analysis of quantization errors in the direct form for finite impulse response digital filters. *IEEE Trans. Audio Electroacoustics*, **21**(4), 354–366.
- Chan D. S. K. and Rabiner L. R. (1973) An algorithm for minimizing roundoff noise in cascade realizations of finite impulse response digital filters. *Bell System Technical J.*, **52**(3), 347–385.
- DeFatta D. J., Lucas J. G. and Hodgkiss W. S. (1988) *Digital Signal Processing: A System Design Approach*. New York: Wiley.
- Gershon A., Gopinath B. and Odlyzko A.M. (1979) Coefficient inaccuracy in transversal filtering. *Bell System Technical J.*, **58**(10), 2401–2416.
- Gold B. and Jordan K. L., Jr (1968) A note on digital filter synthesis. *Proc. IEEE (Lett.)*, **56**, 1717, 1718.
- Gold B. and Jordan K. L., Jr (1969) A direct search procedure for designing finite duration impulse response filters. *IEEE Trans. Audio Electroacoustics*, **17**, 33–36.
- Gold B. and Rader C. M. (1969) *Digital Processing of Signals*. New York: McGraw-Hill.
- Gore A. E. (1986) Cascadable digital signal processor. *New Electronics*, **19**, October, 39–41.

- Heute U. (1977) Comments on Rabiner L. R. A simplified computational algorithm for implementing FIR digital filters. *IEEE Trans. Acoustics, Speech Signal Processing*, **25**, June, 266, 267.
- Hillman G. D. (1987) DSP56200: an algorithm-specific digital signal processor peripheral. *Proc. IEEE*, **75**, September, 1185–1191.
- Knowles J. B. and Olcayto E. M. (1968) Coefficient accuracy and digital filter response. *IEEE Trans. Circuit Theory*, **15**, 31–41.
- Lin K., Frantz G. A. and Simar R. (1987) The TMS320 family of digital signal processors. *Proc. IEEE*, **75**, 1143–1159.
- Lynn P. A. (1970) Economic linear-phase recursive digital filters. *Electronics Lett.*, **6**, 143–145.
- Lynn P. A. and Fuerst W. (1989) *Introductory Digital Signal Processing with Computer Applications*. New York: Wiley.
- Mintzer F. (1982) On half-band, third-band and N th-band FIR filters and their design. *IEEE Trans. Acoustics, Speech Signal Processing*, **30**, 734–738.
- Mitra S. K. and Sherwood R. J. (1972) Canonic realizations of digital filters using the continued fraction expansion. *IEEE Trans. Audio Electroacoustics*, **20**, 185–194.
- Proakis J. G. and Manolakis D. G. (1992) *Introduction to Digital Signal Processing*. New York: Macmillan.
- Rabiner L. R. (1971) Techniques for designing finite-duration impulse response digital filters. *IEEE Trans. Communication Technology*, **19**, 188–195.
- Rabiner L. R. (1973) Approximate design relationships for lowpass FIR digital filters. *IEEE Trans. Audio Electroacoustics*, **21**, 456–460.
- Rabiner L. R. (1977) A simplified computational algorithm for implementing FIR digital filters. *IEEE Trans. Acoustics, Speech Signal Processing*, **25**, June, 259–261.
- Rabiner L. R. and Schafer R. W. (1971) Recursive and nonrecursive realizations of digital filters designed by frequency sampling techniques. *IEEE Trans. Audio Electroacoustics*, **19**, 200–207.
- Rabiner L. R. and Schafer R. W. (1972) Correction to “Recursive and nonrecursive realizations of digital filters designed by frequency sampling techniques”. *IEEE Trans. Audio Electroacoustics (Corresp.)*, **20**, 104, 105.
- Rabiner L. R., Kaiser J. F. and Schafer R. W. (1974) Some considerations in the design of multiband finite impulse response digital filters. *IEEE Trans. Acoustics, Speech Signal Processing*, **22**(6), 462–472.
- Rabiner L. R., McClellan J. H. and Parks T. W. (1975) FIR digital filter design techniques using weighted Chebyshev approximation. *Proc. IEEE*, **63**(4), 595–610.

Приложения

7.A. Программы на С для разработки КИХ-фильтров

В книге [Ifeachor, 2001] имеются следующие программы на С, облегчающие разработку КИХ-фильтров (подробности см. в приложении):

- `fresamp.c` — программа вычисления коэффициентов фильтра по методу частотной выборки;
- `optimal.c` — программа вычисления коэффициентов фильтра с помощью оптимального метода;
- `window.c` — программа вычисления коэффициентов фильтра по методу вырезания;
- `firfilt.c` — программа КИХ-фильтрации данных;
- `ncoeff.c` — программа оценки числа коэффициентов фильтра для оптимального фильтра нижних частот или полосового фильтра.

Для ограничения объема книги ниже приводится только последняя программа, `ncoeff.c` (программа 7А.1), представляющая собой прямую реализацию уравнений, приведенных в разделе 7.6.3. Для применения программы используем ее для оценки длины полосового фильтра, удовлетворяющего следующим спецификациям:

полоса пропускания	1800–3300 Гц,
полосы подавления	0–1400, 3700–5000 Гц,
частота дискретизации	10 кГц,
неравномерность в полосе пропускания	1 дБ,
затухание в полосе подавления	40 дБ.

Программа 7А.1

```

* -----*
*  программа оценки числа коэффициентов                               *
*  оптимального КИХ-фильтра нижних частот или полосового фильтра*
*                                                                 *
*  имя программы: ncoeff.c                                           *
*                                                                 *
*  Manny Ifeachor, 17.10.91                                         *
*                                                                 *
* -----*
* /
#include <stdio.h>
#include <math.h>
#include <dos.h>

int      filter__spec();
double   lpfcoeff();
double   bpfcoeff();
float    dp, ds, df;
int      ftype;

main()

{
    double N;
    ftype=filter__spec();      /* получение спецификаций фильтра */
    switch(ftype){
        case 1:
            N=lpfcoeff(); break;
        case 2:
            N=bpfcoeff(); break;
        default:
            printf("неверно задан тип фильтра \n");
            break;
    }

    printf("Число коэффициентов    \t%f\n",N);
    printf("неравномерность в полосе пропускания в дБ \t%f\n",dp);
    printf("затухание в полосе подавления в дБ \t%f\n",ds);
    printf("\n");
    printf("Для продолжения нажмите <Enter> \n");
}

```

```

        getch();
        exit(0);
    }
    /*-----*/
int      filter__spec()
{
    int itype;
    printf("программа оценки длины оптимального фильтра \n");
    printf("\n");
    printf("выберите тип фильтра \n");
    printf("1    оптимальный фильтр нижних частот \n");
    printf("2    оптимальный полосовой фильтр \n");
    scanf ("%d", &itype);
    printf("\n");
    printf("введите неравномерность в полосе пропускания  
и подавления в обычных единицах \n");
    printf("неравномерность должна быть от 0 до 1\n");
    scanf ("%f%f", &dp, &ds);
    switch(itype){
        case 1:
            printf("введите нормированную ширину  
полосы перехода\n");
            scanf ("%f", &df);
            break;
        case 2:
            printf("введите нормированную ширину полосы  
перехода - меньшую ширину \n");
            scanf ("%f", &df);
            break;
    }
    return(itype);
}
/*-----*/
double    lpfccoeff()
{
    float    ddp, dds, a1, a2, a3, a4, a5, a6, b1, b2;
    double    dinf, ff, t1, t2, t3, t4, NI;
    /*константы */
    a1=0.005309; a2=0.07114; a3=-0.4761; a4=-0.00266;
    a5=-0.5941; a6=-0.4278;
    b1=11.01217; b2=0.5124401;
    ddp=log10(dp);
    dds=log10(ds);
    t1=a1*ddp*ddp;
    t2=a2*ddp;
    t3=a4*ddp*ddp;
    t4=a5*ddp;
    dinf=((t1+t2+a3)*dds) +(t3+t4+a6);
    ff=b1+b2*(ddp-dds);
    NI=((dinf/df)-(ff*df)+1);
    dp=20*log10(1+dp); ds=-20*log10(ds);
}

```

Таблица 7А.1. Запросы, ответы и выход программы `ncoeff.c`

```

программа оценки длины оптимального фильтра
выберите тип фильтра
1    оптимальный фильтр нижних частот
2    оптимальный полосовой фильтр
2
ведите неравномерность в полосе пропускания и подавления в обычных
единицах
неравномерность должна быть от 0 до 1
0.122 0.01
введите нормированную ширину полосы перехода – меньшую ширину
0.04
Число коэффициентов                31.261084
неравномерность в полосе пропускания в дБ    0.999857
затухание в полосе подавления в дБ          40.000000
для продолжения нажмите <Enter>

```

```

        return(NI);
}
/*-----*/
double    bpfcoeff()
{
    float    a1, a2, a3, a4, a5, a6, ddp, dds;
    double    t1, t2, t3, t4, cinf, ginf, Nb;
    a1=0.01201, a2=0.09664, a3=-0.51325; a4=0.00203;
    a5=-0.57054; a6=-0.44314;
    ddp=log10(dp);
    dds=log10(ds);
    t1=a1*ddp*ddp;
    t2=a2*ddp;
    t3=a4*ddp*ddp;
    t4=a5*ddp;
    cinf=dds*(t1+t2+a3)+t3+t4+a6;
    ginf=-14.6*log10(dp/ds)-16.9;
    Nb=(cinf/df) + ginf*df+1;
    dp=20*log10(1+dp); ds=-20*log10(ds);
    return(Nb);
}

```

Из спецификации следует, что нормированная ширина полосы перехода равна 0,04 (450/10 000), неравномерность в полосе пропускания — $0,122 (20 \lg(1 + 1))$, а неравномерность в полосе подавления равна 0,01 ($-20 \lg(40)$). Запросы программы, ответы пользователя и выход программы из предыдущего примера приведены в табл. 7А.1. Число коэффициентов фильтра, 31, в данном случае является лишь оценкой. В большинстве практических ситуаций для соответствия спецификациям требуется большее значение длины фильтра (т.е. большее число коэффициентов), чем выдает программа. В приведенном выше примере реальная длина для соответствия спецификации равна 35. Разработчик всегда должен помнить данный момент, используя программу.

7.Б. Разработка КИХ-фильтра с помощью MATLAB

Прекрасный набор программ и функций разработки и анализа различных типов цифровых КИХ-фильтров содержит средство Signal Processing Toolbox в MATLAB. Доступ к программам и командам производится с помощью команд высокого уровня, что делает Toolbox ценным инструментом при разработке и изучении КИХ-фильтров без “закапывания” в пространное программирование.

В данном разделе иллюстрируется использование некоторых функций и программ MATLAB для разработки КИХ-фильтров с линейной фазовой характеристикой. В частности, иллюстрируется вычисление средствами MATLAB коэффициентов КИХ-фильтров с линейной фазовой характеристикой с использованием методов вырезания, оптимального (Паркса–Мак-Клиллана) и частотной выборки, что дополняет программу на С, рассмотренную в предыдущем разделе.

7.Б.1. Метод взвешивания

Этапы вычисления коэффициентов стандартного частотно-избирательного КИХ-фильтра с линейной фазовой характеристикой с помощью метода взвешивания можно упорядочить следующим образом.

1. Задать необходимую частотную характеристику.
2. Выбрать весовую функцию и оценить число коэффициентов фильтра, N .
3. Получить идеальную частотную характеристику, $h_D(n)$ (усеченную до N значений).
4. Получить N коэффициентов весовой функции, $w(n)$.
5. Получить коэффициенты КИХ-фильтра, воздействуя на частотную характеристику весовой функцией, $h(n) = h_D(n) \times w(n)$.

Для стандартного частотно-избирательного КИХ-фильтра с линейной фазовой характеристикой (нижних частот, верхних частот, полосовые и режекторные фильтры), при разработке которого используется метод вырезания, ключевой командой высокого уровня в Toolbox является команда `fir1`. Синтаксис данной команды (в стандартной форме):

$$b = \text{fir1}(N - 1, F_c).$$

Команда в стандартной форме вычисляет и возвращает коэффициенты N -точечной импульсной характеристики КИХ-фильтра с частотой среза F_c . Команда возвращает коэффициенты в вектор b , упорядоченный по возрастанию отрицательной степени z :

$$b(z) = b(0) + b(1)z^{-1} + b(2)z^{-2} + \dots + b(N - 1)z^{-(N-1)}.$$

Параметр команды $N - 1$ задает порядок фильтра (обычно на единицу меньше числа коэффициентов КИХ-фильтра). Частота среза F_c нормирована на частоту Найквиста (т.е. половину частоты дискретизации).

По умолчанию стандартная команда `fir1` действует на данные весовой функцией Хэмминга и в ней предполагается использование фильтра нижних частот (или полосовой фильтр, если F_c задает более одной частоты среза). Стандартную команду можно

расширить, задав тип фильтра и/или весовую функцию. В этом случае используется такой синтаксис:

```
b = fir1(N - 1, Fc, 'тип фильтра')
b = fir1(N - 1, Fc, window)
b = fir1(N - 1, Fc, 'тип фильтра', window)
```

Для фильтра верхних частот (highpass) тип задается словом “high”, для режекторного (stopband) — словом “stop”. Для полосовых и режекторных фильтров переменная F_c — вектор, который задает частоты среза. Для фильтров верхних частот и режекторных длина фильтра должна быть нечетным целым числом (четные целые не подходят, поскольку, как показано ниже, это приведет к нулевой амплитудной характеристике на частоте Найквиста).

MATLAB поддерживает использование различных весовых функций, включая функции Хэмминга (Hamming), Хеннинга (Hanning), прямоугольную (boxcar), Кайзера (Kaiser) и Чебышева (Chebyshev). Для получения весовых коэффициентов следует использовать такой синтаксис:

```
w = boxcar(N)
w = blackman(N)
w = hamming(N)
w = hanning(N)
w = kaiser(N, beta)
```

На практике команда взвешивания часто вкладывается в команду `fir1` (см. примеры ниже).

Нужно отметить, что из-за различий в реализациях результаты, полученные при разработке КИХ-фильтров на основе методов вырезания с помощью MATLAB, могут отличаться от результатов, полученных с помощью других программ. Например, в MATLAB после вырезания коэффициенты импульсной характеристики могут масштабироваться для получения в середине полосы пропускания единичной амплитудно-частотной характеристики. Чтобы запретить такое поведение, следует добавить слово “noscale”, например, `b = fir1(N - 1, Fc, 'noscale')`. Кроме того, реализация в MATLAB большинства весовых функций может немного меняться, приводя к различным результатам. Разработчик должен знать об этих отличиях и при необходимости делать соответствующие допущения, чтобы их компенсировать.

Пример 7Б.1

Определите коэффициенты КИХ-фильтра нижних частот с линейной фазовой характеристикой с краями полосы пропускания и полосы подавления на частотах 1 и 4,3 кГц соответственно. Используйте весовую функцию Хэмминга, частоту дискретизации предполагайте равной 10 кГц.

Решение

Из табл. 7.3 находим подходящее соотношение между шириной перехода Δf и длиной фильтра N для фильтра на основе весовой функции Хэмминга:

$$N \approx \frac{3,3}{\Delta f}.$$

Таблица 7Б.1. Параметры фильтра из примера 7Б.1

n	Усеченная идеальная импульсная характеристика, $hD(n)$	Весовые коэффициенты, $w(n)$	Коэффициенты фильтра, $h(n)$
0	0,0641	0,0800	0,0053
1	−0,0388	0,1876	−0,0075
2	−0,1052	0,4601	−0,0496
3	0,1235	0,7700	0,0974
4	0,4564	0,9723	0,4544
5	0,4564	0,9723	0,4544
6	0,1235	0,7700	0,0974
7	−0,1052	0,4601	−0,0496
8	−0,0388	0,1876	−0,0075
9	0,0641	0,0800	0,0053

В нашем случае Δf равно 0,33 (из $(4,1 - 1)/10$), так что длина фильтра $N = 10$. Используя тот же подход, что и в тексте главы, принимаем, что реальная частота среза (с учетом эффекта размывания) лежит посередине между заданными краями полос пропускания и подавления, т.е. на частоте 2,65 кГц. В MATLAB частоту среза можно нормировать на половину частоты дискретизации. Тогда f_c (нормированная) = $2,65/5 = 0,53$.

Команды MATLAB приведены в программе 7Б.1. Значения усеченной идеальной импульсной характеристики, коэффициенты весовой функции и финального фильтра представлены в табл. 7Б.1.

Программа 7Б.1. Реализация решения задачи 7Б.1 в форме m-файла MATLAB

```
fc=0.53; % Частота среза (нормированная на Fs/2)
N=10; % Длина фильтра (число отводов)
nd=fir1(N-1,fc,boxcar(N)); % Усеченная идеальная импульсная характеристика
wn=hamming(N); % Вычислить коэффициенты функции Хэмминга
hn=fir1(N-1,fc,wn); % Получить коэффициенты взвешенной функции
```

Пример 7Б.2

Иллюстрация вычисления коэффициентов КИХ-фильтра с использованием весовой функции Кайзера. Определите коэффициенты и изобразите амплитудно-частотную характеристику полосового КИХ-фильтра, используя весовую функцию Кайзера и MATLAB. Фильтр должен удовлетворять следующим спецификациям:

полоса пропускания	150–250 Гц,
ширина полосы перехода	50 Гц,
неравномерность в полосе пропускания	0,1 дБ,
затухание в полосе подавления	60 дБ,
частота дискретизации	1 кГц.

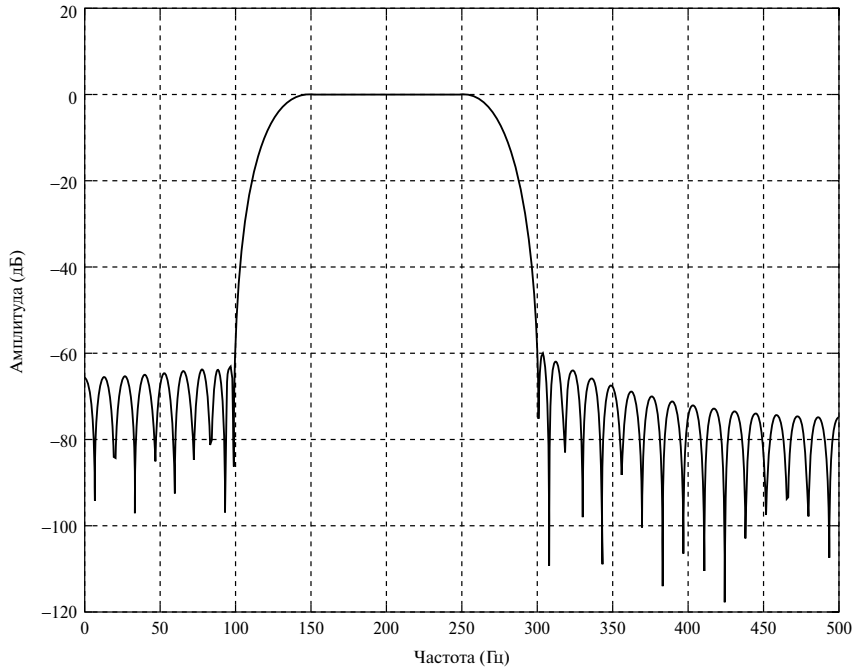


Рис. 7Б.1

Решение

Данная задача идентична примеру 7.4 в тексте главы. В этот раз для решения задачи используем средства MATLAB.

В примере 7.4 была найдена длина фильтра $N = 73$ и параметр неравномерности $\beta = 5,65$. Реализация решения задачи в форме m-файла MATLAB приведена в программе 7Б.2. Коэффициенты фильтра и амплитудный спектр представлены в табл. 7Б.2 и на рис. 7Б.1.

Следует отметить, что в данном примере коэффициенты весовой функции не вычисляются отдельно, просто в команду `fir1` включен тип весовой функции.

Программа 7Б.2. Реализация решения примера 7Б.2 в форме m-файла MATLAB

```

FS=1000; % Частота дискретизации
FN=FS/2; % Частота Найквиста
N=73; % Длина фильтра
beta=5.65; % Параметр неравномерности функции Кайзера
fc1=125/FN; % Нормированные частоты среза
fc2=275/FN;
FC=[fc1 fc2]; % Вектор краевых частот
hn=fir1(N-1,FC,kaizer(N,beta)); % Получить коэффициенты фильтра
[H,f]=freqz(hn,1,512,FS); % Вычислить частотную характеристику
mag=20*log10(abs(H));
plot(f,mag), grid on
xlabel('Частота (Гц)')
ylabel('Амплитудная характеристика (дБ)')
  
```

Таблица 7Б.2. Коэффициенты фильтра из примера 7Б.2

n	$h(n)$	n	$h(n)$	n	$h(n)$	n	$h(n)$
0	-0,0001	10	-0,0005	20	0,0069	30	0,0094
1	-0,0004	11	-0,0044	21	0,0189	31	0,0856
2	-0,0001	12	-0,0022	22	0,0029	32	0,0453
3	-0,0001	13	0,0069	23	0,0044	33	-0,1665
4	-0,0007	14	0,0066	24	0,0188	34	-0,2066
5	0,0005	15	-0,0016	25	-0,0125	35	0,0891
6	0,0023	16	0,0000	26	-0,0520	36	0,2998
7	0,0008	17	0,0022	27	-0,0165		
8	-0,0017	18	-0,0117	28	0,0333		
9	-0,0005	19	-0,0164	29	0,0104		

7.Б.2. Оптимизационные методы

Средство Signal Processing Toolbox в MATLAB содержит несколько программ разработки и функций для создания оптимальных КИХ-фильтров на основе алгоритмов Паркса–Мак–Клиллана и Ремеза. Основной командой для вычисления коэффициентов с помощью оптимального метода является `remez`. Команда может использоваться для разработки многополосных КИХ-фильтров с линейной фазовой характеристикой. В стандартной форме она имеет следующий синтаксис:

$$b = \text{remez}(N - 1, F, M),$$

где N — длина фильтра, F — вектор нормированных граничных частот, а M — вектор желаемой амплитудной характеристики фильтра на заданных граничных частотах. Граничные частоты нормированы на половину частоты дискретизации и лежат в диапазоне от 0 до 1 (частота Найквиста соответствует 1).

Стандартную команду можно расширить, задав, например, относительные коэффициенты колебаний характеристики в полосе пропускания и подавления и/или тип фильтра. Если задаются относительные весовые коэффициенты и требуются амплитуды колебаний характеристики в полосе пропускания и полосе подавления, используется следующий синтаксис:

$$b = \text{remez}(N - 1, F, M, WT),$$

где WT — вектор относительных коэффициентов амплитуд колебаний характеристики в полосах.

Для задания типа требуемого фильтра можно дополнительно добавить флаг `'ftype'`. Существует четыре возможных типа, в зависимости от четности N и типа симметрии коэффициентов фильтра. Фильтры относятся к типу 1, когда длина фильтра нечетная (т.е. когда порядок фильтра, $N - 1$, — четная величина), и к типу 2, когда длина фильтра четная. На использование фильтров типа 1 в качестве стандартных частотно-избирательных фильтров ограничений не существует. Фильтры типа 2 имеют нуль на частоте Найквиста, так что не могут использоваться при разработке фильтров верхних частот и заградительных фильтров. Фильтры типа 3 (N — нечетное) дают реализации преобразования Гильберта, и фильтры типа 4 (N — четное) — дифференциаторы. Для выбора фильтров типа 1 и 2 достаточно задать длину фильтра, но, чтобы указать тип фильтра для фильтров типа 3 и 4, необходимо включить флаг `'hilbert'` или `'differentiator'`.

Пример 7Б.3

Используя оптимальный метод, вычислите коэффициенты и изобразите частотную характеристику полосового КИХ-фильтра с линейной фазовой характеристикой, удовлетворяющего следующим спецификациям:

полоса пропускания	1000–1500 Гц,
полоса перехода	500 Гц,
длина фильтра	41,
частота дискретизации	10 000 Гц.

Решение

Полосы фильтра: 0–500 Гц (нижняя полоса подавления), 1000–1500 Гц (полоса пропускания), 2000–5000 Гц (верхняя полоса подавления). Граничные частоты нормируются на половину частоты дискретизации:

$$500/5000 = 0,1$$

$$1000/5000 = 0,2$$

$$1500/5000 = 0,3$$

$$2000/5000 = 0,4$$

$$5000/5000 = 1$$

Следовательно, вектор нормированных граничных частот F записывается в таком виде:

$$F = [0; 0,1; 0,2; 0,3; 0,4; 1].$$

Нужная амплитудная характеристика равна 1 в полосе пропускания и 0 в полосе подавления, поэтому вектор желаемой амплитудной характеристики записываем в виде

$$M = [001100]$$

Команды MATLAB для расчета коэффициентов и вывода графика амплитудной характеристики фильтра приведены в программе 7Б.3.

Коэффициенты фильтра и амплитудная характеристика приведены в табл. 7Б.3 и на рис. 7Б.2.

Программа 7Б.3. Реализация вычисления коэффициентов оптимального КИХ-фильтра и вывода на экран частотной характеристики в форме m-файла MATLAB

```
%
%
Fs=1000; % Частота дискретизации
N=41; % Длина фильтра
M=[0 0 1 1 0 0]; % Желаемая амплитудная характеристика
F=[0, 0.1, 0.2, 0.3, 0.4, 1]; % Края полос
b=remez(N-1, F, M); % Вычислить коэффициенты фильтра
[H, f]=freqz(b, 1, 512, Fs); % Вычислить частотную характеристику
```

```
mag=20*log10(abs(H)); % фильтра и нарисовать ее
plot(f,mag)
xlabel('Frequency (Hz)')
ylabel('Magnitude (dB)')
```

Таблица 7Б.3. Коэффициенты фильтра из примера 7Б.3

n	$h(n)$	n	$h(n)$
0	-0,0001	11	-0,0044
1	-0,0004	12	-0,0022
2	-0,0001	13	0,0069
3	-0,0001	14	0,0066
4	-0,0007	15	-0,0016
5	0,0005	16	0,0000
6	0,0023	17	0,0022
7	0,0008	18	-0,0117
8	-0,0017	19	-0,0164
9	-0,0005	20	0,0069
10	-0,0005	21	0,0189

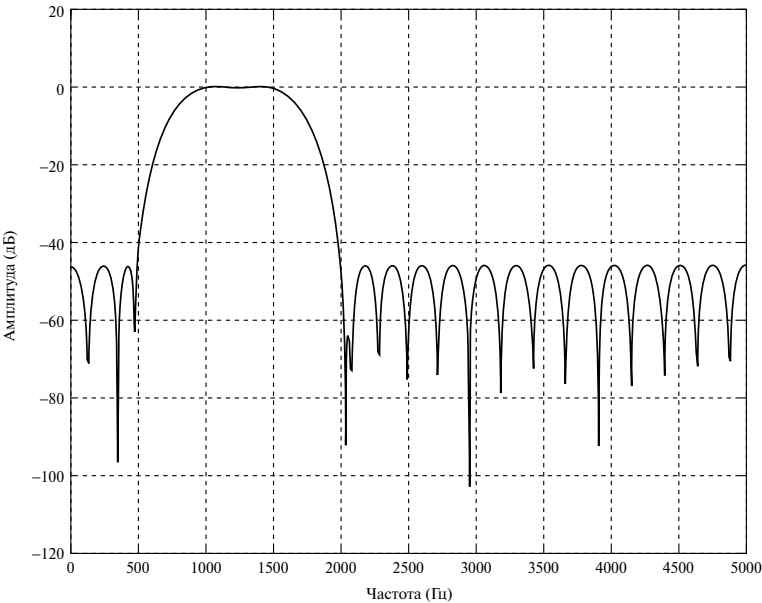


Рис. 7Б.2

Пример 7Б.4

Требуется полосовой фильтр с линейной фазовой характеристикой, удовлетворяющий следующим спецификациям:

полоса пропускания	12–16 Гц,
ширина полосы перехода	2 кГц,
неравномерность в полосе пропускания	1 дБ,
затухание в полосе подавления	45 дБ,
частота дискретизации	50 кГц.

Оцените длину фильтра N и используйте оптимальный метод, чтобы определить его коэффициенты и амплитудно-частотную характеристику. Сравните амплитуды колебаний характеристики в полосе пропускания и подавления с заданными значениями.

Решение

Как и ранее, граничные частоты нужно вначале определить, а затем нормировать на половину частоты Найквиста:

$$\begin{aligned} 10/25 &= 0,4, \\ 12/25 &= 0,48, \\ 16/25 &= 0,64, \\ 18/25 &= 0,72. \end{aligned}$$

Используя эти значения, получаем следующий вектор граничных частот:

$$\begin{aligned} F &= [00, 40, 480, 640, 721] , \\ M &= [001100] . \end{aligned}$$

Для оценки длины фильтра можно использовать команду `remezord`. Для этого неравномерность в полосе пропускания и полосе подавления нужно перевести из децибелов в стандартные линейные единицы:

$$\delta_p = \frac{10^{\frac{A_p}{20}} - 1}{10^{\frac{A_p}{20}} + 1}, \quad \delta_s = 10^{\frac{-A_s}{20}},$$

где A_p и A_s — неравномерность в полосе пропускания и полосе подавления в децибелах.

На основе граничных частот, желаемой амплитудной характеристики, значений амплитуды колебаний и частоты дискретизации оценивается порядок $(N - 1)$, а следовательно, и длина фильтра (см. программу 7Б.4, рис. 7Б.3 и табл. 7Б.4).

Получаем такие оценки параметров фильтра: $N = 40$, весовые коэффициенты 10,22:1:10,22, максимальное отклонение 0,0774. Значение N можно увеличить, чтобы достичь большего затухания в полосе подавления и/или меньшей неравномерности в полосе пропускания (см. программу 7Б.5).

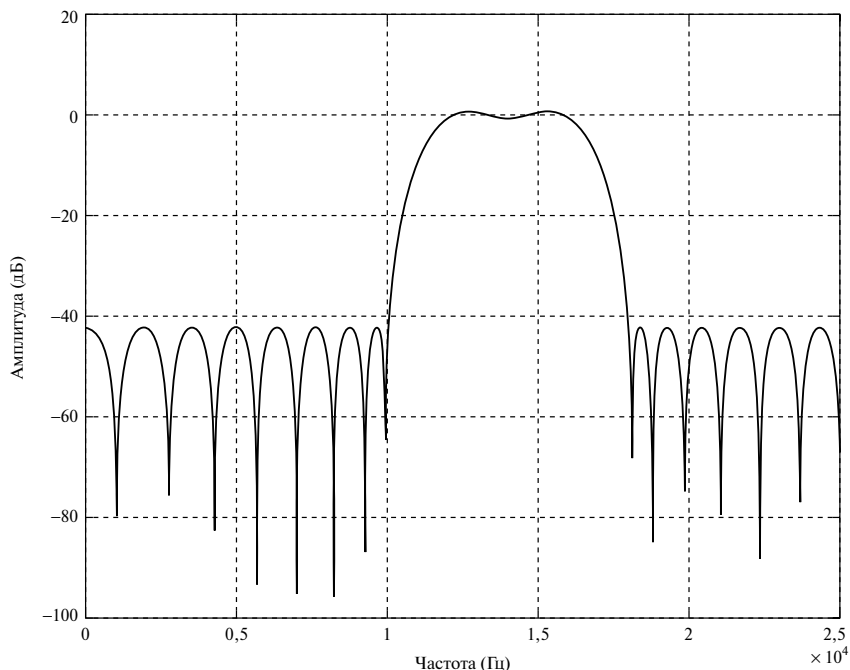


Рис. 7Б.3

Программа 7Б.4. Реализация вычисления коэффициентов оптимального КИХ-фильтра и вывода на экран частотной характеристики (пример 7Б.4) в форме m-файла MATLAB

```
%
%
Fs=50000; % Частота дискретизации
Ap=1; % Неравномерность в полосе пропускания в дБ
As=45; % Затухание в полосе подавления в дБ
M=[0 1 0]; % Желаемая амплитудная характеристика
F=[10000, 12000, 16000, 18000]; % Края полос
dp=(10^(Ap/20)*P1)/(10^(Ap/20)+1); % Неравномерность в полосе пропускания
                                   % и полосе подавления
ds=10^(-As/20); dev=[ds dp ds];
[N1, F0, M0, W] = remezord(F, M, dev, Fs) % Определить порядок фильтра
[b delta] = remez(N1, F0, M0, W); % Вычислить коэффициенты фильтра
[H, f] = freqz(b, 1, 1024, Fs); % Вычислить частотную характеристику
mag = 20*log10(abs(H));          % фильтра и нарисовать ее
plot(f, mag), grid on
xlabel('Частота (Гц)')
ylabel('Амплитуда(дБ)')
```

Таблица 7Б.4. Коэффициенты оптимального КИХ-фильтра из примера 7Б.4

n	$h(n)$	n	$h(n)$
0	0,0005	10	0,0054
1	-0,0017	11	-0,0080
2	-0,0088	12	0,0305
3	0,0139	13	0,0293
4	0,0136	14	-0,0988
5	-0,0273	15	-0,0085
6	-0,0060	16	0,1654
7	0,0363	17	-0,0595
8	-0,0059	18	-0,1854
9	-0,0225	19	0,1411

Программа 7Б.5. Альтернативный m-файл MATLAB для вычисления коэффициентов оптимального КИХ-фильтра и вывода на экран частотной характеристики (пример 7Б.4)

```
%
%
N=44
Fs=50000; % Частота дискретизации
Ap=1; % Неравномерность в полосе пропускания в дБ
As=45; % Затухание в полосе подавления в дБ
M=[0 0 1 1 0 0]; % Желаемая амплитудная характеристика
F=[0, 0.4, 0.48, 0.64, 0.72 1] ; % Края полос
dp=(10^(Ap/20)-1)/(10^(Ap/20)+1);
ds=10^(-As/20); W=[dp/ds, 1, dp/ds];
dev=[ds ds dp dp ds ds];
[b delta] = remez(NP1, F, M, W); % Вычислить коэффициенты фильтра
[N, f] = freqz(b, 1, 1024, Fs); % Вычислить частотную характеристику
mag = 20*log10(abs(N)); % фильтра и нарисовать ее
plot(f, mag), grid on
xlabel('Частота (Гц)')
ylabel('Амплитуда (дБ)')
```

7.Б.3. Метод частотной выборки

Для разработки КИХ-фильтров с произвольными частотными характеристиками, подобных фильтрам, рассмотренным при использовании метода частотной выборки, предназначена команда `fir2`. Синтаксис стандартной команды:

$$b = \text{fir2}(N - 1, F, H)$$

Команда `fir2` вычисляет коэффициенты КИХ-фильтра длины N . Вектор F задает нормированные частоты в интервале от 0 до 1 (причем частоты, как и ранее, нормированы на половину частоты дискретизации). Вектор H определяет желаемую амплитудную характеристику в точках, заданных в F . Оба вектора должны иметь одинаковую длину.

Ниже использование команды `fir2` иллюстрируется на примерах.

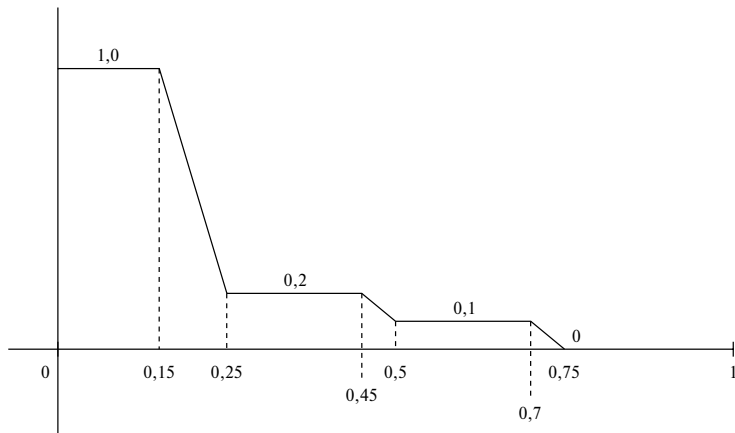


Рис. 7Б.4

Пример 7Б.5

Пример разработки фильтра частотной выборки. КИХ-фильтр частотной выборки с линейной фазовой характеристикой имеет две частотные выборки в полосе перехода. Предполагается, что фильтр имеет 15 отводов и характеризуется следующими частотными выборками:

$$|H(k)| = 1 \quad k = 0, 1, 2, 3$$

$$0,5571 \quad k = 4$$

$$0,0841 \quad k = 5$$

$$0 \quad k = 6, 7$$

Определите коэффициенты фильтра, если частота дискретизации равна 2 кГц.

Решение

Частотные выборки заданы в диапазоне от 0 до половины частоты дискретизации. Следовательно, имеем такие положения точек выборки, нормированные на половину частоты дискретизации: 0, 1/7, 2/7, 3/7, 4/7, 5/7, 6/7, 1.

Программа 7Б.6 представляет собой код MATLAB, в котором частотные выборки используются для определения коэффициентов КИХ-фильтра; амплитудно-частотная характеристика фильтра представлена на рис. 7Б.4, а коэффициенты фильтра — в табл. 7Б.5.

Программа 7Б.6. Реализация вычисления коэффициентов КИХ-фильтра частотной выборки в форме m-файла MATLAB

```
N=15;
fd=[0 1/7 2/7 3/7 4/7 5/7 6/7 1];
Hd=[1 1 1 1 0.5571 0.0841 0 0];
hn=fir2(NP1, fd, Hd);
[H, f] = freqz(hn, 1, 512, Fs);
plot(f, abs(H)), grid on
xlabel('Частота (Гц)')
ylabel('Амплитуда (дБ)')
```

Таблица 7Б.5. Коэффициенты фильтра из примера 7Б.5

n	$h(n)$	n	$h(n)$
0	−0,0001	8	0,2777
1	−0,0006	9	−0,0571
2	0,0017	10	−0,0299
3	0,0128	11	0,0128
4	−0,0299	12	0,0017
5	−0,0571	13	−0,0006
6	0,2777	14	−0,0001
7	0,5910		

Пример 7Б.6

Разработка фильтра с произвольной амплитудной характеристикой. Требуется КИХ-фильтр, аппроксимирующий амплитудно-частотную характеристику, изображенную на рис. 7Б.5.

Определите коэффициенты подходящего КИХ-фильтра и изобразите его амплитудно-частотную характеристику. Частоту дискретизации считайте равной 2 кГц, а длину фильтра — равной 10.

Решение

Желаемая амплитудная характеристика находится на уровне 1 между нормированными частотами 0 и 0,15, на уровне 0,3 между 0,25 и 0,45, на уровне 0,1 между 0,5 и 0,75 и на уровне 0 между 0,85 и 1.

Код MATLAB, реализующий решение задачи, приведен в программе 7Б.7. Для вычисления коэффициентов фильтра используется команда `fir2`, а для определения амплитудной характеристики — команда `freqz`. Коэффициенты не приводятся из соображений экономии места. Амплитудно-частотная характеристика КИХ-фильтра изображена на рис. 7Б.6.

Программа 7Б.7. Вычисление коэффициентов КИХ-фильтра с произвольной амплитудной характеристикой с помощью MATLAB

```

Fs=2000; % Частота дискретизации
N=110; % Длина фильтра
fd=[0 0.15 0.25 0.45 0.5 0.75 0.85 1]; % Точки выборки
Hd=[1 1 0.3 0.3 0.1 0.1 0 0]; % Частотные выборки
hn=fir2(NP1, fd, Hd); % Вычислить импульсную характеристику
[H, f] = freqz(hn, 1, 512, Fs);
plot(f, abs(H)), grid on
xlabel('Частота (Гц)')
ylabel('Амплитуда (дБ)')

```

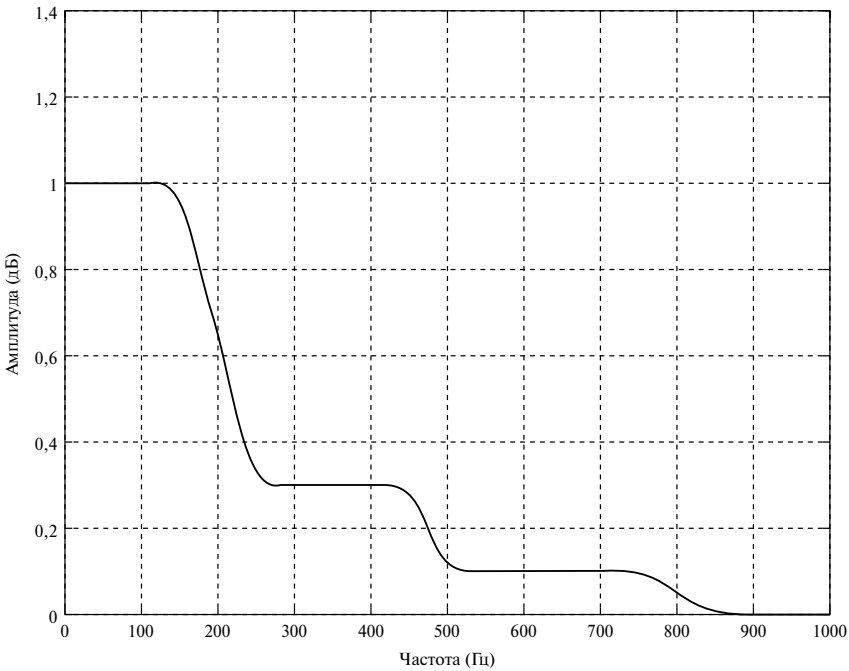


Рис. 7Б.5

Разработка фильтров с бесконечной импульсной характеристикой (БИХ-фильтров)

8.1. Вступление: резюме по основным характеристикам БИХ-фильтров	500
8.2. Этапы разработки цифровых БИХ-фильтров	501
8.3. Спецификация производительности	502
8.4. Методы расчета коэффициентов БИХ-фильтров	504
8.5. Расчет коэффициентов фильтра путем размещения нулей и полюсов	504
8.6. Расчет коэффициентов методом инвариантного преобразования импульсной характеристики	508
8.7. Расчет коэффициентов с помощью согласованного z -преобразования	513
8.8. Расчет коэффициентов с помощью билинейного z -преобразования	517
8.9. Использование для разработки БИХ-фильтров билинейного z -преобразования и классических аналоговых фильтров	527
8.10. Расчет коэффициентов БИХ-фильтра путем отображения полюсов и нулей s -плоскости	544
8.11. Использование программ разработки БИХ-фильтров	553
8.12. Выбор метода расчета коэффициентов БИХ-фильтров	554
8.13. Структуры реализации цифровых БИХ-фильтров	562
8.14. Влияние конечной разрядности на БИХ-фильтры	569
8.15. Реализация БИХ-фильтров	573
8.16. Подробный пример разработки цифрового БИХ-фильтра	574
8.17. Резюме	579
8.18. Примеры использования БИХ-фильтров в цифровых аудиосистемах и измерительных приборах	580
8.19. Примеры применения БИХ-фильтров в связи	582
Задачи	593
Литература	603
Дополнительная литература	603
Приложения	605

В данной главе представлены практические методы разработки цифровых фильтров с бесконечной импульсной характеристикой (БИХ-фильтров), включая популярные методы, которые позволяют конвертировать аналоговые фильтры в эквивалентные цифровые. Приведено простое обобщенное пошаговое руководство по разработке цифровых БИХ-фильтров от спецификации до реализации в программной или аппаратной форме. Для иллюстрации различных аспектов разработки цифровых БИХ-фильтров (в том числе анализа влияния арифметики конечной точности на производительность фильтра и реализацию в реальном времени) разобрано несколько примеров.

Чтобы читатель мог сам вычислить коэффициенты и проанализировать влияние конечной разрядности, приводятся несколько программ на языке С и m-файлов MATLAB. При необходимости читатель может обращаться к главе 6, где рассмотрена общая схема разработки фильтра и приводится сравнительная характеристика БИХ-, КИХ-, цифровых и аналоговых фильтров. В данной главе рассматривается исключительно разработка и применение БИХ-фильтров.

8.1. Вступление: резюме по основным характеристикам БИХ-фильтров

Реальные цифровые БИХ-фильтры характеризуются следующим рекурсивным уравнением:

$$y(n) = \sum_{k=0}^{\infty} h(k)x(n-k) = \sum_{k=0}^N b_k x(n-k) - \sum_{k=1}^M a_k y(n-k), \quad (8.1)$$

где $h(k)$ — импульсная характеристика фильтра, длительность которой теоретически бесконечна, b_k и a_k — коэффициенты фильтра, $x(n)$ и $y(n)$ — вход и выход фильтра. Передаточная функция БИХ-фильтра записывается следующим образом:

$$H(z) = \frac{b_0 + b_1 z^{-1} + \dots + b_N z^{-N}}{1 + a_1 z^{-1} + \dots + a_M z^{-M}} = \frac{\sum_{k=0}^N b_k z^{-k}}{1 + \sum_{k=1}^M a_k z^{-k}}. \quad (8.2)$$

Важной составляющей процесса разработки БИХ-фильтра является поиск таких значений коэффициентов b_k и a_k , чтобы определенные аспекты характеристик фильтра, например, частотная характеристика, вели себя определенным образом. Уравнения БИХ-фильтров приведены в формулах (8.1) и (8.2).

Обратите внимание на то, что в уравнении (8.1) текущая выходная выборка $y(n)$ является функцией прошедших выходов $y(n-k)$, а также текущей и прошедших входных выборок $x(n-k)$, т.е. БИХ-фильтр — это определенная система с обратной связью. Достоинства БИХ-фильтров объясняются именно гибкостью, которую обеспечивает обратная связь. Например, БИХ-фильтр обычно требует меньше коэффициентов, чем КИХ-фильтр при идентичном наборе спецификаций, поэтому БИХ-фильтры используются

тогда, когда важны резкие срезы характеристики. Ценой этого является потенциальная неустойчивость БИХ-фильтра, кроме того, если при разработке не принять надлежащих мер, возможно значительное снижение производительности.

Передаточную функцию БИХ-фильтра $H(z)$, представленную в формуле (8.2), можно факторизовать следующим образом:

$$H(z) = \frac{K(z - z_1)(z - z_2) \dots (z - z_N)}{(z - p_1)(z - p_2) \dots (z - p_M)}, \quad (8.3)$$

где $z_1, z_2, \dots$ — нули $H(z)$, т.е. те значения z , при которых $H(z)$ становится равной нулю, а $p_1, p_2, \dots$ — полюса $H(z)$, т.е. значения z , при которых $H(z)$ бесконечна.

График полюсов и нулей передаточной функции называется *диаграммой нулей и полюсов* и является удобным средством представления и анализа фильтра на комплексной плоскости; подробности см. в главе 3. Чтобы фильтр был устойчивым, все его полюса должны лежать внутри единичной окружности (или совпадать с нулями на единичной окружности). На положения нулей ограничений не существует.

8.2. Этапы разработки цифровых БИХ-фильтров

Разработку БИХ-фильтров можно условно разбить на пять основных этапов.

1. Составление спецификации фильтра, в которой разработчик задает передаточную функцию фильтра (например, указывает, что требуется фильтр нижних частот) и желаемую производительность.
2. Аппроксимация или расчет коэффициентов, когда выбирается один из доступных методов и вычисляются значения коэффициентов b_k и a_k , передаточной функции $H(z)$, которая соответствует спецификациям, предложенным на этапе 1.
3. Выбор подходящей фильтрующей структуры, в которую переводится передаточная функция. Обычно в БИХ-фильтрах используются параллельная структура и/или каскады блоков второго и/или первого порядка.
4. Анализ ошибок, которые могут появиться при представлении коэффициентов фильтра и выполнении арифметических операций, фигурирующих при фильтрации, с помощью конечного числа битов.
5. Реализация, которая включает построение аппаратного обеспечения и/или написание программного кода плюс выполнение собственно фильтрации.

Данные этапы сведены на рис. 8.1. Как видно из рисунка, пять этапов не являются независимыми, и они не всегда выполняются в указанном порядке. Фактически существуют методы, в которых второй, третий и четвертый этапы объединены. В то же время описанный подход гарантирует успешный результат, иногда, правда, для этого требуется несколько итераций, в ходе которых повторяется один или несколько этапов.

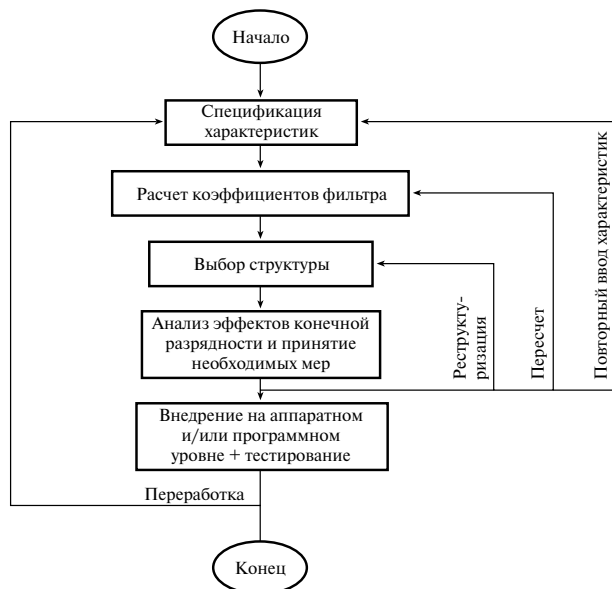


Рис. 8.1. Этапы разработки цифровых фильтров

8.3. Спецификация производительности

Как и большинство других технологических задач, разработка БИХ-фильтров начинается с составления списка требований к производительности. В спецификациях должны указываться 1) характеристики сигнала (тип источников и получателей данных, интерфейс ввода-вывода, скорости передачи данных и длины слов, а также частоты, представляющие практический интерес); 2) частотная характеристика фильтра (желаемые амплитудные и/или частотные характеристики плюс их допуски (если есть), скорость работы); 3) способ реализации (например, как компьютерная программа на языке высокого уровня или система на основе процессора ЦОС, здесь же выбирается процессор обработки сигналов и режим фильтрации (реальное или модельное время)); 4) другие условия разработки (такие как стоимость и разрешенное ухудшение сигнала при прохождении через фильтр). Вообще, большинство из приведенных выше требований зависят от конкретного приложения. Разработчик может не иметь достаточно информации, чтобы изначально полностью определить фильтр, но задать максимально много требований стоит, поскольку это облегчает процесс разработки.

У частотно-избирательных фильтров, таких как фильтры нижних частот и полосовые фильтры, спецификация частотной характеристики часто задается в форме схемы допусков. Пример такой схемы для полосового БИХ-фильтра приведен на рис. 8.2. Заштрихованные горизонтальные полосы обозначают допуски. Для определения частотной характеристики обычно используются следующие параметры:

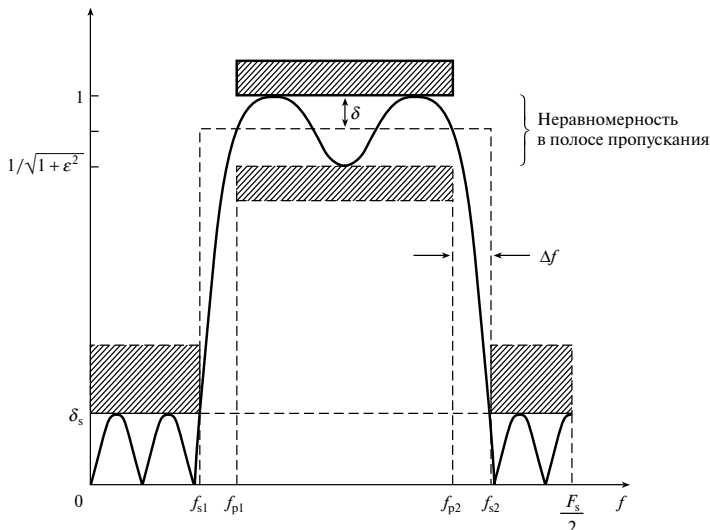


Рис. 8.2. Схема допусков для полосового БИХ-фильтра

- ϵ^2 параметр неравномерности в полосе пропускания;
 δ_p амплитуда отклонений в полосе пропускания;
 δ_s амплитуда отклонений в полосе подавления;
 f_{p1} и f_{p2} граничные частоты полосы пропускания;
 f_{s1} и f_{s2} граничные частоты полосы подавления.

Граничные частоты часто приводятся в нормированной форме, т.е. как доли частоты дискретизации (f/F_s), но далее мы будем задавать их в обычных единицах частоты, герцах или килогерцах, поскольку так более понятно, особенно молодому разработчику. Амплитуды отклонений в полосе пропускания и подавления можно выразить как обычные числа или как величины в децибелах: амплитуда отклонений (неравномерность) в полосе пропускания в децибелах равна

$$A_p = 10 \lg(1 + \epsilon^2) = -20 \lg(1 - \delta_p), \quad (8.4, a)$$

а амплитуда отклонений (затухание) в полосе подавления в децибелах выражается так:

$$A_s = -20 \lg(\delta_s). \quad (8.4, б)$$

Как обсуждалось в главе 6 (см. также рис. 8.2) для БИХ-фильтров неравномерность в полосе пропускания — это разность между минимальным и максимальным отклонением в полосе пропускания. Для КИХ-фильтров неравномерность в полосе пропускания — это разность между идеальной характеристикой и максимальным (или минимальным) отклонением в полосе пропускания. Следовательно, при обсуждении БИХ-фильтров мы будем говорить “неравномерность в полосе пропускания”, подразумевая “удвоенная амплитуда отклонения характеристики в полосе пропускания”.

8.4. Методы расчета коэффициентов БИХ-фильтров

На этом этапе вначале выбирается метод аппроксимации, который затем используется для расчета значений коэффициентов a_k и b_k в уравнении (8.2), при которых спецификации частотной характеристики, полученные на первом этапе разработки, будут удовлетворены.

Для простого получения коэффициентов БИХ-фильтра можно разумно разместить полюса и нули на комплексной плоскости, чтобы получающийся в результате фильтр имел нужную частотную характеристику. Данный подход, известный как *метод размещения нулей и полюсов*, полезен только при разработке простых фильтров, например, узкополосных режекторных фильтров, где параметры фильтра (такие как неравномерность в полосе пропускания) не обязательно задавать точно. Более эффективный подход — вначале разработать аналоговый фильтр, удовлетворяющий желаемой спецификации, а затем преобразовать его в эквивалентный цифровой. Большинство цифровых БИХ-фильтров разрабатываются именно так. Данный подход получил широкое распространение потому, что на настоящий момент в литературе имеется масса информации по аналоговым фильтрам, которую можно использовать при разработке цифровых фильтров. Тремя наиболее распространенными методами конвертации аналоговых фильтров в эквивалентные цифровые являются метод инвариантного преобразования импульсной характеристики, согласованное z -преобразование и билинейное z -преобразование.

В следующих разделах рассмотрены такие методы расчета коэффициентов БИХ-фильтров:

- метод размещения нулей и полюсов;
- метод инвариантного преобразования импульсной характеристики;
- согласованное z -преобразование;
- билинейное z -преобразование.

8.5. Расчет коэффициентов фильтра путем размещения нулей и полюсов

8.5.1. Основные концепции и пример разработки

Если в некоторую точку комплексной плоскости поместить нуль, частотная характеристика в этой точке будет равной нулю. Полюс, с другой стороны, порождает максимум (рис. 8.3). Полюса, расположенные близко к единичной окружности, дают большие пики, тогда как нули, расположенные близко к единичной окружности или лежащие на ней, дают минимумы характеристики. Следовательно, стратегическое размещение полюсов и нулей на комплексной плоскости позволяет получить простой фильтр нижних частот или другой частотно-избирательный фильтр. Более подробно фильтры такого типа описаны в работе [15].

При разработке фильтра стоит помнить один важный момент: чтобы коэффициенты фильтра были действительными, полюса и нули должны либо быть действительны-

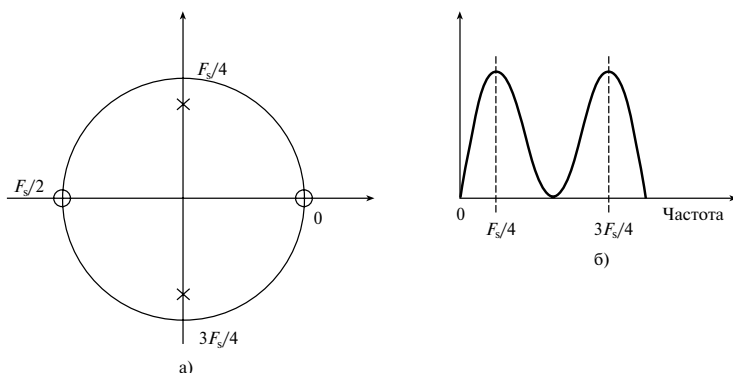


Рис. 8.3. Диаграмма нулей и полюсов простого фильтра (панель а); схематическое изображение частотной характеристики этого фильтра (панель б)

ми, либо образовывать комплексно сопряженные пары. Проиллюстрируем описанный метод на примерах.

Пример 8.1

Иллюстрация расчета коэффициентов фильтра с помощью простого метода нулей и полюсов. Требуется цифровой полосовой фильтр, удовлетворяющий следующим спецификациям:

- 1) полная режекция сигнала на 0 и 250 Гц;
- 2) узкая полоса пропускания, центрированная на 125 Гц;
- 3) ширина полосы пропускания по уровню 3 дБ равна 10 Гц.

Считая частоту дискретизации равной 500 Гц, определите передаточную функцию фильтра, подходящим образом расположив на комплексной плоскости полюса и нули, и запишите разностное уравнение.

Решение

Вначале нужно определить, где на комплексной плоскости поместить полюса и нули. Поскольку полная режекция требуется на 0 и 250 Гц, в соответствующих точках комплексной плоскости следует поместить нули. Эти точки лежат на единичной окружности в местах, соответствующих углам 0° и $360^\circ \times 250/500 = 180^\circ$. Чтобы полоса пропускания была центрирована на 125 Гц, требуется поместить полюс в точках $\pm 360^\circ \times 125/500 = \pm 90^\circ$. Чтобы коэффициенты были действительными, нужна пара комплексно-сопряженных полюсов.

Радиус r полюсов определяется желаемой шириной полосы. Для определения приблизительной ширины полосы (шп) при $r > 0,9$ используется следующее соотношение:

$$r \approx 1 - (\text{шп}/F_s)\pi. \quad (8.5)$$

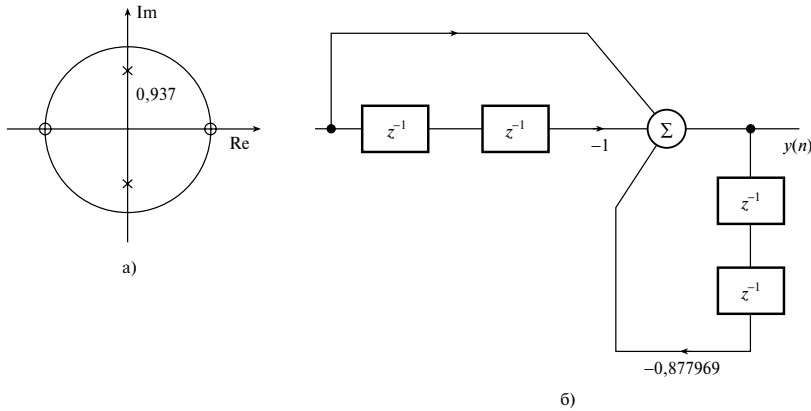


Рис. 8.4. Диаграмма нулей и полюсов (панель а). Блок-схема фильтра (панель б)

В данной задаче $\text{шп} = 10$ Гц и $F_s = 500$ Гц, откуда $r = 1 - (10/500)\pi = 0,937$. Получающаяся диаграмма нулей и полюсов изображена на рис. 8.4, а. С помощью этой диаграммы записываем передаточную функцию:

$$\begin{aligned} H(z) &= \frac{(z-1)(z+1)}{(z-re^{i\pi/2})(z-re^{-i\pi/2})} = \\ &= \frac{z^2-1}{z^2+0,877969} = \frac{1-z^{-2}}{1+0,877969z^{-2}}. \end{aligned}$$

Разностное уравнение:

$$y(n) = -0,877969y(n-2) + x(n) - x(n-2).$$

Сравнивая передаточную функцию $H(z)$ с общим уравнением БИХ-фильтров (8.2), находим, что фильтр представляет собой блок второго порядка со следующими коэффициентами:

$$\begin{aligned} b_0 &= 1 & a_1 &= 0 \\ b_1 &= 0 & a_2 &= 0,877969 \\ b_2 &= -1 \end{aligned}$$

Пример 8.2

Расчет коэффициентов режекторного фильтра путем размещения нулей и полюсов. С помощью метода размещения нулей и полюсов получите передаточную функцию и разностное уравнение простого цифрового режекторного фильтра, удовлетворяющего следующим спецификациям:

частота режекции	50 Гц,
уровень 3 дБ	± 5 Гц,
частота дискретизации	500 Гц.

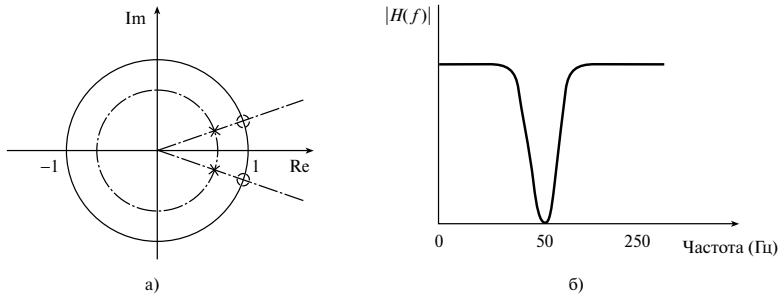


Рис. 8.5. Диаграмма нулей и полюсов фильтра из примера 8.2 (панель а); соответствующая частотная характеристика (панель б)

Решение

- Для режекции компонента на частоте 50 Гц поместим пару комплексных нулей в точках единичной окружности, соответствующих 50 Гц, т.е. в точках с угловыми координатами $360^\circ \times 50/500 = \pm 36^\circ$.
- Чтобы получить узкополосный режекторный фильтр и улучшить амплитудную характеристику по обе стороны от частоты режекции, поместим пару комплексно-сопряженных полюсов на радиусе $r < 1$. Ширина полосы режекции определяется положением полюсов. Используем то же соотношение, что и в примере 8.1, и получим $r = 0,9372$.
- Диаграмма нулей и полюсов приведена на рис. 8.5, а. Из рисунка находим передаточную функцию:

$$\begin{aligned}
 H(z) &= \frac{[z - \exp(-i36^\circ)][z - \exp(i36^\circ)]}{[z - 0,937 \exp(-36^\circ)][z - 0,9372 \exp(36^\circ)]} = \\
 &= \frac{z^2 - 1,6180z + 1}{z^2 - 1,5164z + 0,8783} = \frac{1 - 1,6180z^{-1} + z^{-2}}{1 - 1,5164z^{-1} + 0,8783z^{-2}}.
 \end{aligned}$$

Разностное уравнение:

$$y(n) = x(n) - 1,6180x(n-1) + x(n-2) + 1,5164y(n-1) - 0,8783y(n-2).$$

Сравнивая вид $H(z)$ с формулой (8.2), находим, что режекторный фильтр имеет следующие коэффициенты:

$$\begin{aligned}
 b_0 &= 1 & a_1 &= -1,5164 \\
 b_1 &= -1,6180 & a_2 &= 0,8783 \\
 b_2 &= 1
 \end{aligned}$$

8.6. Расчет коэффициентов методом инвариантного преобразования импульсной характеристики

8.6.1. Основные концепции и пример разработки

В данном методе с помощью преобразования Лапласа из подходящей аналоговой передаточной функции $H(s)$ получают импульсную характеристику $h(t)$. Затем $h(t)$ дискретизируется, а получающаяся функция $h(nT)$ (где T — интервал дискретизации) подвергается z -преобразованию и дает желаемую передаточную функцию $H(z)$. Ниже данный метод иллюстрируется на примерах.

Пример 8.3

Иллюстрация метода инвариантного преобразования импульсной характеристики. С помощью метода инвариантного преобразования импульсной характеристики оцифруйте простой аналоговый фильтр с передаточной функцией

$$H(s) = \frac{C}{s - p}. \quad (8.6)$$

Решение

Импульсная характеристика $h(t)$ находится через обратное преобразование Лапласа:

$$h(t) = L^{-1}[H(s)] = L^{-1}\left(\frac{C}{s - p}\right) = Ce^{pt},$$

где L^{-1} обозначает обратное преобразование Лапласа. Согласно методу инвариантного преобразования импульсной характеристики импульсная характеристика эквивалентного цифрового фильтра $h(nT)$ равна $h(t)$ в дискретные моменты времени $t = nT$, $n = 0, 1, 2, \dots$, т.е.

$$h(nt) = h(t) \Big|_{t=nT} = Ce^{pnT}.$$

Передаточная функция $H(z)$ находится как результат действия z -преобразования на $h(nT)$:

$$\begin{aligned} H(z) &= \sum_{n=0}^{\infty} h(nT)z^{-n} = \sum_{n=0}^{\infty} ce^{pnT}z^{-n} = \\ &= \frac{c}{1 - e^{pT}z^{-1}}. \end{aligned}$$

Значит, используя приведенный результат, можно записать:

$$\frac{C}{s - p} \rightarrow \frac{C}{1 - e^{pT}z^{-1}}. \quad (8.7)$$

Чтобы применить метод инвариантного преобразования импульсной характеристики к БИХ-фильтрам высоких порядков (например, фильтрам M -го порядка) с простыми полюсами, передаточную функцию $H(s)$ вначале нужно разложить на простые дроби (такое разложение описывает цепь фильтров с единственным полюсом):

$$\begin{aligned} H(s) &= \frac{C_1}{s - p_1} + \frac{C_2}{s - p_2} + \dots + \frac{C_M}{s - p_M} = \\ &= \sum_{K=1}^M \frac{C_K}{s - p_K}, \end{aligned} \quad (8.8)$$

где p_K — полюса функции $H(s)$. Каждый член в правой части уравнения (8.8) имеет вид, как в формуле (8.6), так что преобразование (8.8) правомерно. Следовательно,

$$\sum_{K=1}^M \frac{C_K}{s - p_K} \rightarrow \sum_{K=1}^M \frac{C_K}{1 - e^{p_K T} z^{-1}}. \quad (8.9)$$

БИХ-фильтры высоких порядков обычно реализуются как каскад или параллельная структура из стандартных фильтрующих блоков второго порядка. Следовательно, особый интерес представляет вариант $M = 2$. Здесь преобразование (8.9) имеет вид

$$\begin{aligned} \frac{C_1}{s - p_1} + \frac{C_2}{s - p_2} &\rightarrow \frac{C_1}{1 - e^{p_1 T} z^{-1}} + \frac{C_2}{1 - e^{p_2 T} z^{-1}} = \\ &= \frac{C_1 + C_2 - (C_1 e^{p_2 T} + C_2 e^{p_1 T}) z^{-1}}{1 - (e^{p_1 T} + e^{p_2 T}) z^{-1} + e^{(p_1 + p_2) T} z^{-2}}. \end{aligned} \quad (8.10)$$

Если полюса p_1 и p_2 — комплексно-сопряженные, то C_1 и C_2 также будут комплексно-сопряженными, и уравнение (8.10) сводится к такому виду:

$$\begin{aligned} \frac{C_1}{1 - e^{p_1 T} z^{-1}} + \frac{C_1^*}{1 - e^{p_1^* T} z^{-1}} &= \\ &= \frac{2C_{re} - [C_{re} \cos(p_{im} T) + C_{im} \sin(p_{im} T)] 2e^{p_{re} T} z^{-1}}{1 - 2e^{p_{re} T} \cos(p_{im} T) z^{-1} + e^{2p_{re} T} z^{-2}}, \end{aligned} \quad (8.11)$$

где C_{re} и C_{im} — действительная и мнимая части C_1 , p_{re} и p_{im} — действительная и мнимая части p_1 , а $*$ обозначает “комплексно-сопряженное”.

Для большинства практических БИХ-фильтров, которые можно реализовать по схеме инвариантного преобразования импульсной характеристики, для получения коэффициентов передаточной функции достаточно преобразований (8.7), (8.10) и (8.11). В приложении к данной главе приводится программа на языке С для вычисления коэффициентов фильтров по описанной схеме, а показанный ниже пример поможет проиллюстрировать данную концепцию.

Пример 8.4

Применение метода инвариантного преобразования импульсной характеристики в разработке фильтров. Требуется разработать цифровой фильтр, аппроксимирующий следующую нормированную аналоговую передаточную функцию:

$$H(s) = \frac{1}{s^2 + \sqrt{2}s + 1}.$$

Используя метод инвариантного преобразования импульсной характеристики, получите передаточную функцию $H(z)$ цифрового фильтра, предполагая частоту среза по уровню 3 дБ равной 150 Гц и частоту дискретизации 1,28 кГц.

Решение

Перед тем как применять метод инвариантного преобразования импульсной характеристики, следует масштабировать нормированную передаточную функцию. Для этого s меняется на s/α , где $\alpha = 2\pi \times 150 = 942,4778$, в результате чего фильтр получит желаемую характеристику. Итак,

$$H'(s) = H(s) \Big|_{s=s/\alpha} = \frac{\alpha^2}{s^2 + \sqrt{2}\alpha s + \alpha^2} = \frac{C_1}{s - p_1} + \frac{C_2}{s - p_2},$$

где

$$p_1 = \frac{-\sqrt{2}\alpha(1-i)}{2} = -666,4324(1-i), p_2 = p_1^*;$$

$$C_1 = -\frac{\alpha}{\sqrt{2}}i = -666,4324i; C_2 = C_1^*.$$

Поскольку полюса комплексно-сопряженные, для получения передаточной функции дискретного времени $H(z)$ используется преобразование (8.11). В данной задаче $C_{re} = 0$, $C_{im} = -666,4324$, $p_{im}T = 0,5207$, $p_{re}T = -0,5207$, $e^{p_{re}T} = 0,5941$, $\sin(p_{im}T) = 0,4974$, $\cos(p_{im}T) = 0,8675$ и $e^{p_{re}T} = 0,3530$. Подставляя эти значения в уравнение (8.11), получаем $H(z)$:

$$H(z) = \frac{393,9264z^{-1}}{1 - 1,0308z^{-1} + 0,3530z^{-2}}.$$

Если в приведенное уравнение подставить $z = e^{i\omega T}$, значение $H(z)$ на частоте $\omega = 0$ будет равно 1223, что примерно равно частоте дискретизации. Вообще такое большое усиление характерно для фильтров, реализованных по схеме инвариантного преобразования импульсной характеристики, и усиление передаточной функции примерно равно частоте дискретизации, т.е. $1/T$ (следствие дискретизации частотной характеристики). Чтобы уменьшить усиление и избежать переполнения при реализации фильтра, $H(z)$ часто множится на T (или делится на частоту дискретизации). При таком подходе в данной задаче получим следующую передаточную функцию:

$$H(z) = \frac{0,3078z^{-1}}{1 - 1,0308z^{-1} + 0,3530z^{-2}}.$$

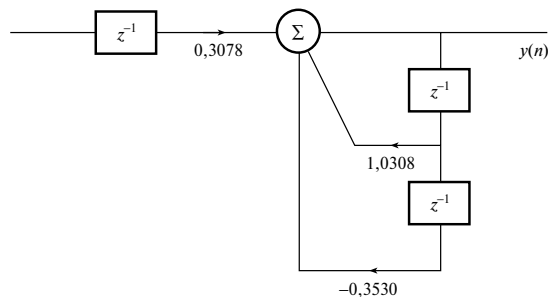


Рис. 8.6. Блок-схема фильтра из примера 8.4

Таким образом,

$$\begin{aligned} b_1 &= 0 & a_1 &= -1,0308 \\ b_2 &= 0,3078 & a_2 &= 0,3530 \end{aligned}$$

Альтернативный метод устранения влияния частоты дискретизации на усиление фильтра — это работа с нормированными частотами. Тогда в последнем примере можно использовать $T = 1$ и $\alpha = 2\pi \times 150/1280 = 0,7363$. Подставляя эти значения в уравнение (8.11), сразу получаем указанную выше желаемую передаточную функцию. Важным преимуществом работы с нормированными частотами является использование значительно более простых чисел. Это также означает, что результаты можно обобщить; соответствующая блок-схема фильтра приведена на рис. 8.6.

8.6.2. Метод инвариантного преобразования импульсной характеристики: резюме

1. Определить нормированную характеристику аналогового фильтра $H(s)$, удовлетворяющую спецификациям желаемого частотного фильтра.
2. При необходимости разложить $H(s)$ на элементарные дроби, чтобы упростить следующий этап.
3. Применить z -преобразование к каждой дроби и получить выражение в форме (8.9).
4. Получить $H(z)$, сгруппировав результаты п. 3 в члены второго порядка и, возможно, один член первого порядка. Если используется реальная частота дискретизации, $H(z)$ нужно затем умножить на T .

8.6.3. Метод инвариантного преобразования импульсной характеристики: замечания

1. Импульсная характеристика дискретного фильтра $h(nT)$ идентична характеристике аналогового фильтра $h(t)$ в дискретные моменты времени $t = nT$, $n = 0, 1, \dots$; см., например, рис. 8.7. Именно по этой причине метод называется *инвариантным преобразованием импульсной характеристики*.

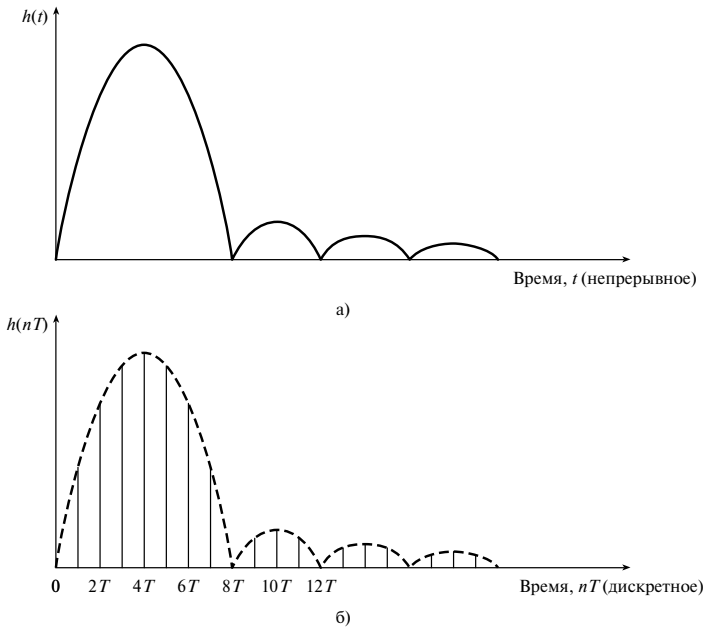


Рис. 8.7. Сравнение импульсной характеристики а) аналогового фильтра $h(t)$ и б) эквивалентного цифрового фильтра $h(nT)$. В методе инвариантного преобразования импульсной характеристики две характеристики идентичны в моменты взятия выборок

2. На частотную характеристику дискретного фильтра, построенного по схеме инвариантного преобразования импульсной характеристики, влияет частота дискретизации. Чтобы частотная характеристика была ближе к характеристике эквивалентного аналогового фильтра, необходима достаточно большая частота дискретизации.
3. Как и для систем дискретизованных данных, спектр фильтра с инвариантным преобразованием импульсной характеристики, соответствующий $H(z)$, аналогичен спектру исходного аналогового фильтра $H(s)$, но повторяется с периодом, равным частоте дискретизации, как показано на рис. 8.8, вызывая таким образом наложение спектров. В то же время, если характеристика исходного аналогового фильтра имеет достаточно резкие срезы, или если ширина полосы характеристики аналогового фильтра была ограничена перед применением метода инвариантного преобразования импульсной характеристики, наложение будет незначительным. Этого же эффекта можно добиться, увеличив частоту дискретизации. Делаем вывод, что метод стоит использовать для фильтров нижних частот с очень резкими срезами и незначительным наложением, поскольку при таких условиях частота дискретизации разумно велика. Данный метод не подходит для фильтров верхних частот или режекторных фильтров, если дополнительно не используется фильтр защиты от наложения спектров.

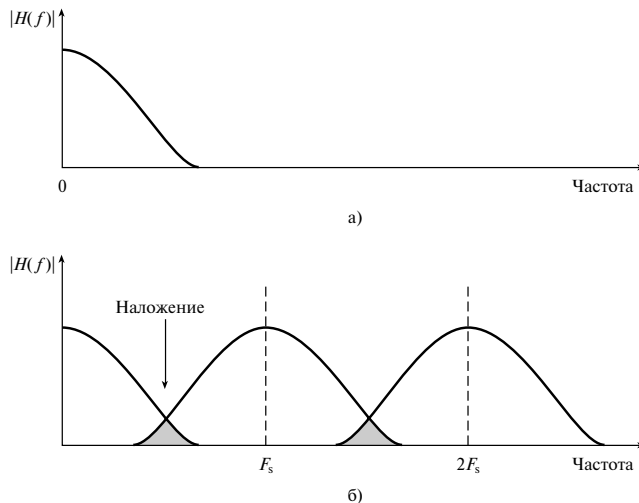


Рис. 8.8. Спектр аналогового фильтра (панель *а*); спектр эквивалентного цифрового фильтра, полученного методом инвариантного преобразования импульсной характеристики, на котором виден эффект наложения (панель *б*)

8.7. Расчет коэффициентов с помощью согласованного z -преобразования

8.7.1. Основные концепции и примеры разработки

Согласованное z -преобразование позволяет преобразовать аналоговый фильтр в эквивалентный цифровой. В данном методе каждый полюс и нуль аналогового фильтра непосредственно переводятся с s - на z -плоскость (комплексную плоскость):

$$(s - a) \rightarrow (1 - z^{-1}e^{aT}), \quad (8.12)$$

где T — период дискретизации. Преобразование (8.12) отображает полюс (или нуль), находящийся в точке $s = a$ s -плоскости, в полюс (или нуль) комплексной плоскости, находящийся в точке $z = e^{aT}$.

Для аналоговых фильтров высоких порядков передаточная функция имеет несколько полюсов и/или нулей, которые нужно отобразить с s - на z -плоскость. Для аналогового фильтра наивысшего порядка с различными полюсами и нулями передаточную функцию можно записать в следующем виде:

$$H(s) = \frac{(s - z_1)(s - z_2) \dots (s - z_M)}{(s - p_1)(s - p_2) \dots (s - p_N)}, \quad (8.13)$$

где z_k и p_k — нули и полюса $H(s)$ соответственно.

Далее на каждый множитель действуем согласованным z -преобразованием:

$$\begin{aligned}(s - z_k) &\rightarrow (1 - z^{-1}e^{z_k T}), \\ (s - p_k) &\rightarrow (1 - z^{-1}e^{p_k T}).\end{aligned}$$

В БИХ-фильтрах высокого порядка основной составляющей является фильтрующий блок второго порядка. Следовательно, особый интерес представляет случай, когда в уравнении (8.13) $M = N = 2$. При этом аналоговая передаточная функция сводится к виду

$$H(s) = \frac{(s - z_1)(s - z_2)}{(s - p_1)(s - p_2)}. \quad (8.14)$$

Применяя к этой функции согласованное z -преобразование, получаем

$$\frac{(s - z_1)(s - z_2)}{(s - p_1)(s - p_2)} \rightarrow \frac{1 - (e^{z_1 T} + e^{z_2 T})z^{-1} + e^{(z_1 + z_2)T}z^{-2}}{1 - (e^{p_1 T} + e^{p_2 T})z^{-1} + e^{(p_1 + p_2)T}z^{-2}}. \quad (8.15)$$

Если полюса и нули звена второго порядка формируют комплексно-сопряженные пары, тогда $p_2 = p_1^*$ и $z_2 = z_1^*$ и правая часть уравнения (8.15) сводится к виду

$$\frac{1 - 2e^{z_{re}T} \cos(z_{im}T)z^{-1} + e^{z_{re}T}z^{-2}}{1 - 2e^{p_{re}T} \cos(p_{im}T)z^{-1} + e^{p_{re}T}z^{-2}}, \quad (8.16)$$

где z_{re} и z_{im} , p_{re} и p_{im} — действительная и мнимая части z_1 и p_1 соответственно.

На практике аналоговые фильтрующие блоки второго порядка удобнее представить в знакомой форме рациональной дроби:

$$H(s) = \frac{(s - z_1)(s - z_2)}{(s - p_1)(s - p_2)} = \frac{A_0 + A_1s + A_2s^2}{B_0 + B_1s + B_2s^2}.$$

В такой форме полюса и нули $H(s)$ определяются следующими выражениями:

$$p_{1,2} = -\frac{B_1}{2B_2} \pm \left[\left(\frac{B_1}{2B_2} \right)^2 - \frac{B_0}{B_2} \right]^{\frac{1}{2}}, \quad (8.17, a)$$

$$z_{1,2} = -\frac{A_1}{2A_2} \pm \left[\left(\frac{A_1}{2A_2} \right)^2 - \frac{A_0}{A_2} \right]^{\frac{1}{2}}. \quad (8.17, б)$$

На практике формулы (8.17, а) и (8.17, б) позволяют по данной передаточной функции аналогового фильтра сразу определить, где находятся полюса и нули (а следовательно, их действительную и мнимую части). Определив действительную и мнимую части нулей и полюсов $H(s)$, с помощью уравнения (8.15) или (8.16) можно вычислить передаточную функцию $H(z)$ эквивалентного дискретного фильтра.

Пример 8.5

Дана нормированная передаточная функция аналогового фильтра:

$$H(s) = \frac{1}{s^2 + \sqrt{2}s + 1}.$$

Получите передаточную функцию $H(z)$ эквивалентного цифрового фильтра, используя метод согласованного z -преобразования. Частота среза по уровню 3 дБ равна 150 Гц, частота дискретизации — 1,28 кГц.

Решение

Частоту среза можно выразить как $\omega_c = 2\pi \times 150 = 942,4778$ рад/с. Передаточная функция денормированного аналогового фильтра получается заменой s на s/ω_c :

$$\begin{aligned} H'(s) &= H(s) \Big|_{s=\frac{s}{\omega_c}} = \\ &= \frac{\omega_c^2}{s^2 + \sqrt{2}\omega_c s + \omega_c^2}. \end{aligned}$$

Полюса фильтра расположены в точках

$$\begin{aligned} p_{1,2} &= -\frac{\sqrt{2}\omega_c}{2} \pm \left[\left(\frac{\sqrt{2}\omega_c}{2} \right)^2 - \omega_c^2 \right]^{\frac{1}{2}} = \\ &= -\frac{\sqrt{2}\omega_c}{2} \pm \omega_c \left[\left(\frac{\sqrt{2}}{2} \right)^2 - 1 \right]^{\frac{1}{2}} = \\ &= -\frac{\sqrt{2}\omega_c}{2} (1 \mp i). \end{aligned}$$

В данной задаче действительная и мнимая части полюсов равны

$$p_{re} = -\frac{\sqrt{2}\omega_c}{2} = -666,4324, \quad p_{im} = \frac{\sqrt{2}\omega_c}{2}i = 666,4324i.$$

Следовательно, $p_{re}T = -0,5206503$, $p_{im}T = 0,5206503$, $\cos(p_{im}T) = 0,867496$ и $e^{p_{re}T} = 0,594134$. В результате получаем такую передаточную функцию:

$$H(z) = \frac{8,8876 \times 10^5}{1 - 1,030818z^{-1} + 0,594134z^{-2}}.$$

8.7.2. Метод согласованного z -преобразования: резюме

1. Определить подходящую аналоговую передаточную функцию $H(s)$, удовлетворяющую спецификациям искомого цифрового фильтра.
2. Найти положение полюсов и нулей $H(s)$. При этом может потребоваться факторизация аналоговой передаточной функции $H(s)$.
3. Отобразить полюса и нули с s - на z -плоскость, используя формулу (8.12). Для блоков второго порядка можно использовать формулы (8.15) и (8.16).
4. Объединить уравнения, записанные на z -плоскости, для получения передаточной функции $H(z)$.

8.7.3. Метод согласованного z -преобразования: замечания

1. Метод согласованного z -преобразования требует знания положений полюсов и нулей аналогового фильтра. Для получения этой информации можно факторизовать аналоговую передаточную функцию $H(s)$. После этого применить преобразование относительно просто.
2. Метод согласованного z -преобразования и метод инвариантного преобразования импульсной характеристики дают дискретные фильтры с идентичными знаменателями. Сравните, например, знаменатели согласованного z -преобразования (8.15) и результат преобразования по схеме инвариантного преобразования импульсной характеристики (8.10). Это также очевидно, если сравнить знаменатели передаточных функций, полученных в примерах 8.4 и 8.5.
3. В цифровых фильтрах полезная полоса частот лежит в диапазоне от нуля до частоты Найквиста (половина частоты дискретизации), тогда как в аналоговых фильтрах фигурируют частоты в диапазоне от нуля до бесконечности. Следовательно, отображение, которое производит согласованное z -преобразование, сжимает бесконечную аналоговую полосу частот в конечную. Это приводит к искажению частотных характеристик эквивалентных цифровых фильтров по сравнению с аналоговыми фильтрами. Для фильтров, получающихся при использовании согласованного z -преобразования, характерно меньшее затухание по сравнению с аналоговыми фильтрами. К данному вопросу мы еще вернемся в разделе 8.12, где будет показано, как можно использовать данную особенность.
4. Если аналоговый фильтр имеет полюса на частотах, близких к частоте Найквиста, или нули на частотах, превышающих ее, частотная характеристика получающегося цифрового фильтра будет искажена в результате наложения (см. далее). В таких случаях характеристика аналогового фильтра, которая приходится на диапазон за частотой Найквиста, существенна. Чтобы отобразить эту часть характеристики в желаемую полосу частот, используется процесс неявной дискретизации.
5. Согласованное z -преобразование также не подходит для оцифровки фильтров с одними полюсами, поскольку они не имеют нулей за частотой Найквиста. Несколько улучшить ситуацию в этом случае можно, добавив нули в точке $z = -1$ (т.е. на частоте Найквиста).

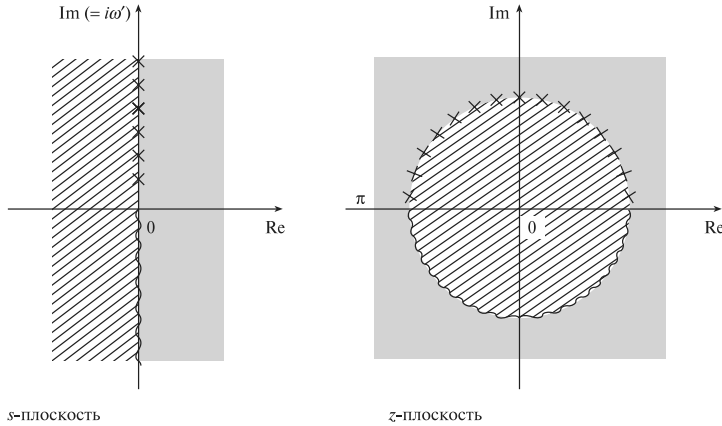


Рис. 8.9. Иллюстрация отображения с s -плоскости на комплексную (z) плоскость с использованием билинейного z -преобразования. Обратите внимание на то, что положительная часть оси $i\omega'$ на s -плоскости (т.е. точки от $s = 0$ до $s = i\infty$) отображается в верхнюю половину единичной окружности, а отрицательная часть оси $i\omega'$ переводится в нижнюю половину

Тема влияния согласованного z -преобразования на частотную характеристику фильтра будет продолжена в разделе 8.12.

8.8. Расчет коэффициентов с помощью билинейного z -преобразования

8.8.1. Основные концепции и примеры разработки

Данный метод является, пожалуй, самым важным методом получения коэффициентов БИХ-фильтра. В нем для преобразования характеристики аналогового фильтра $H(s)$ в характеристику эквивалентного цифрового фильтра применяется следующая замена:

$$s = k \frac{z - 1}{z + 1}, \quad k = 1 \text{ или } \frac{2}{T}. \quad (8.18, a)$$

Приведенное выше преобразование отображает аналоговую передаточную функцию $H(s)$, записанную на s -плоскости, в дискретную передаточную функцию $H(z)$ комплексной плоскости, как показано на рис. 8.9. Обратите внимание на то, что на рисунке вся ось $i\omega$ s -плоскости отображается в единичную окружность, левая половина s -плоскости отображается внутрь единичной окружности, а правая — в область снаружи единичной окружности. Таким образом, устойчивый аналоговый фильтр с полюсами в левой половине s -плоскости перейдет в цифровой фильтр с полюсами внутри единичной окружности.

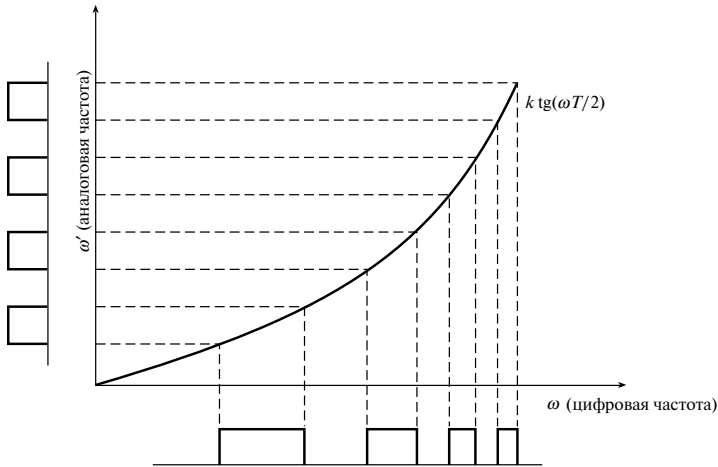


Рис. 8.10. Связь между аналоговыми и цифровыми частотами, демонстрирующая эффект деформации. Обратите внимание на то, что равноотстоящие аналоговые полосы после преобразования в цифровую область на высоких частотах сжимаются и располагаются плотнее

К сожалению, прямая замена s в $H(s)$, как она записана в формуле (8.18, а) может привести к получению цифрового фильтра с нежелательной характеристикой. Это легко показать, сделав в уравнении (8.18, а) замену $z = e^{i\omega T}$ и $s = i\omega'$. Упрощая, находим, что аналоговая частота ω' и цифровая частота ω связаны соотношением

$$\omega' = k \operatorname{tg} \left(\frac{\omega T}{2} \right), \quad k = 1 \text{ или } \frac{2}{T}. \quad (8.18, б)$$

Зависимость (8.18, б) схематически изображена на рис. 8.10. Видно, что связь аналоговой частоты ω' с цифровой частотой ω почти линейна при малых значениях ω , но становится нелинейной при больших значениях ω , что приводит к искажению (или деформации) цифровой частотной характеристики. Обратите внимание на то, что полосы пропускания аналогового фильтра слева имеют постоянную ширину и их центры располагаются через равные промежутки, тогда как полосы пропускания цифрового эквивалента несколько сгущены. Для компенсации этого эффекта аналоговый фильтр (одна или несколько критичных частот) обычно предварительно деформируется перед применением билинейного преобразования. Например, при разработке фильтра нижних частот предварительной деформации часто подвергаются частота среза или граничная частота:

$$\omega'_p = k \operatorname{tg} \left(\frac{\omega_p T}{2} \right), \quad (8.19)$$

где

ω_p = заданная частота среза,

ω'_p = предварительно деформированная частота среза,

$$k = 1 \text{ или } \frac{T}{2},$$

T — период дискретизации.

8.8.2. Метод билинейного z -преобразования: резюме

Для стандартных частотно-избирательных БИХ-фильтров можно следующим образом обобщить этапы использования билинейного z -преобразования.

1. На основе спецификаций цифрового фильтра определить подходящий нормированный аналоговый фильтр-прототип с передаточной функцией $H(s)$.
2. Определить и деформировать граничные или критичные частоты нужного фильтра. Для фильтров нижних или верхних частот существует единственная граничная частота, или частота среза (скажем, ω_p). Для полосовых и режекторных фильтров имеем верхнюю и нижнюю граничные частоты полосы пропускания ω_{p1} и ω_{p2} , каждую из которых нужно деформировать (также могут задаваться граничные частоты полосы подавления):

$$\omega'_p = \operatorname{tg} \left(\frac{\omega_p T}{2} \right); \quad (8.20, a)$$

$$\omega'_{p1} = \operatorname{tg} \left(\frac{\omega_{p1} T}{2} \right); \quad \omega'_{p2} = \operatorname{tg} \left(\frac{\omega_{p2} T}{2} \right). \quad (8.20, б)$$

3. Денормировать аналоговый фильтр-прототип, заменив s в передаточной функции с помощью одного из следующих преобразований (в зависимости от типа требуемого фильтра):

$$s = \frac{s}{\omega'_p} \quad \text{нижних частот в нижних частот,} \quad (8.21, a)$$

$$s = \frac{\omega'_p}{s} \quad \text{нижних частот в верхних частот,} \quad (8.21, б)$$

$$s = \frac{s^2 + \omega_0^2}{Ws} \quad \text{нижних частот в полосовой,} \quad (8.21, в)$$

$$s = \frac{Ws}{s^2 + \omega_0^2} \quad \text{нижних частот в режекторный.} \quad (8.21, г)$$

Здесь

$$\omega_0^2 = \omega'_{p2} \omega'_{p1}, \quad W = \omega'_{p2} - \omega'_{p1}.$$

4. Применить билинейное z -преобразование и получить передаточную функцию нужного цифрового фильтра $H(z)$, следующим образом заменив s в масштабированной (т.е. денормированной) передаточной функции $H'(s)$:

$$s = \frac{z - 1}{z + 1}.$$

Пример 8.6

Фильтр нижних частот. Требуется разработать цифровой фильтр нижних частот, аппроксимирующий следующую передаточную функцию:

$$H(s) = \frac{1}{s^2 + \sqrt{2}s + 1}.$$

Используя метод билинейного z -преобразования, получите передаточную функцию $H(z)$ цифрового фильтра, если частота среза по уровню 3 дБ равна 150 Гц, а частота дискретизации равна 1,28 кГц.

Решение

Используя критическую частоту $\omega_p = 2\pi \times 150$ рад/с и $F_s = 1/T = 1,28$ кГц, получаем следующую деформированную критическую частоту:

$$\omega'_p = \operatorname{tg}(\omega_p T/2) = 0,3857.$$

Масштабированный аналоговый фильтр характеризуется функцией

$$\begin{aligned} H'(s) = H(s) \Big|_{s=s/\omega'_p} &= \frac{1}{(s/\omega'_p)^2 + \sqrt{2}s/\omega'_p + 1} = \\ &= \frac{\omega'^2_p}{s^2 + \sqrt{2}\omega'_p s + \omega'^2_p} = \frac{0,1488}{s^2 + 0,5455s + 0,1488}. \end{aligned}$$

После применения билинейного z -преобразования получаем такой результат:

$$\begin{aligned} H(z) = H'(s) \Big|_{s=\frac{z-1}{z+1}} &= \frac{0,0878z^2 + 0,1756z + 0,0878}{z^2 - 1,0048z + 0,3561} = \\ &= \frac{0,0878(1 - 2z^{-1} + z^{-2})}{1 - 1,0048z^{-1} + 0,3561z^{-2}}. \end{aligned}$$

Пример 8.7

Фильтр верхних частот. Нормированная передаточная функция простого аналогового RC-фильтра верхних частот записывается как

$$H(s) = \frac{1}{s + 1}.$$

Начав с уравнения на s -плоскости, определите с помощью метода билинейного z -преобразования передаточную функцию эквивалентного фильтра верхних частот, работающего в дискретном времени. Частоту дискретизации считайте равной 150 Гц, а частоту среза — 30 Гц.

Решение

Критичной частотой для данного цифрового фильтра является $\omega_p = 2\pi \times 30$ рад/с. Частота среза после деформации равна $\omega'_p = \text{tg}(\omega_p T/2)$. При $T = 1/150$ Гц $\omega'_p = \text{tg}(\pi/5) = 0,7265$.

Используя преобразование “фильтр нижних частот в фильтр верхних частот”, представленное в формуле (8.21, а), получаем следующую денормированную аналоговую передаточную функцию:

$$H'(s) = H(s) \Big|_{s=\omega'_p/s} = \frac{1}{\omega'_p/s + 1} = \frac{2}{s + 0,7265}.$$

Для получения передаточной функции на комплексной плоскости применим билинейное z -преобразование:

$$H(z) = H'(s) \Big|_{s=(z-1)/(z+1)} = \frac{(z-1)/(z+1)}{(z-1)/(z+1) + 0,7265}.$$

Упрощая, получаем

$$H(z) = 0,5792 \frac{1 - z^{-1}}{1 + 0,1584z^{-1}}.$$

Коэффициенты фильтра дискретного времени:

$$\begin{aligned} b_0 &= 0,5792, & a_1 &= 0,1584, \\ b_1 &= -0,5792. \end{aligned}$$

Пример 8.8

Полосовой фильтр. Требуется полосовой фильтр дискретного времени с характеристиками фильтра Баттерворта, удовлетворяющий приведенным ниже спецификациям. Для получения коэффициентов фильтра используйте метод билинейного z -преобразования.

полоса пропускания 200–300 Гц,
 частота дискретизации 2 кГц,
 порядок фильтра, N 2.

Решение

Требуется нормированный аналоговый фильтр нижних частот первого порядка (поскольку преобразование полосы частот для полосовых фильтров, выраженное в формуле (8.21, в), удвоит порядок фильтра). Следовательно,

$$H(s) = \frac{1}{s+1}.$$

Критичные частоты после предварительной деформации:

$$\omega'_{p1} = \operatorname{tg} \left(\frac{\omega_{p1}}{2} \right) = \operatorname{tg} \left(\frac{2\pi \times 200}{2 \times 200} \right) = 0,3249,$$

$$\omega'_{p2} = \operatorname{tg} \left(\frac{\omega_{p2}}{2} \right) = \operatorname{tg} \left(\frac{2\pi \times 300}{2 \times 200} \right) = 0,5095,$$

$$\omega^2 = \omega'_{p1} \omega'_{p2} = 0,1655,$$

$$W = \omega'_{p2} - \omega'_{p1} = 0,1846.$$

Используя преобразование “фильтр нижних частот в полосовой фильтр” (формулу (8.21, в)), получаем

$$\begin{aligned} H'(s) &= H(s) \Big|_{s=\frac{s^2+\omega_0^2}{Ws}} = \frac{1}{\frac{s^2+\omega_0^2}{Ws} + 1} = \\ &= \frac{Ws}{s^2 + Ws + \omega_0^2}. \end{aligned}$$

После применения билинейного z -преобразования получаем

$$H(z) = H'(s) \Big|_{s=\frac{z-1}{z+1}} = \frac{W \left(\frac{z-1}{z+1} \right)}{\left(\frac{z-1}{z+1} \right)^2 + W \left(\frac{z-1}{z+1} \right) + \omega_0^2}.$$

Подставляя значения ω_0^2 и W и упрощая, получаем

$$H(z) = 0,1367 \frac{1 - z^{-2}}{1 - 1,2362z^{-1} + 0,7265z^{-2}}.$$

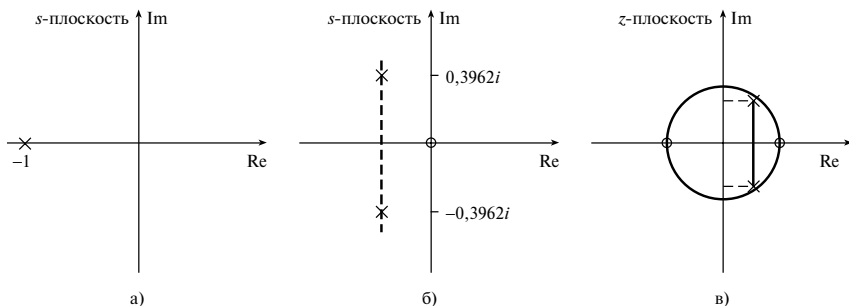


Рис. 8.11. Диаграмма нулей и полюсов для а) фильтра-прототипа нижних частот, б) промежуточного аналогового полосового фильтра и в) дискретного полосового фильтра, полученного путем преобразования полосы

Диаграмма нулей и полюсов нормированного фильтра-прототипа нижних частот, аналогового полосового фильтра и цифрового полосового фильтра дискретного времени изображены на рис. 8.11. Отметим, что преобразование “фильтр нижних частот в полосовой фильтр” ввело по одному нулю в начале координат s -плоскости и на бесконечности. Затем билинейное z -преобразование отобразило эти нули в $z = \pm 1$. Таким образом, нули дискретного полосового фильтра — $z = 1$ и $z = -1$. Его полюса расположены в точках $z = 0,6040 \pm j0,6015i$. Нули аналогового полосового фильтра находятся в точке $s = 0$ и на бесконечности (на рисунке не показано), а полюса — в точках $s = -j0,0923 \pm j0,3962i$.

На практике БИХ-фильтры высоких порядков (например, $N > 3$) обычно реализуются как каскады или параллельные соединения фильтрующих звеньев второго и/или первого порядков, что позволяет снизить влияние конечной разрядности на производительность фильтра (см. далее). Следовательно, после преобразования аналогового фильтра в дискретную форму получающуюся передаточную функцию $H(z)$, если она имеет большой порядок, нужно выразить в факторизованном виде (для каскадной реализации) или как сумму членов второго и/или первого порядка (для параллельной реализации). Для упрощения данной задачи $H(s)$ изначально можно выразить в факторизованном виде, а затем преобразовать каждый множитель отдельно. После этого все полученные множители можно объединить или перегруппировать в формат, подходящий для нужной реализации. По сути, такой подход использован в примере 8.15.

8.8.3. Метод билинейного z -преобразования: замечания

По сути, метод билинейного преобразования включает два отдельных преобразования. Во-первых, нормированная аналоговая передаточная функция масштабируется (в частотной области) посредством следующей замены аргумента s :

$$s = \frac{s}{\omega'_p}, \quad (8.22, a)$$

где

$$\omega'_p = k \operatorname{tg} \left(\frac{\omega_p T}{2} \right), \quad k = 1 \text{ или } \frac{2}{T}. \quad (8.22, б)$$

Во-вторых, применяется билинейное z -преобразование, при котором s в новой передаточной функции заменяется величиной

$$s = k \frac{z - 1}{z + 1}.$$

1. Во многих работах (например, [19]) в двух приведенных выше операциях используется множитель $k = 2/T$. Следует помнить, что $k = 1$ и $k = 2/T$ приводят к одинаковым результатам, поскольку в любом случае k сокращается. Продемонстрируем это на примере. Рассмотрим простой фильтр:

$$H(s) = \frac{1}{s + 1}.$$

Предположим, что цифровой фильтр должен иметь частоту среза ω_p , тогда $H(s)$ следует масштабировать следующим образом:

$$\omega'_p = k \operatorname{tg} \left(\frac{\omega_p T}{2} \right).$$

Получаем такую передаточную функцию:

$$H'(s) = H(s)|_{s=s/\omega'_p} = \frac{1}{s/k \operatorname{tg}(\omega_p T/2) + 1}.$$

Далее заменяем s на $k(z - 1)/(z + 1)$:

$$H(z) = H'(s)|_{s=k(z-1)/(z+1)} = \frac{1}{k(z - 1)/(z + 1)/k \operatorname{tg}(\omega_p T/2) + 1}$$

Из приведенных выкладок видно, что множитель k сокращается, и не важно, какой величине он равен, 1 или $2/T$.

2. Для повышения вычислительной эффективности два преобразования можно объединить в одно:

$$s = \operatorname{ctg} \left(\frac{\omega_p T}{2} \right) \frac{z - 1}{z + 1}. \quad (8.23)$$

Данный подход иллюстрируется в примере 8.15.

3. Для фильтров верхних и нижних частот порядок $H(z)$ равен порядку $H(s)$. Например, если функция $H(z)$ выведена из функции $H(s)$ аналогового фильтра второго порядка, то $H(z)$ также будет описывать систему второго порядка. Для полосовых и заградительных фильтров порядок $H(z)$ будет вдвое больше порядка $H(s)$. Данная связь иногда используется для сокращения алгебраических вычислений в методе билинейного z -преобразования (см., например, [22]).
4. На практике иногда бывает так, что передаточную функцию на s -плоскости $H(s)$ существующего аналогового фильтра нужно преобразовать в функцию эквивалентного полосового фильтра дискретного времени. Такая ситуация обычна, например, при разработке цифровых аудиосистем, где аналоговый фильтр, ранее успешно использовавшийся для выравнивания, теперь нужно преобразовать в цифровой эквивалент [5].

В подобных случаях аналоговая передаточная функция реального фильтра уже дана, так что билинейное z -преобразование можно применить непосредственно после предварительной деформации и прямого частотного масштабирования характеристики аналогового фильтра нижних частот в характеристику цифрового фильтра нижних частот. Вопросы, возникающие при разработке, иллюстрируются ниже на примере.

Пример 8.9

Иллюстрация метода билинейного z -преобразования при наличии аналоговой передаточной функции реального фильтра. Определите, используя метод билинейного преобразования, коэффициенты фильтра дискретного времени, который предназначен для обработки аудиосигналов в цифровом микшере, при такой настройке управляющих сигналов, когда при добротности $Q = 2$ фильтр на частоте 5 кГц дает (максимальное) усиление на 6,02 дБ. Частота дискретизации равна 48 кГц, а передаточная функция эквивалентного аналогового фильтра в s -представлении имеет такой вид:

$$H(s) = \frac{s^2 + (3 + k)\frac{\omega_0}{Q}s + \omega_0^2}{s^2 + (3 - k)\frac{\omega_0}{Q}s + \omega_0^2},$$

где

$$k = 3 \left(\frac{G - 1}{G + 1} \right),$$

ω_0 — частота усиления,

G — коэффициент усиления,

Q — добротность.

Решение

Усиление 6,02 дБ соответствует

$$G = 10^{\frac{6,02}{20}} = 1,9999; \quad k = 0,9999.$$

Передаточная функция на s -плоскости записывается теперь так:

$$H(s) = \frac{s^2 + 4\frac{\omega_0}{Q}s + \omega_0^2}{s^2 + 2\frac{\omega_0}{Q}s + \omega_0^2}.$$

Для получения аналоговой частотной характеристики в этом уравнении делается замена $s = i\omega$.

Теперь деформированная частота усиления вычисляется как

$$\omega'_0 = \operatorname{tg} \left(\frac{\omega_0 T}{2} \right) = 0,339454,$$

а коэффициент деформации шкалы частот (согласно [5]) записывается как

$$p = \frac{\omega_0}{\operatorname{tg}\left(\frac{\omega_0 T}{2}\right)} = \frac{\omega_0}{0,339454}.$$

Итак, деформированная аналоговая передаточная функция равна

$$H'(s) = H(s) \big|_{s=ps} = \frac{p^2 s^2 + 4 \frac{\omega_0}{Q} p s + \omega_0^2}{p^2 s^2 + 2 \frac{\omega_0}{Q} p s + \omega_0^2}.$$

Далее применяем билинейное z -преобразование:

$$\begin{aligned} H(z) &= H'(s) \big|_{s=\frac{z-1}{z+1}} = \\ &= \frac{\left(\frac{z-1}{z+1}\right)^2 + 0,6789085 \left(\frac{z-1}{z+1}\right) + 0,115229}{\left(\frac{z-1}{z+1}\right)^2 + 0,339454 \left(\frac{z-1}{z+1}\right) + 0,115229} = \\ &= \frac{1,233352 - 1,216444z^{-1} + 0,29994z^{-2}}{1 - 1,216444z^{-1} + 0,5332946z^{-2}}. \end{aligned}$$

Пример 8.10

Простой режекторный LRC-фильтр имеет следующую нормированную передаточную функцию на s -плоскости:

$$H(s) = \frac{s^2 + 1}{s^2 + s + 1}.$$

Определите передаточную функцию эквивалентного фильтра дискретного времени, используя метод билинейного z -преобразования. Частота режекции равна 50 Гц, частота дискретизации — 500 Гц.

Решение

Поскольку передаточная функция на s -плоскости уже известна, применять преобразование “фильтр нижних частот в режекторный фильтр” нет смысла, поскольку это будет двойным преобразованием. Критическая частота в этой задаче равна

$$\omega'_p = \operatorname{tg}\left(\frac{\omega_p T}{2}\right) = \operatorname{tg}\left(\frac{2\pi \times 50}{500 \times 2}\right) = 0,3249196.$$

Масштабированная передаточная функция на s -плоскости записывается как

$$\begin{aligned} H'(s) &= H(s) \Big|_{s \rightarrow \frac{s}{\omega_p}} = \\ &= \frac{\left(\frac{s}{\omega_p'}\right)^2 + 1}{\left(\frac{s}{\omega_p'}\right)^2 + \frac{s}{\omega_p'} + 1} = \\ &= \frac{s^2 + 0,105572}{s^2 + 0,3249196s + 0,105572}. \end{aligned}$$

Применяем билинейное z -преобразование:

$$\begin{aligned} H(z) &= H'(z) \Big|_{s \rightarrow \frac{z-1}{z+1}} = \\ &= \frac{\left(\frac{z-1}{z+1}\right)^2 + 0,105572}{\left(\frac{z-1}{z+1}\right)^2 + 0,3249196\frac{z-1}{z+1} + 0,105572}. \end{aligned}$$

8.9. Использование для разработки БИХ-фильтров билинейного z -преобразования и классических аналоговых фильтров

Во многих практических ситуациях аналоговая передаточная функция $H(s)$, по которой вычисляется $H(z)$, может быть неизвестной, и ее нужно определить по спецификациям желаемых цифровых фильтров. В стандартных задачах частотно-избирательной цифровой фильтрации (т.е. включающих фильтры нижних частот, верхних частот, полосовые и режекторные) $H(s)$ можно получить на основе классических фильтров с характеристиками Баттерворта, Чебышева или эллиптическими (см. рис. 8.12). В данном разделе представлены методы разработки БИХ-фильтров по таким классическим аналоговым фильтрам. Будут подробно рассмотрены основные концепции и на примерах проиллюстрированы вопросы разработки. Вначале кратко описаны важные особенности классических аналоговых фильтров, которые нужно знать при разработке БИХ-фильтров. Рассматриваются только фильтры-прототипы нижних частот, поскольку, как будет показано ниже, фильтры других типов обычно выводятся из нормированных фильтров нижних частот.

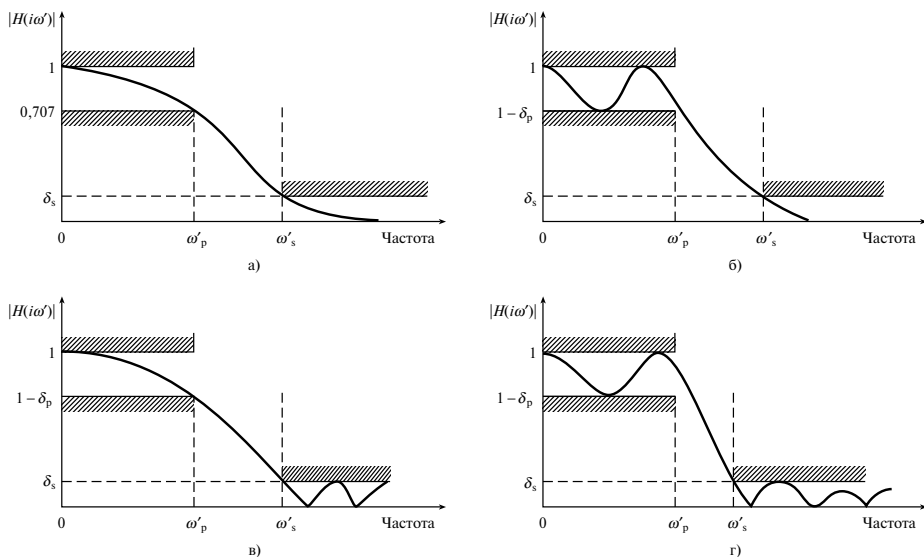


Рис. 8.12. Схематические частотные характеристики некоторых классических аналоговых фильтров: а) фильтр Баттерворта; б) фильтр Чебышева I типа; в) фильтр Чебышева II типа; г) эллиптический фильтр

8.9.1. Характерные особенности классических аналоговых фильтров

8.9.1.1. Фильтр Баттерворта

Фильтр нижних частот Баттерворта характеризуется следующим квадратом амплитудно-частотной характеристики:

$$|H(\omega)|^2 = \frac{1}{1 + \left(\frac{\omega}{\omega_p^p}\right)^{2N}}, \quad (8.24)$$

где N — порядок фильтра, а ω_p^p — частота среза фильтра нижних частот по уровню 3 дБ (для нормированного фильтра-прототипа $\omega_p^p = 1$). Амплитудно-частотная характеристика типичного фильтра Баттерворта изображена на рис. 8.12, а и видно, что она монотонна в полосе пропускания и полосе подавления. Характеристика называется *максимально плоской* из-за ее изначальной пологости (с нулевым наклоном на постоянной составляющей).

Порядок фильтра N определяется соотношением

$$N \geq \frac{\lg\left(\frac{10^{\frac{A_s}{10}} - 1}{10^{\frac{A_p}{10}} - 1}\right)}{2 \lg\left(\frac{\omega_s^p}{\omega_p^p}\right)}, \quad (8.25)$$

где A_p и A_s соответственно — неравномерность в полосе пропускания и затухание в полосе подавления в децибелах, а ω_s^p — граничная частота полосы подавления.

Передаточная функция нормированного аналогового фильтра Баттерворта $H(s)$ содержит нули на бесконечности и полюса, равномерно распределенные по окружности единичного радиуса на s -плоскости (см. [14, 23]):

$$s_k = e^{\pi i(2k+N-1)/2N} = \cos \left[\frac{(2k+N-1)\pi}{2N} \right] + i \sin \left[\frac{(2k+N-1)\pi}{2N} \right], \quad (8.26)$$

$$k = 1, 2, \dots, N.$$

Полюса располагаются комплексно-сопряженными парами и лежат в левой части s -плоскости.

8.9.1.2. Фильтр Чебышева

Альтернативный путь получения подходящей аналоговой передаточной функции $H(s)$ предлагает характеристика Чебышева. Существует два типа фильтров Чебышева, тип I и тип II, отличающиеся следующими особенностями (рис. 8.12, б и 8.12, в).

- Тип I: характеристика с равными колебаниями в полосе пропускания, монотонная в полосе подавления.
- Тип II: характеристика с равными колебаниями в полосе подавления, монотонная в полосе пропускания.

Фильтры Чебышева типа I, например, характеризуются следующим квадратом амплитудно-частотной характеристики:

$$|H(\omega')|^2 = \frac{K}{1 + \epsilon^2 C_N^2(\omega'/\omega'_p)}, \quad (8.27, a)$$

где $C_N(\omega'/\omega'_p)$ — полином Чебышева с равными колебаниями в полосе пропускания, N — порядок полинома (также порядок фильтра), а ϵ определяет неравномерность в полосе пропускания, которая в децибелах выражается так:

$$\begin{aligned} \text{неравномерность в полосе пропускания} &\leq 10 \lg(1 + \epsilon^2) = \\ &= -20 \lg(1 - \delta_p). \end{aligned} \quad (8.27, б)$$

Типичная амплитудная характеристика фильтра Чебышева первого типа показана на рис. 8.12, б. Передаточная функция фильтра Чебышева $H(s)$ зависит от желаемой неравномерности в полосе пропускания и порядка фильтра N . Величина N определяется соотношением

$$N \geq \frac{\cosh^{-1} \left(\frac{10^{\frac{A_s}{10}} - 1}{10^{\frac{A_p}{10}} - 1} \right)}{\cosh^{-1} \left(\frac{\omega_s^p}{\omega_p^p} \right)}, \quad (8.28)$$

где A_p и A_s соответственно — неравномерность в полосе пропускания и затухание в полосе подавления в децибелах, ω_s^p — граничная частота полосы подавления.

Полюса нормированного ФНЧ Чебышева лежат на эллипсе в s -плоскости, а их координаты определяются выражением [23]

$$s_k = \operatorname{sh}(\alpha) \cos(\beta_k) + i \operatorname{csh}(\alpha) \sin(\beta_k), \quad (8.29)$$

где

$$\alpha = \frac{1}{N} \operatorname{sh}^{-1} \left(\frac{1}{\epsilon} \right); \beta_k = \frac{(2k + N - 1)\pi}{2N}, \quad k = 1, 2, \dots, N.$$

8.9.1.3. Эллиптический фильтр

Характеристика эллиптического фильтра имеет равные колебания в полосе пропускания и подавления (см. рис. 8.12, ε). Данный фильтр характеризуется следующим квадратом амплитудно-импульсной характеристики:

$$|H(\omega')|^2 = \frac{K}{1 + \epsilon^2 G_N^2(\omega')}, \quad (8.30)$$

где $G_N(\omega')$ — рациональная функция Чебышева. В отличие от фильтров Баттерворта и Чебышева, простого выражения для полюсов эллиптического фильтра не существует, взамен этого есть процедуры вычисления положений полюсов (например, см. [2, 7, 14]). Нули эллиптического фильтра нижних частот полностью мнимые.

Эллиптическая характеристика дает наиболее эффективные фильтры с точки зрения амплитудной характеристики. Она позволяет получать фильтры наименьшего порядка при данном наборе спецификаций, и в первую очередь при разработке БИХ-фильтров следует попробовать именно эллиптическую характеристику. Исключение составляют те случаи, когда важна фазовая характеристика (тогда стоит использовать характеристику Баттерворта).

Во многие книги по аналоговой разработке включены таблицы полиномиальных представлений $H(s)$ для фильтров Баттерворта, Чебышева и эллиптических, где они приводятся в нормированной форме, так что их можно использовать в билинейном преобразовании. На практике, впрочем, для вычисления $H(z)$ по данной $H(s)$ обычно используются программные пакеты, что будет продемонстрировано ниже.

8.9.2. Методология билинейного z -преобразования с использованием классических аналоговых фильтров

Если ФНЧ-прототипа не существует, схема билинейного z -преобразования реализуется следующим образом.

1. Деформировать граничные или критичные частоты цифрового фильтра, как описывалось выше.
2. На основе спецификации цифрового фильтра и классических характеристик фильтров найти подходящий аналоговый фильтр-прототип нижних частот. Данный этап включает использование одного из уравнений преобразования частоты (в зависимости от типа цифрового фильтра — нижних частот, верхних частот, полосовой или

режекторный) для определения спецификаций фильтра-прототипа нижних частот. На основе полученных данных определяется порядок фильтра-прототипа, а следовательно, его передаточная функция $H(s)$.

3. Денормировать передаточную функцию аналогового ФНЧ-прототипа $H(s)$, используя преобразование частоты и масштабирование, и получить новую передаточную функцию $H'(s)$, как описывалось выше.
4. Применить билинейное z -преобразование и получить передаточную функцию искомого цифрового фильтра $H(z)$, заменив s в частотно-масштабированной передаточной функции $H'(s)$, как описывалось выше.

Рассмотрим теперь основные концепции всех типов фильтров (нижних частот, верхних частот, полосовых и режекторных).

8.9.2.1. Фильтр нижних частот: основные концепции

Преобразование передаточной функции фильтра нижних частот в передаточную функцию фильтра нижних частот происходит по следующему закону:

$$s = \frac{s}{\omega'_p}.$$

Если в этом выражении заменить s на $i\omega$ и записать частоты фильтра-прототипа как ω^p , а частоты разрабатываемого фильтра нижних частот как $\omega_{\text{нч}}$ (чтобы как-то их различать), приведенное выше выражение переходит в такую форму:

$$i\omega^p = i\frac{\omega_{\text{нч}}}{\omega'_p}, \text{ т.е. } \omega^p = \frac{\omega_{\text{нч}}}{\omega'_p}. \quad (8.31)$$

Уравнение (8.31) определяет связь между частотами фильтра-прототипа и фильтра нижних частот, который требуется разработать. Зная критичные частоты денормированного фильтра нижних частот, можно использовать формулу (8.31) и найти критичные частоты фильтра-прототипа, а следовательно, определить его спецификации.

Фильтр-прототип имеет три критичные частоты: 0, граничная частота полосы пропускания; ω_p^p (это практически всегда 1); и граничная частота полосы подавления, ω_s^p :

- 1) если $\omega_{\text{нч}} = 0$, $\omega^p = 0$ (из уравнения (8.31));
- 2) если $\omega_{\text{нч}} = \omega'_p$ (т.е. граничной частоте полосы пропускания), $\omega^p = \omega'_p/\omega'_p = 1 = \omega_p^p$;
- 3) если $\omega_{\text{нч}} = \omega'_s$, $\omega^p = \omega'_s/\omega'_p = \omega_s^p$.

Итак, критичными частотами для фильтра-прототипа являются 0, 1, ω'_s/ω'_p .

Связь частот денормированного фильтра нижних частот и фильтра-прототипа показана на рис. 8.13.

8.9.2.2. Фильтр верхних частот: основные концепции

Используя преобразование “НЧ-фильтр в ВЧ-фильтр”, $s = \omega_p'/s$ и обозначив через $\omega_{\text{вч}}$ частоты денормированного фильтра верхних частот, а через ω^p — частоты ФНЧ-

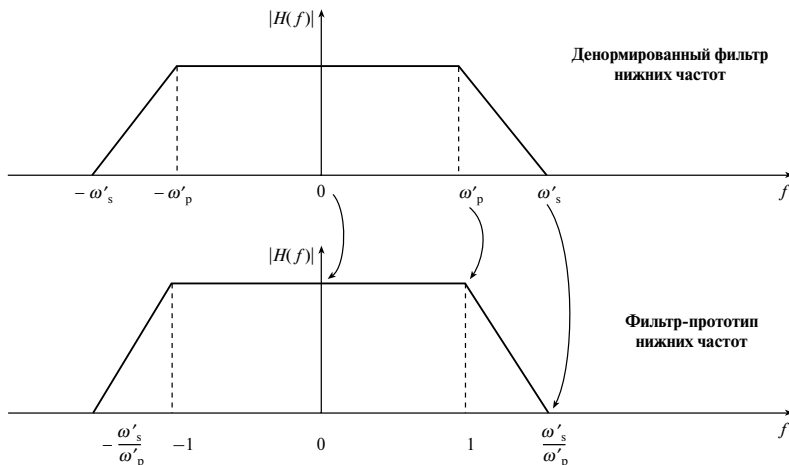


Рис. 8.13. Связь между частотами денормированного ФНЧ и ФНЧ-прототипа

прототипа (как и выше), получим следующую связь между частотами ФНЧ-прототипа и нужного фильтра верхних частот:

$$\omega^p = -\frac{\omega'_p}{\omega_{\text{ВЧ}}}. \quad (8.32)$$

Используя формулу (8.32), критичные частоты ФНЧ-прототипа можно выразить через частоты искомого фильтра верхних частот:

- 1) если $\omega_{\text{ВЧ}} = 0$, $\omega^p = \infty$ (используем формулу (8.32));
- 2) если $\omega_{\text{ВЧ}} = \omega'_p$ (т.е. граничной частоте полосы пропускания), $\omega^p = 1$.
- 3) если $\omega_{\text{ВЧ}} = \omega'_s$, $\omega^p = -\frac{\omega'_p}{\omega'_s}$;
- 4) если $\omega_{\text{ВЧ}} = -\omega'_p$, $\omega^p = 1$;
- 5) если $\omega_{\text{ВЧ}} = -\omega'_s$, $\omega^p = \frac{\omega'_p}{\omega'_s}$.

Следовательно, при разработке фильтра верхних частот тремя критическими частотами ФНЧ-прототипа являются 0, 1 и ω'_p/ω'_s .

Критичные частоты ФНЧ-прототипа и их связь с частотами денормированного фильтра верхних частот изображены на рис. 8.14. Отметим, что преобразование “НЧ-фильтр в ВЧ-фильтр” переводит частоты фильтра верхних частот: 0 отображается в бесконечность, ω_p — в единицу, а бесконечность — в нуль.

8.9.2.3. Полосовые фильтры: основные концепции

Преобразование “фильтр нижних частот в полосовой фильтр” записывается следующим образом:

$$s = \frac{s^2 + \omega_0^2}{W s}.$$

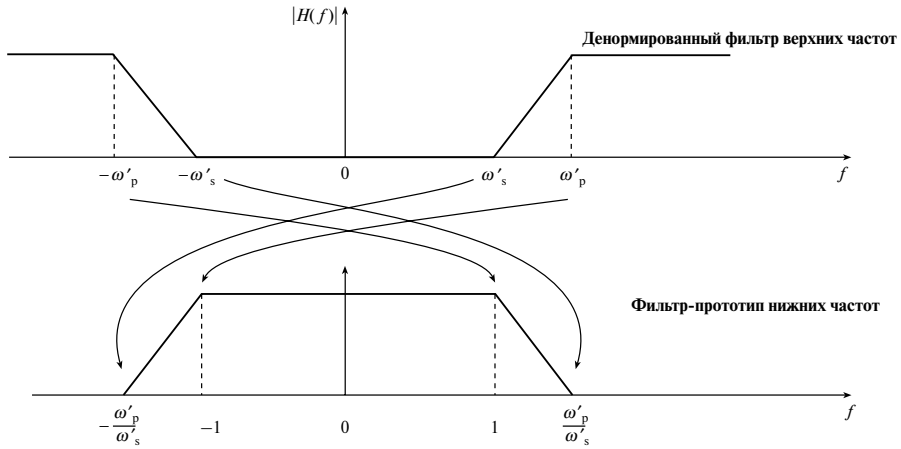


Рис. 8.14. Связь частот денормированного ФВЧ и ФНЧ-прототипа

Согласно этому правилу частоты полосового фильтра $\omega_{\text{пп}}$ и частоты ФНЧ-прототипа ω^p связаны следующим соотношением:

$$i\omega^p = \frac{(i\omega_{\text{пп}})^2 + \omega_0^2}{iW\omega_{\text{пп}}},$$

т.е.

$$\omega^p = \frac{\omega_{\text{пп}}^2 - \omega_0^2}{W\omega_{\text{пп}}}. \quad (8.33)$$

Полосовой фильтр имеет четыре критичные, или граничные, частоты и центральную частоту:

$\omega'_{p1}, \omega'_{p2}$ = верхняя и нижняя граничные частоты полосы пропускания

$\omega'_{s1}, \omega'_{s2}$ = верхняя и нижняя граничные частоты полосы поглощения

ω_0 = центральная частота ($\omega_0^2 = \omega'_{p1}\omega'_{p2}$)

Используя соотношение (8.33), граничные частоты ФНЧ-прототипа можно выразить через граничные частоты полосового фильтра:

- 1) если $\omega_{\text{пп}} = \omega'_{s1}$, $\omega^p = \omega'_{s1} = \frac{\omega'^2_{s1} - \omega_0^2}{W\omega'_{s1}}$;
- 2) если $\omega_{\text{пп}} = \omega'_{p1}$, $\omega^p = \frac{\omega'^2_{p1} - \omega_0^2}{W\omega'_{p1}} = \frac{\omega'^2_{p1} - \omega'_{p1}\omega'_{p2}}{(\omega'_{p2} - \omega'_{p1})\omega'_{p1}} = -1$;
- 3) если $\omega_{\text{пп}} = \omega'_{p2}$, $\omega^p = \frac{\omega'^2_{p2} - \omega_0^2}{W\omega'_{p2}} = \frac{\omega'^2_{p2} - \omega'_{p1}\omega'_{p2}}{(\omega'_{p2} - \omega'_{p1})\omega'_{p2}} = 1$;
- 4) если $\omega_{\text{пп}} = \omega'_{s2}$, $\omega^p = \omega'_{s2} = \frac{\omega'^2_{s2} - \omega_0^2}{W\omega'_{s2}}$;
- 5) если $\omega_{\text{пп}} = \omega_0$, $\omega^p = \frac{\omega_0^2 - \omega_0^2}{W\omega_0^2}$;
- 6) $\omega^p_s = \min(\omega'^p_{s1}, \omega'^p_{s2})$.

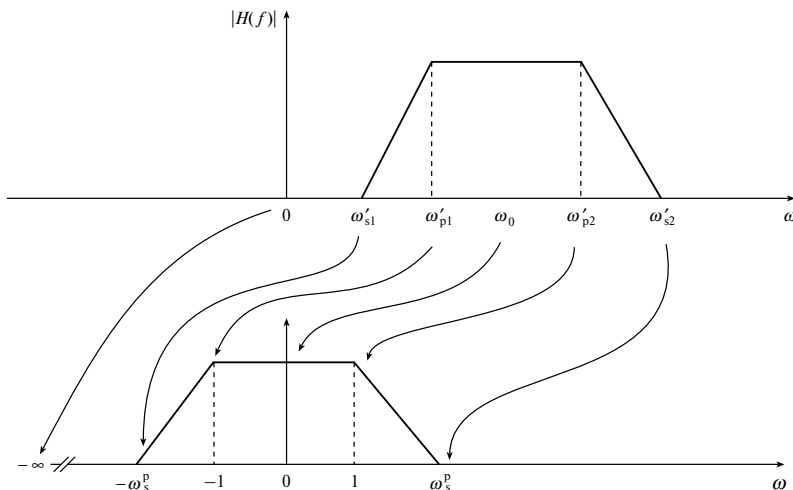


Рис. 8.15. Отображение ФНЧ-прототипа в полосовой фильтр

Следовательно, важными критическими частотами ФНЧ-прототипа являются

$$0, 1, \min(\omega_{s1}'^p, |\omega_{s2}'^p|).$$

Отображение частот полосового фильтра в частоты ФНЧ-прототипа изображено на рис. 8.15. Отметим, что центральная частота полосового фильтра отображается в нулевую частоту фильтра-прототипа, а верхние граничные частоты полосы пропускания и полосы подавления ω_{p2}' и ω_{s2}' соответственно переводятся в положительные граничные частоты полосы пропускания и полосы подавления фильтра-прототипа. С другой стороны, нижние граничные частоты полосы пропускания и полосы подавления ω_{p1}' и ω_{s1}' соответственно переводятся в отрицательные граничные частоты полосы пропускания и полосы подавления фильтра-прототипа. На практике используется граничная частота полосы подавления фильтра-прототипа, равная меньшей из двух частот полосы подавления $\omega_{s2}'^p$ и $\omega_{s1}'^p$, как отмечалось выше. Неравномерности в полосе пропускания и затухания в полосе подавления ФНЧ-прототипа равны соответствующим величинам цифрового полосового фильтра.

По спецификациям ФНЧ-прототипа можно определить порядок N передаточной функции фильтра, используя, например, уравнение (8.25) или (8.28). Порядок полосового фильтра равен удвоенному порядку ФНЧ-прототипа, т.е. $2N$. Полюса фильтра-прототипа определяются из уравнений (8.26) и (8.29). Для фильтров Баттерворта и Чебышева (тип I) нули ФНЧ-прототипа расположены на бесконечности, а для эллиптического фильтра они полностью мнимые. Зная положения полюсов и нулей (или имея стандартные таблицы для классических фильтров), можно вычислить передаточную функцию фильтра-прототипа.

8.9.2.4. Режекторные фильтры: основные концепции

Преобразование “фильтр нижних частот в режекторный фильтр” записывается следующим образом:

$$s = \frac{W s}{s^2 + \omega_0^2}.$$

Частота режекции $\omega_{\text{пр}}$ и частота фильтра-прототипа ω^p связаны соотношением

$$i\omega^p = \frac{iW\omega_{\text{пр}}}{(i\omega_{\text{пр}})^2 + \omega_0^2},$$

т.е.

$$\omega^p = \frac{W\omega_{\text{пр}}}{\omega_0^2 - \omega_{\text{пр}}^2}. \quad (8.34)$$

Из соотношения (8.34) можно определить граничные частоты ФНЧ-прототипа по известным частотам нужного цифрового режекторного фильтра. Напомним, что режекторный фильтр имеет четыре граничные частоты — ω'_{p1} , ω'_{p2} (нижняя и верхняя граничные частоты полосы пропускания), ω'_{s1} , ω'_{s2} (нижняя и верхняя граничные частоты полосы подавления) и центральную частоту ω_0 ($\omega_0^2 = \omega'_{p1}\omega'_{p2}$):

- 1) если $\omega_{\text{пр}} = \omega'_{p1}$, $\omega^p = \frac{W\omega'_{p1}}{\omega_0^2 - \omega_{p1}^2} = \frac{(\omega'_{p2} - \omega'_{p1})\omega'_{p1}}{\omega'_{p1}\omega'_{p2} - \omega_{p1}^2} = 1$;
- 2) если $\omega_{\text{пр}} = \omega'_{s1}$, $\omega^p = \omega_s^{p(1)} = \frac{W\omega'_{s1}}{\omega_0^2 - \omega_{s1}^2}$;
- 3) если $\omega_{\text{пр}} = \omega'_{s2}$, $\omega^p = \omega_s^{p(2)} = \frac{W\omega'_{s2}}{\omega_0^2 - \omega_{s2}^2}$;
- 4) если $\omega_{\text{пр}} = \omega_0$, $\omega^p = \frac{W\omega_0}{\omega_0^2 - \omega_0^2} = \infty$;
- 5) если $\omega_{\text{пр}} = \omega'_{p2}$, $\omega^p = \frac{W\omega'_{p2}}{\omega_0^2 - \omega_{p2}^2} = \frac{(\omega'_{p2} - \omega'_{p1})\omega'_{p2}}{\omega'_{p1}\omega'_{p2} - \omega_{p2}^2} = -1$.

Таким образом, граничная частота полосы подавления ФНЧ-прототипа $\omega_s^p = \min(\omega_s^{p(1)}, \omega_s^{p(2)})$, а его граничная частота полосы пропускания равна 1. Неравномерность в полосе пропускания и затухание в полосе подавления равны соответственно A_p и A_s . Отображение частот режекторного фильтра в частоты фильтра нижних частот показано на рис. 8.16. Видно, что верхние граничные частоты полосы подавления и полосы пропускания в режекторном фильтре отображаются в отрицательные частоты фильтра-прототипа, тогда как нижние граничные частоты отображаются в положительные частоты фильтра-прототипа.

Итак, существенными критичными частотами фильтра нижних частот являются

$$0, 1, \omega_s^p (\text{где } \omega_s^p = \min(\omega_s^{p(1)}, \omega_s^{p(2)})).$$

Из спецификации ФНЧ-прототипа можно, используя уравнения (8.25) и (8.28), определить порядок N и передаточную функцию фильтра. Порядок режекторного фильтра равен удвоенному порядку фильтра-прототипа, т.е. $2N$.

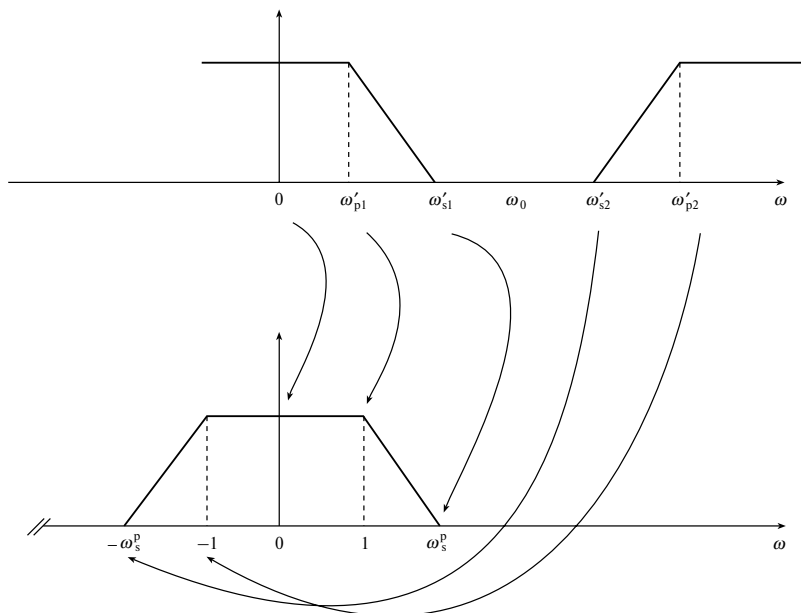


Рис. 8.16. Связь частот денормированного режекторного фильтра и фильтра нижних частот, используемого при разработке как прототип

8.9.3. Примеры разработки фильтров нижних частот, верхних частот, полосовых и режекторных

Пример 8.11

Фильтр нижних частот. Требуется цифровой фильтр нижних частот, удовлетворяющий следующим спецификациям:

полоса пропускания	0–500 Гц,
полоса подавления	2–4 кГц,
неравномерность в полосе пропускания	3 дБ,
затухание в полосе подавления	20 дБ,
частота дискретизации	8 кГц.

Определите следующие величины:

- 1) граничные частоты полосы пропускания и полосы подавления для подходящего аналогового фильтра-прототипа нижних частот;
- 2) порядок N фильтра-прототипа;
- 3) коэффициенты, а следовательно, передаточную функцию фильтра дискретного времени, используя билинейное z -преобразование.

Считайте, что аналоговый фильтр имеет характеристику Баттерворта.

Решение

1. На основе спецификации находим деформированные частоты:

$$\omega'_p = \operatorname{tg} \left(\frac{2\pi \times 500}{2 \times 8000} \right) = 0,198912,$$

$$\omega'_s = \operatorname{tg} \left(\frac{2\pi \times 2000}{2 \times 8000} \right) = 1,$$

$$\omega_s^p = \frac{\omega'_s}{\omega'_p} = 1/0,198912 = 5,0273.$$

Следовательно, имеем такие деформированные граничные частоты полосы пропускания и полосы подавления ФНЧ-прототипа: 0; 1; 5,0273.

2. Используем уравнение (8.25) и параметры, найденные в п. 1, чтобы определить порядок фильтра.

Итак, $10^{A_s/10} - 1 = 10^{20/10} - 1 = 99$; $10^{A_p/10} - 1 = 10^{3/10} - 1 = 0,9952623$;

$$\lg \left(\frac{99}{0,9952623} \right) = 1,997697.$$

Для ФНЧ-прототипа

$$\omega_p^p = 1; \omega_s^p = 5,0273; \lg \left(\frac{\omega_s^p}{\omega_p^p} \right) = 2 \lg(5,0273) = 1,40266$$

$$N \geq \frac{1,997697}{1,40266} = 1,424. \text{ Пусть } N = 2.$$

3. Полюса фильтра-прототипа (из уравнения (8.26)) расположены в точках

$$s_{p,1} = \cos \left[\frac{(2+2-1)\pi}{4} \right] + i \sin \left[\frac{(2+2-1)\pi}{4} \right] = -\frac{\sqrt{2}}{2} + i\frac{\sqrt{2}}{2}$$

$$s_{p,2} = -\frac{\sqrt{2}}{2} - i\frac{\sqrt{2}}{2}.$$

Передаточная функция на s -плоскости $H(s)$ записывается следующим образом:

$$H(s) = \frac{1}{(s - s_{p,1})(s - s_{p,2})} = \frac{1}{s^2 + \sqrt{2}s + 1}.$$

Частотно-масштабированная передаточная функция на s -плоскости записывается так:

$$\begin{aligned} H'(s) &= H(s) \Big|_{\frac{s}{\omega_p}} = \frac{1}{\left(\frac{s}{\omega_p}\right)^2 + \sqrt{2}\frac{s}{\omega_p} + 1} = \\ &= \frac{\omega_p'^2}{s^2 + \sqrt{2}s\omega_p' + \omega_p'^2}. \end{aligned}$$

Применяя билинейное z -преобразование, получаем такой результат:

$$H(z) = H'(s) \Big|_{s=\frac{z-1}{z+1}} = \frac{\omega_p'^2}{\left(\frac{z-1}{z+1}\right)^2 + \sqrt{2}\omega_p' \left(\frac{z-1}{z+1}\right) + \omega_p'^2} =$$

$$= \frac{\omega_p'^2(z+1)^2}{(z-1)^2 + \sqrt{2}\omega_p'(z-1)(z+1) + \omega_p'^2(z+1)^2}.$$

Упрощаем, делим числитель и знаменатель на z^2 и получаем такую передаточную функцию:

$$H(z) = \frac{\omega_p'^2}{1 + \sqrt{2}\omega_p' + \omega_p'^2} \times \frac{1 + 2z^{-1} + z^{-2}}{1 + \frac{2(\omega_p'^2-1)z^{-1}}{1+\sqrt{2}\omega_p'+\omega_p'^2} + \frac{(1-\sqrt{2}\omega_p'+\omega_p'^2)z^{-2}}{1+\sqrt{2}\omega_p'+\omega_p'^2}}.$$

Подставляя следующие значения параметров:

$$1 + \sqrt{2}\omega_p' + \omega_p'^2 = 1,32087; \omega_p'^2 - 1 = -0,96043$$

$$1 - \sqrt{2}\omega_p' + \omega_p'^2 = 0,758285; \omega_p'^2 = 0,0395659$$

в приведенную выше формулу, получаем

$$H(z) = \frac{0,02995(1 + 2z^{-1} + z^{-2})}{1 - 1,4542z^{-1} + 0,57408z^{-2}}.$$

Пример 8.12

Фильтр верхних частот. Требуется цифровой фильтр верхних частот, удовлетворяющий следующим спецификациям:

полоса пропускания	2–4 Гц,
полоса подавления	0–500 кГц,
неравномерность в полосе пропускания	3 дБ,
затухание в полосе подавления	20 дБ,
частота дискретизации	8 кГц.

Определите следующие величины:

- 1) граничные частоты полосы пропускания и полосы подавления для подходящего аналогового фильтра-прототипа нижних частот;
- 2) порядок N фильтра-прототипа;
- 3) коэффициенты, а следовательно, передаточную функцию фильтра дискретного времени, используя билинейное z -преобразование.

Считайте, что аналоговый фильтр имеет характеристику Баттерворта.

Решение

1. На основе спецификации находим деформированные частоты:

$$\omega'_s = \operatorname{tg} \left(\frac{2\pi \times 500}{2 \times 8000} \right) = 0,198912,$$

$$\omega'_p = \operatorname{tg} \left(\frac{2\pi \times 2000}{2 \times 8000} \right) = 1,$$

$$\omega_s^p = \frac{\omega'_p}{\omega'_s} = 1/0,198912 = 5,0273.$$

Следовательно, имеем такие деформированные граничные частоты полосы пропускания и полосы подавления ФНЧ-прототипа: 0; 1; 5,0273.

2. Используем уравнение (8.25) и параметры, найденные в п. 1, чтобы определить порядок фильтра.

Итак, $10^{A_s/10} - 1 = 10^{20/10} - 1 = 99$; $10^{A_p/10} - 1 = 10^{3/10} - 1 = 0,9952623$;

$$\lg \left(\frac{99}{0,9952623} \right) = 1,997697.$$

Для ФНЧ-прототипа

$$\omega_p^p = 1; \omega_s^p = 5,0273; \lg \left(\frac{\omega_s^p}{\omega_p^p} \right) = 2 \lg(5,0273) = 1,40266$$

$$N \geq \frac{1,997697}{1,40266} = 1,424. \text{ Пусть } N = 2.$$

3. Полюса фильтра-прототипа (из уравнения (8.26)) расположены в точках

$$s_{p,1} = \cos \left[\frac{(2+2-1)\pi}{4} \right] + i \sin \left[\frac{(2+2-1)\pi}{4} \right] = -\frac{\sqrt{2}}{2} + i\frac{\sqrt{2}}{2}$$

$$s_{p,2} = -\frac{\sqrt{2}}{2} - i\frac{\sqrt{2}}{2}.$$

Передаточная функция на s -плоскости $H(s)$ записывается следующим образом:

$$H(s) = \frac{1}{(s - s_{p,1})(s - s_{p,2})} = \frac{1}{s^2 + \sqrt{2}s + 1}.$$

Частотно-масштабированная передаточная функция на s -плоскости записывается следующим образом:

$$\begin{aligned} H'(s) &= H(s) \Big|_{\frac{\omega'_p}{s}} = \frac{1}{\left(\frac{\omega'_p}{s}\right)^2 + \sqrt{2}\frac{\omega'_p}{s} + 1} = \\ &= \frac{s^2}{s^2 + \sqrt{2}s\omega'_p + \omega_p'^2}. \end{aligned}$$

Применяя билинейное z -преобразование, получаем такой результат:

$$\begin{aligned} H(z) = H'(s) \Big|_{s=\frac{z-1}{z+1}} &= \frac{\left(\frac{z-1}{z+1}\right)^2}{\left(\frac{z-1}{z+1}\right)^2 + \sqrt{2}\omega'_p \left(\frac{z-1}{z+1}\right) + \omega_p'^2} = \\ &= \frac{(z-1)^2}{(z-1)^2 + \sqrt{2}\omega'_p(z-1)(z+1) + \omega_p'^2(z+1)^2}. \end{aligned}$$

Упрощаем, делим числитель и знаменатель на z^2 и получаем такую передаточную функцию:

$$H(z) = \frac{1}{1 + \sqrt{2}\omega'_p + \omega_p'^2} \times \frac{1 - 2z^{-1} + z^{-2}}{1 + \frac{2(\omega_p'^2 - 1)z^{-1}}{1 + \sqrt{2}\omega'_p + \omega_p'^2} + \frac{(1 - \sqrt{2}\omega'_p + \omega_p'^2)z^{-2}}{1 + \sqrt{2}\omega'_p + \omega_p'^2}}.$$

Подставляя следующие значения параметров:

$$1 + \sqrt{2}\omega'_p + \omega_p'^2 = 3,41421; \omega_p'^2 - 1 = 0$$

$$1 - \sqrt{2}\omega'_p + \omega_p'^2 = 0,585786; \omega_p'^2 = 1$$

в приведенную выше формулу, получаем

$$H(z) = \frac{0,029289(1 - 2z^{-1} + z^{-2})}{1 + 0,17157z^{-2}}.$$

Пример 8.13

Полосовой фильтр. Требуется цифровой полосовой фильтр с амплитудно-частотной характеристикой типа Баттерворта, удовлетворяющий следующим спецификациям:

нижняя граничная частота полосы пропускания	200 Гц,
верхняя граничная частота полосы пропускания	300 Гц,
нижняя граничная частота полосы подавления	50 Гц,
верхняя граничная частота полосы подавления	450 Гц,
неравномерность в полосе пропускания	3 дБ,
затухание в полосе подавления	20 дБ,
частота дискретизации	8 кГц.

Определите следующие величины:

- 1) граничные частоты полосы пропускания и полосы подавления для подходящего аналогового фильтра-прототипа нижних частот;
- 2) порядок N фильтра-прототипа;
- 3) коэффициенты, а следовательно, передаточную функцию фильтра дискретного времени, используя билинейное z -преобразование.

Решение

Деформированные критичные частоты полосового фильтра равны

$$\begin{aligned}\omega_0 &= 1, & W &= 0,6498, \\ \omega'_{p1} &= 0,7265, & \omega'_{p2} &= 1,37638, \\ \omega'_{s1} &= 0,1584, & \omega'_{s2} &= 6,3138.\end{aligned}$$

Следовательно, граничные частоты фильтра-прототипа равны (используются соотношения, приведенные выше)

$$\omega_p^p = 1; \quad \omega_s^p = 9,4721.$$

Итак, требуется НЧ-фильтр-прототип с $\omega_p^p = 1$; $\omega_s^p = 9,4721$; $A_p = 3$ дБ; $A_s = 20$ дБ.

Из уравнения (8.25) находим порядок фильтра-прототипа:

$$\begin{aligned}10^{A_s/10} - 1 &= 10^{20/10} - 1 = 99; \quad 10^{A_p/10} - 1 = 10^{3/10} - 1 = 0,9952623; \\ \lg\left(\frac{99}{0,9952623}\right) &= 1,9976976; \quad \frac{\omega_s^p}{\omega_p^p} = 9,4721; \quad 2\lg(9,4721) = 1,95289 \\ N &= \frac{1,9976976}{1,95289} = 1,0229.\end{aligned}$$

Величина N должна быть целой, поэтому для простоты используем значение $N = 1$. Передаточная функция на s -плоскости ФНЧ первого порядка записывается как

$$H(s) = \frac{1}{s + 1}.$$

Используя преобразование “фильтр нижних частот в полосовой фильтр” из таблицы, получаем

$$\begin{aligned}H'(s) &= H(s) \Big|_{s=\frac{s^2+\omega_0^2}{Ws}} = \frac{1}{\left(\frac{s^2+\omega_0^2}{Ws}\right) + 1} = \\ &= \frac{Ws}{s^2 + Ws + \omega_0^2}.\end{aligned}$$

Результат применения билинейного z -преобразования таков:

$$\begin{aligned}H(z) &= H'(s) \Big|_{s=\frac{z-1}{z+1}} = \\ &= \frac{W\left(\frac{z-1}{z+1}\right)}{\left(\frac{z-1}{z+1}\right)^2 + W\left(\frac{z-1}{z+1}\right) + \omega_0^2}.\end{aligned}$$

Подставляя значения ω_0^2 и W , окончательно получаем

$$H(z) = \frac{0,2452(1 - z^{-2})}{1 + 0,5095z^{-2}}.$$

Пример 8.14

Заграждающий фильтр. Требуется цифровой режекторный БИХ-фильтр с амплитудно-частотной характеристикой типа Баттерворта, удовлетворяющий следующим спецификациям:

нижняя полоса пропускания	0–50 Гц,
верхняя полоса пропускания	450–500 Гц,
полоса подавления	200–300 Гц,
неравномерность в полосе пропускания	3 дБ,
затухание в полосе подавления	20 дБ,
частота дискретизации	8 кГц.

Определите следующие величины:

- 1) граничные частоты полосы пропускания и полосы подавления для подходящего аналогового фильтра-прототипа нижних частот;
- 2) порядок N фильтра-прототипа;
- 3) коэффициенты, а следовательно, передаточную функцию фильтра дискретного времени, используя билинейное z -преобразование.

Решение

Деформированные критичные частоты режекторного фильтра равны

$$\begin{aligned}\omega_0 &= 1, & W &= 6,1554, \\ \omega'_{p1} &= 0,1584, & \omega'_{p2} &= 6,3138, \\ \omega'_{s1} &= 0,7265, & \omega'_{s2} &= 1,37638.\end{aligned}$$

Следовательно, граничные частоты фильтра-прототипа равны (используются соотношения, приведенные выше)

$$\omega_p^p = 1; \quad \omega_s^p = 9,4721.$$

Итак, требуется НЧ-фильтр-прототип с $\omega_p^p = 1$; $\omega_s^p = 9,4721$; $A_p = 3$ дБ; $A_s = 20$ дБ.

Из уравнения (8.25) находим порядок фильтра-прототипа:

$$\begin{aligned}10^{A_s/10} - 1 &= 10^{20/10} - 1 = 99; & 10^{A_p/10} - 1 &= 10^{3/10} - 1 = 0,9952623; \\ \lg\left(\frac{99}{0,9952623}\right) &= 1,9976976; & \frac{\omega_s^p}{\omega_p^p} &= 9,4721; & 2\lg(9,4721) &= 1,95289 \\ N &= \frac{1,9976976}{1,95289} = 1,0229.\end{aligned}$$

Величина N должна быть целой, и для простоты используем значение $N = 1$. Передаточная функция на s -плоскости ФНЧ первого порядка записывается как

$$H(s) = \frac{1}{s + 1}.$$

Используя преобразование “фильтр нижних частот в режекторный фильтр” из таблицы, получаем

$$\begin{aligned} H'(s) &= H(s) \Big|_{s=\frac{Ws}{s^2+\omega_0^2}} = \frac{1}{\left(\frac{Ws}{s^2+\omega_0^2}\right) + 1} = \\ &= \frac{s^2 + \omega_0^2}{s^2 + Ws + \omega_0^2}. \end{aligned}$$

Результат применения билинейного z -преобразования таков:

$$\begin{aligned} H(z) &= H'(s) \Big|_{s=\frac{z-1}{z+1}} = \\ &= \frac{\left(\frac{z-1}{z+1}\right)^2 + \omega_0^2}{\left(\frac{z-1}{z+1}\right)^2 + W\left(\frac{z-1}{z+1}\right) + \omega_0^2}. \end{aligned}$$

Подставляя значения ω_0^2 и W , окончательно получаем

$$H(z) = \frac{0,2452(1 + z^{-2})}{1 - 0,5095z^{-2}}.$$

Пример 8.15

Получите передаточную функцию цифрового фильтра нижних частот, который удовлетворяет следующим спецификациям:

полоса пропускания	0–60 Гц,
полоса подавления	>85 Гц,
затухание в полосе подавления	>15 дБ.

Частоту дискретизации считайте равной 256 Гц и используйте характеристику Баттерворта.

Решение

На данном примере иллюстрируется объединение согласно формуле (8.23) этапов 4 и 5 процесса билинейного z -преобразования.

1. Критичные частоты цифрового фильтра:

$$\begin{aligned} \omega_1 T &= \frac{2\pi f_1}{F_s} = \frac{2\pi 60}{256} = 2\pi \times 0,2344, \\ \omega_2 T &= \frac{2\pi f_2}{F_s} = \frac{2\pi 85}{256} = 2\pi \times 0,3320. \end{aligned}$$

2. Деформированные аналоговые частоты:

$$\omega'_1 = \operatorname{tg} \left(\frac{\omega_1 T}{2} \right) = 0,906347; \quad \omega'_2 = \operatorname{tg} \left(\frac{\omega_2 T}{2} \right) = 1,71580.$$

3. Далее нужно получить $H(s)$ с характеристикой Баттерворта, частотой среза по уровню 3 дБ 0,906347 и откликом 85 Гц, уменьшенным на 15 дБ. При затухании 15 дБ и неравномерности в полосе пропускания 3 дБ из уравнения (8.25) получаем $N = 2,68$. Поскольку N должно быть целым, выбираем $N = 3$. Далее записываем нормированный фильтр третьего порядка:

$$\begin{aligned} H(s) &= \frac{1}{(s+1)(s^2+s+1)} = \frac{1}{s+1} \frac{1}{s^2+s+1} = \\ &= H_1(s)H_2(s). \\ \operatorname{ctg} \left(\frac{\omega_1 T}{2} \right) &= \operatorname{ctg} \left(\frac{2\pi \times 0,2344}{2} \right) = 1,103155. \end{aligned}$$

Выполняя преобразование в два этапа (отдельно для каждого сомножителя $H(s)$), находим

$$\begin{aligned} H_2(z) &= H_2(s) \Big|_{s=\operatorname{ctg}(\omega_1 T/2)[(z-1)/(z+1)]} = \\ &= 0,3012 \frac{1+2z^{-1}+z^{-2}}{1-0,1307z^{-1}+0,3355z^{-2}}, \end{aligned}$$

результат, которой мы уже получали ранее, правда, после значительных выкладок. Подобным образом вычисляем $H_1(z)$:

$$H_1(z) = 0,4754 \frac{1+z^{-1}}{1-0,0490z^{-1}}.$$

Далее функции $H_1(z)$ и $H_2(z)$ можно объединить и получить желаемую передаточную функцию $H(z)$:

$$H(z) = H_1(z)H_2(z) = 0,1432 \frac{1+3z^{-1}+3z^{-2}+z^{-3}}{1-0,1801z^{-1}+0,3419z^{-2}-0,0165z^{-3}}.$$

8.10. Расчет коэффициентов БИХ-фильтра путем отображения полюсов и нулей s -плоскости

8.10.1. Основные концепции

Альтернативным, а возможно, даже более мощным и гибким методом расчета коэффициентов $H(z)$ конкретного БИХ-фильтра является отображение отдельных полюсов и нулей подходящего аналогового фильтра с s - на z -плоскость с последующим выводом коэффициентов цифрового фильтра по нулям и полюсам на z -плоскости. Данный

подход использован в нескольких коммерческих программах и удобен при разработке фильтров высокого порядка.

Процедура вычисления коэффициентов БИХ-фильтра путем отображения полюсов и нулей с s -плоскости на z -плоскость описана ниже.

8.10.1.1. Этап 1

Как и ранее, разработчик начинает с нормированного аналогового фильтра нижних частот N -го порядка с характеристикой типа Баттерворта, Чебышева или эллиптической в зависимости от требований к разработке. Затем с помощью уравнения (8.26) для фильтра Баттерворта или уравнения (8.29) для фильтра Чебышева находят полюса нормированного фильтра нижних частот. Для эллиптического фильтра каждый полюс комплексный и в общем случае имеет вид

$$s_{l,k} = \alpha_{p,k} + i\beta_{p,k}. \quad (8.35)$$

Для фильтров Баттерворта и Чебышева (тип I) нули фильтра-прототипа находятся на бесконечности, а для эллиптического фильтра они полностью мнимые. Вообще, положения нулей нормированного фильтра нижних частот определить легче, чем положения полюсов.

8.10.1.2. Этап 2

Далее полюса и нули нормированного аналогового фильтра нижних частот преобразуются в полюса и нули фильтра нижних частот, верхних частот, полосового или режекторного фильтра, при этом используется подходящее преобразование из набора (8.21, a – e).

Фильтры нижних и верхних частот

Для цифрового фильтра нижних или верхних частот N полюсов нормированного фильтра нижних частот преобразуются следующим образом:

$$s_{l,k} = s_{1,k}/\omega'_p \quad k = 1, 2, \dots, N \quad (8.36, a)$$

(фильтр нижних частот в фильтр нижних частот),

$$s_{h,k} = \omega'_p/s_{l,k} \quad k = 1, 2, \dots, N \quad (8.36, б)$$

(фильтр нижних частот в фильтр верхних частот),

где ω'_p — желаемая граничная частота полосы пропускания, $s_{l,k}$ — полюса аналогового фильтра нижних частот, а $s_{h,k}$ — полюса аналогового фильтра верхних частот.

Сходство уравнений (8.36, a и $б$) очевидно, и оно объясняется дуальностью характеристик фильтров верхних частот и нижних частот. При N четном будет $N/2$ пар комплексных полюсов. При нечетном N будет $(N - 1)/2$ пар комплексных полюсов и один действительный полюс.

Для классических фильтров (Баттерворта, Чебышева и эллиптического) преобразования (8.36, a и $б$) переводят нули ФНЧ-прототипа на мнимую ось s -плоскости. В фильтрах Баттерворта и Чебышева нули фильтра-прототипа расположены на бесконечности.

В любом случае преобразование переводит нули с бесконечности на бесконечность (для фильтров нижних частот) или с бесконечности в начало координат (для фильтров верхних частот), как показано на рис. 8.17, *a(ii)* и 8.17, *б(ii)*.

Полосовые и режекторные фильтры

При разработке полосовых цифровых фильтров полюса аналогового полосового фильтра получаются из полюсов нормированного фильтра-прототипа нижних частот:

$$s_{l,k} = \frac{s_{b,k}^2 + \omega_0^2}{W s_{b,k}}, \quad (8.37)$$

где $s_{l,k}$ — полюса аналогового фильтра-прототипа нижних частот, $s_{b,k}$ — пары полюсов промежуточного аналогового полосового фильтра, $W = \omega'_2 - \omega'_1$ — ширина полосы пропускания фильтра, а $\omega_0^2 = \omega'_1 \omega'_2$ определяет центральную частоту полосы пропускания. Из уравнения (8.37) получаем следующее квадратное уравнение на $s_{b,k}$:

$$s_{b,k}^2 - W s_{l,k} s_{b,k} + \omega_0^2 = 0. \quad (8.38)$$

Решая его относительно $s_{b,k}$, вычисляем следующее выражение для полюсов аналогового полосового фильтра:

$$s_{b,k} = \frac{W}{1} \left[s_{l,k} \pm \left(s_{l,k}^2 - \frac{2\omega_0^2}{W^2} \right)^{1/2} \right]. \quad (8.39)$$

Из формулы (8.39) видно, что каждый полюс аналогового фильтра нижних частот дает пару полюсов аналогового полосового фильтра, что объясняется наличием члена s^2 в формуле преобразования. Вообще, полюс аналогового фильтра нижних частот $s_{l,k}$ комплексный, так что в результате действий с этой величиной также получится комплексное значение. Вычислять корень квадратный из комплексного числа следует аккуратно, если, конечно, нужно получить точное решение (см. приложение 8В).

Для режекторных цифровых фильтров можно использовать следующее преобразование:

$$s_{l,k} = \frac{W s_{r,k}}{s_{r,k}^2 + \omega_0^2}, \quad (8.40)$$

где $s_{l,k}$ — полюса аналогового прототипа, $s_{r,k}$ — полюса промежуточного аналогового режекторного фильтра, W — ширина полосы подавления, а ω_0^2 — центральная частота полосы подавления. Из уравнения (8.40) получаем следующее выражение для полюсов аналогового режекторного фильтра через полюса фильтра-прототипа.

$$s_{r,k} = \frac{W}{2} \left[s_{l,k}^{-1} \pm \left(s_{l,k}^{-2} - \frac{2\omega_0^2}{W^2} \right)^{1/2} \right]. \quad (8.41)$$

Не должно удивлять, что уравнение (8.41) по форме схоже с уравнением (8.39), исключая инверсию полюсов фильтра нижних частот вследствие дуальности полосового и режекторного фильтров.

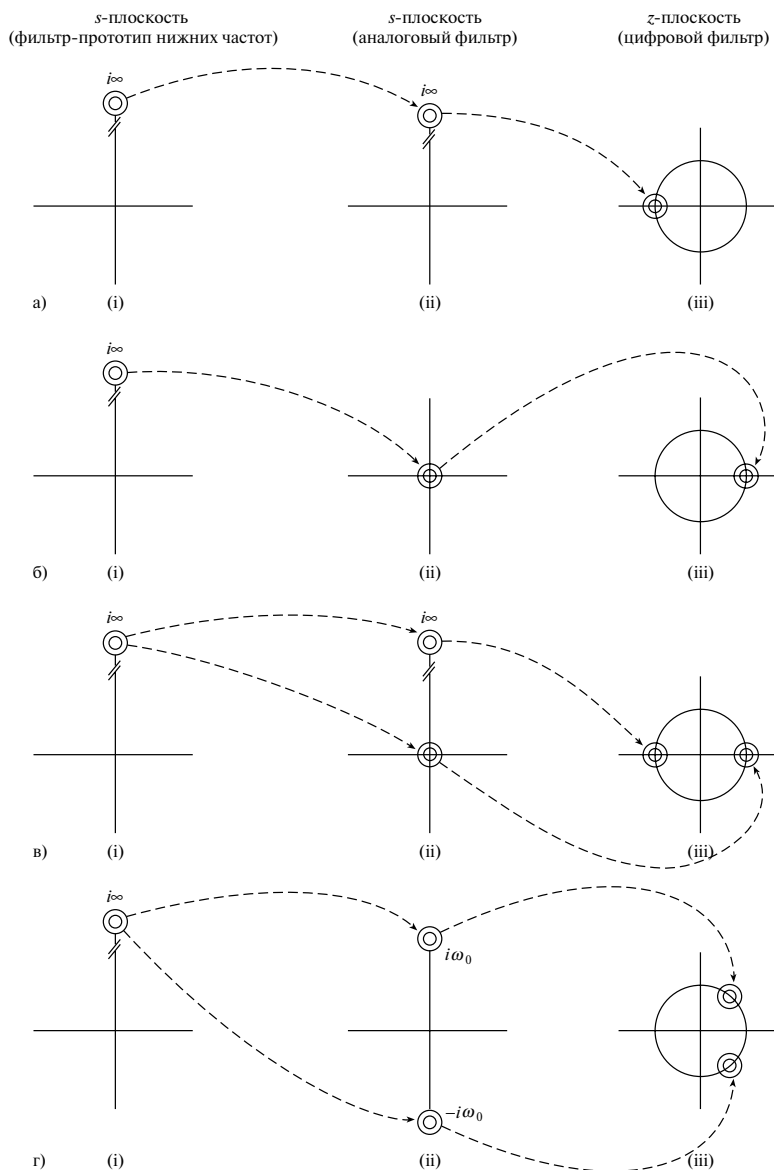


Рис. 8.17. Отображение нулей фильтра-прототипа нижних частот второго порядка в нули а) фильтра нижних частот; б) фильтра верхних частот; в) полосового фильтра; г) режекторного фильтра

Как и в ситуации с фильтрами нижних и верхних частот, преобразования (8.39) и (8.41) переводят нули на мнимую ось. Для фильтров Баттерворта и Чебышева преобразование “фильтр нижних частот в полосовой фильтр” отображает N нулей ФНЧ-прототипа с бесконечности на бесконечность и в начало координат s -плоскости (см. рис. 8.17, *в*). С другой стороны, преобразование “фильтр нижних частот в режекторный фильтр” переводит N нулей фильтра-прототипа бесконечности в точки $\pm i\omega_0$ s -плоскости; см. рис. 8.17, *г*. Для эллиптических полосовых и режекторных фильтров преобразование отображает полностью мнимые нули фильтра-прототипа в другие точки на ось $i\omega$.

8.10.1.3. Этап 3

После указанных действий применяется билинейное z -преобразование, переводящее полюса и нули с s -плоскости на цифровую z -плоскость. Каждый полюс $s_{p,k}$ s -плоскости отображается согласно следующему правилу:

$$z_{p,k} = \frac{1 + s_{p,k}}{1 - s_{p,k}}. \quad (8.42)$$

Подобным образом каждый нуль s -плоскости преобразованного аналогового фильтра следующим образом отображается на z -плоскость:

$$z_{z,k} = \frac{1 + s_{z,k}}{1 - s_{z,k}}. \quad (8.43)$$

Отображение нулей s -плоскости на z -плоскость с помощью билинейного z -преобразования иллюстрируется на рис. 8.17, *а–г* (фильтры Баттерворта и Чебышева). Отметим, в частности, что для фильтров нижних частот (рис. 8.17, *а(ii)*) нули располагаются на бесконечности s -плоскости, и билинейное z -преобразование переводит их в точку $z = -1$ z -плоскости (рис. 8.17, *а(iii)*), тогда как для полосового фильтра нули s -плоскости (рис. 8.17, *в(ii)*) отображаются с начала координат в точку $z = 1$, а с бесконечности — в точку $z = -1$.

Для эллиптических фильтров нижних частот, верхних частот, полосовых и режекторных $s_{z,k}$ — мнимые, и билинейное z -преобразование переводит эти точки на единичную окружность в z -плоскости. Вообще, нули z -плоскости для всех классических фильтров (Баттерворта, Чебышева и эллиптического), полученные методом билинейного z -преобразования, лежат на единичной окружности вне зависимости от типа фильтра. В результате коэффициенты числителя $H(z)$ классических фильтров всегда целые $(0, \pm 1, \pm 2)$.

8.10.1.4. Этап 4

Финальный этап — это определение коэффициентов числителя и знаменателя фильтрующих звеньев второго и первого порядка. Для этого следующим образом объединяются пары комплексно-сопряженных нулей:

$$H_i(z) = \frac{(z - z_{z,k})(z - z_{z,k}^*)}{(z - z_{p,k})(z - z_{p,k}^*)} = \quad (8.44, a)$$

$$= \frac{1 + b_{1i}z^{-1} + b_{2i}z^{-2}}{1 + a_{1i}z^{-1} + a_{2i}z^{-2}}. \quad (8.44, б)$$

Вообще, пара комплексных полюсов или нулей, расположенных в точках $\alpha \pm i\beta$, дает квадратный по z полином следующего вида:

$$\begin{aligned} [z - (\alpha + i\beta)][z - (\alpha - i\beta)] &= z^2 - 2\alpha z + \alpha^2 + \beta^2 = \\ &= 1 - 2\alpha z^{-1} + (\alpha^2 + \beta^2)z^{-2}. \end{aligned} \quad (8.45, а)$$

Отдельный полюс на действительной оси (т.е. $z = \pm\alpha$) дает множитель вида

$$z \pm \alpha = 1 \pm \alpha z^{-1}. \quad (8.45, б)$$

Довольно часто фильтр N -го порядка имеет N действительных корней на действительной оси в z -плоскости. В этом случае можно объединять в пары нули z -плоскости, чтобы числитель каждого звена фильтра описывался квадратным выражением вида

$$1 \pm 2z^{-1} + z^{-2}. \quad (8.45, в)$$

Общая передаточная функция $H(z)$ записывается как

$$H(z) = K H_1(z) H_2(z) \dots H_M(z),$$

где K — коэффициент усиления, который используется для подгонки амплитудной характеристики в полосе пропускания до нужного уровня. В большинстве случаев значение K выбирается так, чтобы максимальный отклик в полосе пропускания был равен единице.

8.10.2. Примеры

Пример 8.16

В определенном приложении ЦОС требуется цифровой полосовой фильтр второго порядка с характеристикой Баттерворта и полосой пропускания между 200 и 300 Гц с частотой дискретизации 2 кГц. Определите передаточную функцию цифрового фильтра путем отображения полюсов и/или нулей подходящего аналогового ФНЧ-прототипа с s -плоскости на z -плоскость.

Представьте с соответствующими пометками диаграмму нулей и полюсов прототипа, промежуточного аналогового полосового фильтра и цифрового полосового фильтра.

Решение

Поскольку порядок фильтра удваивается при преобразовании в полосовой фильтр, требуется найти нормированный фильтр нижних частот первого порядка. Следовательно,

$$H(s) = \frac{1}{s+1}.$$

Передаточная функция имеет единственный нуль в точке $s_{l,1} = -1$. Далее $F_s = 2 \text{ кГц} = 1/T$. Следовательно, деформированные граничные частоты имеют вид

$$\omega'_1 = \text{tg} \left(\frac{\omega_1 T}{2} \right) = \text{tg} \left(\frac{2\pi \times 200}{2 \times 2000} \right) = 0,3249,$$

$$\omega'_2 = \text{tg} \left(\frac{\omega_2 T}{2} \right) = \text{tg} \left(\frac{2\pi \times 300}{2 \times 2000} \right) = 0,5095.$$

Итак, находим ω_0 и W :

$$\omega_0^2 = \omega'_1 \omega'_2 = 0,1655, \quad W = \omega'_2 - \omega'_1 = 0,1846.$$

Согласно формуле (8.39) один полюс фильтра нижних частот преобразуется в два полюса полосового фильтра:

$$\begin{aligned} s_{b,1} &= \frac{0,1846}{2} \left\{ -1 + \left[(-1)^2 - \frac{4 \times 0,1655}{(0,1846)^2} \right]^{1/2} \right\} = \\ &= -0,0923 + 0,4172i; \\ s_{b,2} &= \frac{0,1846}{2} \left\{ -1 - \left[(-1)^2 - \frac{4 \times 0,1655}{(0,1846)^2} \right]^{1/2} \right\} = \\ &= -0,0923 - 0,4172i = s_{d,1}^*. \end{aligned}$$

Из билинейного z -преобразования (формула (8.42)) получаем

$$\begin{aligned} z_{p,1} &= \frac{1 - 0,0923 + 0,4172i}{1 + 0,0923 - 0,4172i} = 0,5979 + 0,6103i \\ z_{p,2} &= z_{p,1}^*. \end{aligned}$$

Фильтр-прототип имеет нуль на бесконечности, который при преобразовании “фильтр нижних частот в полосовой фильтр” отображается в начало координат и на бесконечность s -плоскости. Следовательно $s_{z,1} = 0$, $s_{z,2} = \infty$. Билинейное z -преобразование переводит эти нули в точки z -плоскости $z = 1$ и $z = -1$.

$$s_{z,1} \rightarrow z_{z,1} = 1; s_{z,2} \rightarrow z_{z,2} = -1.$$

Далее, зная полюса и нули, определяем дискретную передаточную функцию $H(z)$:

$$\begin{aligned} H(z) &= \frac{(z-1)(z+1)}{(z-z_{p,1})(z-z_{p,2})} = \\ &= \frac{z^2-1}{z^2-1,1958z+0,7995} = \frac{1-z^{-2}}{1-1,1958z^{-1}+0,7995z^{-2}}. \end{aligned}$$

Полученная передаточная функция идентична функции из примера 8.9 с точностью до постоянного множителя. Диаграммы нулей и полюсов также идентичны приведенным на рис. 8.11.

Пример 8.17

Начав с подходящего аналогового фильтра нижних частот, найдите передаточную функцию цифрового фильтра Чебышева верхних частот в факторизированном виде, который удовлетворяет следующим спецификациям:

граничная частота полосы пропускания	15 кГц,
затухание на частоте 18 кГц	>30 дБ,
неравномерность в полосе пропускания	1 дБ,
частота дискретизации	48 кГц.

Решение

Деформированные критические частоты:

$$\begin{aligned} \omega'_p &= \operatorname{tg} \left(\frac{15\,000\pi}{48\,000} \right) = 1,4966, \\ \omega'_s &= \operatorname{tg} \left(\frac{18\,000\pi}{48\,000} \right) = 2,4142. \end{aligned}$$

На основе спецификаций полосы пропускания находим $\epsilon = 0,3493$. Порядок подходящего фильтра Чебышева нижних частот равен 5 (ближайшее целое).

При $\alpha = 1/Nsh^{-1}(1/\epsilon) = 0,3548$, $sh(\alpha) = 0,3623$ и $csh(\alpha) = 1,0636$ полюса левой полуплоскости нормированного фильтра Чебышева нижних частот расположены в точках ($\omega'_p = 1$)

$$\begin{aligned} s_{l,1} &= 0,3623 \cos \left[\frac{(2+5-1)\pi}{10} \right] + 1,0636 \sin \left[\frac{(2+5-1)\pi}{10} \right] i = \\ &= -0,11196 + 1,0115i; \\ s_{l,2} &= 0,3623 \cos \left[\frac{(4+5-1)\pi}{10} \right] + 1,0636 \sin \left[\frac{(4+5-1)\pi}{10} \right] i = \\ &= -0,2931 + 0,6252i; \end{aligned}$$

$$\begin{aligned}
s_{l,3} &= 0,3623 \cos \left[\frac{(6+5-1)\pi}{10} \right] + 1,0636 \sin \left[\frac{(6+5-1)\pi}{10} \right] i = \\
&= -0,3623; \\
s_{l,4} &= 0,3623 \cos \left[\frac{(8+5-1)\pi}{10} \right] + 1,0636 \sin \left[\frac{(8+5-1)\pi}{10} \right] i = \\
&= -0,2931 - 0,6252i; \\
s_{l,5} &= 0,3623 \cos \left[\frac{(10+5-1)\pi}{10} \right] + 1,0636 \sin \left[\frac{(10+5-1)\pi}{10} \right] i = \\
&= -0,11196 - 1,0115i.
\end{aligned}$$

Из приведенных данных видна симметрия полюсов и то, что $s_{l,1}$ и $s_{l,5}$ (а также $s_{l,2}$ и $s_{l,4}$) формируют комплексно-сопряженную пару. Обратите также внимание на то, что все полюса лежат в левой полуплоскости s -плоскости, что является необходимым условием устойчивости.

Для преобразования полюсов прототипа в полюса нужного фильтра верхних частот используется уравнение (8.36, б):

$$\begin{aligned}
s_{h,1} &= -0,1618 - 1,4616i, \\
s_{h,2} &= -0,92013 - 1,9625i, \\
s_{h,3} &= -4,1306, \\
s_{h,4} &= s_{h,2}^* = -0,92013 + 1,9625i, \\
s_{h,5} &= s_{h,1}^* = -0,1618 + 1,4616i.
\end{aligned}$$

Далее полюса отображаются с s - на z -плоскость с помощью билинейного z -преобразования. Из уравнения (8.42) полюса z -плоскости после билинейного z -преобразования записываются так (рассматриваются только полюса над действительной осью):

$$\begin{aligned}
z_{h,1} &= -0,3335 + 0,8386i, \\
z_{h,2} &= -0,4906 + 0,5207i, \\
z_{h,3} &= -0,6102.
\end{aligned}$$

Все нули $z_{z,k}$ расположены в точке $z = 1$. Далее по полюсам и нулям вычисляются коэффициенты звеньев фильтра второго и первого порядка (уравнения (8.45, а и б)):

$$\begin{aligned}
b_{11} &= -2, & b_{12} &= -2, & b_{13} &= -1, \\
b_{21} &= 1, & b_{22} &= 1, & b_{23} &= 0, \\
a_{11} &= 0,6670, & a_{12} &= 0,9812, & a_{13} &= 0,6102, \\
a_{21} &= 0,8145, & a_{22} &= 0,5118, & a_{23} &= 0.
\end{aligned}$$

Наконец, передаточная функция записывается как

$$H(z) = KH_1(z)H_2(z)H_3(z),$$

где

$$H_1(z) = \frac{1 - 2z^{-1} + z^{-2}}{1 + 0,6670z^{-1} + 0,8145z^{-2}},$$

$$H_2(z) = \frac{1 - 2z^{-1} + z^{-2}}{1 + 0,9812z^{-1} + 0,5118z^{-2}},$$

$$H_3(z) = \frac{1 - z^{-1}}{1 + 0,6102z^{-1}}.$$

8.11. Использование программ разработки БИХ-фильтров

Какой бы метод мы не выбрали, очевидно, что метод билинейного z -преобразования включает значительные алгебраические манипуляции со множеством потенциальных ошибок. В настоящее время существуют компьютерные программы вычисления коэффициентов фильтра с помощью билинейного метода, в которых требуется только задать необходимые параметры [7, 10, 11, 14, 18]. Большинство компьютерных программ в литературе написаны на языке FORTRAN, хотя в настоящее время от этого языка отказываются в пользу более современных C и BASIC. Программу на языке C вычисления коэффициентов фильтра с помощью билинейного z -преобразования можно, например, найти в книге [Ifeachor, 2001]. Использование этой программы иллюстрируется на следующем примере. В приложении 8Б показано, как для разработки цифровых БИХ-фильтров можно использовать MATLAB.

Пример 8.18

Иллюстрация использования программы разработки БИХ-фильтров. Вычислите коэффициенты цифрового аудиофильтра с характеристикой Чебышева, удовлетворяющего следующим спецификациям:

полоса пропускания	0–2,5 кГц,
граничная частота полосы подавления	2820 кГц,
неравномерность в полосе пропускания	0,47 дБ,
частота дискретизации	10 кГц,
порядок фильтра	4.

Решение

Используя программу, получаем следующий выход:

k	A_k	B_k
0	$1,000\,000 \times 10^0$	$1,934\,410 \times 10^{-1}$
1	$-2,516\,884 \times 10^{-1}$	$3,783\,311 \times 10^{-1}$
2	$1,054\,118 \times 10^0$	$5,241\,429 \times 10^{-1}$
3	$-2,406\,030 \times 10^{-1}$	$3,783\,311 \times 10^{-1}$
4	$1,985\,861 \times 10^{-1}$	$1,934\,410 \times 10^{-1}$

8.12. Выбор метода расчета коэффициентов БИХ-фильтров

При использовании метода инвариантного преобразования импульсной характеристики после оцифровки аналогового фильтра импульсная характеристика исходного аналогового фильтра сохраняется, а амплитудно-частотная характеристика — нет. Вследствие присущего методу наложения он не подходит для разработки фильтров верхних частот или режекторных фильтров. Метод билинейного z -преобразования, с другой стороны, дает весьма эффективные фильтры и прекрасно подходит для расчета коэффициентов частотно-избирательных фильтров. Он позволяет разрабатывать цифровые фильтры с такими широко известными классическими характеристиками, как Баттерворта, Чебышева и эллиптические. Цифровые фильтры, полученные с помощью метода билинейного z -преобразования, будут, в общем случае, сохранять специфические особенности амплитудной характеристики аналогового фильтра (например, граничные частоты, неравномерность в полосе пропускания и затухание в полосе подавления), но не свойства, связанные с временной областью. Метод инвариантного преобразования импульсной характеристики хорош для моделирования аналоговых систем с характеристиками фильтров нижних частот, но для частотно-избирательных БИХ-фильтров наилучшим является билинейный метод. Согласованное z -преобразование порождает большинство проблем, присущих методу инвариантного преобразования импульсной характеристики. Если поставленная задача проста, то простым и эффективным методом расчета коэффициентов является метод размещения нулей и полюсов.

8.12.1. Эффект Найквиста

Три метода преобразования аналоговых фильтров в эквивалентные фильтры дискретного времени (а именно, согласованное z -преобразование, метод инвариантного преобразования импульсной характеристики и билинейное z -преобразование) могут в некоторых случаях значительно влиять на характеристики фильтра (например, амплитудную характеристику, фазовую и групповую задержки и т.д.). Как отмечалось ранее, доступная полоса частот аналогового фильтра простирается от нуля до бесконечности, тогда как для цифрового фильтра — от нуля до частоты Найквиста (половина частоты дискретизации). Следовательно, амплитудно-частотная характеристика цифровых

фильтров, разработанных с помощью любого из названных методов, может значительно отличаться от характеристики аналогового фильтра, поскольку теперь аналоговая полоса частот (от нуля до бесконечности) сжата до узкой полосы (от нуля до частоты Найквиста). Данное отличие характеристик представляет искажение, которое иногда называют *эффектом Найквиста*.

Во многих приложениях эффект Найквиста не вреден, просто он вызывает определенное дополнительное искажение. Впрочем, если в некоторой ситуации следует сохранить характеристику аналогового фильтра, например, при профессиональной или полупрофессиональной работе с аудио [5], данный эффект все же представляет нежелательное искажение. В таких случаях при выборе метода преобразования аналогового фильтра в эквивалентный фильтр дискретного времени может учитываться степень искажения. При выборе метода также следует учитывать влияние метода на такие характеристики фильтра, как групповая задержка и импульсная характеристика [19].

В данном разделе кратко рассматриваются последствия эффекта Найквиста. В конце главы приведено несколько задач, позволяющих читателю изучить относительные преимущества методов преобразования аналоговых фильтров в эквивалентные цифровые.

Пример 8.19

Требуется фильтр нижних частот дискретного времени с характеристикой Баттерворта согласно следующей спецификации:

частота среза	300 Гц,
порядок фильтра	5,
частота дискретизации	1000 Гц.

1. С помощью MATLAB получите и изобразите следующие характеристики:
 - а) амплитудно-частотную характеристику и характеристику групповой задержки (используйте метод инвариантного преобразования импульсной характеристики);
 - б) амплитудно-частотную характеристику и характеристику групповой задержки (используйте метод билинейного z -преобразования).
2. Сравните два метода с позиции искажения амплитудной характеристики вследствие эффекта Найквиста.

Решение

- 1, а. Начнем с разработки аналогового фильтра, который будет служить эталоном.

Расчет аналогового фильтра производится с помощью программы 8.1. Полученная амплитудно-частотная характеристика показана на рис. 8.18.

Амплитудно-частотная характеристика и характеристика групповой задержки эквивалентного дискретного фильтра показаны на рис. 8.19, а и б. Для расчета этого фильтра использовалась программа 8.2. Для получения характеристики групповой задержки использовалась команда `grpdelay` из MATLAB Signal Processing Toolbox.

- 1, б. Амплитудно-частотная характеристика и групповая задержка эквивалентного дискретного фильтра, разработанного с помощью метода билинейного z -преобразования, показаны на рис. 8.20, а и б. Для расчета этого фильтра использовалась программа 8.3.
2. Если сравнить амплитудно-частотные характеристики фильтра, полученного методом инвариантного преобразования импульсной характеристики, и фильтра, полученного с помощью билинейного z -преобразования, очевидно, что первый фильтр дает меньшее, а второй — большее затухание возле частоты Найквиста. Например, на частоте 500 Гц фильтр, полученный методом инвариантного преобразования импульсной характеристики, дает затухание порядка 17 дБ, тогда как фильтр, полученный методом билинейного z -преобразования, имеет затухание более 200 дБ.

Программа 8.1. Программа разработки аналогового фильтра

```
%
% Название программы: EX8-1.m
%
FN=1000/2;
fc=300; % Частота среза
N=5; % Порядок фильтра
[z, p, k]=butter(N); % Создать аналоговый фильтр
w=linspace(0, FN/fc, 1000); % Вывести на экран характеристику фильтра
h=freqs(k*poly(z), poly(p), w); f=fc*w; plot(f, 20*log10(abs(h))), grid
ylabel('Амплитудная характеристика (дБ)')
xlabel('Частота (Гц)')
```

Программа 8.2. Программа разработки фильтра с помощью метода инвариантного преобразования импульсной характеристики

```
%
% Название программы: EX8-2.m
%
Fs=1000; % Частота дискретизации
fc=300; % Частота среза
WC=2*pi*fc; % Частота среза в радианах
N=5;
[b,a]=butter(N, WC, 's'); % Создать аналоговый фильтр
[z,p,k]=butter(N, WC, 's');
[bz, az]=impinvar(b, a, Fs); % Определить коэффициенты БИХ-фильтра
[h, f]=freqz(bz, az, 512, Fs);
plot(f, 20*log10(abs(h))), grid
ylabel('Амплитудная характеристика (дБ)')
xlabel('Частота (Гц)')
```

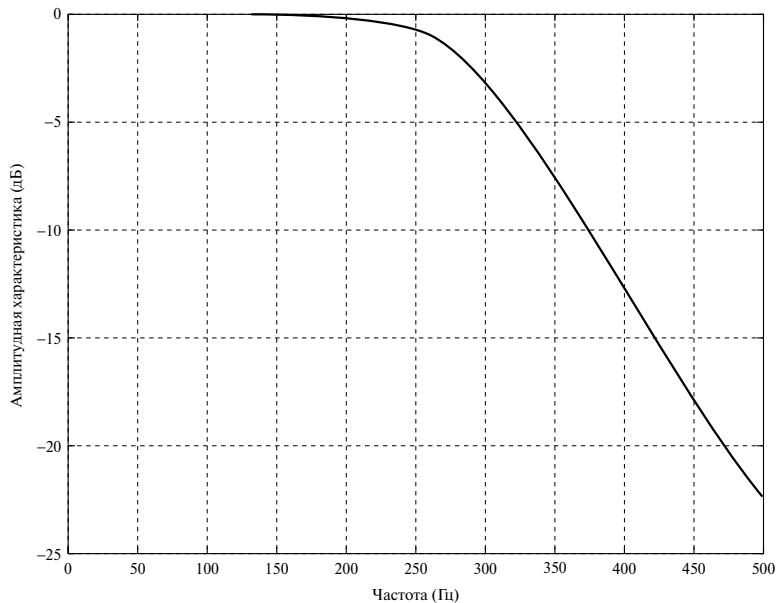


Рис. 8.18. Амплитудная характеристика аналогового фильтра

Программа 8.3. Программа разработки фильтра с помощью билинейного z -преобразования

```
%
% Название программы: EX8-3.m
%
Fs1000;    % Частота дискретизации
FN=Fs/2;
fc=300;    % Частота среза
N=5;
[z, p, k]=butter(N, fc/FN);
[h, f]=freqz(k*poly(z), poly(p), 512, Fs);
plot(f, 20*log10(abs(h))), grid
ylabel('Амплитудная характеристика (дБ)')
xlabel('Частота (Гц)')
```

Анализ фильтра, полученного с помощью билинейного z -преобразования, показывает, что передаточная функция имеет пять нулей на частоте Найквиста (см. рис. 8.20, в), которые отвечают за быстрый спад амплитудной характеристики. Такая ситуация характерна для дискретных фильтров, полученных с помощью билинейного z -преобразования, и это является причиной, по которой данные цифровые фильтры дают большее затухание, чем исходные аналоговые фильтры. Дискретный фильтр, полученный методом инвариантного преобразования импульсной характеристики, наоборот, имеет нуль в начале координат и нуль вне единичной окружности. Получившееся искажение представляет отличие амплитудных характеристик двух дискретных и аналогового фильтров.

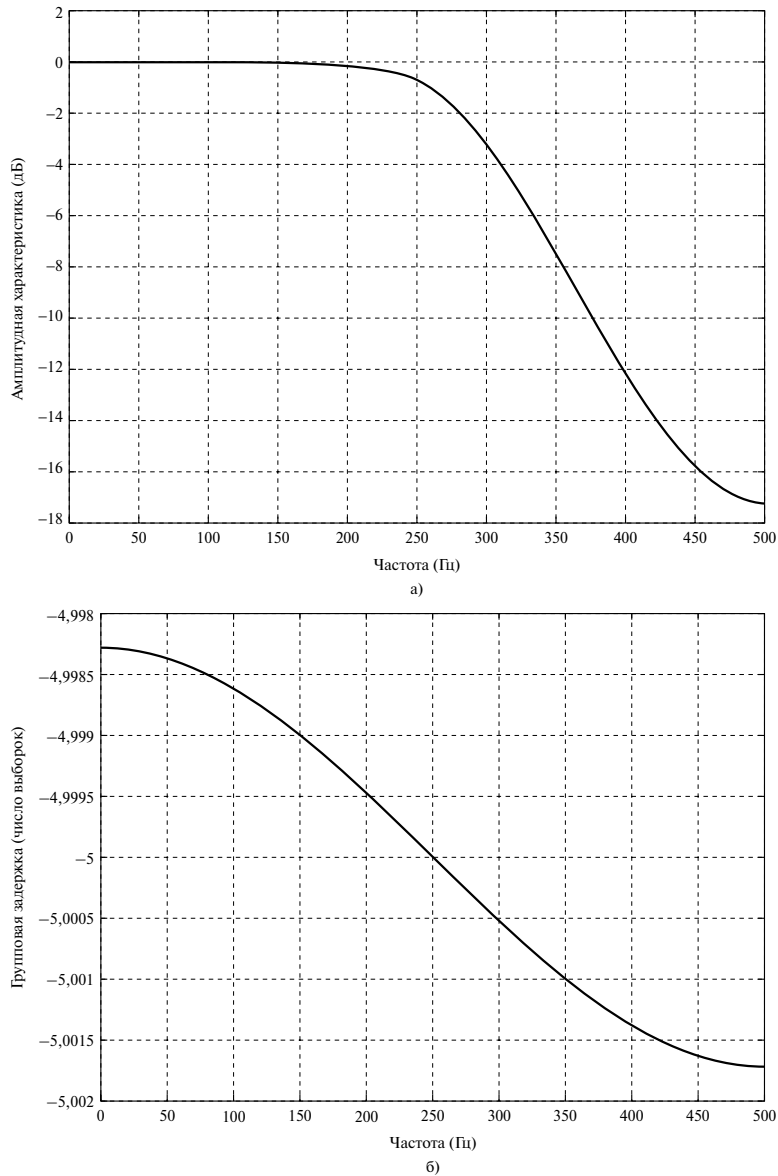


Рис. 8.19. Амплитудно-частотная характеристика инвариантного преобразования импульсной характеристики (панель *а*); групповая задержка фильтра, полученного методом инвариантного преобразования импульсной характеристики (панель *б*)

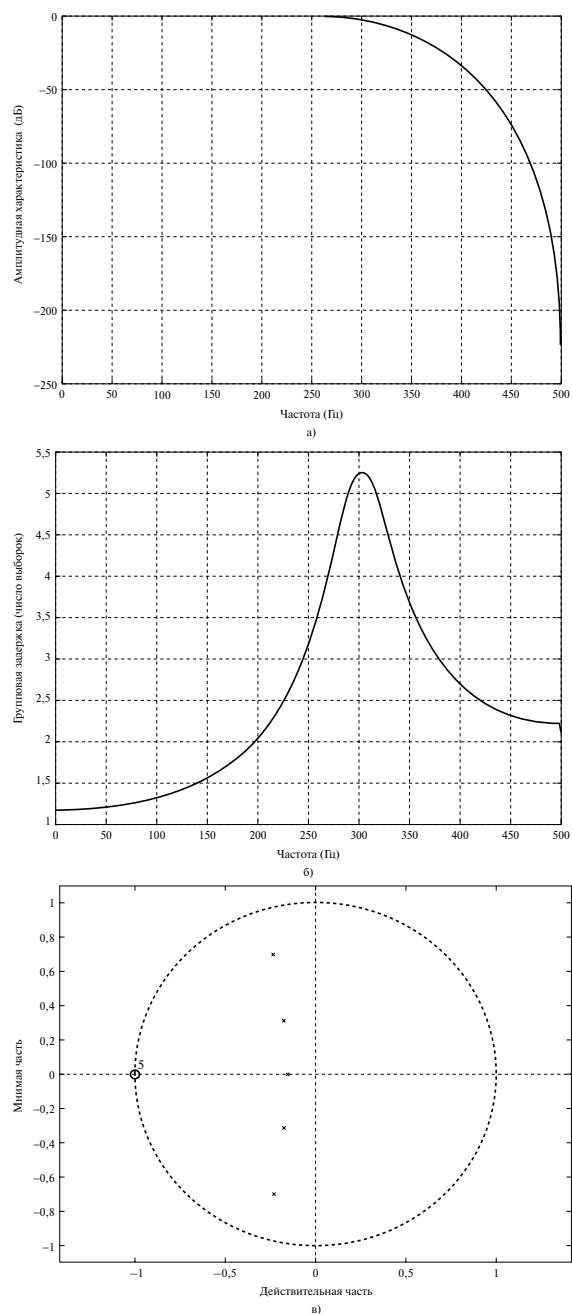


Рис. 8.20. Амплитудно-частотная характеристика билинейного z -преобразования (панель а); групповая задержка фильтра, полученного методом билинейного z -преобразования (панель б); диаграмма нулей и полюсов фильтра, полученного методом билинейного z -преобразования (панель в)

Во многих случаях искажения амплитудной характеристики фильтров, полученных методом согласованного z -преобразования или инвариантного преобразования импульсной характеристики, и фильтров, полученных методом билинейного z -преобразования, противоположны по знаку. Первая группа фильтров дает меньшее затухание, чем вторая. При низких и средних частотах (относительно частоты Найквиста) амплитудные характеристики фильтров нижних частот и верхних частот, разработанных с помощью согласованного z -преобразования и метода инвариантного преобразования импульсной характеристики, разумно близки к характеристикам исходных аналоговых фильтров, но искажаются возле частоты Найквиста. То же справедливо и для фильтров, разработанных с помощью билинейного z -преобразования, только искажение характеристики ближе к частоте Найквиста происходит в противоположном смысле. Для простого и эффективного подавления искажения характеристики фильтров нижних и верхних частот можно объединить билинейное z -преобразование с согласованным z -преобразованием либо методом инвариантного преобразования импульсной характеристики, усреднив их коэффициенты (см. [4, 5]). При таком подходе фильтр нижних частот или полосовой фильтр вначале разрабатывается отдельно с помощью билинейного z -преобразования и согласованного z -преобразования. Затем коэффициенты этих двух фильтров объединяются для получения коэффициентов нового фильтра:

$$\begin{aligned} b'_k &= [b_k(BZT) + b_k(MZT)] / 2, \quad k = 0, 1, 2, \\ a'_k &= [a_k(BZT) + a_k(MZT)] / 2, \quad k = 0, 1, 2. \end{aligned}$$

На рис. 8.21 сравниваются характеристики двух дискретных фильтров (билинейное z -преобразование и согласованное z -преобразование). Данные фильтры предназначены для обработки аудиосигналов, причем требуется эффективно и оперативно генерировать коэффициенты фильтра при слабом искажении характеристики. Как и ранее, фильтр, основанный на билинейном z -преобразовании, дает более резкую амплитудно-частотную характеристику за частотой 10 кГц (частота Найквиста — 24 кГц), тогда как согласованное z -преобразование дает меньшее затухание. Оба отклонения амплитудной характеристики представляют в данной задаче искажение. Усредненная характеристика (рис. 8.21), с другой стороны, менее искажена. На диаграмме нулей и полюсов на рис. 8.22 сравниваются положения полюсов и нулей фильтра, основанного на согласованном z -преобразовании, и фильтра, основанного на комбинированном согласованном z -преобразовании и билинейном z -преобразовании. Как и следовало ожидать, процесс усреднения приводит к чередованию полюсов и нулей фильтра, построенного на согласованном z -преобразовании. В частности, он вводит “мягкий” нуль на частоте Найквиста. Нуль называется мягким, поскольку он расположен глубоко внутри единичной окружности и поэтому не дает амплитудной характеристике быстро снизиться до нуля, как при билинейном z -преобразовании.

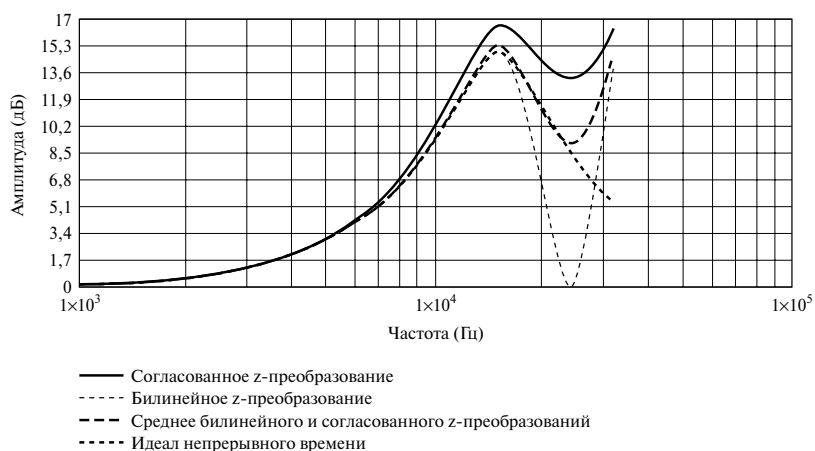


Рис. 8.21. Амплитудная характеристика фильтров, полученных методами согласованного z -преобразования, билинейного z -преобразования, усредненного согласованного/билинейного z -преобразования и аналогового эталона, 15 кГц, $Q = 2$, усиление 15 дБ

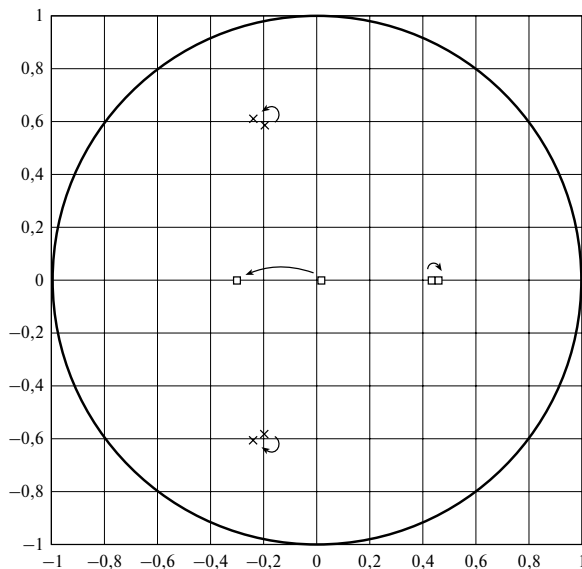


Рис. 8.22. Сравнение графиков нулей и полюсов фильтров, полученных путем согласованного z -преобразования и усредненного согласованного/билинейного z -преобразования. Стрелочками указано движение полюсов/нулей при переходе от первого фильтра ко второму

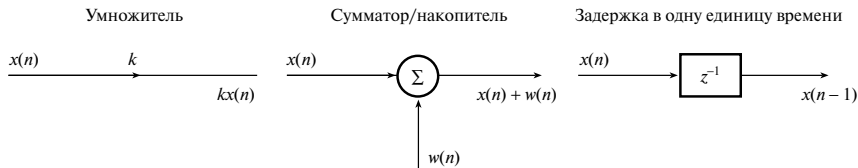


Рис. 8.23. Основные элементы структуры фильтра

8.13. Структуры реализации цифровых БИХ-фильтров

Реализация включает преобразование данной передаточной функции $H(z)$ в подходящую фильтрующую структуру. Для отображения структуры фильтра обычно используются блок-схемы или функциональные схемы, на которых показана вычислительная процедура, которая будет воплощена в цифровом фильтре. Основными элементами структуры реализации фильтров являются умножители, сумматоры и элементы задержки (рис. 8.23).

Напомним, что БИХ-фильтр характеризуется следующими уравнениями:

$$H(z) = \sum_{k=0}^N b_k z^{-k} / \left(1 + \sum_{k=1}^M a_k z^{-k} \right) \quad M \geq N, \quad (8.46, a)$$

$$y(n) = \sum_{k=0}^{N-1} b_k x(n-k) - \sum_{k=1}^M a_k y(n-k). \quad (8.46, б)$$

Реализация уравнения (8.46) в прямой форме приведена на рис. 8.24, где для простоты $N = M$. Отметим, что коэффициенты, использованные на диаграмме, аналогичны коэффициентам передаточной функции, но коэффициенты знаменателя имеют противоположный знак. Если фильтр имеет большой порядок, например, $M > 3$, прямая реализация фильтра (как на рис. 8.24) очень чувствительна к эффектам конечной разрядности, и в таких случаях ее следует избегать. На практике $H(z)$ обычно разбивается на меньшие фрагменты, обычно звенья второго и/или первого порядка, которые затем соединяются каскадом или параллельно.

8.13.1. Практические стандартные блоки БИХ-фильтров

Примеры практических стандартных блоков второго порядка, используемых при реализации БИХ-фильтров высоких порядков, изображены на рис. 8.25. Первый блок (рис. 8.25, а) часто называется *каноническим звеном* (или прямой формой 2), поскольку он имеет минимальное число элементов задержки. Данное биквадратное звено характеризуется следующими уравнениями:

$$w(n) = x(n) - \sum_{k=1}^2 a_k w(n-k), \quad (8.47, a)$$

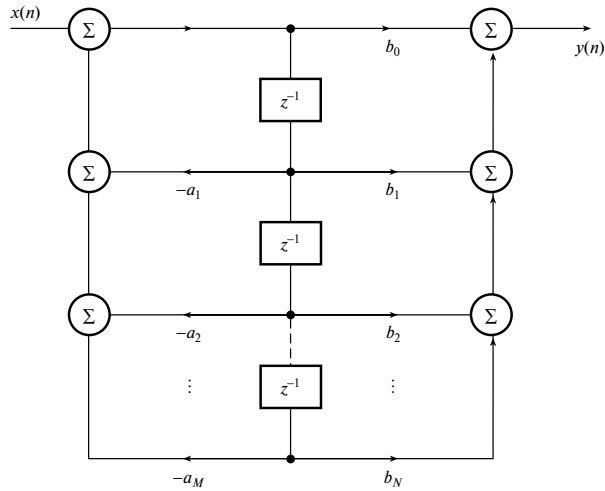


Рис. 8.24. Реализация БИХ-фильтра в прямой форме

$$y(n) = \sum_{k=0}^2 b_k w(n-k), \quad (8.47, б)$$

$$H(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}}. \quad (8.47, в)$$

Второе звено фильтра (рис. 8.25, б) называется *прямой реализацией* БИХ-уравнения второго порядка. Оно характеризуется следующими уравнениями:

$$y(n) = \sum_{k=0}^2 b_k x(n-k) - \sum_{k=1}^2 a_2 y(n-k), \quad (8.48, а)$$

$$H(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}}. \quad (8.48, б)$$

Каноническое звено (рис. 8.25, а) более популярно, поскольку сигнал в нем незначительно искажается шумом округления, а само звено требует минимального числа элементов памяти, хотя и подвержено внутреннему переполнению. Чтобы избежать этого переполнения, вход звена необходимо масштабировать. Для прямой формы (см. рис. 8.25, б) масштабирование не обязательно, поскольку это звено имеет всего один сумматор и может быть более выгодным в реализациях, когда масштабирование нежелательно, как, например, в системах качественного цифрового аудио [6]. Кроме того, при определенных условиях прямая форма лучше канонической с точки зрения шумовых характеристик.

Связанная форма обладает некоторыми преимуществами, касающимися эффектов конечной разрядности [9], но требует больше вычислительных усилий и ее трудно использовать при воплощении передаточных функций, коэффициенты числителей которых имеют второй порядок.

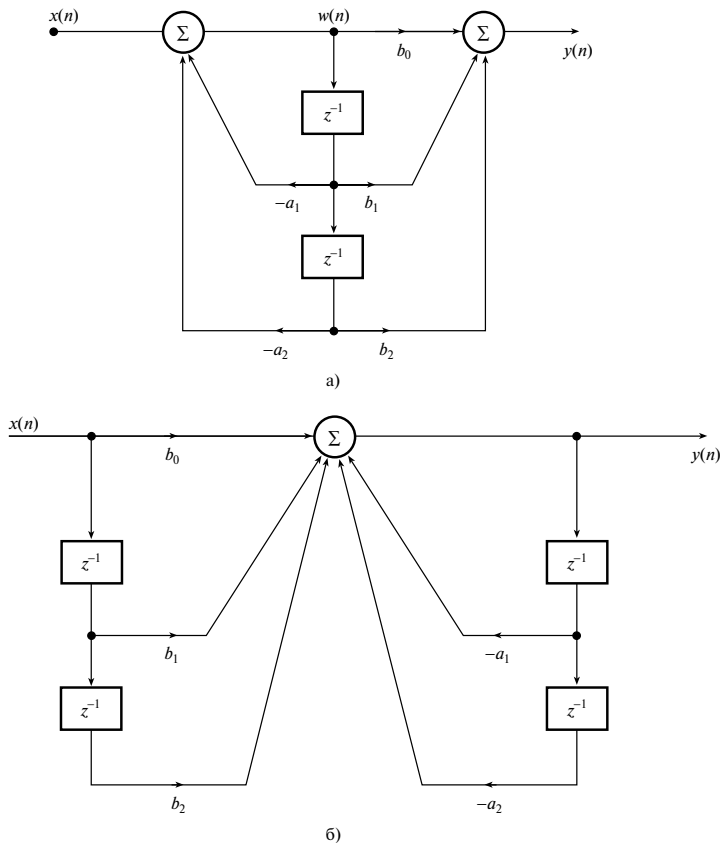


Рис. 8.25. Стандартные блоки структуры реализации БИХ-фильтров: а) каноническое звено второго порядка; б) звено второго порядка в прямой форме

Фильтрующие блоки, представленные на рис. 8.25, — это общие звенья второго порядка. Из них можно вывести несколько других блоков. Например, если коэффициенты числителя a_1 и a_2 на рис. 8.25, а равны нулю, имеем чисто рекурсивную структуру. С другой стороны, если для вычисления коэффициентов фильтра использовались эллиптические функции, коэффициент a_2 равен единице. Кроме того, в любой из приведенных выше структур можно положить $a_s = b_2 = 0$ и получить звено первого порядка.

На рис. 8.26 показаны переупорядоченные каноническое звено второго порядка и звено второго порядка в прямой реализации. Рисунки получены из рис. 8.25, а и б соответственно — переставлены сумматоры и соединительные узлы и обращены направления стрелок. Хотя передаточные функции звеньев на рис. 8.26 от перестановки не изменились, свойства, касающиеся эффектов конечной разрядности, изменились значительно. Вообще, существуют и другие структуры, менее чувствительные к эффектам конечной длины слова, но обычно они очень сложны. В качестве примеров таких структур можно назвать блоки минимального шума, блоки с изменяемыми параметрами и решетчатую структуру.

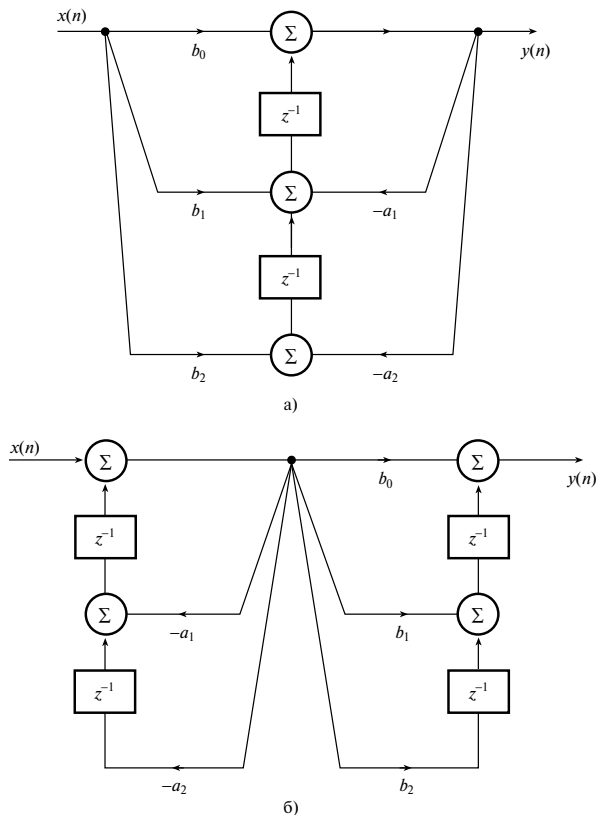


Рис. 8.26. Переупорядоченное каноническое звено второго порядка (панель а); переупорядоченное звено второго порядка в прямой реализации (панель б)

8.13.2. Каскадная и параллельная реализации БИХ-фильтров больших порядков

На практике передаточные функции больших порядков реализуются как каскады или параллельные комбинации описанных выше стандартных блоков второго и/или первого порядка. Обычно каскадная реализация передаточной функции факторизуется в $N/2$ множителей второго порядка:

$$\begin{aligned}
 H(z) &= \prod_{k=1}^{N/2} \left[\frac{b_{0k} + b_{1k}z^{-1} + b_{2k}z^{-2}}{1 + a_{1k}z^{-1} + a_{2k}z^{-2}} \right] = \\
 &= \prod_{k=1}^{N/2} \frac{N_k(z)}{D_k(z)},
 \end{aligned}
 \tag{8.49, а}$$

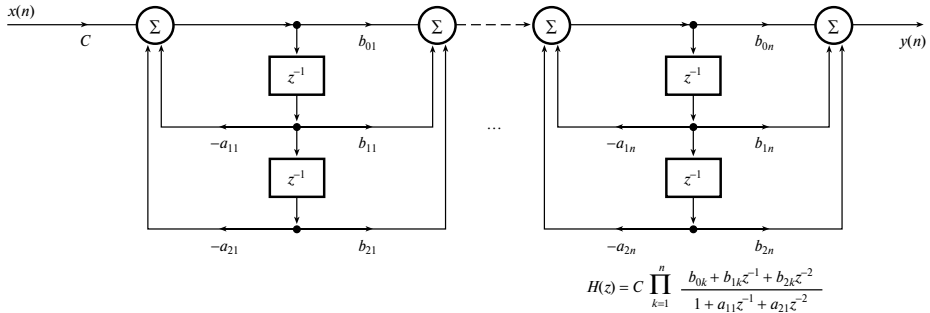


Рис. 8.27. Каскадная реализация

где

$$\begin{aligned} N_k(z) &= b_{0k} + b_{1k}z^{-1} + b_{2k}z^{-2} \\ D_k(z) &= 1 + a_{1k}z^{-1} + a_{2k}z^{-2} \end{aligned}, \quad (8.49, б)$$

а порядок фильтра N предполагается четным. Если N нечетное, то одна из функций $H_k(z)$ будет описывать звено первого порядка.

Каждый сомножитель $H_k(z)$ можно реализовать с использованием одного из стандартных блоков, а затем полученные блоки соединить каскадом (рис. 8.27). При каскадной реализации возникают три проблемы: 1) как образовать пары “множитель числителя-множитель знаменателя”; 2) в каком порядке объединять отдельные звенья; 3) как масштабировать уровни сигналов в различных внутренних точках фильтра, чтобы уровни не стали слишком низкими или высокими.

Упорядочить множители числителя и знаменателя можно по-разному. Например, фильтр четвертого порядка можно факторизовать в два звена второго порядка, а затем упорядочить их одним из четырех различных способов:

$$\begin{aligned} 1) \quad H(z) &= \frac{N_1 z}{D_1(z)} \frac{N_2 z}{D_2(z)}, \\ 2) \quad H(z) &= \frac{N_2 z}{D_2(z)} \frac{N_1 z}{D_1(z)}, \\ 3) \quad H(z) &= \frac{N_1 z}{D_2(z)} \frac{N_2 z}{D_1(z)}, \\ 4) \quad H(z) &= \frac{N_2 z}{D_1(z)} \frac{N_1 z}{D_2(z)}, \end{aligned}$$

причем каждое $N_k(z)$ и $D_k(z)$ — это полином второго порядка, определенный в уравнении (8.49, б). В первом случае первое звено фильтра состоит из пары числитель-знаменатель $N_1(z)$ и $D_1(z)$, а второе звено — из пары $N_2(z)$ и $D_2(z)$. Очевидно, что

число возможных способов разбиения на пары достаточно велико. Как правило, для фильтра N -го порядка число различных разбиений равно

$$\left(\frac{N}{2}\right)!^2. \quad (8.50)$$

Чтобы избежать появления большой амплитудной характеристики на частоте, соответствующей полюсу, применяется следующее эмпирическое правило: объединять в пару $N_i(z)$ с $D_k(z)$, если нули $N_i(z)$ близки к полюсам $D_k(z)$, и располагать в конце каскада звенья второго порядка с полюсами, наиболее близкими к единичной окружности [12]. Кроме того, разработано немало эффективных схем организации пар и упорядочения звеньев фильтра, основанных на том, какое упорядочение дает наилучшее отношение сигнал-шум (см. главу 13).

При параллельной реализации передаточная функция N -го порядка $H(z)$ раскладывается в ряд элементарных дробей:

$$H(z) = C + \sum_{k=1}^{N/2} H_k(z), \quad (8.51)$$

где

$$C = \frac{b_N}{a_N}, \quad H_k(z) = \frac{b_{0k} + b_{1k}z^{-1}}{1 + a_{1k}z^{-1} + a_{2k}z^{-2}}.$$

Как и ранее, звено второго порядка можно реализовать с использованием описанных ранее стандартных блоков, как показано на рис. 8.28. Стоит отметить, что в параллельной реализации коэффициент числителя при z^{-2} равен нулю. В параллельной структуре порядок соединения звеньев не важен. Более того, масштабирование выполняется легче и его можно провести независимо для каждого блока (см. далее), а отношения сигнал-шум сравнимы с наилучшими значениями для каскадной реализации [12]. В то же время нули параллельных структур более чувствительны к ошибкам квантования коэффициентов. Следует также отметить, что чувствительность нулей параллельной структуры к квантованию коэффициентов кажется более серьезной, когда длина коэффициентов уменьшается до 5 бит или меньше. Похоже, что для длин коэффициентов от 12 бит отличие параллельной и каскадной структур для большинства фильтров менее значительно. В то же время, важным преимуществом каскадного метода является то, что примерно от 25 до 50% коэффициентов фильтра являются простыми целыми числами (0 , ± 1 или ± 2), если фильтр выведен из классических аналоговых фильтров посредством билинейного z -преобразования. Эта особенность привлекательна в системах, имеющих только примитивные арифметические возможности, где число умножений должно быть малым. Отметим также, что большинство доступных программных пакетов рассчитывают коэффициенты для каскадной реализации, но не для параллельной структуры. По этим причинам каскадный метод в настоящее время более популярен.

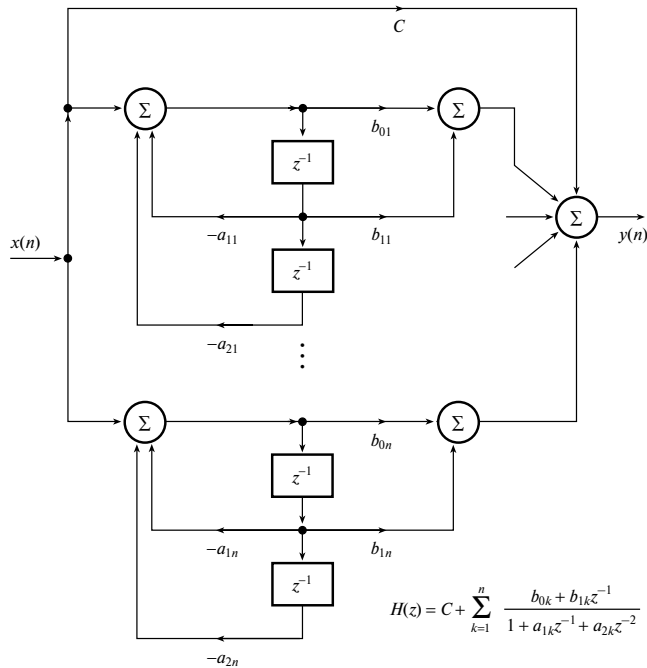


Рис. 8.28. Параллельная реализация

Пример 8.20

Используя звенья второго и первого порядка, перепишите представленную ниже передаточную функцию для каскадной (1) и параллельной (2) реализации фильтра

$$H(z) = \frac{0,1432(1 + 3z^{-1} + 3z^{-2} + z^{-3})}{1 - 0,1801z^{-1} + 0,3419z^{-2} - 0,0165z^{-3}}.$$

Решение

1. Для каскадной реализации $H(z)$ выражается в факторизованном виде:

$$H(z) = 0,1432 \frac{1 + 2z^{-1} + z^{-2}}{1 - 0,1307z^{-1} + 0,3355z^{-2}} \frac{1 + z^{-1}}{1 - 0,0490z^{-1}}.$$

2. Для параллельной реализации $H(z)$ выражается как сумма звеньев второго и первого порядков:

$$H(z) = \frac{1,2916 - 0,08407z^{-1}}{1 - 0,131z^{-1} + 0,3355z^{-2}} + \frac{10,1764}{1 - 0,049z^{-1}} - 8,7107.$$

Диаграммы реализации каскадного и параллельного представлений изображены на рис. 8.29, *a* и *б* соответственно. Коэффициенты параллельной реализации получены с помощью программы на языке C, приведенной в главе 4.

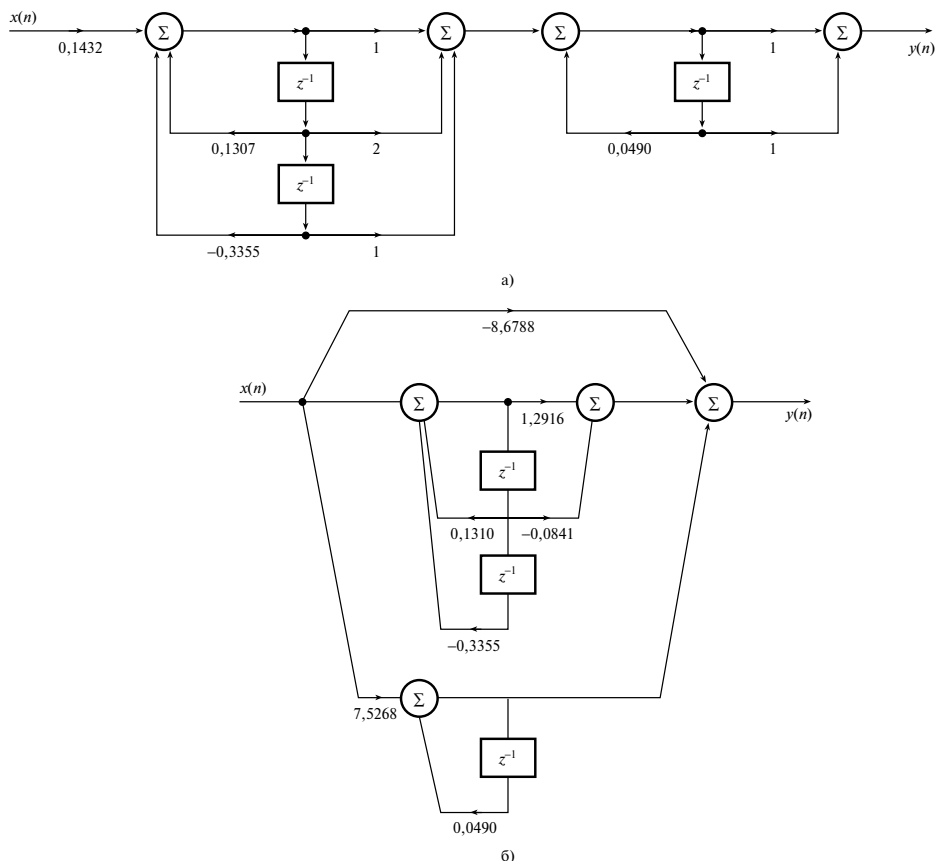


Рис. 8.29. Реализации фильтра из примера 8.20: а) каскадная; б) параллельная

8.14. Влияние конечной разрядности на БИХ-фильтры

Полученные ранее (см. разделы 8.4–8.10) коэффициенты a_k и b_k имеют бесконечную или очень высокую точность, обычно 6–7 десятичных разрядов. Когда же цифровой БИХ-фильтр реализуется в такой малой системе, как 8-битовый микрокомпьютер, при представлении коэффициентов и выполнении арифметических операций, указанных в разностном уравнении, появляются ошибки. Данные ошибки отрицательно сказываются на производительности фильтра и в крайних случаях приводят к его неустойчивости.

Перед воплощением БИХ-фильтра в программной или аппаратной форме важно выяснить степень ухудшения производительности за счет эффектов конечной разрядности и найти “лекарство”, если ухудшение неприемлемо. Вообще, влияние названных ошибок можно уменьшить до приемлемых уровней, используя больше битов, но такое решение иногда приводит к увеличению стоимости фильтра.

Перечислим основные искажающие факторы цифровых БИХ-фильтров.

- Шум квантования АЦП, проистекающий от представления выборок входных данных $x(n)$ малым числом битов.
- Ошибки квантования коэффициентов, вызванные представлением коэффициентов БИХ-фильтров конечным числом битов.
- Ошибки переполнения, которые возникают при сложении или промежуточном суммировании частичных результатов в регистре ограниченной длины.
- Ошибки округления результата, когда выход $y(n)$ и результаты внутренних арифметических операций округляются (или усекаются) до разрешенной длины слова.

Степень ухудшения фильтра зависит, во-первых, от длины слова и типа арифметики, используемой для выполнения операции фильтрации, во-вторых, метода, используемого для квантования коэффициентов и переменных фильтра, и, в-третьих, структуры фильтра. Зная данные факторы, разработчик может оценить влияние конечной разрядности на производительность фильтра и при необходимости принять меры. В зависимости от реализации фильтра, некоторые эффекты могут быть незначительными. Например, если фильтр реализуется как программа на языке высокого уровня преимущественно на больших компьютерах, ошибки квантования коэффициентов и округления несущественны. При обработке в реальном времени входные и выходные сигналы, коэффициенты фильтра и результаты арифметических операций представляются с использованием слов конечной длины (обычно 8, 12 и 16 бит). В таких случаях практически всегда необходимо проанализировать влияние квантования на производительность фильтра.

Влияние конечной разрядности на производительность более сложно проанализировать для БИХ-, чем для КИХ-фильтров из-за наличия обратной связи. В то же время, использование MATLAB (см. приложение 8Б) позволяет для определенных фильтров получить практические решения. Влияние каждого из четырех перечисленных источников ошибок рассматривается ниже в этой главе. Более подробный анализ влияния конечной длины слова на производительность БИХ-фильтров и других систем цифровой обработки сигналов приводится в главе 13.

8.14.1. Ошибки квантования коэффициентов

Напомним, что БИХ-фильтр характеризуется следующим уравнением:

$$H(z) = \frac{\sum_{k=0}^M b_k z^{-k}}{1 + \sum_{k=1}^M a_k z^{-k}}.$$

Когда коэффициенты квантуются до конечного числа битов, например, 8 или 16, квантованную передаточную функцию можно записать следующим образом:

$$[H(z)]_q = \frac{\sum_{k=0}^M [b_k]_q z^{-k}}{1 + \sum_{k=1}^M [a_k]_q z^{-k}}, \quad (8.52)$$

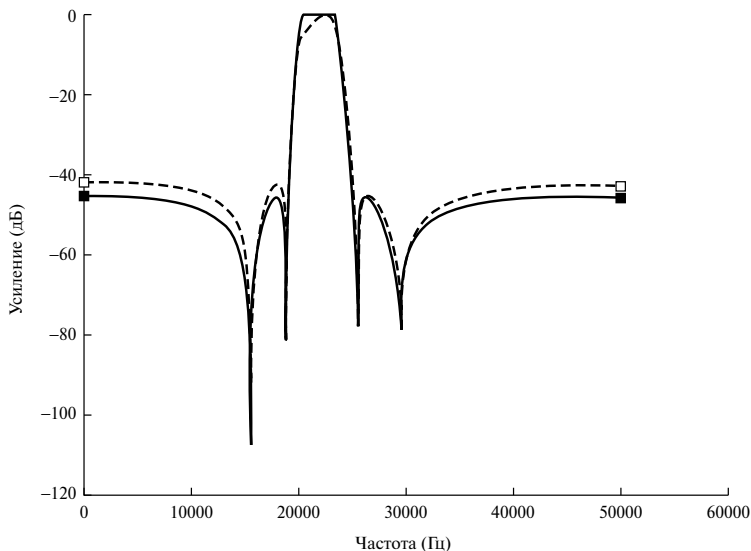


Рис. 8.30. Практическое влияние квантования коэффициентов на частотную характеристику:
 ■ — неквантованная характеристика; □ — квантованная характеристика (5 бит)

где

$$[b_k]_q = b_k + \Delta b_k; \quad [a_k]_q = a_k + \Delta a_k,$$

$\Delta b_k, \Delta a_k$ — изменение соответственно коэффициентов b_k и a_k ,

q — “квантованный коэффициент”.

Основное влияние квантования коэффициентов фильтра с использованием конечного числа битов проявляется в изменении положения полюсов и нулей $H(z)$ на комплексной плоскости. Это может привести к следующим последствиям:

- неустойчивость или потенциальная неустойчивость фильтров больших порядков с узкими полосами перехода и полюсами, близкими к единичной окружности;
- изменение желаемой частотной характеристики, как показано на рис. 8.30.

Квантованный фильтр всегда следует анализировать, чтобы убедиться, что используемой длины слова достаточно для обеспечения устойчивости и получения удовлетворительной частотной характеристики. Влияние квантования коэффициентов на частотную характеристику иллюстрируется ниже на примере. Более подробный анализ влияния квантования коэффициентов на производительность фильтра проводится в главе 13.

Пример 8.21

Полосовой цифровой фильтр планируется использовать для цифрового восстановления тактовой синхронизации на скорости 4,8 Кбод при частоте дискретизации 153,6 кГц. Фильтр характеризуется следующей передаточной функцией:

$$H(z) = \frac{1}{1 + a_1 z^{-1} + a_2 z^{-2}},$$

где

$$a_1 = -1,957558 \text{ и } a_2 = 0,995913.$$

Оцените влияние квантования коэффициентов до 8 бит на положение полюсов, а следовательно, центральной частоты.

Решение

Вначале найдем положения всех полюсов неквантованного фильтра. Радиус r и угол θ полюса записываются как

$$r = \sqrt{a_2}, \quad \theta = \cos^{-1} \left(-\frac{a_1}{2r} \right).$$

Следовательно,

$$r = \sqrt{0,995913} = 0,99795 \text{ и } \theta = \cos^{-1} \left(\frac{1,957558}{2 \times 0,99795} \right) = 11,25^\circ.$$

Найденные величины соответствуют центральной частоте

$$\left(\frac{11,25}{360} \right) \times 153,6 \times 10^3 = 4,7999 \text{ кГц.}$$

Поскольку один коэффициент больше единицы, выделим 1 бит для хранения бита знака, 1 бит для целой и 6 бит — для дробной части коэффициента. Таким образом, после квантования до 8 бит коэффициенты приобретают такие значения:

$$a_1 = -1,957558 \times 2^6 = -125 (\equiv 10000100)$$

$$a_2 = 0,995913 \times 2^6 = 63 \text{ (максимальная положительная дробная часть)} (\equiv 00111111)$$

В дробной записи значения квантованных коэффициентов имеют такой вид:

$$a_1 = -\frac{125}{64} = -1,953125; \quad a_2 = \frac{63}{64} = 0,984375.$$

Новое положение полюса: $r = 0,992156$, $\theta = 10,17^\circ$; центральная частота смещается в положение

$$f_0 = \left(\frac{10,17}{360} \right) \times 153,6 \times 10^3 = 4,34 \text{ кГц.}$$

8.15. Реализация БИХ-фильтров

Выход БИХ-фильтра $y(n)$ вычисляется для каждой входной выборки $x(n)$. Учитывая каскадную реализацию с использованием звеньев второго порядка в прямой форме, записываем следующее уравнение фильтрации:

$$y(n) = \sum_{k=0}^2 b_k x(n-k) - \sum_{k=1}^2 a_k y(n-k).$$

Из данного уравнения ясно видно, что для реализации фильтра требуются следующие компоненты:

- память (например, ПЗУ) для хранения коэффициентов фильтра;
- память (например, ОЗУ) для хранения текущих и прошедших входных и выходных выборок $\{x(n), x(n-1), \dots\}$ и $\{y(n), y(n-1), \dots\}$;
- аппаратный или программный умножитель (умножители);
- сумматор или арифметико-логическое устройство (АЛУ).

В современной цифровой обработке сигналов в реальном времени для фильтрации эффективно используются такие процессоры ЦОС, как TMS320C50. В данные процессоры встроены все стандартные блоки, в том числе аппаратные умножители. В некоторых приложениях привлекательна альтернативная реализация — с помощью таких стандартных 8- или 16-битовых микропроцессоров, как процессоры семейства Motorola 6800 или 68000. В дополнение к аппаратуре обработки сигналов разработчик должен обеспечить подходящие интерфейсы ввода-вывода (например, при преобразовании аналоговый-цифровой-аналоговый сигнал) цифровой аппаратуры в зависимости от типа источника и получателя данных. Данный подход можно назвать *аппаратной реализацией*.

При обработке сигналов в модельном времени или автономно для реализации фильтра используется подходящий язык высокого уровня. В этом случае фильтр часто реализуется на языке C или FORTRAN и запускается на универсальном компьютере, таком как персональный компьютер или мэйнфрейм, где все стандартные блоки уже сконфигурированы. Следовательно, обработку в модельном времени можно описать как чисто программную реализацию.

8.15.1. Требования к вычислениям

Разработчик также должен анализировать влияние на цифровой фильтр, построенный на используемом процессоре, требований к вычислению. Основными вычислительными операциями в фильтре являются умножение, сложение, накопление и задержки или сдвиги. Например, фильтр, состоящий из звеньев второго порядка, обычно требует четыре операции умножения, четыре операции сложения и несколько операций сдвига и обращения к ЗУ. Если фильтрация выполняется в реальном времени, например, с частотой 44,1 кГц (цифровое аудио), арифметические операции нужно выполнять каждые $1/(44,1 \text{ кГц})$. Кроме того, нужны допуски для учета других служебных издержек, таких

как выборка выходных данных, запоминание или извлечение из памяти фильтрованных выборок, а также других вспомогательных служебных операций.

8.16. Подробный пример разработки цифрового БИХ-фильтра

Данный пример иллюстрирует некоторые концепции, представленные в данной главе. В частности, показана реализация пятиэтапной процедуры разработки.

Этап 1: спецификация фильтра

Разработайте и реализуйте цифровой БИХ-фильтр, используя программный пакет. Учтите, что конечная реализация будет выполнена на основе TMS320C50, и фильтр должен удовлетворять следующим спецификациям:

частота дискретизации	15 кГц,
полоса пропускания	0–3 кГц,
ширина полосы перехода	450 Гц,
неравномерность в полосе пропускания	0,5 дБ,
затухание в полосе подавления	45 дБ.

Примечание: целевая плата имеет 12-битовые АЦП и ЦАП.

Этап 2: вычисление коэффициентов

Используя программу разработки БИХ-фильтров из книги [Ifeachor, 2001], находим, что спецификации соответствует эллиптический фильтр четвертого порядка, разработанный с помощью билинейного преобразования. Выход программы разработки приводится ниже.

	Знаменатель	Числитель
	A_k	B_k
1	$1,000000E+00$	$5,846399E-02$
2	$-1,325263E+00$	$1,359507E-01$
3	$1,480202E+00$	$1,820297E-01$
4	$-7,841098-01$	$1,359506E-01$
5	$2,339270E-01$	$5,846398E-02$

	Полюса		Коэффициенты
	Мнимая часть	z^{-1}	z^{-2}
Действительная часть			
0,247967	0,836885	-0,495935	0,761864
0,414664	0,367559	-0,829328	0,307046

	Нули		Коэффициенты
	Мнимая часть	z^{-1}	z^{-2}
Действительная часть			
-0,337859	0,941197	0,675718	1,000000
-0,824828	0,565383	1,649656	1,000000

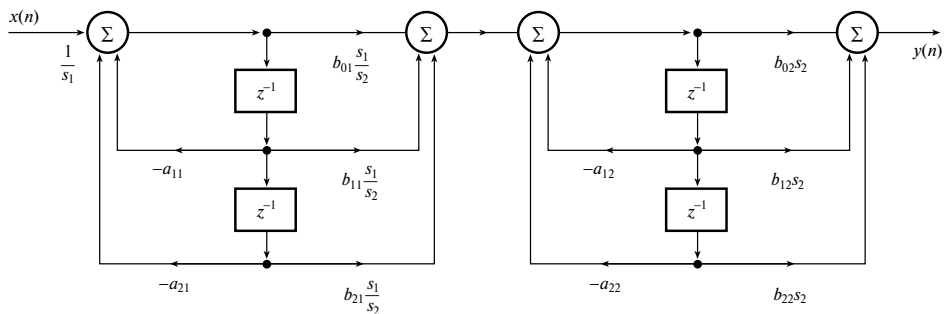


Рис. 8.31. Диаграмма реализации фильтра из раздела 8.16

На основе приведенных данных находим, что передаточная функция фильтра, реализованного в прямой форме, записывается как

$$H(z) = \frac{0,05846399 + 0,1359507z^{-1} + 0,1820979z^{-2} + 0,1359506z^{-3} + 0,05846398z^{-4}}{1 - 1,325263z^{-1} + 1,480202z^{-2} - 0,784109z^{-3} + 0,233927z^{-4}}.$$

Этап 3: выбор структуры реализации

Как объяснялось выше, реализация $H(z)$ в прямой форме очень чувствительна ко многим неблагоприятным эффектам конечной разрядности, например, таким, как ошибки квантования коэффициентов. Следовательно, $H(z)$ лучше разбить на меньшие звенья, а затем соединить их, например, в каскадную или параллельную структуру. Если используется каскадная структура, то $H(z)$ разбивается на два звена второго порядка $H_1(z)$ и $H_2(z)$:

$$H(z) = H_1(z)H_2(z),$$

где

$$H_1(z) = \frac{b_{01} + b_{11}z^{-1} + b_{21}z^{-2}}{1 + a_{11}z^{-1} + a_{21}z^{-2}},$$

$$H_2(z) = \frac{b_{02} + b_{12}z^{-1} + b_{22}z^{-2}}{1 + a_{12}z^{-1} + a_{22}z^{-2}}.$$

Диаграмма реализации фильтра изображена на рис. 8.31, где каждое звено фильтра реализовано с использованием стандартной биквадратной структуры. Соответствующие наборы разностных уравнений, определяющих, как выполняется операция фильтрации, приведены ниже.

Звено фильтра 1

$$\begin{aligned}w_1(n) &= (1/s_1)x(n) - a_{11}w_1(n-1) - a_{21}w_1(n-2), \\y_1(n) &= b_{01}w_1(n)s_1/s_2 + b_{11}w_1(n-1)s_1/s_2 + b_{21}w_1(n-2)s_1/s_2.\end{aligned}$$

Звено фильтра 2

$$\begin{aligned}w_2(n) &= y_1(n) - a_{12}w_2(n-1) - a_{22}w_2(n-2), \\y_2(n) &= b_{02}w_2(n)s_2 + b_{12}w_2(n-1)s_2 + b_{22}w_2(n-2)s_2.\end{aligned}$$

Точные значения коэффициентов a_{ij} и b_{ij} зависят от разбиения на пары полиномов числителя и знаменателя $H(z)$ и упорядочения звеньев фильтра второго порядка, использованных для реализации полиномов. Наилучший выбор пар и сортировки можно определить, только проанализировав влияние конечной разрядности.

Этап 4: анализ влияния конечной разрядности

Основываясь на данной спецификации, предположим, что будет использоваться арифметика с дополнением до двух с фиксированной запятой, и что каждый коэффициент будет квантоваться через округление до 16 бит.

Основная цель на этом этапе — оценить влияние различных ошибок квантования на производительность фильтра и определить наилучшую конфигурацию фильтра с позиции отношения сигнал-шум. При оценке рассматриваются следующие источники ошибок (подробнее см. главу 13):

- ошибки переполнения;
- ошибки округления;
- ошибки квантования коэффициентов.

Чтобы избежать переполнения на выходе сумматоров, показанных на рис. 8.31, вводятся подходящие масштабные множители, которые выписаны перед сумматорами.

Поскольку $H(z)$ описывает фильтр четвертого порядка, который нужно реализовать как два звена второго порядка, множители числителя и знаменателя этой функции можно упорядочить четырьмя различными способами:

$$\begin{aligned}H_A(z) &= \frac{N_1(z)}{D_1(z)} \frac{N_2(z)}{D_2(z)}, \\H_B(z) &= \frac{N_2(z)}{D_2(z)} \frac{N_1(z)}{D_1(z)}, \\H_C(z) &= \frac{N_1(z)}{D_2(z)} \frac{N_2(z)}{D_1(z)}, \\H_D(z) &= \frac{N_2(z)}{D_1(z)} \frac{N_1(z)}{D_2(z)},\end{aligned}$$

Таблица 8.1. Масштабные множители четырех конфигураций фильтра

Фильтр	Масштабный множитель	L_1	L_2	L_∞
A	s_1	5,524 844	1,608 890	4,379 544
	s_2	11,821 571	3,677 381	7,262 393
B	s_1	2,479 158	1,359 467	2,175 539
	s_2	18,908 47	10,880 490	12,548 114
C	s_1	2,479 158	1,359 467	2,175 539
	s_2	11,821 571	10,880 490	7,262 393
D	s_1	5,524 844	1,608 890	4,379 544
	s_2	18,908 47	5,727 459	12,548 114

где

$$\begin{aligned}
 N_1(z) &= 1 + 0,675718z^{-1} + z^{-2}, \\
 N_2(z) &= 1 + 1,649656z^{-1} + z^{-2}, \\
 D_1(z) &= 1 - 0,495935z^{-1} + 0,761864z^{-2}, \\
 D_2(z) &= 1 - 0,829328z^{-1} + 0,307046z^{-2}.
 \end{aligned}$$

Каждая из четырех возможных конфигураций фильтра будет иметь различные масштабные множители, а также различное отношение сигнал-шум округления. Задача этого этапа — определить наилучший для фильтра выбор пар и сортировки с точки зрения отношения сигнал-шум. Ошибки переполнения и округления тесно связаны, так что влияние масштабирования и округления следует анализировать одновременно.

Используя программу анализа влияния конечной разрядности, получаем масштабные множители четырех возможных фильтров, которые сведены в табл. 8.1 (анализ основан на нормах L_1 , L_2 и L_∞ , в данном примере в качестве критерия использовалась норма L_1). Для фильтра четвертого порядка, реализованного в виде каскада двух канонических звеньев второго порядка, на выходе шум округления после масштабирования записывается как

$$\sigma_o^2 = \frac{q^2}{12} [3s_1^2 \|H_1(z)H_2(z)\|_2^2 + 5s_2^2 \|H(z)\|_2^2 + 3],$$

где q — шаг квантования, $\|\cdot\|_2^2$ обозначает квадрат нормы L_2 , $H_1(z)$ — передаточная функция первого звена фильтра, $H_2(z)$ — передаточная функция второго звена фильтра, s_1 и s_2 — масштабные множители соответственно первого и второго звеньев фильтра.

Шумовые характеристики всех четырех конфигураций при квантовании коэффициентов до 16 бит (после масштабирования) приведены в табл. 8.2. Очевидно, что фильтр В имеет наилучшие характеристики по фактору шума округления. Масштабированная передаточная функция этого фильтра записывается следующим образом:

$$H(z) = H'_B(z) = \frac{s_1}{s_2} \frac{N_2(z)}{D_2(z)} \frac{N_1(z)}{D_1(z)} s_2 =$$

Таблица 8.2. Сравнение мощности шума округления четырех конфигураций фильтра

Фильтр	Мощность шума
A	$703q^2$
B	$326,378q^2$
C	$382,32q^2$
D	570,453

$$\begin{aligned} &= 0,1311136 \frac{1 + 1,649656z^{-1} + z^{-2}}{1 - 0,829328z^{-1} + 0,307046z^{-2}} \times \\ &\times \frac{1 + 0,675718z^{-1} + z^{-2}}{1 - 0,495935z^{-1} + 0,761864z^{-2}} 10,880490. \end{aligned}$$

Проанализируем далее влияние ошибок квантования коэффициентов. В частности, проверим, достаточно ли заданной длины коэффициентов для обеспечения устойчивости и удовлетворения спецификациям частотной характеристики. Поскольку полюса не расположены слишком близко к единичной окружности, для обеспечения устойчивости 16-битовых коэффициентов достаточно. Например, для первого звена фильтра программа анализа влияния конечной разрядности показывает, что устойчивость обеспечивают уже 3 бит, и что квантование коэффициентов до 16 бит приводит к изменению радиуса полюсов всего на 0,00048%. Данная программа также показывает, что для того, чтобы удерживать частотную характеристику в пределах допусков, достаточно всего 12 бит. При 16-битовых коэффициентах характеристика фильтра, по сути, идентична характеристике неквантованного фильтра. Частотная характеристика и диаграмма нулей и полюсов неквантованного фильтра изображены на рис. 8.32.

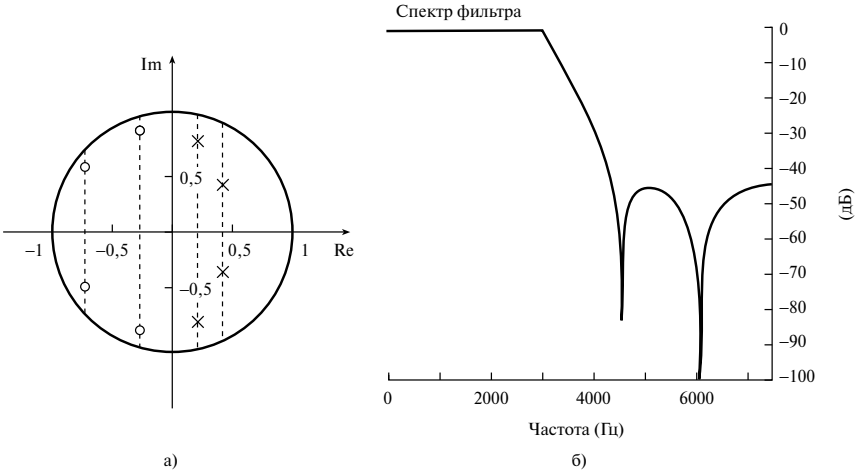


Рис. 8.32. Диаграмма нулей и полюсов (панель а) и частотная характеристика фильтра из раздела 8.16 (панель б)

Чтобы свести влияние шума округления к минимуму, накопление сумм произведений и квантование после накопления проводится в регистрах двойной длины.

Этап 5

Квантованные коэффициенты (получены умножением масштабированных коэффициентов на 2^{15}) вводятся в приведенную в приложении программу БИХ-фильтрации на основе TMS320C25. Более подробно программы БИХ-фильтрации и их разработка рассмотрены в главе 12.

8.17. Резюме

Разработку БИХ-фильтров можно разделить на пять независимых этапов (см. рис. 8.1). Спецификации фильтров часто зависят от приложения, но в любом случае должны включать детали, касающиеся граничных частот, допусков амплитудной характеристики, частоты дискретизации требований ко входу-выходу. Если разрабатываются фильтры со стандартными характеристиками, их коэффициенты можно получить с помощью билинейного z -преобразования. Этот подход, а также другие полезные методы получения коэффициентов, описаны в данной главе и проиллюстрированы на многих примерах. Для реализации БИХ-фильтров больших порядков используются каскадные или параллельные комбинации звеньев второго или первого порядка, что позволяет минимизировать изменение положений полюсов и нулей, вызванное влиянием конечной разрядности. Если это нужно, для борьбы с переполнением во внутренних узлах фильтра вход каждого звена масштабируется.

Производительность цифрового БИХ-фильтра ограничена числом битов, использованных в его реализации. Существует четыре основных источника ошибок: 1) входное квантование; 2) квантование коэффициентов; 3) округление произведений и 4) переполнение при сложении. В данной главе представлены методы, с помощью которых можно анализировать влияние этих факторов на производительность фильтра, и средства, позволяющие при необходимости свести это влияние к минимуму или устранить совсем. При этом длины коэффициентов должно быть достаточно для минимизации влияния квантования коэффициентов на частотную характеристику и предотвращения неустойчивости. Кстати, вопрос устойчивости БИХ-фильтра следует рассматривать всегда. БИХ-фильтр, устойчивый при реализации с бесконечной точностью, может стать неустойчивым, если его реализовать с конечной точностью. Например, при обработке аудиосигналов низкой частоты необходимыми называют 24-битовые коэффициенты. В большинстве других случаев для минимизации влияния конечной разрядности достаточно представить коэффициенты 16 или более битами и выполнять арифметические операции в регистрах двойной длины.

Ошибки округления или усечения, вызванные арифметическими операциями конечной точности, создают в фильтре такие нелинейные эффекты, как предельные циклы, в силу чего выход фильтра осциллирует даже при отсутствии входа (или при постоянном входе). Влияние ошибок округления на производительность фильтра можно описать через отношение сигнал-шум и выход фильтра. Снижение отношения сигнал-шум вслед-

ствие ошибок округления можно нейтрализовать с помощью схемы формирования спектра шума (ESS, подробности см. в главе 13). Главная задача подобных схем — это свести на нуль “усиливающее” влияние полюсов фильтра на ошибки округления. Цена такой коррекции — увеличить число операций умножения и сложения, несмотря на то, что схемы ESS первого порядка с целыми коэффициентами вычислительно более эффективны.

8.18. Примеры использования БИХ-фильтров в цифровых аудиосистемах и измерительных приборах

В данном разделе рассматриваются приложения, в которых БИХ-фильтры применяются или могли бы применяться.

8.18.1. Цифровые аудиосистемы

Цифровые фильтры нашли применение во многих областях цифрового аудио, особенно в системах с высококачественными цифровыми источниками, такими как проигрыватель компакт-дисков и цифровые аудиопленки. Цифровая обработка сигналов также позволяет имитировать акустические свойства различных мест, таких как концертные залы, джаз-клубы и дискотеки. БИХ-фильтры используются при графическом выравнивании, регулировке тембра, выравнивании каналов, формировании спектра шума в АЦП/ЦАП и расщеплении полос.

Например, в цифровых графических эквалайзерах БИХ-фильтры используются для расщепления всего диапазона аудиочастот на отдельные полосы, что позволяет полностью подстраивать тона воспроизводимого звука под персональные вкусы, а не только регулировать верхние и нижние звуковые частоты. Обычный пятиполосный графический эквалайзер разбивает диапазон аудиочастот на пять полос с центральными частотами 100 Гц, 330 Гц, 3,3 кГц, 10 кГц и 16 кГц и позволяет изменять уровень сигнала в каждой полосе в пределах ± 10 дБ.

Простая графическая схема графического выравнивания показана на рис. 8.33.

8.18.2. Цифровое управление

С осознанием выгод, которые несет цифровая обработка сигналов, и с понижением цен на процессоры происходит переход к цифровым контроллерам, которые обеспечивают и большую гибкость, и большую точность. На рис. 8.34 демонстрируются принципы цифрового управления аналоговой установкой $H(s)$, которой может быть, например, машина или мотор. В общем случае цифровой контроллер имеет бесконечную импульсную характеристику.

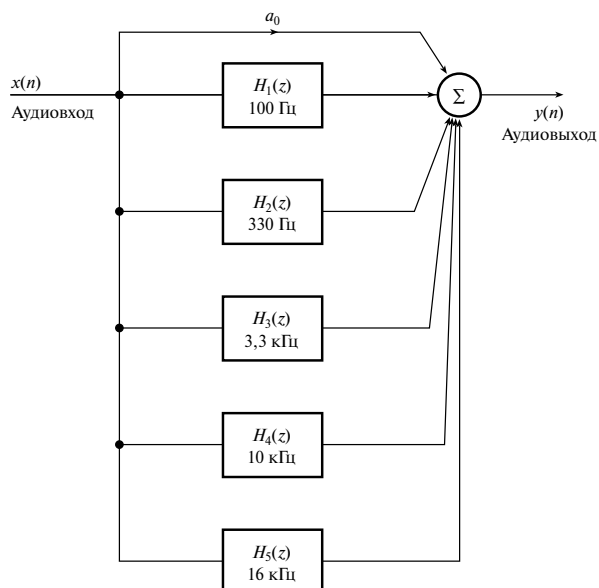


Рис. 8.33. Упрощенная диаграмма полностью цифрового графического эквалайзера. Основной компонент — блок параллельных БИХ-фильтров с различными центральными частотами. Усиление каждого фильтра можно регулировать независимо, например, с помощью скользящего потенциометра, скажем, в диапазоне ± 10 дБ

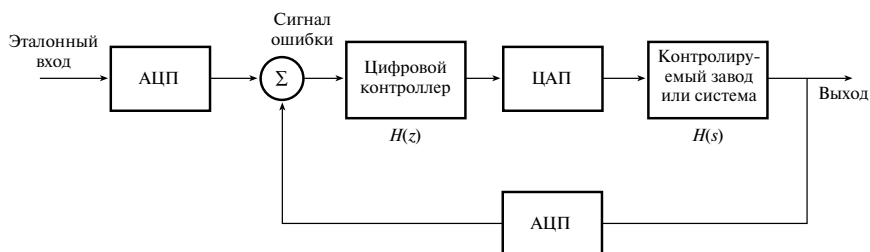


Рис. 8.34. Принципы цифрового управления аналоговым оборудованием

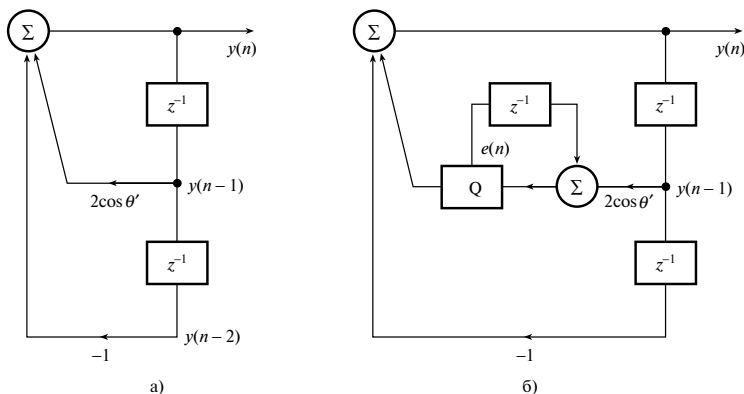


Рис. 8.35. Простой цифровой осциллятор (панель а); простой цифровой осциллятор со схемой формирования формы ошибки первого порядка (панель б)

8.18.3. Цифровые частотные осцилляторы

БИХ-фильтры используются также для генерации точных сигналов. В этом подходе использован тот факт, что БИХ-фильтры с полюсами на единичной окружности неустойчивы. Рассмотрим рис. 8.35, а, где изображен простой гармонический осциллятор. Полюса БИХ-фильтра расположены в точках $e^{\pm i\theta}$, а частота осцилляции определяется соотношением

$$\theta' = \omega_0 T_B,$$

где T — период дискретизации. Коэффициент фильтра $2\cos\theta'$ может принимать только целые значения, для чего он считается как целая часть от $2^B \times 2\cos\theta'$ (B — число битов).

Основной вопрос при генерации цифрового сигнала с помощью БИХ-фильтров связан с влиянием конечной разрядности. Например, из-за квантования коэффициентов частоты будут расположены неравномерно, а квантование произведения приводит к накоплению ошибок округления, которое вскоре делает генератор сигналов бесполезным. В то же время, использование схем ESS позволяет свести эти ошибки к минимуму. Пример осциллятора, в котором используется техника ESS, значительно снижающая влияние шума округления, приведен на рис. 8.35, б [1].

8.19. Примеры применения БИХ-фильтров в связи

Из-за своих характерных особенностей БИХ-фильтры широко используются в цифровой связи. В цифровой телефонии [8], например, с помощью импульсно-кодовой модуляции (pulse-code modulation — PCM) можно одновременно вести передачу по нескольким речевым каналам. Каждый канал после ограничения полосы выбирается

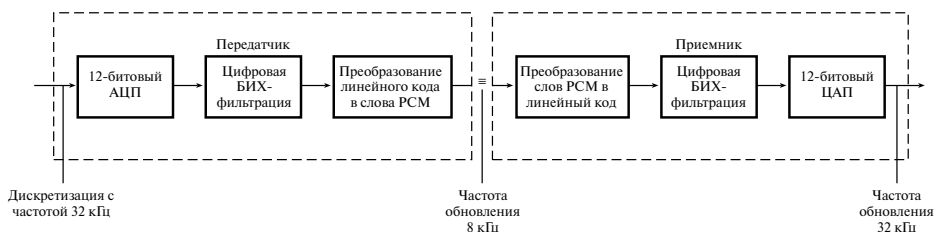


Рис. 8.36. Канал РСМ, демонстрирующий возможное использование БИХ-фильтров как фильтров защиты от наложения спектров (передатчик) и фильтров подавления зеркальных частот (приемник)

с частотой 8 кГц и кодируется с использованием либо A -, либо μ -характеристики. После приема РСМ-данные восстанавливаются в аналоговой форме и проходят фильтрацию с целью подавления зеркальных частот. Для выполнения необходимой фильтрации при передаче и приеме могут использоваться цифровые БИХ-фильтры (см. рис. 8.36). В таком случае фильтрация выполняется с более высокой частотой дискретизации, например, 32 кГц, а затем полученный сигнал переводится в стандартную форму РСМ.

В следующих двух разделах более подробно рассмотрены два специфических приложения БИХ-фильтров в цифровой связи — в цифровых кнопочных телефонах и при восстановлении тактовой частоты при передаче данных.

8.19.1. Генерация и прием сигналов в цифровых кнопочных телефонах

Великолепным применением БИХ-фильтров является полностью цифровой двухтональный многочастотный кнопочный приемник [13, 17].

В современных телефонных системах информация, необходимая для установки связи и поддержания соединения, обычно поставляется с помощью многочастотного кода. Как правило, телефонный аппарат генерирует два тона — один тон низкой частоты и один тон высокой частоты (рис. 8.37).

Генератор тональной частоты можно реализовать с использованием пары программируемых БИХ-осцилляторов второго порядка (рис. 8.38). После нажатия кнопки на основе кода набранной цифры выбираются подходящие коэффициенты фильтра и иницируется (вызывается из ПЗУ) режим, при котором генерируется пара тонов (один высокой частоты и один низкой). Тоны складываются и получается двухтональный сигнал. Как для цифрового генератора гармонических сигналов, производительность кнопочного генератора можно улучшить, используя схемы обратной связи по ошибке.

При приеме информация оцифровывается с частотой 8 кГц, а затем разделяется на низкую и высокую полосы частот с помощью полосовых БИХ-фильтров. Чтобы обнаружить тон, следует проверить уровень сигнала, проведя комбинированную полосовую фильтрацию, полностью очистить сигнал и пропустить его через фильтр нижних частот. Чтобы идентифицировать тон низкой частоты, полоса низких частот расщепляется на четыре полосы (используется два из четырех полосовых фильтров). То же делается и с полосой высоких частот. Затем полученные восемь уровней передаются к логической схеме принятия решения, где и определяется полученный код.

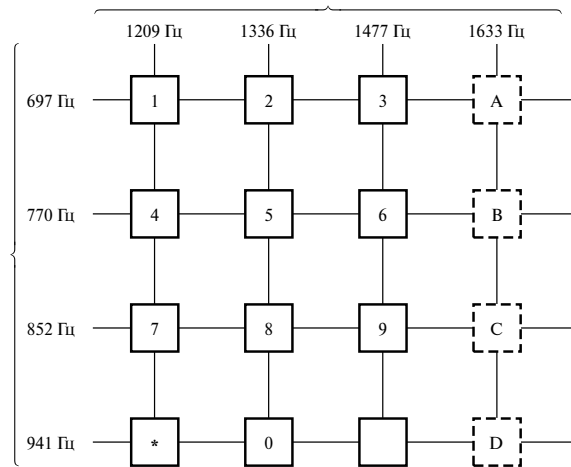


Рис. 8.37. Упрощенная схема клавишного поля 4×4 кнопочного телефона. Кнопки, изображенные пунктиром, недоступны. При нажатии кнопки генерируется пара тонов — один из низкочастотной группы, другой из высокочастотной. Например, при нажатии кнопки 9 генерируются тоны 825 и 1477 Гц (согласно [17])

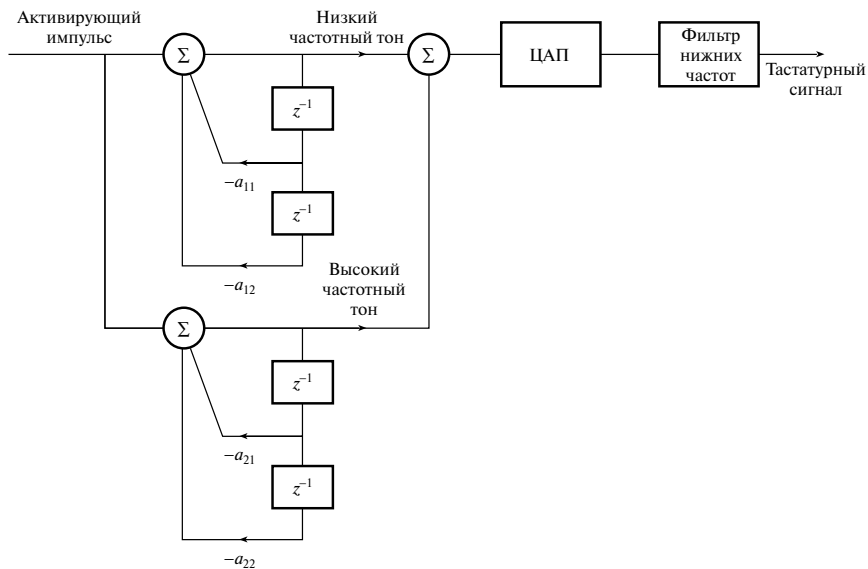


Рис. 8.38. Кнопочный генератор (согласно [17]). Код набранной цифры используется для выбора коэффициентов фильтра и начальных условий, а следовательно, частоты осциллятора

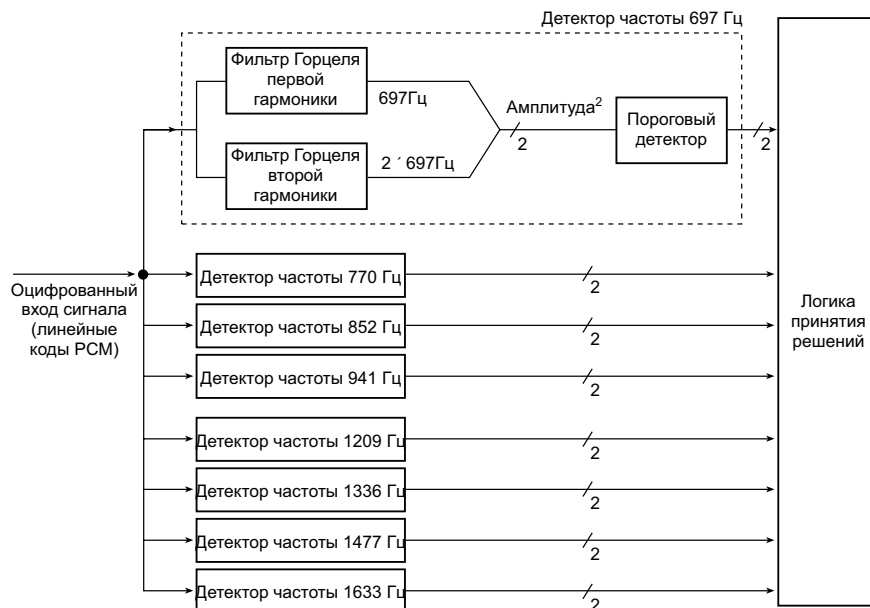


Рис. 8.39. Принципы декодирования ДТМЧ с использованием фильтров Горцеля (согласно [17])

8.19.2. Цифровая телефония: детектирование двухтонального многочастотного сигнала с использованием алгоритма Горцеля

При детектировании двухтональных многочастотных (ДТМЧ) сигналов могут использоваться не только стандартные БИХ-фильтры, но и алгоритм Горцеля (Goertzel) (см. [3, 16, 17]). Алгоритм Горцеля — это специальная реализация дискретного преобразования Фурье (ДПФ) в форме БИХ-фильтра. Схема детектирования ДТМЧ-сигналов на основе алгоритма Горцеля изображена на рис. 8.39. Она включает параллельный блок из восьми пар фильтров Горцеля. Каждая пара фильтров детектирует тон ДТМЧ-сигнала и его вторую гармонику. Вторая гармоника нужна, чтобы различать тоны ДТМЧ и речь. Речевой сигнал, в отличие от сигнала ДТМЧ, включает значительные гармоники четного порядка. Затем каждый выход фильтра возводится в квадрат, и эта величина принимается в качестве меры интенсивности сигнала на каждой из восьми частот ДТМЧ и их вторых гармониках. Принятую цифру определяет наиболее мощная пара сигналов из группы низких и высоких частот.

Каждый фильтр Горцеля — это полосовой БИХ-фильтр второго порядка с высокой добротностью, который характеризуется следующей передаточной функцией (рис. 8.40):

$$H_k(z) = \frac{1 - W_N^k z^{-1}}{1 - 2 \cos\left(\frac{2\pi k}{N}\right) z^{-1} + z^{-2}}, \quad (8.53, a)$$

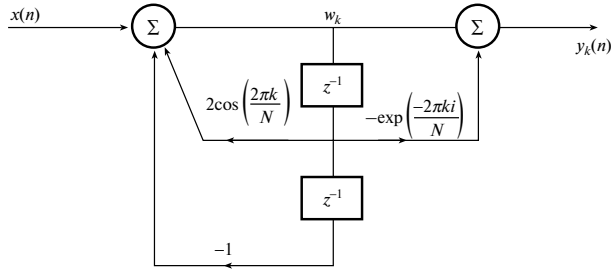


Рис. 8.40. Структура фильтра Горцеля второго порядка

где

$$W_N^k = \exp\left(-\frac{2\pi ki}{N}\right).$$

Фильтр описывается следующими разностными уравнениями:

$$\nu_k(n) = 2 \cos\left(\frac{2\pi k}{N}\right) \nu_k(n-1) - \nu_k(n-2) + x(n), \quad n = 0, 1, \dots, N \quad (8.53, б)$$

$$y_k(n) = \nu_k(n) - W_N^k \nu_k(n-1), \quad (8.53, в)$$

где

$$w_k(-1) = w_k(-2) = 0.$$

При детектировании тонов ДТМЧ-сигнала требуется только амплитуда сигнала (информация о фазе игнорируется), так что алгоритм Горцеля модифицируется так, чтобы на выход подавался только квадрат амплитуды:

$$|y_k(N)|^2 = \nu_k^2(N) + \nu_k^2(N-1) - 2 \cos\left(\frac{2\pi k}{N}\right) \nu_k(N) \nu_k(N-1). \quad (8.53, з)$$

Модифицированный алгоритм ДТМЧ-детектирования (уравнения (8.53, б и з)) требует для вычисления амплитуды каждого ДТМЧ-сигнала только один действительный коэффициент, т.е.

$$2 \cos\left(\frac{2\pi k}{N}\right),$$

и при этом не нужно использовать комплексную арифметику. Выход фильтра по формуле (8.53, з) вычисляется только раз для каждого тона ДТМЧ для $n = N$, т.е. в конце итераций в цепи обратной связи фильтра (уравнение (8.53, б)).

В число достоинств алгоритма Горцеля входит то, что для определения амплитуды сигнала он требует только один действительный коэффициент на каждую частоту ДТМЧ, использует мало памяти и выполняется очень быстро. В отличие от схем с БПФ, для начала работы этого алгоритма не нужен полный набор данных, выборки обрабатываются по мере поступления. При использовании БПФ значение N обычно ограничивается

Таблица 8.3. Параметры схемы декодирования ДТМЧ (согласно [17]). Частота дискретизации — 8 кГц. Размер ДПФ — 250 (первая гармоника), 210 (вторая гармоника)

<i>Частота ДТМЧ (Гц)</i>	<i>Разрешение по частоте (первая гармоника)</i>	<i>Разрешение по частоте (вторая гармоника)</i>
697	18	35
770	20	39
852	22	43
941	24	47
1209	31	61
1336	34	67
1477	38	74
1633	42	82

(как правило, это степень двойки). В алгоритме Горцеля может использоваться любое целое значение N , хотя выбор N — это компромисс между разрешением по частоте и временем вычисления.

Значение коэффициента фильтра, а следовательно, обнаруженную частоту определяют размер ДПФ N и число элементов разрешения по частоте k .

Размер ДПФ N , дискретное разрешение по частоте k , интервал дискретизации T и частота тона ДТМЧ f_k связаны следующим соотношением:

$$f_k = \frac{k}{NT}.$$

Частота дискретизации и частоты тонов выбираются согласно международным стандартам. Размер ДПФ N может быть разным. Параметры некоторых возможных схем декодирования перечислены в табл. 8.3 (см. [17]).

Следует отметить, что полюса фильтра Горцеля расположены на единичной окружности, следовательно, фильтр чувствителен к влиянию конечной разрядности, и это нельзя игнорировать. Более того, если число частот, которые нужно определить, относительно велико, более подходящим может быть БПФ.

Пример 8.22

1. Дискретное преобразование Фурье (ДПФ) информационной последовательности $x(n)$ ($n = 0, 1, \dots, N-1$) можно определить как

$$X(k) = \sum_{m=0}^{N-1} x(m) W_N^{km}, \quad k = 0, 1, \dots, N-1,$$

где W_N^{km} — настраиваемый параметр.

- А. Начав с приведенного уравнения, покажите, что передаточную функцию на z -плоскости $H_k(z)$ фильтра Горцеля для детектирования тонов ДТМЧ можно выразить в следующей рекурсивной форме:

$$H_k(z) = \frac{1 - W_N^k z^{-1}}{1 - 2 \cos\left(\frac{2\pi k}{N}\right) z^{-1} + z^{-2}}.$$

- Б. Выведите выражения для квадрата амплитуды сигнала на выходе фильтра Горцеля $|y_k(n)|^2$ в дискретные моменты времени $n = N$ и покажите, что в модифицированном алгоритме Горцеля не требуется комплексная арифметика.

2. Схема детектирования тонов ДТМЧ для системы кнопочного телефона основана на спецификациях, приведенных в табл. 8.3, и в ней используется фильтр Горцеля второго порядка.

Вычислите коэффициенты фильтра Горцеля для декодирования цифр в приемнике, если набрано “99”.

Решение

1, а. Итак,

$$X(k) = \sum_{m=0}^{N-1} x(m)W_N^{km}.$$

Используя периодичность настраиваемого множителя, можно следующим образом записать уравнение ДПФ (поскольку $W_N^{-kN} = 1$):

$$\begin{aligned} X(k) &= W_N^{-kN} \sum_{m=0}^{N-1} x(m)W_N^{km} = \\ &= \sum_{m=0}^{N-1} x(m)W_N^{-k(N-m)}. \end{aligned} \quad (8.54)$$

Уравнение (8.54) имеет ту же форму, что и выражение для свертки. Следовательно, если определить информационную последовательность $y_k(n)$ следующим образом:

$$y_k(n) = \sum_{m=0}^{N-1} x(m)W_N^{-k(n-m)}, \quad (8.55)$$

видно, что $y_k(n)$ можно рассматривать как выход КИХ-фильтра, вход которого $x(m)$, а N коэффициентов характеристики $h_k(n)$ записываются следующим образом:

$$h_k(n) = W_N^{-kn}. \quad (8.56)$$

Сравнивая формулы (8.54) и (8.55), видим, что выход фильтра при $n = N$ равен Фурье-образу $X(k)$ на частоте k :

$$X(k) = y_k(n)|_{n=N}.$$

Передаточная функция фильтра на z -плоскости записывается так:

$$\begin{aligned} H_k(z) &= \sum_{n=0}^{\infty} W_N^{-kn} z^{-n} = \\ &= \frac{1}{1 - W_N^{-k} z^{-1}}. \end{aligned} \quad (8.57)$$

Это фильтр первого порядка с единственным комплексным полюсом (поскольку настраиваемые параметры — это комплексные величины) на единичной окружности в точке $z = W_N^{-k}$. Чтобы избежать комплексной арифметики, отдельные полюса объединяются в пару комплексно-сопряженных полюсов и получается

звено фильтра второго порядка. Для этого уравнение (8.57) можно умножить на величину

$$\left(\frac{1 - W_N^k z^{-1}}{1 - W_N^k z^{-1}} \right),$$

в результате чего получаем

$$H_k(z) = \frac{1 - W_N^k z^{-1}}{1 - 2 \cos\left(\frac{2\pi k}{N}\right) z^{-1} + z^{-2}}.$$

- 1, б. На основе рис. 8.40 записываем следующее двухэтапное уравнение для фильтра Горцея:

$$\nu_k(n) = 2 \cos\left(\frac{2\pi k}{N}\right) \nu_k(n-1) - \nu_k(n-2) + x(n)$$

$$y_k(n) = \nu_k(n) - W_N^k \nu_k(n-1).$$

При $n = N$

$$\begin{aligned} y_k(N) &= \nu_k(N) - W_N^k \nu_k(N-1) = \\ &= \nu_k(N) - \nu_k(N-1) \left[\cos\left(\frac{2\pi k}{N}\right) - i \sin\left(\frac{2\pi k}{N}\right) \right] = \\ &= \nu_k(N) - \nu_k(N-1) \cos\left(\frac{2\pi k}{N}\right) + i \nu_k(N-1) \sin\left(\frac{2\pi k}{N}\right). \end{aligned}$$

$$\begin{aligned} |y_k(N)|^2 &= (\text{действительная часть})^2 + (\text{мнимая часть})^2 = \\ &= \left[\nu_k(N) - \nu_k(N-1) \cos\left(\frac{2\pi k}{N}\right) \right]^2 + \left[\nu_k(N-1) \sin\left(\frac{2\pi k}{N}\right) \right]^2 = \\ &= \nu_k^2(N) - 2\nu_k(N)\nu_k(N-1) \cos\left(\frac{2\pi k}{N}\right) + \\ &\quad + \nu_k^2(N-1) \cos^2\left(\frac{2\pi k}{N}\right) + \nu_k^2(N-1) \sin^2\left(\frac{2\pi k}{N}\right) = \\ &= \nu_k^2(N) - 2\nu_k(N)\nu_k(N-1) \cos\left(\frac{2\pi k}{N}\right) + \\ &\quad + \nu_k^2(N-1) \left[\cos^2\left(\frac{2\pi k}{N}\right) + \sin^2\left(\frac{2\pi k}{N}\right) \right] = \\ &= \nu_k^2(N) + \nu_k^2(N-1) - 2 \cos\left(\frac{2\pi k}{N}\right) \nu_k(N)\nu_k(N-1). \end{aligned} \tag{8.58}$$

2. Тоны ДТМЧ для цифры “9” — 1477 и 852 Гц. Детектирование каждого тона требует пары БИХ-фильтров Горцея. Для тона 1477 Гц используются частоты № 38 и 74:

$$a_1 = 2 \cos\left(\frac{2\pi \times 38}{205}\right) = 0,79; \quad a_2 = -1$$

$$a'_1 = 2 \cos\left(\frac{2\pi \times 74}{210}\right) = -1,1996; \quad a_2 = -1.$$

Для тона 852 Гц коэффициенты равны

$$a_1 = 1,5623, a_2 = -1; \quad a'_1 = 0,5, a'_2 = -1.$$

8.19.3. Восстановление тактовой частоты в сфере передачи данных

Во множестве систем цифровой передачи данных на большие расстояния критичным моментом является генерация в приемнике тактовых импульсов с нужной частотой и фазой, чтобы переданные данные можно было верно декодировать. Как правило, для извлечения таких тактовых импульсов используются принятые данные.

Для восстановления тактовых импульсов традиционно применяются аналоговые цепи, например, использующие контуры фазовой автоподстройки частоты (ФАПЧ), но для таких систем характерен дрейф тактовой частоты с возрастом и температурой. Более того, некоторые цепи не подходят для задач, включающих пакетную передачу, из-за их медленного отклика, также их нельзя использовать в системах с несколькими скоростями передачи [21].

Поток входных данных обычно скремблируется передатчиком (чтобы он мог предоставлять информацию о тактовых импульсах в периоды холостой передачи) и кодируются, причем каждый код представляет символ. Затем коды передаются на так называемой *скорости передачи символов*.

Принципы восстановления тактовой частоты на скорости передачи символов с использованием методов цифровой обработки сигналов показаны на рис. 8.41. Поток данных проходит суммирование по модулю 2 (схема исключаящего ИЛИ) с собственной версией, задержанной на половину тактового периода, в результате (точка С) получается выход, который содержит информацию об изменении уровня на скорости передачи символов. Затем данные передаются минимально устойчивому полосовому БИХ-фильтру. Импульсная характеристика такого фильтра спадает очень медленно со временем, давая “затухающие колебания” на частоте ω_0 , центральной частоте фильтра. Использование минимально устойчивого фильтра гарантирует наличие выхода даже при отсутствии переходов в потоке входных данных в течение разумно длительных периодов времени. Частота дискретизации фильтра выбирается кратной скорости передачи символов. Желаемая символьная тактовая частота выводится из выхода фильтра путем детектирования переходов через нуль (точка Е на рис. 8.41). Для представления в форме дополнения до двух это реализуется тривиально — исследуются знаки информационных выборок на выходе цифрового фильтра.

Для восстановления символьной тактовой частоты можно использовать простой БИХ-фильтр с одними полюсами, реализованный в форме, представленной на рис. 8.42. Фильтр характеризуется следующей передаточной функцией:

$$\begin{aligned} H(z) &= \frac{1}{[z - r \exp(-i\omega_0 T)][z - r \exp(i\omega_0 T)]} = \\ &= \frac{1}{z^2 - 2r \cos(\omega_0 T)z - r^2}, \end{aligned}$$

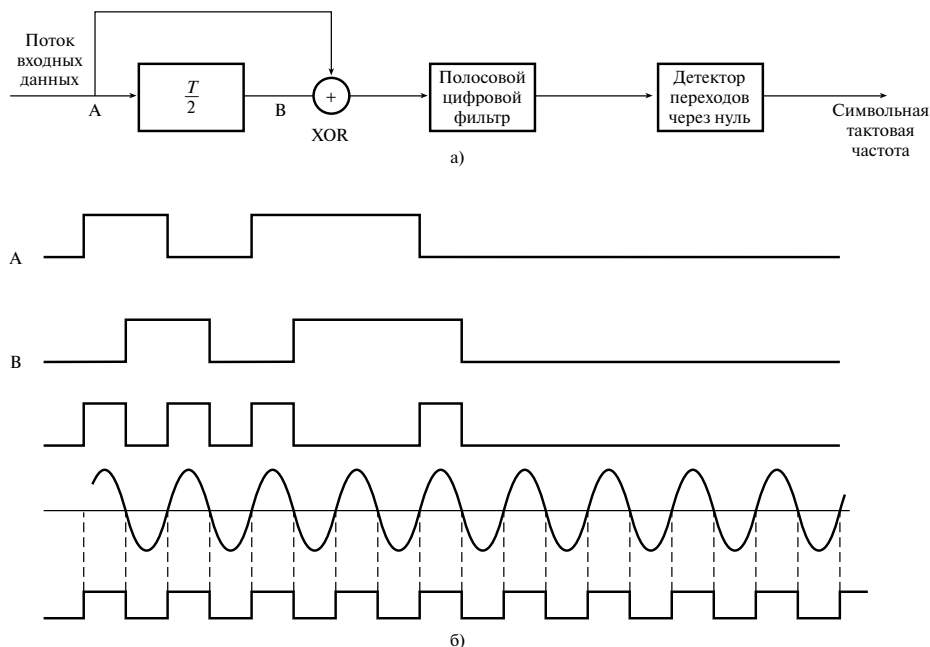


Рис. 8.41. Иллюстрация принципов восстановления символьной тактовой частоты в сфере передачи данных

где ω_0 — центральная частота полосового фильтра, r — радиус полюса, а T — величина, обратная к частоте дискретизации. Величина ω_0 обычно выбирается равной или очень близкой к символьной тактовой частоте, которую нужно восстановить, а частота дискретизации кратна центральной частоте. Ширина полосы фильтра определяется радиусом полюса (см. уравнение (8.4)). Чтобы гарантировать медленный спад импульсной характеристики, полюс обычно располагается очень близко к единичной окружности, как правило, в диапазоне $0,99 < r < 1$. Как обсуждалось в разделе 8.5.1 (уравнение (8.5)), радиус полюса r и ширина полосы фильтра bw связаны соотношением

$$r \approx 1 - (bw/F_s)\pi,$$

где $F_s = 1/T$ — частота дискретизации.

Например, для восстановления символьной тактовой частоты гипотетического модела на 4 800 бод можно использовать фильтр со следующими параметрами:

скорость передачи данных	4,8 Кбод,
центральная частота фильтра, f_0	4,8 кГц,
частота дискретизации	153,6 кГц,
ширина полосы, bw	100 Гц.

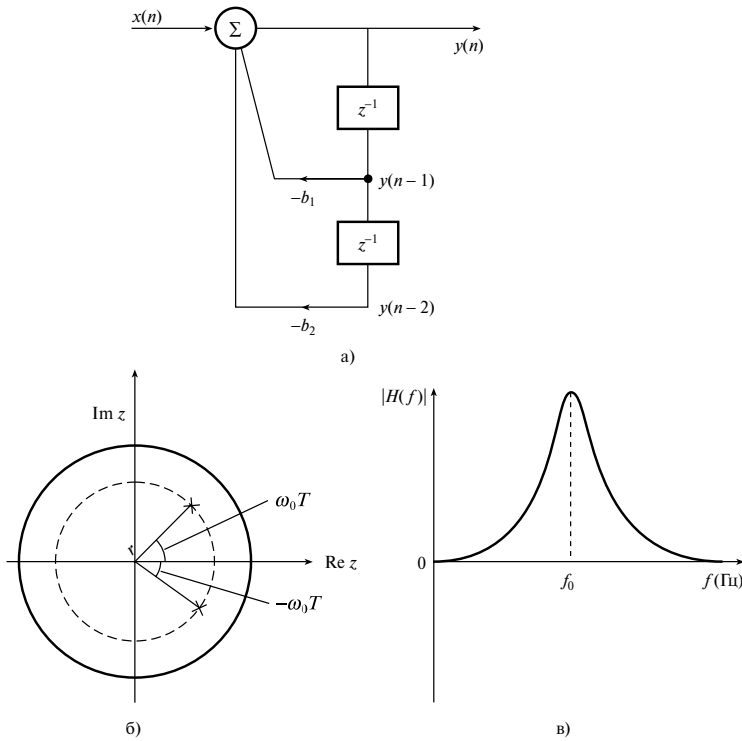


Рис. 8.42. Структура БИХ-фильтра для восстановления символьной тактовой частоты (панель а); диаграмма полюсов (панель б); спектр фильтра (панель в)

В этом случае радиус полюса (из приведенного выше уравнения) $r = 0,9795469$ и угол полюса $\omega_0 T = 2\pi f_0 T = (2\pi \times 4,8 \times 10^3 / 153,6 \times 10^3) = 0,19635$ радиан $\approx 11,25^\circ$. Итак, получаем следующую передаточную функцию:

$$H(z) = \frac{1}{z^2 - 1,957558z + 0,995913}.$$

Как обсуждалось ранее в этой главе, для того, чтобы правильно настроить фильтр, следует рассмотреть влияние конечной разрядности. В частности вход фильтра нужно масштабировать, чтобы избежать автоколебаний на выходе вследствие переполнения. Кроме того, получению “чистого” тактового генератора может способствовать простая схема формирования шума округления. Чтобы улучшить производительность практической системы восстановления тактовой частоты, входными данными которой являются нули и единицы, следует использовать второй каскад фильтра (см. [21]).

Задачи

8.1. Полюса и нуль фильтра нижних частот расположены в следующих точках:

$$\text{нуль : } -0,5; \text{ полюса : } 0,370; 0,6 \pm 0,5i.$$

1. Изобразите диаграмму нулей и полюсов.
 2. Получите передаточную функцию $H(z)$.
- 8.2. Оцифруйте с использованием метода инвариантного преобразования импульсной характеристики аналоговый фильтр с передаточной функцией

$$H(s) = \frac{\alpha}{s(s + \alpha)}, \quad \alpha = 0,5.$$

Частоту дискретизации предполагайте равной 1 (все величины нормированны).

- 8.3. На цифровом компьютере нужно смоделировать аналоговую систему со следующей нормированной характеристикой:

$$H(s) = \frac{1}{s^2 + \sqrt{2}s + 1}.$$

Получите передаточную функцию, используя

- а) метод инвариантного преобразования импульсной характеристики;
- б) метод билинейного преобразования.

Частоту дискретизации предполагайте равной 5 кГц, частота среза по уровню 3 дБ равна 1 кГц.

- 8.4. С помощью метода билинейного преобразования определите передаточную функцию и разностное уравнение цифрового эквивалента РС-фильтра, показанного на рис. 8.43. Пусть частота дискретизации равна 150 Гц, а частота среза — 30 Гц.
- 8.5. Требуется цифровой полосовой БИХ-фильтр согласно спецификации, приведенной ниже. Начав с подходящего нормированного аналогового фильтра нижних частот: 1) получите коэффициенты $H(z)$ в факторизованной форме, используя подходящее преобразование полосы и билинейное z -преобразование; 2) изобразите диаграммы нулей и полюсов аналогового фильтра и окончательного цифрового полосового фильтра.
- | | |
|----------------------------|--------------|
| полоса пропускания | 8–10 кГц, |
| частота дискретизации | 32 кГц, |
| порядок полосового фильтра | 4, |
| тип характеристики фильтра | Баттерворта. |

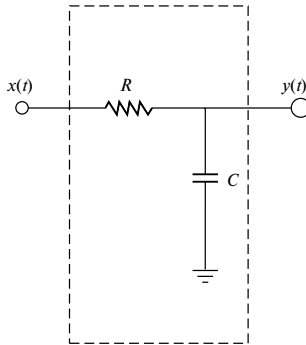


Рис. 8.43

- 8.6. 1. Прокомментируйте практическую реализацию “эффекта деформации”, связанного с разработкой методом билинейного z -преобразования, когда характеристика оцифровываемого аналогового фильтра близка к частоте Найквиста. Приведите пример специфического приложения, где данный эффект имеет существенное значение.
2. Требуется разработать БИХ-фильтр верхних частот с характеристикой Баттерворта, удовлетворяющий следующим спецификациям:
- | | |
|--------------------------------------|-----------|
| полоса пропускания | 2–4 кГц, |
| полоса подавления | 0–500 Гц, |
| неравномерность в полосе пропускания | 3 дБ, |
| затухание в полосе подавления | 20 дБ, |
| частота дискретизации | 8 кГц. |

Определите

- а) граничные частоты полосы пропускания и полосы подавления подходящего аналогового фильтра-прототипа нижних частот. Ответ должен включать подробности относительно основных концепций того, как из спецификаций получен фильтр-прототип;
 - б) порядок N фильтра-прототипа нижних частот;
 - в) передаточную функцию, а следовательно, коэффициенты БИХ-фильтра, используя метод билинейного z -преобразования.
- 8.7. Требуется, чтобы цифровой полосовой фильтр с характеристикой Баттерворта удовлетворял следующим спецификациям:
- | | |
|-----------------------|-------------|
| полоса пропускания | 200–300 Гц, |
| частота дискретизации | 2000 Гц, |
| порядок фильтра | 2. |
- 1, а. Начав с подходящего аналогового фильтра-прототипа нижних частот, определите передаточную функцию цифрового фильтра с использованием билинейного z -преобразования.
 - 1, б. Объясните с помощью диаграммы нулей и полюсов, как полюса и нули фильтра-прототипа последовательно отображаются с s - на z -плоскость.

2. Предположим, что фильтр из п. 1 нужно реализовать на устройстве с арифметикой с фиксированной запятой. Оцените последствия квантования коэффициентов до 8 бит на положение полюсов, а следовательно, центральную частоту.

Примечание: преобразование фильтра нижних частот в полосовой фильтр происходит по следующему закону:

$$s \rightarrow \frac{s^2 + \omega_0^2}{Bs},$$

где

$$\omega_0^2 = \omega'_1 \omega'_2; \quad B = \omega'_2 - \omega'_1.$$

Здесь ω'_1 , ω'_2 и ω_0 — верхняя и нижняя граничные частоты и центральная частота соответственно.

- 8.8. Используя метод билинейного преобразования и эллиптическую характеристику, получите подходящие коэффициенты цифрового БИХ-фильтра, удовлетворяющего следующим спецификациям:

полоса пропускания	4–12 кГц,
полосы подавления	0–3,4 кГц, 12,6–16 кГц,
неравномерность в полосе пропускания	<0,1 дБ,
затухание в полосе подавления	>30 дБ,
частота дискретизации	32 кГц.

Определите подходящую длину коэффициентов, гарантирующую устойчивость фильтра и наличие частотной характеристики в данных пределах.

- 8.9. Разработайте и реализуйте в программном виде цифровой фильтр нижних частот согласно следующей спецификации:

граница полосы пропускания	2,5 кГц,
граница полосы подавления	3 кГц,
отклонение в полосе пропускания	<0,1 дБ,
затухание в полосе подавления	>60 дБ,
частота дискретизации	15 кГц.

- 8.10. С помощью билинейного преобразования получите коэффициенты цифрового фильтра, с максимально плоской характеристикой в полосе пропускания, от 0 до 4 кГц, и имеющего затухание не меньше 25 дБ на частотах свыше 10 кГц. Пусть частота дискретизации равна 32 кГц.

- 8.11. Ниже приводятся требования к определенному фильтру нижних частот:

полоса пропускания	0–30 Гц,
граница полосы подавления	50 Гц,
затухание в полосе подавления	>40 дБ на $f > 50$ Гц,
частота дискретизации	256 Гц.

Предполагая, что фильтр имеет характеристику Баттерворта, определите с помощью билинейного преобразования передаточную функцию фильтра $H(z)$. Получите диаграмму реализации фильтра в каскадной форме, используя звенья второго и/или первого порядка.

- 8.12. В определенной цифровой системе обработки сигналов реального времени требуется полосовой цифровой фильтр с характеристикой Баттерворта. Фильтр должен удовлетворять следующим требованиям:

полоса пропускания	0,3–3,4 кГц,
полосы подавления	0–0,2 и 4–8 кГц,
затухание в полосе подавления	25 дБ,
частота дискретизации	32 кГц.

Получите подходящую передаточную функцию фильтра, используя метод билинейного преобразования.

- 8.13. Требуется, чтобы цифровой фильтр устранял дрейф базовой линии и артефакты, вызванные движением тела в определенном биомедицинском приложении. Фильтр должен удовлетворять следующим требованиям:

полоса пропускания	1–30 Гц,
полосы подавления	0–0,5 и 40–128 Гц,
неравномерность в полосе пропускания	<0,1 дБ,
затухание в полосе подавления	>30 дБ,
частота дискретизации	256 Гц.

Определите порядок подходящего БИХ-фильтра и его передаточную функцию $H(z)$.

- 8.14. Для устранения сигнала-помехи требуется узкополосный режекторный фильтр. Фильтр должен удовлетворять следующим спецификациям:

граница полосы пропускания	45 и 55 Гц,
неравномерность в полосе пропускания	<0,1 дБ,
затухание в полосе подавления	>50 дБ,
частота дискретизации	500 Гц.

Вычислите коэффициенты фильтра.

- 8.15. Определите, используя метод инвариантного преобразования импульсной характеристики, передаточную функцию и разностное уравнение цифрового эквивалента однополосного RC-фильтра нижних частот. Частота дискретизации равна 150 Гц, частота среза по уровню 3 дБ — 30 Гц.
- 8.16. Стандартное аналоговое звено фильтра второго порядка с простыми полюсами можно выразить как

$$\frac{A_0 + A_1 s}{B_0 + B_1 s + B_2 s^2} = \frac{C_1}{s + p_1} + \frac{C_2}{s + p_2},$$

где C_1 и C_2 — коэффициенты элементарных дробей, а p_1 и p_2 — полюса на s -плоскости. Предполагается, что инвариантное преобразование импульсной характеристики звена фильтра второго порядка записывается так:

$$\begin{aligned} \frac{A_0 + A_1 s}{B_0 + B_1 s + B_2 s^2} &\rightarrow \frac{c_1 + c_2 - (c_1 e^{-p_1 T} + c_2 p^{-p_1 T}) z^{-1}}{1 - (e^{-p_1 T} + e^{-p_2 T}) z^{-1} + e^{-(p_1 + p_2) T} z^{-2}} = \\ &= \frac{a_0 - a_1 z^{-1}}{1 + b_1 z^{-1} + b_2 z^{-2}}, \end{aligned}$$

где T — интервал дискретизации.

1. Найдите выражения для p_1 , p_2 , C_1 и C_2 через A_0 , A_1 , B_0 , B_1 и B_2 .
2. Получите выражения для коэффициентов a_0 , a_1 , b_1 и b_2 , если полюса комплексно-сопряженные.
3. Повторите п. 2 при действительных неравных полюсах.
4. Для данной ниже нормированной аналоговой передаточной функции используйте полученные выше результаты и найдите коэффициенты эквивалентного фильтра дискретного времени. Частоту дискретизации предполагайте равной 10 кГц, частоту среза — 2 кГц.

$$H(s) = \frac{1}{s^2 + \sqrt{2}s + 1}.$$

- 8.17. Начав с подходящего аналогового фильтра нижних частот с характеристикой Чебышева, получите передаточную функцию цифрового режекторного фильтра, удовлетворяющего следующим спецификациям:

полоса подавления	10–15 кГц,
частота дискретизации	50 кГц,
неравномерность в полосе пропускания	0,5 дБ,
порядок фильтра	6.

- 8.18. Начав с подходящего аналогового фильтра нижних частот с характеристикой Чебышева и приведенного ниже биквадратного преобразования, определите с помощью метода отображения нулей и полюсов с s -плоскости на z -плоскость передаточную функцию цифрового полосового фильтра, удовлетворяющего следующим требованиям:

полоса подавления	10–15 кГц,
частота дискретизации	50 кГц,
неравномерность в полосе пропускания	0,5 дБ,
порядок фильтра	6.

Альтернативами билинейному z -преобразованию при разработке полосовых и режекторных фильтров являются следующие биквадратные преобразования [9, 10]:

$$\begin{aligned} s &= \operatorname{ctg} \left[\frac{(\omega_2 - \omega_1)T}{2} \right] \left[\frac{z^2 - 2z \cos \gamma + 1}{z^2 - 1} \right] && \text{нижних частот в полосовой} \\ s &= \operatorname{tg} \left[\frac{(\omega_2 - \omega_1)T}{2} \right] \left[\frac{z^2 - 1}{z^2 - 2z \cos \gamma + 1} \right] && \text{нижних частот в режекторный,} \end{aligned}$$

где

$$\cos \gamma = \cos \left[\frac{(\omega_2 + \omega_1)T}{2} \right] / \cos \left[\frac{(\omega_2 - \omega_1)T}{2} \right],$$

ω_1 и ω_2 — нижняя и верхняя граничные частоты (полосы пропускания для полосовых фильтров, полосы подавления для режекторных фильтров), а γ — центральная частота.

- 8.19. Аналоговый фильтр нижних частот характеризуется парой полюсов на s -плоскости в точках

$$p_{1,2} = -1,4 \pm 1,2936i.$$

Данный фильтр желательно преобразовать в цифровой полосовой фильтр с граничными частотами 3 и 5 кГц и частотой дискретизации 15 кГц. Зная, как цифровой фильтр нижних частот преобразовывается в полосовой фильтр, и используя билинейное z -преобразование в форме (8.40), определите

- а) полюса и нули цифрового полосового фильтра;
 - б) передаточную функцию фильтра в факторизованной форме.
- 8.20. Схема детектирования двухтонального многочастотного (ДТМЧ) сигнала в цифровом телефоне использует ряд фильтров Горцеля второго порядка для извлечения тонов ДТМЧ и их вторых гармоник. Если тоны для цифры “0” — 941 и 1336 Гц, определите значения коэффициентов в цепи обратной связи фильтров Горцеля для нижнего тона, 941 Гц, если для основной и второй гармоники используются соответственно значения $N = 205$ и $N = 210$, а номера соответствующих дискретных частот — 24 и 47.
- 8.21. 1. Объясните с помощью диаграммы принципы схемы детектирования двухтонального многочастотного (ДТМЧ) сигнала в кнопочных телефонах с использованием алгоритма Горцеля.
- А. Дискретное преобразование Фурье (ДПФ) последовательности данных $x(n)$, $n = 0, 1, \dots, N - 1$ можно определить следующим образом:

$$X(k) = \sum_{m=0}^{N-1} x(m) W_N^{km}, \quad k = 0, 1, \dots, N - 1,$$

где W_N^{km} — настраиваемый параметр.

- Б. Начав с приведенного выше уравнения, покажите, что передаточную функцию на z -плоскости $H_k(z)$ фильтра Горцеля, который используется для детектирования тонов ДТМЧ, можно записать в следующей рекурсивной форме:

$$H_k(z) = \frac{1 - W_N^k z^{-1}}{1 - 2 \cos \left(\frac{2\pi k}{N} \right) z^{-1} + z^{-2}}.$$

- В. Выведите выражение для квадрата амплитуды выхода фильтра Горцеля $|y_k(n)|^2$ в дискретное время $n = N$, а следовательно, покажите, что в модифицированном алгоритме Горцеля не требуется комплексная арифметика.

2. Схема детектирования тонов ДТМЧ для системы кнопочного телефона основана на спецификациях, приведенных в табл. 8.3, и использует фильтр Горцеля второго порядка.

Вычислите коэффициенты фильтра Горцеля для декодирования цифр в приемнике, если набран номер “00”.

Сравните вычислительную сложность алгоритма Горцеля с двоичным алгоритмом БПФ при использовании их для детектирования тонов ДТМЧ. Из результатов должно следовать, что в данном приложении нужно использовать алгоритм Горцеля. Укажите предположения, сделанные при решении задачи.

Задачи для решения с помощью MATLAB

- 8.22. Требуется, чтобы цифровой полосовой фильтр с характеристикой Чебышева удовлетворял следующим спецификациям:

полоса пропускания	1200–1800 Гц,
затухание в полосе подавления	>30 дБ,
неравномерность в полосе пропускания	<0,5 дБ,
ширина полосы перехода	400 Гц,
частота дискретизации	7,5 кГц.

Используя подходящую программу, получите коэффициенты фильтра.

- 8.23. Аналоговый фильтр нужно преобразовать в эквивалентный цифровой, который будет работать с частотой дискретизации 256 Гц. Аналоговый фильтр имеет следующую передаточную функцию:

$$H(s) = \frac{1}{s^3 + 2s^2 + 2s + 1}.$$

1. Получите подходящие коэффициенты для цифрового фильтра.
 2. Предполагая, что цифровой фильтр нужно реализовать с использованием каскадной структуры, изобразите подходящую блок-схему реализации и выведите разностное уравнение.
 3. Повторите п. 2 для параллельной структуры.
- 8.24. БИХ-фильтр имеет следующую передаточную функцию:

$$H(z) = \frac{0,1436 + 0,2871z^{-1} + 0,1436z^{-2}}{1 - 1,8353z^{-1} + 0,9748z^{-2}}.$$

1. Определите положения полюсов и нулей и изобразите диаграмму нулей и полюсов.
2. Определите радиальное расстояние полюсов от начала координат.
3. Оцените число битов, требуемых для представления каждого коэффициента, если
 - а) требуется поддерживать устойчивость;
 - б) амплитудная характеристика в полосе пропускания на должна меняться более чем на 1%.

8.25. Сравните согласованное z -преобразование, инвариантное преобразование импульсной характеристики и билинейное z -преобразование с точки зрения

- а) влияния эффекта Найквиста на амплитудную характеристику, фазовую и групповую задержки;
- б) распределения диаграмм нулей и полюсов.

В качестве инструментов при исследовании используйте MATLAB и следующие фильтры.

1. Фильтр нижних частот.

Эллиптический цифровой фильтр нижних частот со следующими спецификациями:

полоса пропускания	0–1 кГц,
полоса подавления	3–5 кГц,
неравномерность в полосе пропускания	1 дБ,
затухание в полосе подавления	60 дБ,
частота дискретизации	10 кГц.

2. Фильтр верхних частот.

Эллиптический цифровой фильтр верхних частот со следующими спецификациями:

полоса подавления	0–1 кГц,
полоса пропускания	3–5 кГц,
неравномерность в полосе пропускания	1 дБ,
затухание в полосе подавления	60 дБ,
частота дискретизации	10 кГц.

3. Полосовой фильтр.

А. Цифровой полосовой фильтр Баттерворта со следующими спецификациями:

полоса пропускания	200–300 Гц,
частота дискретизации	2000 Гц,
порядок фильтра	8.

Б. Цифровой полосовой фильтр Баттерворта со следующими спецификациями:

полоса пропускания	800–900 Гц,
частота дискретизации	2000 Гц,
порядок фильтра	8.

4. Заграждающий фильтр.

Эллиптический цифровой режекторный фильтр со следующими спецификациями:

полосы пропускания	0–15 и 30–50 кГц,
полоса подавления	20–25 кГц,
неравномерность в полосе пропускания	0,2 дБ,
затухание в полосе подавления	40 дБ,
частота дискретизации	100 кГц.

8.26. Сравните особенности цифровых фильтров Баттерворта, Чебышева типа I, Чебышева типа II и эллиптического фильтра с точки зрения

- а) распределения их полюсов и нулей;
- б) порядка фильтра;
- в) ширины полосы перехода, неравномерности в полосе пропускания и полосе подавления амплитудно-частотной характеристики;
- г) характеристик фазовой и групповой задержки.

При исследовании используйте следующие фильтры.

1. Фильтр нижних частот:

полоса пропускания	0–500 Гц,
полоса подавления	2–4 кГц,
неравномерность в полосе пропускания	3 дБ,
затухание в полосе подавления	20 дБ,
частота дискретизации	8 кГц.

2. Фильтр верхних частот:

полоса подавления	0–500 Гц,
полоса пропускания	2–4 кГц,
неравномерность в полосе пропускания	3 дБ,
затухание в полосе подавления	20 дБ,
частота дискретизации	8 кГц.

3. Полосовые фильтры:

	нижняя граничная частота полосы пропускания	250 Гц,
	верхняя граничная частота полосы пропускания	300 Гц,
	нижняя граничная частота полосы подавления	50 Гц,
а)	верхняя граничная частота полосы подавления	450 Гц,
	неравномерность в полосе пропускания	3 дБ,
	затухание в полосе подавления	20 дБ,
	частота дискретизации	1 кГц.
	полосы подавления	0–15 и 30–50 кГц,
б)	полоса пропускания	20–25 кГц,
	неравномерность в полосе пропускания	0,2 дБ,
	затухание в полосе подавления	40 дБ,
	частота дискретизации	100 кГц.

4. Режекторные фильтры:

а)	полосы пропускания	0–50 и 450–500 кГц,
	полоса подавления	250–300 кГц,
	неравномерность в полосе пропускания	3 дБ,
	затухание в полосе подавления	20 дБ,
	частота дискретизации	1 кГц.
б)	полосы пропускания	0–15 и 30–50 кГц,
	полоса подавления	20–25 кГц,
	неравномерность в полосе пропускания	0,2 дБ,
	затухание в полосе подавления	40 дБ,
	частота дискретизации	100 кГц.

8.27.1, а. С помощью билинейного z -преобразования и MATLAB определите передаточную функцию фильтра, предназначенного для аудиообработки сигнала в цифровом микшере, если при добротности 2 усиление (максимальное) на частоте 48 кГц равно 6,02 дБ. Частота дискретизации равна 48 кГц, а передаточная функция эквивалентного аналогового фильтра для s -плоскости приведена ниже. Изобразите амплитудную характеристику, характеристики фазовой и групповой задержки и диаграмму нулей и полюсов дискретного фильтра.

- 1, б. Повторите задачу для частоты дискретизации 96 кГц.
2. Повторите задачи 1, а и б, используя метод согласованного z -преобразования.
3. Повторите задачи 1, а и б, используя метод инвариантного преобразования импульсной характеристики.
4. Сравните результаты, полученные в задачах 1 и 3.

$$H(s) = \frac{s^2 + (3 + k)\frac{\omega_0}{Q}s + \omega_0^2}{s^2 + (3 - k)\frac{\omega_0}{Q}s + \omega_0^2},$$

где

$$k = 2 \left(\frac{G - 1}{G + 1} \right); \quad \omega_0 = \text{частота усиления}$$

G = коэффициент усиления; Q = добротность.

- 8.28. 1. Напишите на языке С псевдокод для детектирования тонов ДТМЧ и их вторых гармоник. Укажите использованные предположения.
2. Повторите п. 1, используя MATLAB.
3. С помощью MATLAB сгенерируйте подходящие частотные тоны и используйте их для проверки программы MATLAB, написанной для детектирования тонов ДТМЧ.

Литература

1. Abu-el-Haija A. and Al-Ibrahim M. M. (1986) Improving performance of digital sinusoidal oscillators by means of error feedback circuits. *IEEE Trans. Circuits and Systems*, **33**(4), 373–380.
2. Antoniou A. (1979) *Digital Filters Analysis and Design*. New York: McGraw-Hill.
3. Chen C. J. (1996) *Modified Goertzel Algorithm in DTMF Detection using the TMS320C80*, Texas Instruments, Application Report SPRA066, June.
4. Clark R. J., Ifeachor E. C. and Rogers G. M. (1996) Real-time equaliser coefficient realisation with minimised computational load and distortion. Preprint 4360, 101st Audio Engineering Society Convention.
5. Clark R. J., Ifeachor E. C., Rogers G. M. and Van Eetvelt P. W. J. (2000) Techniques for generating digital equaliser coefficients. *J. Audio Engineering Society*, **48**(4), 281–298.
6. Dattorro J. (1988) The implementation of recursive digital filters for high-fidelity audio. *J. Audio Engineering Society*, **36**(11), 851–878.
7. DeFatta D. J., Lucas J. G. and Hodgkiss W. S. (1988) *Digital Signal Processing*. New York: Wiley.
8. Feeney S. L., Kiebert R. B., Mina K. V. and Tewksbury S. K. (1971) Design of digital filters for an all digital frequency division multiplex-time division multiplex translator. *IEEE Trans. Circuit Theory*, **18**, 702–711.
9. Gold B. and Rader C. M. (1969) *Digital Processing of Signals*. New York: McGraw-Hill.
10. Gray A. H. and Markel J. D. (1976) A computer program for designing elliptic filters. *IEEE Trans. Acoustics, Speech and Signal Processing*, **24**(6), 529–538.
11. IEEE (1979) *Programs for Digital Signal Processing*. New York: IEEE Press.
12. Jackson L. B. (1986) *Digital Filters and Signal Processing*. Boston MA: Kluwer.
13. Jackson L. B., Kaiser J. F. and McDonald H. S. (1968) An approach to the implementation of digital filters. *IEEE Trans. Audio and Electroacoustics*, **16**(3), 413–421.
14. Jong M. T. (1982) *Methods of Discrete Signal System Analysis*. New York: McGraw-Hill.
15. Lynn P. A. and Fuerst W. (1989) *Introductory Digital Signal Processing with Computer Applications*. Chichester: Wiley.
16. Marvin C. (1990) General-purpose tone decoding and DTMF detection, in *Theory, Algorithms, and Implementations, Digital Signal Processing Applications with the TMS320 Family*, Vol. 2, literature number SPRA016, Texas Instruments.
17. Mock P. (1985) Add DTMF generation and decoding to DSP- μ p designs. *EDN*, **30**.
18. Parks T. W. and Burrus C. S. (1987) *Digital Filter Design*. New York: Wiley.
19. Rabiner L. R. and Gold B. (1975) *Theory and Applications of Digital Signal Processing*. Englewood Cliffs NJ: Prentice-Hall.
20. Rader C. M. and Gold B. (1967) Effects of parameter quantization on the poles of a digital filter. *Proc. IEEE*, **55**, 688, 689.
21. Smithson P. (1992) Clock recovery for a satellite data modem, University of Plymouth (personal communication).
22. Stanley W. D., Dougherty G. R. and Dougherty R. (1984) *Digital Signal Processing*, 2nd edn. Reston VA: Reston Publishing, Inc.
23. Stearns S. D. and Hush D. R. (1990) *Digital Signal Analysis*, 2nd edn. Englewood Cliffs NJ: Prentice-Hall.

Дополнительная литература

- Abu-el-Haija A. I. and Peterson A. M. (1979) An approach to eliminate roundoff errors in digital filters. *IEEE Trans. Acoustics, Speech and Signal Processing*, **27**, 195–198.
- Ahmed N. and Natarajan T. (1983) *Discrete-time Signals and Systems*. Reston VA: Reston Publishing, Inc.
- Allen J. (1975) Computer architecture for signal processing. *Proc. IEEE*, **63**(4), 624–648.

- Arjmand M. and Roberts R. A. (1981) On comparing hardware implementations of fixed point digital filters. *IEEE Circuits Systems Mag.*, **3**(2), 2–8.
- Avenhaus E. (1972) Filters with coefficients of limited wordlength. *IEEE Trans. Audio Electroacoustics*, **20**, 206–212.
- Barnes C. W., Tran B. N. and Leung S. H. (1985) On the statistics of fixed-point roundoff error. *IEEE Trans. Acoustics, Speech and Signal Processing*, **33**, 595–606.
- Bellanger M. (1984) *Digital Processing of Signals. Theory and Practice*. New York: Wiley.
- Chang T. L. (1978) A low roundoff noise digital filter structure. In *Proc. IEEE Int. Symp. on Circuits and Systems*, May 1978, pp. 1004–1008.
- Chang T. L. (1979) Error-feedback digital filters. *Electronics Lett.*, **15**, 348–349.
- Chang T. L. (1980) Comments on “An approach to eliminate roundoff errors in digital filters”. *IEEE Trans. Acoustics, Speech and Signal Processing*, **28**(2), 244, 245.
- Chang T. L. (1981) Suppression of limit cycles in digital filters designed with one magnitude-truncation quantizer. *IEEE Trans. Circuits and Systems*, **28**(2), 107–111.
- Chang T. L. (1981) On low-roundoff noise and low-sensitivity digital filter structures. *IEEE Trans. Acoustics, Speech and Signal Processing*, **29**(5), 1077–1080.
- Chang T. L. and White S. A. (1981) An error cancellation digital-filter structure and its distributed-arithmetic implementation. *IEEE Trans. Circuits and Systems*, **28**(4), 339–342.
- Charalambous C. and Best M. J. (1974) Optimization of recursive digital filters with finite wordlengths. *IEEE Trans. Acoustics, Speech and Signal Processing*, **22**(6), 424–431.
- Chassaing R. and Horning D. W. (1990) *Digital Signal Processing with the TMS320C25*. New York: Wiley.
- Claasen T. A. C. M. and Kristiansson L. O. G. (1975) Necessary and sufficient conditions for the absence of overflow phenomena in a second order recursive digital filter. *IEEE Trans. Acoustics, Speech and Signal Processing*, **23**(6), 509–515.
- Claasen T. A. C. M., Mecklenbrauker W. F. G. and Peek J. B. H. (1973) Second-order digital filter with only one magnitude-truncation quantiser and having practically no limit cycles. *Electronics Lett.*, **9**, 531, 532.
- Claasen T. A. C. M., Mecklenbrauker W. F. G. and Peek J. B. H. (1973) Some remarks on the classification of limit cycles in digital filters. *Philips Research Rep.*, **28**, 297–305.
- Claasen T., Mecklenbrauker W. F. G. and Peek J. B. H. (1976) Effects of quantization and overflow in recursive digital filters. *Philips Research Rep.*, **28**, 297–305.
- Clark R. J., Ifeachor E. C., Rogers G. M. and Van Eetvelt P. W. J. (2000) Techniques for generating digital equaliser coefficients. *Journal of Audio Engineering Society*, **48**(4), 281–298.
- Crochiere R. E. (1975) A new statistical approach to the coefficient wordlength problem for digital filters. *IEEE Trans. Circuits and Systems*, **22**, 190–196.
- Crochiere R. E. and Oppenheim A. V. (1975) Analysis of linear digital networks. *Proc. IEEE*, **63**(4), 581–594.
- Diniz P. S. R. and Antoniou A. (1985) Low-sensitivity digital filter structures which are amenable to error-spectrum shaping. *IEEE Trans. Circuits and Systems*, **32**(10), 1000–1007.
- Elliot D. F. (ed.) (1987) *Handbook of Digital Signal Processing*. London: Academic Press.
- IEEE (1978) Digital Signal Processing II. Institute of Electrical and Electronics Engineers.
- Jackson L. B. (1970) On the interaction of roundoff noise and dynamic range in digital filters. *BSTJ*, **49**(2), 159–184.
- Jackson L. B. (1976) Roundoff noise bounds derived from coefficient sensitivities for digital filters. *IEEE Trans. Circuits and Systems*, **23**(8), 481–485.
- Knowles J. B. and Olcayto E. M. (1968) Coefficient accuracy and digital filter response. *IEEE Trans. Circuit Theory*, **15**, 31–41.
- Liu B. (1971) Effect of finite wordlength on the accuracy of digital filters — a review. *IEEE Trans. Circuit Theory*, **18**, 670–677.
- Liu B. and Kaneko T. (1969) Error analysis of digital filters realized with floating-point arithmetic. *Proc. IEEE*, **57**(10), 1735–1747.
- Markel J. D. and Gray A. H. (1975) Fixed-point implementation algorithms for a class of orthogonal polynomial filter structures. *IEEE Trans. Acoustics, Speech and Signal Processing*, **23**(5), 486–494.
- Markel J. D. and Gray A. H. (1975) Roundoff noise characteristics of a class of orthogonal polynomial structures. *IEEE Trans. Acoustics, Speech and Signal Processing*, **23**(5), 473–486.
- Motorola (1988) *Digital Stereo 10-band Graphic Equalizer Using the DSP56001*. Motorola Application Note.

- Mullis C. T. and Roberts R. A. (1976) Round-off noise in digital filters: frequency transformations and invariants. *IEEE Trans. Acoustics, Speech and Signal Processing*, **24**(6), 538–550.
- Munson D. C. and Liu B. (1980) Low-noise realization for narrow-band recursive digital filters. *IEEE Trans. Acoustics, Speech and Signal Processing*, **28**, 41–54.
- Nagle H. T. and Nelson V. P. (1981) Digital filter implementation on 16 bit microcomputers. *IEEE Micro*, **1**, 23–41.
- Oppenheim A. V. and Schaffer R. W. (1975) *Digital Signal Processing*. Englewood Cliffs NJ: Prentice-Hall.
- Oppenheim A. V. and Weinstein, C. J. (1972) Effects of finite register length in digital filtering and the fast Fourier transform. *Proc. IEEE*, **60**, 957–976.
- Peled A., Liu B. and Steiglitz K. (1974) A new hardware realization of digital filters. *IEEE Trans. Acoustics, Speech and Signal Processing*, **22**, 456–462.
- Peled A., Liu B. and Steiglitz K. (1975) A note on implementation of digital filters. *IEEE Trans. Acoustics, Speech and Signal Processing*, **23**, 387–389.
- Rabiner L. R., Cooley J. W., Helms H. D., Jackson L. B., Kaiser J. F., Rader C. M., Schaffer R. W., Steiglitz K. and Weinstein C. J. (1972) Terminology in digital signal processing. *IEEE Trans. Audio and Electroacoustics*, **20**, 322–337.
- Sandberg I. W. and Kaiser J. F. (1972) A bound on limit cycles in fixed-point implementations of digital filters. *IEEE Trans. Audio and Electroacoustics*, **20**, 110–112.
- Schmalzel J. L., Heine D. N. and Ahmed N. (1980) Some pedagogical considerations of digital filter hardware implementation. *IEEE Circuits and Systems Mag.*, **2**(1), 4–13.
- Sim P. K. and Pang K. K. (1985) Effects of input-scaling on the asymptotic overflow-stability properties of second recursive digital filters. *IEEE Trans. Circuits and Systems*, **32**(10), 1008–1015.
- Steiglitz K. (1971) Designing short-word recursive digital. *Proc. 9th Ann. Allerton Conf. on Circuit and System Theory*, 6–8 October, 1971, pp. 778–788.
- Steiglitz K., Bede L. and Liu B. (1976) An improved algorithm for ordering poles and zeros of fixed-point recursive digital filters. *IEEE Trans. Acoustics, Speech and Signal Processing*, **24**, 341–343.
- Taylor F. J. (1983) *Digital Filter Design Handbook*. New York: Marcel Dekker.
- Thong T. (1976) Finite wordlength effects in the ROM digital filter. *IEEE Trans. Acoustics, Speech and Signal Processing*, **24**, 436, 437.
- Thong T. and Liu B. (1977) Error spectrum shaping in narrowband recursive digital filters. *IEEE Trans. Acoustics, Speech and Signal Processing*, **25**, 200–203.
- Williamson D. and Sridharan S. (1985) An approach to coefficient wordlength reduction in digital filters. *IEEE Trans. Circuits and Systems*, **32**(9), 893–903.

Приложения

8.А. Программы на С для разработки БИХ-фильтров

При работе над этим изданием было написано несколько программ разработки цифровых БИХ-фильтров на языке С, но из-за недостатка места в окончательный вариант книги вошла только программа вычисления коэффициентов методом инвариантного преобразования импульсной характеристики. Остальные программы (для билинейного преобразования; вычисления коэффициентов для классических фильтров (Баттерворта, Чебышева и эллиптических) с помощью билинейного преобразования; анализа влияния конечной разрядности для БИХ-фильтров) можно найти на компакт-диске к книге [Ifeachor, 2001]. Кроме того в Web (подробности см. в предисловии) можно найти m-файлы MATLAB и использовать их для выполнения аналогичных заданий.

8.A.1. Реализации на С инвариантного преобразования импульсной характеристики

Обратите внимание на то, что в программах на С коэффициенты A_k и B_k аналоговой передаточной функции и коэффициенты a_k и b_k дискретных фильтров — это векторы коэффициентов числителя и знаменателя соответственно. В отличие от обозначений в программах, в уравнениях (8A.1)–(8A.8) эти коэффициенты обозначены так, чтобы согласовываться с коэффициентами, использованными в m-файлах MATLAB.

Реализация инвариантного преобразования импульсной характеристики на С приведена в программе 8A.1. Подытожим вначале концепции, на которых основана программа, а затем проиллюстрируем ее использование на примере. Ниже будет рассматриваться звено фильтра второго порядка, поскольку оно является основным стандартным блоком цифровых БИХ-фильтров.

Рассмотрим общую передаточную функцию второго порядка на s -плоскости, которую нужно преобразовать в дискретную передаточную функцию:

$$H(s) = \frac{B_0 + B_1 s}{A_0 + A_1 s + A_2 s^2}. \quad (8A.1)$$

Эквивалентная передаточная функция на z -плоскости — это также звено второго порядка, имеющее вид

$$H(z) = \frac{b_0 + b_1 z^{-1}}{1 + a_1 z^{-1} + a_2 z^{-2}}. \quad (8A.2)$$

Программа 8A.1. Метод инвариантного преобразования импульсной характеристики (*Примечание:* по сравнению с уравнениями, приведенными выше, коэффициенты A и B , а также a и b имеют обратное значение)

```

/*-----*
*   Метод инвариантного преобразования импульсной характеристики   *
*   *                                                                 *
*   Перед использованием программы аналоговую передаточную функцию *
*   следует нормировать                                              *
*   30.10.92                                                         *
*   *                                                                 *
*-----*
*/
#include <stdio.h>
#include <math.h>
#include <dos.h>

void dfilter();
double T;
double a0, a1, a2, b0, b1, b2;
double p1, p2, pr, pi;
double c1, c2, cr, ci;
float A0, A1, B0, B1, B2, temp;
main()
{

```

```

    /* инициализация коэффициентов */
    A0=0; A1=0; B0=1; B1=0; B2=0;
    a0=0; a1=0; b0=1; b1=0; b2=0;
    c1=0; c2=0; p1=0; p2=0; a2=0;

    /* считываются коэффициенты s-плоскости */
    printf("impulse invariant discrete filters \n");
    printf("\n");
    printf("enter s-plane coefficients \n");
    printf("enter denominator coeffs: B0, B1, B2 \n");
    scanf("%f %f %f", &B0, &B1, &B2);
    printf("enter numerator coeffs: A0, A1 \n");
    scanf("%f %f", &A0, &A1);
    T=1;
    dfilter();
    printf("\n");
    printf("press enter to continue\n");
    getch();
    exit(0);
}
/*-----*/
void dfilter()
{
    /* находятся полюса на s-плоскости */
    temp = B1*B1 - 4*B0*B2;
    if(B2==0){ /* один полюс */
        p1=-B0/B1;
        a0=A0/B1;
        b1=-exp(p1*T);
    }
    if(temp.0){ /* действительные неравные полюса */
        pr=-B1/(2*B2);
        pi=(pr*pr)-B0/B2;
        pi=sqrt(pi);
        p1=pr+pi;
        p2=pr-pi;
        c1=(A0+A1*p1)/((p1-p2)*B2);
        c2=A1/B2-c1;
        a0=c1+c2;
        a1=-(c1*exp(p2*T) + c2*exp(p1*T));
        b1=-exp(p1*T)-exp(p2*T);
        b2=exp((p1+p2)*T);
    }
    if(temp,0){ /* комплексно-сопряженные полюса */
        pr=-B1/(2*B2);
        pi=(pr*pr)-B0/B2;
        pi=sqrt(-pi);
        cr=A1/(B2*2);
        ci=-(A0+A1*pr)/(2*pi*B2);
        a0=2*cr;
        a1=-(cr*cos(pi*T)+ci*sin(pi*T))*2*exp(pr*T);
        b1=-2*exp(pr*T)*cos(pi*T);
    }
}

```

```

        b2=exp(2*pr*T);
    }
    printf("discrete filter coeffs: \n");
    printf("a0 a1 a2: \t%f %f %f \n", a0, a1, a2);
    printf("b0 b1 b2: \t%f %f %f \n", b0, b1, b2);
}

```

Для данных коэффициентов аналоговой передаточной функции $H(s)$ программа на языке С (программа (8А.1)) вычисляет коэффициенты эквивалентной передаточной функции в z -области $H(z)$. Чтобы понять, как работает программа, установим связь между коэффициентами $H(s)$ и $H(z)$.

Используя разложение на элементарные дроби, передаточную функцию на s -плоскости в формуле (8А.1) можно выразить следующим образом:

$$\frac{B_0/A_2 + (B_1/A_2)s}{A_0/A_2 + (A_1/A_2)s + s^2} = \frac{c_1}{s - p_1} + \frac{c_2}{s - p_2}, \quad (8А.3)$$

где p_1 и p_2 — полюса $H(s)$ на s -плоскости, которые определяются выражением

$$p_{1,2} = \frac{-A_1}{2A_2} \pm \left[\left(\frac{A_1}{2A_2} \right)^2 - \frac{A_0}{A_2} \right]^{1/2}. \quad (8А.4)$$

Умножая обе части уравнения (8А.3) на $(s - p_1)(s - p_2)$ и приравнивая коэффициенты при s и свободных членах, получим

$$\frac{B_0}{A_2} = -(c_1 p_2 + c_2 p_1), \quad (8А.5, a)$$

$$\frac{B_1}{A_2} = c_1 + c_2. \quad (8А.5, б)$$

Из этой системы находим c_1 и c_2 :

$$c_1 = \frac{B_0 + B_1 p_1}{(p_1 - p_2) A_2}, \quad (8А.6, a)$$

$$c_2 = \frac{B_1}{A_2} - c_1. \quad (8А.6, б)$$

Применяя к выражению для $H(s)$ в форме (8А.3) инвариантное преобразование импульсной характеристики, получаем дискретную передаточную функцию $H(z)$:

$$\begin{aligned} H(z) &= \frac{c_1 + c_2 - (c_1 e^{p_2 T} + c_2 e^{p_1 T}) z^{-1}}{1 - (e^{p_1 T} + e^{p_2 T}) z^{-1} + e^{(p_1 + p_2) T} z^{-2}} = \\ &= \frac{b_0 + b_1 z^{-1}}{1 + a_1 z^{-1} + a_2 z^{-2}}, \end{aligned} \quad (8А.7)$$

где

$$\begin{aligned} b_0 &= c_1 + c_2, & b_1 &= -(c_1 e^{p_2 T} + c_2 e^{p_1 T}) \\ a_1 &= -(e^{p_1 T} + e^{p_2 T}), & a_2 &= e^{(p_1 + p_2) T}, \end{aligned}$$

p_1 и p_2 определены в уравнении (8А.4), а c_1 и c_2 — в уравнении (8А.6).

Итак, зная коэффициенты фильтра второго порядка на s -плоскости (т.е. A_0 , A_1 , A_2 , B_0 и B_1), коэффициенты эквивалентного дискретного фильтра можно получить непосредственно с помощью приведенных выше соотношений. Вычисление коэффициентов $H(z)$ в уравнении (8А.7) зависит от типа полюсов на s -плоскости p_1 и p_2 . На практике возможны три варианта: 1) действительные и неравные; 2) пара комплексно-сопряженных; 3) действительные равные (т.е. совпадающие полюса). Поскольку третий случай встречается редко и является более сложным, рассмотрим только первые два.

В первом случае уравнение (8А.7) можно использовать непосредственно для получения коэффициентов $H(z)$. Во втором случае можно применять упрощенную форму (8А.7) и избежать комплексной арифметики. Используя свойства полюсов, перепишем уравнение (8А.7) для второго варианта в виде

$$\begin{aligned} H(z) &= \frac{(c_1 + c_1^*) - (c_1 e^{p_1^* T} + c_1^* e^{p_1 T}) z^{-1}}{1 - (e^{p_1 T} + e^{p_1^* T}) z^{-1} + e^{(p_1 + p_1^*) T} z^{-2}} \\ &= \frac{2c_{Re} - [c_{Re} \cos(p_{Im} T) + c_{Im} \sin(p_{Im} T)] 2e^{p_{Re} T} z^{-1}}{1 - 2e^{p_{Re} T} \cos(p_{Im} T) z^{-1} + e^{2p_{Re} T} z^{-2}}, \end{aligned}$$

где p_{Re} — действительная часть p_1 , p_{Im} — мнимая часть p_1 , c_{Re} — действительная часть c_1 , а c_{Im} — мнимая часть c_1 . Из уравнения (8А.4) находим действительную и мнимую части p_1 :

$$p_{Re} = -\frac{A_1}{2A_2}, \quad p_{Im} = \left\{ -\left[\left(\frac{A_1}{2A_2} \right)^2 - \frac{A_0}{A_2} \right] \right\}^{1/2}.$$

Из уравнения (8А.6) находим коэффициент элементарной дроби c_1 :

$$c_1 = \frac{B_1}{2A_2} - \frac{B_0 + B_1 p_{Re}}{2p_{Im} A_2} i = c_{Re} + i c_{Im}.$$

Следовательно, если полюса $H(s)$ являются комплексно-сопряженными, в стандартной форме коэффициенты передаточной функции второго порядка на z -плоскости записываются так:

$$\begin{aligned} b_0 &= 2c_{Re}, \quad b_1 = -[c_{Re} \cos(p_{Im} T) + c_{Im} \sin(p_{Im} T)] 2e^{p_{Re} T} \\ a_1 &= -2e^{p_{Re} T} \cos(p_{Im} T), \quad a_2 = e^{2p_{Re} T}. \end{aligned}$$

Пример 8А.1

Используем программу 8А.1 для расчета коэффициентов дискретного фильтра из примера 8.4.

Предполагается, что частоты нормированы. При частоте дискретизации 1280 Гц и частоте среза 150 Гц нормированная частота среза равна 150/1280. Передаточная

функция вначале масштабируется путем замены s на s/α , где $\alpha = 2\pi \times 150/1280 = 0,73631$:

$$H'(s) = \frac{\alpha^2}{\alpha^2 + \sqrt{2}\alpha s + s^2} = \frac{1}{1 + (\sqrt{2}/\alpha)s + (1/\alpha^2)s^2} = \frac{1}{1 + 1,920675s + 1,84496s^2}.$$

Ниже приводятся запросы и выход программы. Рассчитанные дискретные коэффициенты идентичны значениям, полученным в примере 8.4.

```
impulse invariant discrete filters
enter s-plane coefficients
enter denominator coeffs: A0, A1, A2
1 1.920675 1.84496
enter numerator coeffs: B0, B1
1 0
discrete filter coeffs:
b0 b1 b2: 0.000000 0.307718 0.000000
a0 a1 a2: 1.000000 -1.030953 0.353088
```

Приведенный листинг позволяет сразу записать передаточную функцию на z -плоскости:

$$H(z) = \frac{0,307718z^{-1}}{1 - 1,030953z^{-1} + 0,353088z^{-2}}.$$

Коэффициенты идентичны значениям, полученным в примере 8.4.

8.Б. Разработка БИХ-фильтров с помощью MATLAB

Средство MATLAB Signal Processing Toolbox предлагает много полезных функций для разработки и анализа классических БИХ-фильтров (например, Баттерворта, Чебышева типа I и II, эллиптических фильтров) для данного набора спецификаций (например, граничные частоты полос пропускания и подавления, неравномерность в полосе пропускания и затухание в полосе подавления).

В частности, Toolbox предлагает функции преобразования классических аналоговых фильтров в эквивалентные фильтры дискретного времени.

Напомним, что критичным этапом разработки цифрового БИХ-фильтра является расчет коэффициентов. Для классических цифровых БИХ-фильтров можно следующим образом подытожить действия на этом этапе.

1. Задать дискретный фильтр.
2. Определить подходящий аналоговый фильтр нижних частот, например, с характеристикой Баттерворта, Чебышева типа I, Чебышева типа II или эллиптической характеристикой.
3. Преобразовать аналоговый фильтр-прототип в фильтр нижних частот, верхних частот, полосовой или режекторный.
4. Перевести преобразованный фильтр в эквивалентный фильтр дискретного времени (например, используя метод инвариантного преобразования импульсной характеристики или билинейного z -преобразования).

Средство Signal Processing Toolbox предлагает несколько высокоуровневых функций для одновременного или поочередного выполнения этапов 2–4. Приведем, например, синтаксис команды MATLAB для создания фильтра нижних частот, верхних частот, полосового или режекторного фильтра с характеристикой Баттерворта (этапы 1 и 2):

```
[b,a]=butter(N, Wc, options)
[z,p,k]=butter(N, Wc, options)
```

Первая команда определяет коэффициенты числителя и знаменателя фильтра Баттерворта дискретного времени N -го порядка с частотой среза по уровню 3 дБ (или граничной частотой) W_c , нормированной на частоту Найквиста. Коэффициенты числителя и знаменателя фильтра возвращаются в векторы b и a соответственно по возрастающим отрицательным степеням z .

Если опустить слово “options”, команда по умолчанию рассчитывает фильтр нижних частот (если W_c является вектором частот, по умолчанию рассчитывается полосовой фильтр). Для вычисления коэффициентов фильтра верхних частот и режекторного фильтра нужно соответственно использовать опции “high” и “stop”. Для полосовых и режекторных фильтров W_c — это двухкомпонентный вектор, задающий частоты среза (или граничные частоты):

$$W_c = [\omega_{c1}, \omega_{c2}],$$

где $\omega_{c1} < \omega < \omega_{c2}$ — полоса пропускания (полосовые фильтры) или полоса подавления (режекторные фильтры).

Вторая команда возвращает положения нулей (z) и полюсов (p) в ПДСК и коэффициент усиления фильтра k .

Подобные команды существуют и для других классических фильтров. Например, для фильтров Чебышева типа I и типа II и эллиптических фильтров используются команды MATLAB со следующим синтаксисом:

```
[b,a]=cheby1(N, Ap, Wc, options)
[z,p,k]=cheby1(N,Ap,Wc, options)
[b,a]=cheby2(N, As, Wc, options)
[z,p,k]=cheby2(N,As,Wc, options)
[b,a]=ellip(N, Ap, As, Wc, options)
[z,p,k]=ellip(N,Ap,As,Wc, options)
```

Здесь Ap и As — неравномерность в полосе пропускания и затухание в полосе подавления в децибелах соответственно.

Существуют и другие полезные команды, которые можно использовать для выполнения промежуточных задач в процессе вычисления коэффициентов. Например, команды `buttord`, `chebylors` и `ellipord` можно использовать для определения порядка подходящего фильтра. Параметры подходящих аналоговых фильтров-прототипов нижних частот можно определить с помощью команд `butterp`, `cheby1p`, `cheby2p` и `ellipp`.

Пример 8Б.1

Разработка простого фильтра нижних частот с использованием инвариантного преобразования импульсной характеристики и MATLAB. Требуется БИХ-фильтр нижних частот с характеристикой Баттерворта, удовлетворяющий следующим спецификациям:

частота среза 150 Гц,
 частота дискретизации 1,28 кГц,
 порядок фильтра, N 2.

1. Определите, используя инвариантное преобразование импульсной характеристики и MATLAB:
 - а) коэффициенты, полюса и нули частотно-масштабированного аналогового фильтра;
 - б) коэффициенты, полюса и нули дискретного БИХ-фильтра. Запишите его передаточную функцию.
2. Постройте амплитудно-частотную характеристику и диаграмму нулей и полюсов дискретного фильтра.

Решение

Реализация решения задачи в форме m-файла MATLAB приведена в программе 8Б.1.

- 1, а. Используя m-файл, получаем такие коэффициенты, полюса и нули аналогового фильтра:

Коэффициенты: $b = 1,0e + 005 * [0; 0; 8,8826],$
 $a = 1,0e + 005 * [0,00001; 0,0133; 8,8826].$

Полюса: $1,0e + 002 * (-6,6643 \pm 6,6643i).$

Нули: Нет.

Усиление: $8,8826e + 005.$

- 1, б. Используя m-файл, получаем такие коэффициенты, полюса и нули аналогового фильтра:

Коэффициенты: $b = [0; 0,3078; 0],$
 $a = [1,0000; -1,0308; 0,3530].$

Полюса: $0,5154 \pm 0,2955i.$

Нули: 0.

На основании коэффициентов записываем передаточную функцию фильтра:

$$H(z) = \frac{0,3078z^{-1}}{1 - 1,0308z^{-1} + 0,3553z^{-2}}.$$

Результаты идентичны данным, полученным в примере 8.4.

2. Амплитудно-частотная характеристика и диаграмма нулей и полюсов фильтра изображены на рис. 8Б.1.

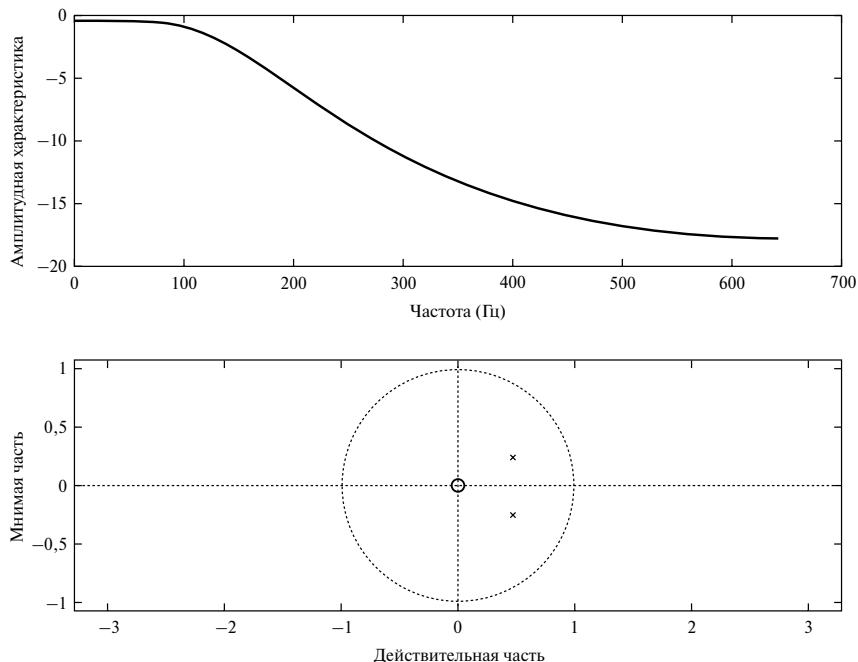


Рис. 8Б.1

Программа 8Б.1. Реализация решения примера 8Б.1 в форме m-файла MATLAB

```
%
% Название программы: EX8B1.m
% Простой фильтр нижних частот
N=2;           % Порядок фильтра
Fs=1280;       % Частота дискретизации
fc=150;        % Частота среза
WC=2*pi*fc;    % Частота среза в радианах
[b, a]=butter (N,WC,'s'); % Создать аналоговый фильтр
[z, p, k]=butter (N, WC, 's');
[bz, az]=impinvar (b, a, Fs); % Преобразовать в дискретный фильтр
subplot (2,1,1) % Вывести на экран амплитудно-частотную характеристику
[H, f]=freqz(bz, az, 512, Fs);
plot(f, 20*log10(abs(H)))
xlabel('Frequency (Hz)')
ylabel('Magnitude Response (dB)')
subplot(2,1,2) % Вывести на экран диаграмму нулей и полюсов
zplane(bz, az)
zz=roots(bz); % Определить полюса и нули
pz=roots(az);
```

Пример 8Б.2

Разработка простого фильтра нижних частот с использованием билинейного z -преобразования и MATLAB. Требуется цифровой БИХ-фильтр с характеристикой Баттерворта, удовлетворяющий следующим спецификациям:

частота среза 150 Гц,
 частота дискретизации 1,28 кГц,
 порядок фильтра, N 2.

1. Определите, используя билинейное z -преобразование и MATLAB, коэффициенты, полюса и нули дискретного фильтра.
2. Изобразите амплитудно-частотную характеристику и диаграмму нулей и полюсов дискретного фильтра.

Решение

Соответствующий m-файл MATLAB приведен в программе 8Б.2.

1. С помощью m-файла получаем следующие векторы коэффициентов (b и a) нулей и полюсов (z и p) БИХ-фильтра:

$$b = [0.0878, 0.1756, 0.0878]$$

$$a = [1.0000, -1.0048, 0.3561]$$

$$z = [-1, -1]$$

$$p = [0.5024 \pm 0.3220i]$$

$$k = 0.0878$$

Используя коэффициенты, записываем передаточную функцию:

$$H(z) = \frac{0,0878(1 + 2z^{-1} + z^{-2})}{1 - 1,0308z^{-1} + 0,3553z^{-2}}.$$

2. Амплитудно-частотная характеристика и диаграмма нулей и полюсов изображены на рис. 8Б.2.

Программа 8Б.2. Решение примера 8Б.2 в форме m-файла MATLAB

```
%
% Название программы: EX8B2.m
% Простой фильтр нижних частот
%
N=2; % Порядок фильтра
Fs=1280; % Частота дискретизации
FN=Fs/2;
fc=150; % Частота среза
Fc=fc/FN; % Нормированная частота среза
[b, a]=butter (N,Fc); % Создать и оцифровать аналоговый фильтр
[z, p, k]=butter (N, Fc);
subplot (2,1,1) % Вывести на экран амплитудно-частотную характеристику
```

```
[H, f]=freqz(b, a, 512, Fs);
plot(f, abs(H))
xlabel('Frequency (Hz)')
ylabel('Magnitude Response (dB)')
subplot(2,1,2) % Вывести на экран диаграмму нулей и полюсов
zplane(b, a)
```

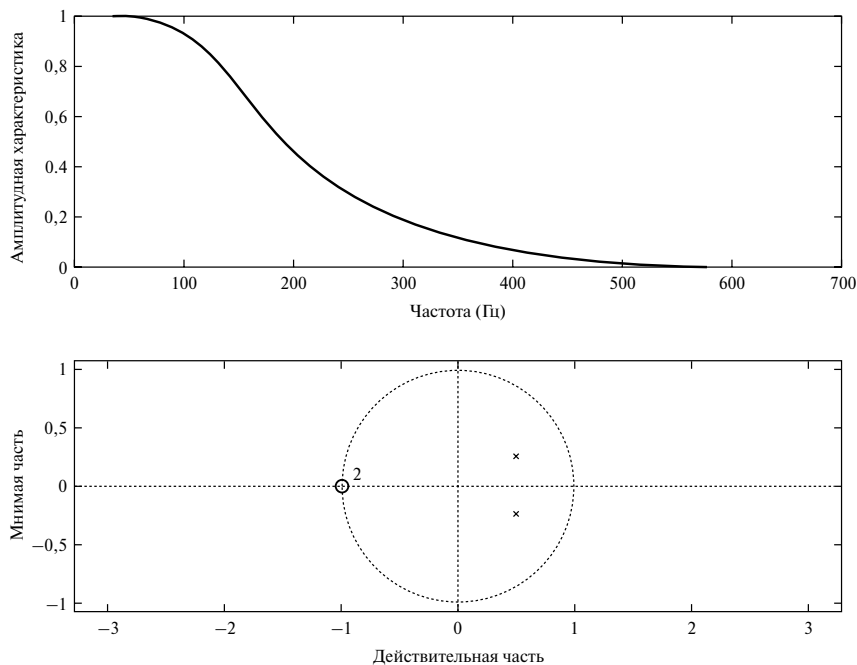


Рис. 8Б.2

Пример 8Б.3

Разработка простого полосового фильтра с использованием билинейного z -преобразования и MATLAB.

1. Вычислите коэффициенты дискретного полосового фильтра с характеристикой Баттерворта, который удовлетворяет следующим спецификациям:
 полоса пропускания 200–300 Гц,
 частота дискретизации 2000 Гц,
 порядок фильтра 8.
2. Изобразите амплитудно-частотную характеристику и диаграмму нулей и полюсов.

Решение

Соответствующий m -файл MATLAB приведен в программе 8Б.3.

1. Коэффициенты b и a , полюса, нули и коэффициенты усиления z , p и k соответственно приведены ниже (отметим, что порядок фильтра, использованный в m-файле, равен половине заданного в задаче; для полосовых и режекторных фильтров порядок равен $2N$):

$$b = [0,004; 0; -0,0017; 0; 0,0025; 0; -0,0017; 0; 0,0004],$$

$$a = [1,000; -5,1408; 13,1256; -20,9376; 22,6982; 17,0342; 8,6867; 2,7672; 0,4383],$$

$$z = [1; 1; 1; 1; -1; -1; -1; -1],$$

$$p = [0,5601 \pm 0,7475i; 0,5800 \pm 0,6286i; 0,6656 \pm 0,5628i; 0,7647 \pm 0,5648i],$$

$$k = 4,1660e - 004.$$

2. Амплитудно-частотная характеристика и диаграмма нулей и полюсов фильтра изображены на рис. 8Б.3.

Программа 8Б.3. Реализация решения примера 8Б.3 в форме m-файла MATLAB

```
%
% Название программы: EX8B3.m
% Простой полосовой фильтр
%
N=2; % Порядок фильтра
Fs=2000; % Частота дискретизации
FN=Fs/2;
fc1=200/FN;
fc2=300/FN;
[b, a]=butter(4,[fc1, fc2]); % Создать и оцифровать аналоговый фильтр
[z, p, k]=butter(4,[fc1, fc2]);
subplot(2,1,1) % Вывести на экран амплитудно-частотную характеристику
[H, f]=freqz(b, a, 512, Fs);
plot(f, abs(H))
xlabel('Frequency (Hz)')
ylabel('Magnitude Response (dB)')
subplot(2,1,2) % Вывести на экран диаграмму нулей и полюсов
zplane(b, a)
```

Пример 8Б.4

Разработка фильтра нижних частот с заданными краевыми частотами полосы пропускания и полосы подавления и неравномерностью/затуханием. Требуется цифровой фильтр нижних частот с характеристикой Баттерворта, удовлетворяющий следующим спецификациям:

полоса пропускания	0–500 Гц,
полоса подавления	2–4 Гц,
неравномерность в полосе пропускания	3 дБ,
затухание в полосе подавления	20 дБ,
частота дискретизации	8 кГц.

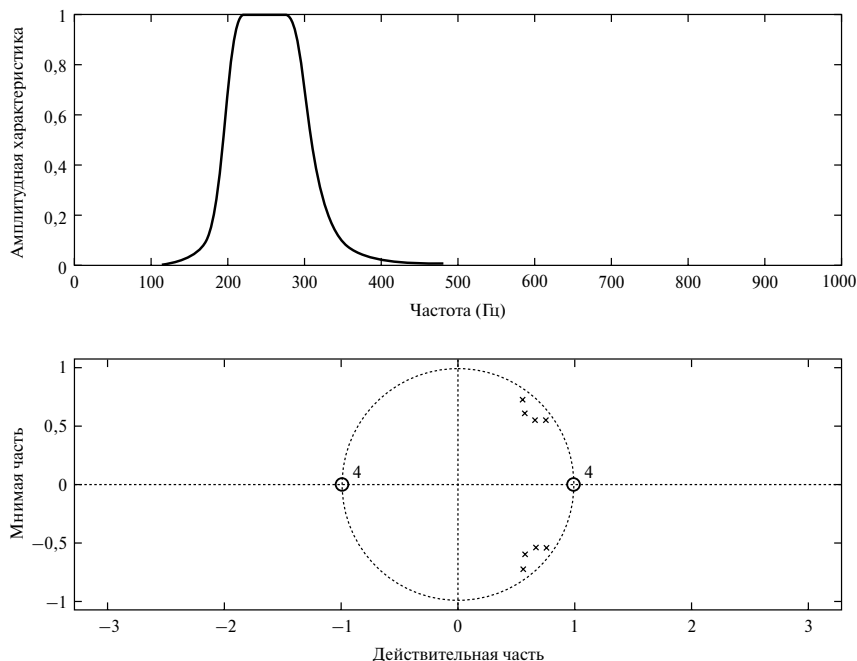


Рис. 8Б.3

1. Определите порядок N и коэффициенты фильтра дискретного времени, используя билинейное z -преобразование и MATLAB.
2. Изобразите амплитудно-частотную характеристику и диаграмму нулей и полюсов фильтра.

Решение

Первая часть задания подобна примеру 8.11, который был решен вручную. Теперь используем для решения MATLAB, поскольку это позволяет определить порядок аналогового фильтра (команда `buttord`), полюса и нули аналогового ФНЧ-прототипа (команда `buttap`), параметры преобразованного аналогового фильтра (команда `butter`) и коэффициенты фильтра дискретного времени (команда `bilinear`). Впрочем, мы используем вариант команды `butter` для цифровой области, поскольку так при вычислении коэффициентов фильтра все промежуточные этапы выполняются автоматически.

1. Реализация решения задачи в форме m-файла MATLAB приведена в программе 8Б.4. Порядок N , нули и полюса (zz и pz), коэффициент усиления (kz) и коэффициенты фильтра (b и a) равны

$$N = 2,$$

$$zz = [-1, -1],$$

$$pz = [0, 7271 \pm j0, 2130i],$$

$$\begin{aligned}
 kz &= 0,0300, \\
 b &= [0,0300; \quad 0,0599; \quad 0,0300], \\
 a &= [1,0000; \quad -1,4542; \quad 0,5741].
 \end{aligned}$$

Полученные коэффициенты идентичны коэффициентам, которые были вычислены вручную в примере 8.11, а передаточная функция записывается так:

$$H(z) = \frac{0,030(1 + 2z^{-1} + z^{-2})}{1 - 1,4542z^{-1} + 0,5741z^{-2}}.$$

2. Амплитудно-частотная характеристика и диаграмма нулей и полюсов изображены на рис. 8Б.4.

Программа 8Б.4. Реализация решения примера 8Б.4 в форме m-файла MATLAB

```

%
% Название программы: EX8B4.m
% Фильтр нижних частот
%
Fs=8000; % Частота дискретизации
Ap=3;
As=20;
wp=500/4000;
ws=2000/4000;
[N, wc]=butterord(wp, ws, Ap, As); % Определить порядок фильтра
[zz, pz, kz]=butter(N, 500/4000); % Оцифровать фильтр
[b, a]=butter(N, 500/4000);
subplot(2,1,1) % Вывести на экран амплитудно-частотную характеристику
[H, f]=freqz(b, a, 512, Fs);
plot(f, abs(H))
xlabel('Frequency (Hz)')
ylabel('Magnitude Response (dB)')
subplot(2,1,2) % Вывести на экран диаграмму нулей и полюсов
zplane(b, a)

```

Пример 8Б.5

Разработка фильтра верхних частот с заданными краевыми частотами полосы пропускания и полосы подавления и неравномерностью/затуханием. Цифровой фильтр верхних частот с характеристикой Баттерворта должен удовлетворять следующим спецификациям:

полоса пропускания	2–4 кГц,
полоса подавления	0–500 Гц,
неравномерность в полосе пропускания	3 дБ,
затухание в полосе подавления	20 дБ,
частота дискретизации	8 кГц.

Определите следующие величины:

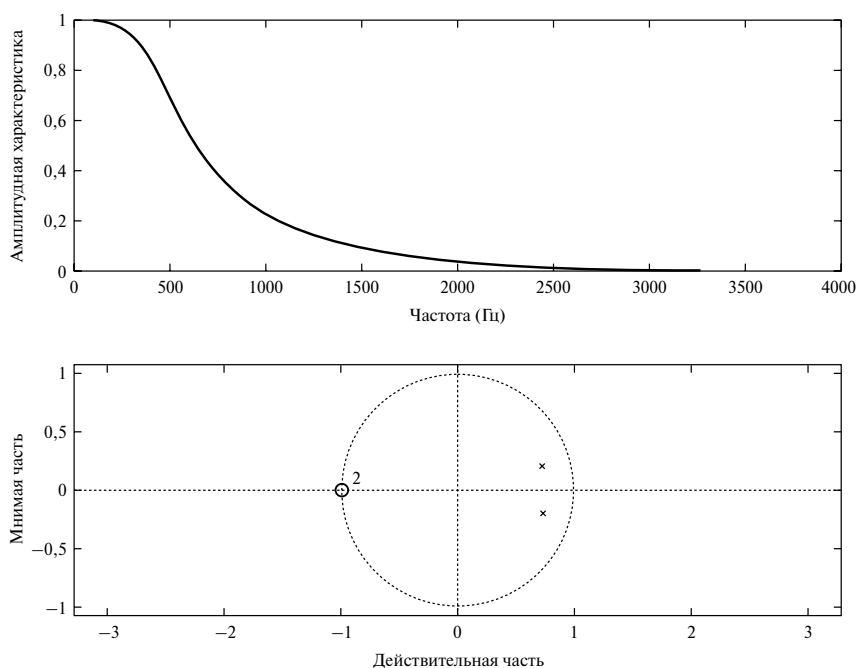


Рис. 8Б.4

- 1) граничные частоты полосы пропускания и полосы подавления подходящего аналогового фильтра-прототипа нижних частот;
- 2) порядок N фильтра-прототипа;
- 3) коэффициенты, а следовательно, передаточную функцию фильтра дискретного времени с использованием билинейного z -преобразования.

Решение

1. Реализация решения задачи в форме m-файла MATLAB приведена в программе 8Б.5.
2. Порядок N , нули и полюса (zz и pz), коэффициент усиления (kz) и коэффициенты фильтра (b и a) равны

$$N = 2,$$

$$zz = [1, 1],$$

$$pz = \pm 0,4142i,$$

$$kz = 0,2929,$$

$$b = [0,2929; \quad -0,5858; \quad 0,2929],$$

$$a = [1,0000; \quad 0,0000; \quad 0,1716].$$

3. Используя полученные коэффициенты, записываем передаточную функцию:

$$H(z) = \frac{0,2929(1 - 2z^{-1} + z^{-2})}{1 + 0,1716z^{-2}}.$$

Амплитудно-частотная характеристика и диаграмма нулей и полюсов изображена на рис. 8Б.5.

Программа 8Б.5. Реализация решения примера 8Б.5 в m-файле MATLAB

```
%
% Название программы: EX8B5.m
%
Fs=8000; % Частота дискретизации
Ap=3;
As=20;
wp=2000/4000;
ws=500/4000;
[N, wc]=buttord(wp, ws, Ap, As); % Определить порядок фильтра
[zz, pz, kz]=butter(N, 2000/4000, 'high'); % Оцифровать фильтр
[b, a]=butter(N, 2000/4000, 'high');
subplot(2,1,1) % Вывести на экран амплитудно-частотную характеристику
[H, f]=freqz(b, a, 512, Fs);
plot(f, abs(H))
xlabel('Frequency (Hz)')
ylabel('Magnitude Response (dB)')
subplot(2,1,2) % Вывести на экран диаграмму нулей и полюсов
zplane(b, a)
```

Пример 8Б.6

Разработка полосового фильтра с заданными краевыми частотами полосы пропускания и полосы подавления и неравномерностью в полосе пропускания/подавления. Цифровой полосовой фильтр с характеристикой Баттерворта должен удовлетворять следующим спецификациям:

нижняя граничная частота полосы пропускания	200 Гц,
верхняя граничная частота полосы пропускания	300 Гц,
нижняя граничная частота полосы подавления	50 Гц,
верхняя граничная частота полосы подавления	450 Гц,
неравномерность в полосе пропускания	3 дБ,
затухание в полосе подавления	20 дБ,
частота дискретизации	1 кГц.

- Определите, используя метод билинейного z -преобразования и MATLAB
 - порядок фильтра N ;
 - полюса, нули, коэффициент усиления и передаточную функцию фильтра дискретного времени.

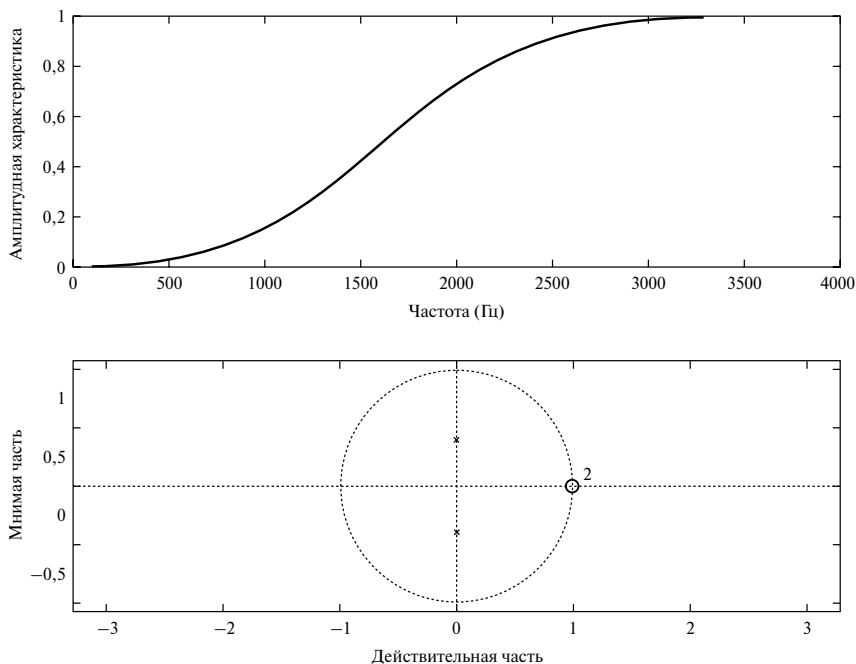


Рис. 8Б.5

- Изобразите амплитудно-частотную характеристику и диаграмму нулей и полюсов фильтра.

Решение

- Реализация решения примера в форме m-файла MATLAB приведена в программе 8Б.6. Используя этот m-файл, получаем следующие значения для нулей и полюсов (zz и kz), коэффициента усиления (kz) и коэффициентов (b и a) дискретного фильтра:

$N = 2$ (порядок полосового фильтра равен $2 * N$, т.е.4),

$$zz = [1, 1, -1, -1],$$

$$pz = [-0,1884 \pm 0,7791i; 0,1884 \pm 0,7791i],$$

$$kz = 0,0675,$$

$$b = [0,0675; 0; -0,1349; 0; 0,0675],$$

$$a = [1,0000; -0,0000; 1,1430; -0,0000; 0,4128].$$

Соответствующая передаточная функция $H(z)$ равна

$$H(z) = \frac{0,0675(1 - 2z^{-2} + z^{-4})}{1 + 1,143z^{-2} + 0,4128z^{-4}}.$$

2. Если $N = 1$ (как в примере 8.11), нули (zz), полюса (pz), коэффициент усиления (kz) и коэффициенты фильтра (b и a) принимают такие значения:

$$\begin{aligned} zz &= [-1, 1], \\ pz &= [0, 0000 \pm 0, 7138i], \\ kz &= [0, 2452], \\ b &= [0, 2452; 0; -0, 2452], \\ a &= [1, 0000; -0, 0000; 0, 5095]. \end{aligned}$$

Соответствующая передаточная функция $H(z)$ равна

$$H(z) = \frac{0,2452(1 - z^{-2})}{1 + 0,5095z^{-2}}.$$

Амплитудно-частотная характеристика и диаграмма нулей и полюсов фильтра приведены на рис. 8Б.6.

Программа 8Б.6. Реализация решения примера 8Б.6 в форме m-файла MATLAB

```
%
% Название программы: EX8B6.m
% Полосовой фильтр
%
Fs=1000; % Частота дискретизации
Ap=3;
As=20;
Wp=[200/500, 300/500]; % Граничные частоты
Ws=[50/500, 450/500];
[N, Wc]=buttord(Wp, Ws, Ap, As); % Определить порядок фильтра
[zz, pz, kz]=butter(N, Wp); % Оцифровать фильтр
[b, a]=butter(N, Wp);
subplot(2,1,1) % Вывести на экран амплитудно-частотную характеристику
[H, f]=freqz(b, a, 512, Fs);
plot(f, abs(H))
xlabel('Frequency (Hz)')
ylabel('Magnitude Response (dB)')
subplot(2,1,2) % Вывести на экран диаграмму нулей и полюсов
zplane(b, a)
```

Пример 8Б.7

Разработка режекторного фильтра с заданными краевыми частотами полосы пропускания и полосы подавления и неравномерностью в полосе пропускания/подавления. Цифровой режекторный фильтр с характеристикой Баттерворта должен удовлетворять следующим спецификациям:

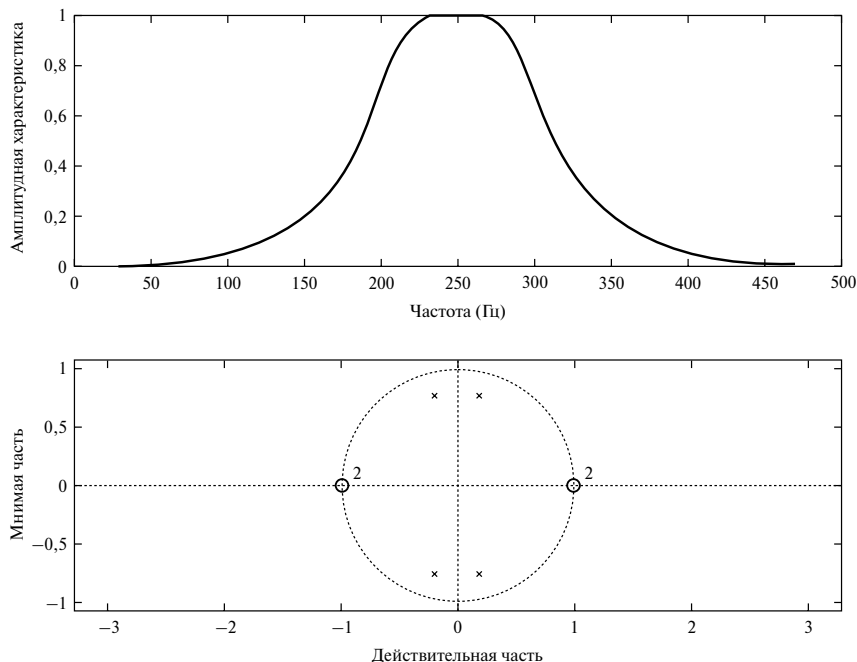


Рис. 8Б.6

нижняя полоса пропускания	0–50 Гц,
верхняя полоса пропускания	450–500 Гц,
полоса подавления	200–300 Гц,
неравномерность в полосе пропускания	3 дБ,
затухание в полосе подавления	20 дБ,
частота дискретизации	1 кГц.

Определите

- граничные частоты полос пропускания и подавления подходящего аналогового фильтра-прототипа нижних частот;
- порядок фильтра-прототипа N ;
- коэффициенты, а следовательно, передаточную функцию фильтра дискретного времени, используя метод билинейного z -преобразования.

Решение

Реализация решения задачи в форме m-файла MATLAB приведена в программе 8Б.6. Используя этот m-файл, получаем следующие значения для нулей и полюсов (zz и kz), коэффициента усиления (kz) и коэффициентов (b и a) дискретного фильтра:

$$N = 2 \text{ (порядок полосового фильтра равен } 2 * N, \text{ т.е.4),}$$

$$zz = \text{две пары нулей в } i \text{ и } -i,$$

$$pz = [-0,1884 \pm 0,7791i; 0,1884 \pm 0,7791i],$$

$$kz = 0,6389,$$

$$b = [0,6389; -0,0000; 1,2779; -0,0000; 0,6389],$$

$$a = [1,0000; -0,0000; 1,1430; -0,0000; 0,4128].$$

Соответствующая передаточная функция фильтра $H(z)$ равна

$$H(z) = \frac{0,6389(1 + 2z^{-2} + z^{-4})}{1 + 1,143z^{-2} + 0,4128z^{-4}}.$$

Если $N = 1$ (как в примере 8.12), передаточная функция становится равной

$$H(z) = \frac{0,7548(1 + z^{-2})}{1 + 0,5095z^{-2}}.$$

Амплитудно-частотная характеристика и диаграмма нулей и полюсов фильтра приведены на рис. 8Б.7.

Программа 8Б.7. Реализация решения примера 8Б.7 в форме m-файла MATLAB

```
%
% Название программы: EX8B7.m
% Заграждающий фильтр
%
Fs=1000; % Частота дискретизации
Ap=3;
As=20;
Wp=[50/500, 450/500]; % Граничные частоты
Ws=[200/500, 300/500];
[N, Wc]=buttord(Wp, Ws, Ap, As); % Определить порядок фильтра
[zz, pz, kz]=butter(N, Wp, 'stop'); % Оцифровать фильтр
[b, a]=butter(N, Wp, 'stop');
subplot(2,1,1) % Вывести на экран амплитудно-частотную характеристику
[H, f]=freqz(b, a, 512, Fs);
plot(f, abs(H))
xlabel('Frequency (Hz)')
ylabel('Magnitude Response (dB)')
subplot(2,1,2) % Вывести на экран диаграмму нулей и полюсов
zplane(b, a)
```

Пример 8Б.8

Пример разработки эллиптического цифрового полосового фильтра. Требуется цифровой фильтр, удовлетворяющий следующим спецификациям частотной характеристики:

полоса пропускания	20,5–23,5 кГц,
полоса подавления	0–19, 25–50 кГц,
неравномерность в полосе пропускания	$\leq 0,25$ дБ,
затухание в полосе подавления	>45 дБ,
частота дискретизации	100 кГц.

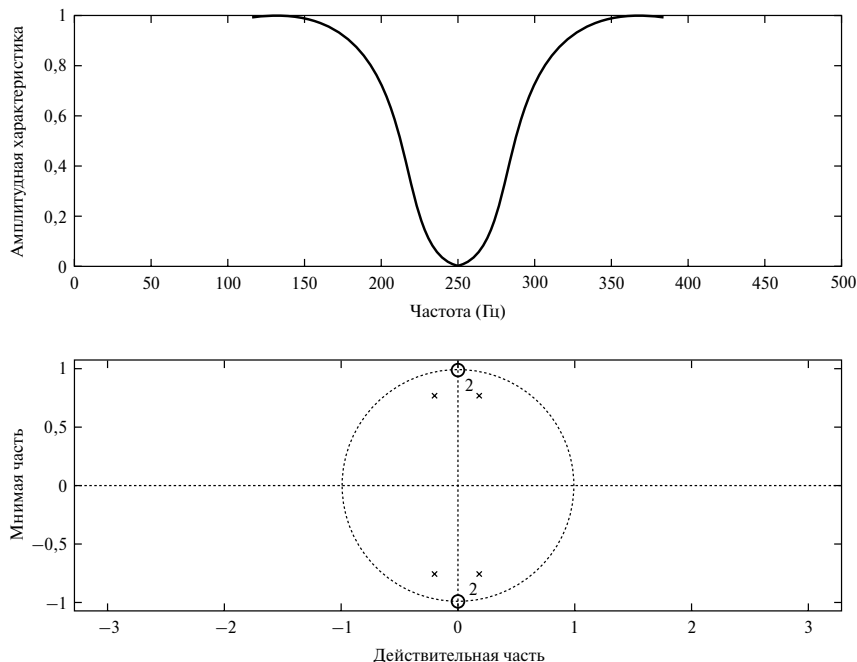


Рис. 8Б.7

1. Определите, используя билинейное z -преобразование и MATLAB, подходящую передаточную функцию фильтра в форме звеньев второго порядка, пригодных для каскадной реализации.
2. Получите и изобразите амплитудно-частотную характеристику и диаграмму нулей и полюсов фильтра.
Предполагайте использование эллиптической характеристики.

Решение

1. Реализация решения задачи в форме m-файла MATLAB приведена в программе 8Б.8. Для определения порядка подходящего эллиптического фильтра-прототипа использована команда `ellipord`. Затем применяется команда `ellip`, определяющая коэффициенты, полюса и нули передаточной функции. Команда `zp2sos` переводит передаточную функцию в звенья второго порядка. Коэффициенты звеньев второго порядка возвращаются в виде матрицы **sos**.

Коэффициенты фильтра (b и a), полюса и нули (p и z), коэффициент усиления (k) и матрица звеньев второго порядка **sos** приведены ниже:

Порядок фильтра, $N = 4$

$$b = [0,0061; -0,0083; 0,0236; -0,0221; 0,0351; -0,0221; 0,0236; -0,0083; 0,0061]$$

$$a = [1,0000; -1,4483; 4,4832; -4,2207; 6,6475; -3,9458; 3,9187; -1,1828; 0,7634]$$

$$z = [-0,0118 \pm 0,9999i; 0,3737 \pm 0,9275i; -0,2553 \pm 0,9669i; 0,5663 \pm 0,8242i]$$

$$p = [0,0872 \pm 0,9793i; 0,1352 \pm 0,9404i; 0,2795 \pm 0,9432i; 0,2223 \pm 0,9246i]$$

$$k = 0,0061$$

$sos =$

0,1203	0,0614	0,1203	1,0000	-0,2705	0,9026
0,2051	-0,2323	0,2051	1,0000	-0,4446	0,9042
0,3740	0,0088	0,3740	1,0000	-0,1744	0,9665
0,6642	-0,4965	0,6642	1,0000	-0,5589	0,9678

Матрица sos — это матрица 4×6 звеньев второго порядка, которая имеет следующий формат.

b_{01}	b_{11}	b_{21}	a_{01}	a_{11}	a_{21}
b_{02}	b_{12}	b_{22}	a_{02}	a_{12}	a_{22}
b_{03}	b_{13}	b_{23}	a_{03}	a_{13}	a_{23}
b_{04}	b_{14}	b_{24}	a_{04}	a_{14}	a_{24}

Очевидно, что коэффициенты звеньев второго порядка идут рядами, по одному ряду на звено.

- Амплитудно-частотная характеристика и диаграмма нулей и полюсов фильтра приведены на рис. 8Б.8.

Программа 8Б.8. Реализация решения примера 8Б.8 в форме m-файла MATLAB

```
%
% Название программы: EX8B8.m
%
Ap=0.25;
As=45;
Fs=100000;
Wp=[20500/50000, 23500/50000]; % Граничные частоты
Ws=[19000/50000, 25000/50000];
[N, Wc]=ellipord(Wp, Ws, Ap, As); % Определить порядок фильтра
[b, a]=ellip(N, Ap, As, Wc); % Определить коэффициенты фильтра
[z, p, k]=ellip(N, Ap, As, Wc); % Определить полюса и нули
sos=zp2sos(z,p,k); % Преобразовать в звенья второго порядка
subplot(2,1,1) % Вывести на экран амплитудно-частотную характеристику
[H, f]=freqz(b, a, 512, Fs);
plot(f, 20*log10(abs(H)))
xlabel('Frequency (Hz)')
ylabel('Magnitude Response (dB)')
subplot(2,1,2) % Вывести на экран диаграмму нулей и полюсов
zplane(b, a)
```

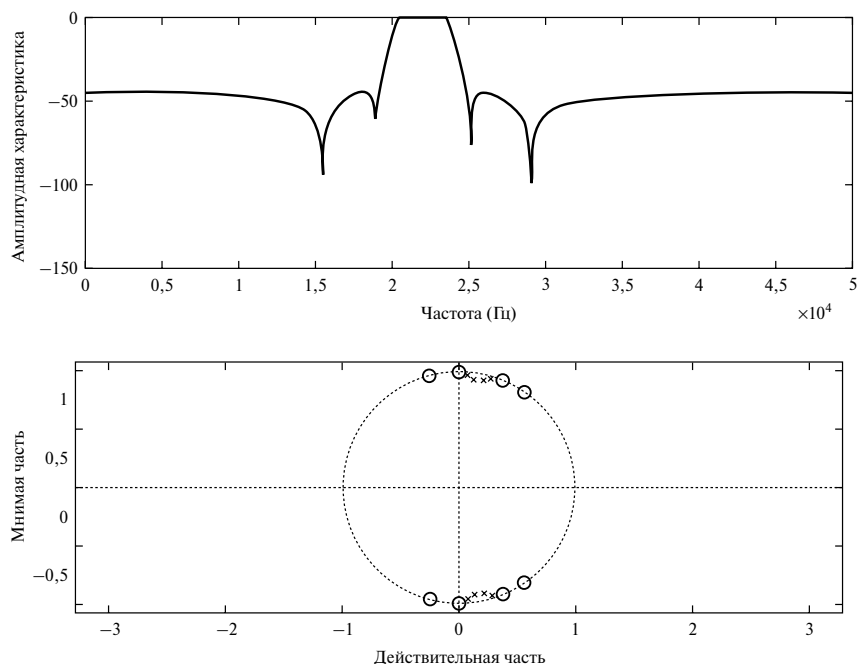


Рис. 8Б.8

8.В.1. Вычисление квадратных корней из комплексных величин с использованием действительной арифметики

Пусть квадратный корень данного комплексного числа $x + iy$ равен $u + iv$:

$$u + iv = (x + iy)^{1/2}. \quad (8B.1)$$

Возводя выражение (8B.1) в квадрат, получаем

$$u^2 - v^2 + i2uv = x + iy.$$

Приравнявая действительную и мнимую части, получаем

$$u^2 - v^2 = x, \quad (8B.2, a)$$

$$2uv = y. \quad (8B.2, б)$$

Из уравнения (8B.2, б) следует, что

$$v = y/2u. \quad (8B.2, \text{ в})$$

Подстановка уравнения (8B.2, в) в (8B.2, а) дает

$$u^2 - y^2/4u^2 = x \quad (8B.3)$$

и, упростив последнее выражение, получаем

$$4u^4 - 4xu^2 - y^2 = 0.$$

Данное уравнение квадратно по u^2 и имеет следующие решения:

$$\begin{aligned} u^2 &= \frac{x}{2} + \frac{1}{8}(16x^2 + 16y^2)^{1/2} = \\ &= \frac{x + (x^2 + y^2)^{1/2}}{2}. \end{aligned}$$

(Отметим, что отрицательное решение недопустимо, так как $u^2 \geq 0$.) Вычисляя из последнего уравнения u и используя уравнение (8B.2, в), получаем

$$u = \left[\frac{x + (x^2 + y^2)^{1/2}}{2} \right]^{1/2} \quad (8B.4, \text{ а})$$

или

$$v = y/2u \quad (8B.4, \text{ б})$$

$$u = - \left[\frac{x + (x^2 + y^2)^{1/2}}{2} \right]^{1/2} \quad (8B.5, \text{ а})$$

$$v = y/2u. \quad (8B.5, \text{ б})$$

Итак, получаем следующий алгоритм вычисления квадратного корня из $x + iy$.

- Этап 1: пусть $u + iv = (x + iy)^{1/2}$.
- Этап 2: $u = \pm \left[\frac{x + (x^2 + y^2)^{1/2}}{2} \right]^{1/2} = \pm \left(\frac{x + |x + iy|}{2} \right)^{1/2}$.
- Этап 3: $v = \pm y/2u$.

Например,

$$(-1 + i)^{1/2} = 0,44509 + 1,09868i \text{ и } -0,44509 - 1,09868i,$$

$$(-1 - i)^{1/2} = 0,44509 - 1,09868i \text{ и } -0,44509 + 1,09868i.$$

Цифровая обработка сигналов при нескольких скоростях

9

9.1. Введение	629
9.2. Концепции обработки при нескольких скоростях	632
9.3. Разработка практических конвертеров частоты дискретизации	640
9.4. Программная реализация дециматоров	651
9.5. Программная реализация интерполяторов	655
9.6. Преобразование частоты дискретизации с использованием многофазного фильтра	663
9.7. Примеры применения	668
9.8. Резюме	683
Задачи	684
Литература	691
Дополнительная литература	691
Приложения	693

В данной главе с практической точки зрения рассматривается важная тема обработки сигналов при нескольких скоростях. Основные концепции исследуются и иллюстрируются на конкретных примерах. Чтобы облегчить разработку и программную реализацию методов обработки при нескольких скоростях, предлагается набор программ на языке C.

9.1. Введение

Растущая потребность в современных цифровых системах обработки данных с более чем одной частотой дискретизации привела к развитию новой подобласти цифровой обработки сигналов, известной как *обработка данных при нескольких скоростях* (multirate processing) [4–9]. При такой обработке данных используются две основные операции: децимация и интерполяция, позволяющие эффективно чередовать скорости передачи данных. Децимация уменьшает частоту дискретизации, эффективно сжимая данные и оставляя только необходимую информацию. Интерполяция, наоборот, увеличивает частоту дискретизации. Часто преобразование данных под новую частоту передачи облегчает их обработку (например, делает вычислительно более эффективной) или обеспечивает совместимость с другой системой. Приведем очевидный пример: если умень-

шить частоту дискретизации сигнала со 100 до 10 кГц без потери желаемой информации, то затраты вычислительных ресурсов при последующей обработке сигнала разом уменьшатся в 10 раз. Другой пример: если требуется воспроизвести аудиокомпакт-диск, имеющий частоту дискретизации 44,1 кГц, в студии, поддерживающей частоту 48 кГц, информацию с данного компакт-диска нужно вначале перевыбрать с частотой 48 кГц, используя методы обработки при нескольких скоростях.

Цель данной главы и дополняющих ее задач — обеспечить прикладное понимание теории, практики и сфер применения обработки при нескольких скоростях. В частности, отмечены следующие моменты.

1. *Теория.* Читатель должен изучить теорию преобразования частоты дискретизации, в частности
 - принципы уменьшения частоты дискретизации (*децимация* (или *прореживание*) — выборка с понижением частоты и цифровая фильтрация с целью защиты от наложения спектров);
 - принципы увеличения частоты дискретизации (*интерполяция* — выборка с увеличением частоты и цифровая фильтрация с целью защиты от зеркальных частот);
 - теорию преобразования частоты в несколько этапов;
 - принципы многофазной фильтрации.
2. *Практика.* Читатель должен уметь разрабатывать (на уровне блок-схем) практические конвертеры скорости передачи при данном наборе спецификаций, например, знать, как
 - задать, разработать и проанализировать фильтры конвертеров;
 - определить параметры конвертеров;
 - оценить вычислительную эффективность системы обработки при нескольких скоростях;
 - реализовать конвертер частоты дискретизации.
3. *Применение.* В главе описаны принципы и области применения методов обработки при нескольких скоростях в сфере аудиотехнологий, цифровой связи и биомедицине, в частности:
 - обработка аудиосигналов — АЦП/ЦАП с выборкой с запасом (oversampling), воспроизведение компакт-дисков и сбор данных;
 - цифровая связь — трансмультиплексоры, приемники;
 - биомедицина — узкополосная фильтрация для обработки электрокардиограмм и электроэнцефалограмм плода.

Данная глава построена на многих источниках, особенно, работах Рабинера (Rabiner) и Крошера (Crochiere), которые внесли значительный вклад в сферу обработки данных при нескольких скоростях, за что мы им особо благодарны.

9.1.1. Некоторые области промышленного применения обработки данных при нескольких скоростях

Достоинств обработки данных при нескольких скоростях множество, и они широко используются во многих современных системах, причем число этих систем постоянно

растет. Разработчики высококачественных систем сбора и хранения данных все чаще обращаются к методам обработки при нескольких скоростях, позволяющим избежать дорогих аналоговых фильтров защиты от наложения спектров и эффективно обрабатывать сигналы с различными ширинами полос и частотами дискретизации. Во всех этих приложениях используется следующий факт: если аналоговый сигнал выбирается (дискретизируется) с частотой, значительно превышающей частоту, определяемую теоремой о дискретном представлении (Котельникова), то для ограничения полосы частот после оцифровки сигнала потребуется значительно более простой аналоговый фильтр защиты от наложения спектров. После перевода сигнала в цифровую форму его можно быстро сжать до желаемой скорости передачи с использованием описанного ниже подхода. Хороший пример системы, построенной по такому принципу, — магнитофон EDR8000 (Earth Data, Великобритания).

В сфере обработки речи методы обработки при нескольких скоростях применяются для уменьшения требуемой памяти или скорости передачи речевых данных. Оценки параметров речи, которые будут храниться или передаваться, вычисляются при очень низкой частоте дискретизации. Когда требуется, с помощью представления с низкой частотой дискретизации исходную речь можно восстановить с значительно большей частотой.

Потребность сферы цифрового аудио в недорогих аналого-цифровых преобразователях (АЦП) с высокой разрешающей способностью привела к переходу от традиционных методов последовательного приближения к методам выборки с запасом (oversampling) [1–3, 12, 14]. Например, при выборке с запасом шум квантования, присущий АЦП, распространяется по более широкому диапазону частот, и внутрисполосный шум значительно уменьшается, эффективно увеличивая число битов АЦП. Кроме того, в новых сериях АЦП улучшенной эффективности используется дельта-сигма-модуляция из-за ее простоты (например, она не требует усилителей выборки-хранения) и низкой стоимости. В большинстве подобных (если не во всех) недорогих АЦП с высоким разрешением (18, 20, 24 бит) применяется обработка при нескольких скоростях. В качестве примеров таких устройств можно привести CS532X (Crystal Semiconductor) и DSP56ADCx (Motorola).

Обработка при нескольких скоростях применяется и для эффективной реализации функций цифровой обработки сигналов. Например, реализация узкополосных цифровых КИХ-фильтров с использованием обычной ЦОС представляет серьезную проблему, поскольку такие фильтры требуют очень большого числа коэффициентов для удовлетворения строгих спецификаций их частотной характеристики. Использование методов обработки при нескольких скоростях позволяет получать очень эффективные реализации, разрешая выполнение фильтрации при значительно более низкой скорости, что намного понижает порядок фильтра. Методы обработки при нескольких скоростях также используются во многих других сферах, включая создание распространенных проигрывателей компакт-дисков. Подробности применения обработки при нескольких скоростях в проигрывателе компакт-дисков и многих других устройствах рассмотрены далее в этой главе.

9.2. Концепции обработки при нескольких скоростях

Простой, но слишком наивный подход к изменению частоты дискретизации цифрового сигнала — это вернуть его к аналоговому виду, а затем повторно оцифровать его с иной частотой. Процессу преобразования цифровой-аналоговый-цифровой присущи такие ошибки, как ошибки квантования и наложения, которые значительно искажают сигнал. Значит, поскольку сигнал уже дан в цифровой форме, обрабатывать его лучше цифровым образом, если, конечно, преобразование в аналоговую форму не обязательно, как, например, когда адресатом информации является громкоговоритель. Обработка сигналов при нескольких скоростях — это, по сути, эффективный метод цифрового изменения частоты дискретизации сигнала, позволяющий использовать сильные стороны традиционной цифровой обработки сигналов. Например, значительную фильтрацию для защиты от наложения спектров и подавления зеркальных частот в системе ЦОС реального времени можно провести в цифровой области, что позволит добиться и резкой амплитудной, и линейной фазовой характеристик.

Фундаментальными операциями обработки сигналов при нескольких скоростях являются процессы децимации и интерполяции, позволяющие уменьшать и увеличивать частоту дискретизации без существенного влияния искажающих факторов, таких как квантование и наложение. Подробности этих базовых операций приводятся ниже.

9.2.1. Уменьшение частоты дискретизации: децимация с целым шагом

На рис. 9.1, *а* представлена блок-схема процесса децимации сигнала $x(n)$ с целым шагом M . На ней изображены цифровой фильтр защиты от наложения спектров $h(k)$ и схема сжатия (компрессор) частоты дискретизации, обозначенная стрелкой, направленной вниз, и коэффициентом децимации M . Компрессор частоты дискретизации снижает частоту дискретизации с F_s до F_s/M . Для предотвращения наложения при более низкой частоте выборки используется цифровой фильтр, предварительно ограничивающий полосу входного сигнала до $F_s/2M$. Таким образом, сигнал $x(n)$ вначале ограничивается по полосе (требования к фильтру для децимации подробно рассмотрены в разделе 9.3.1). Снижение частоты дискретизации достигается за счет того, что из каждых M выборок фильтруемого сигнала $w(n)$ отбрасывается $M - 1$ выборка. Вход и выход процесса децимации связаны следующим соотношением:

$$y(m) = w(mM) = \sum_{k=-\infty}^{\infty} h(k)x(mM - k), \quad (9.1, a)$$

где

$$w(n) = \sum_{k=-\infty}^{\infty} h(k)x(n - k). \quad (9.1, б)$$

На рис. 9.1, *б* описанный процесс иллюстрируется для простого случая $M = 3$, где из каждых трех выборок $x(n)$ отбрасываются две. Отметим, что, по сути, децимация — это операция сжатия данных.

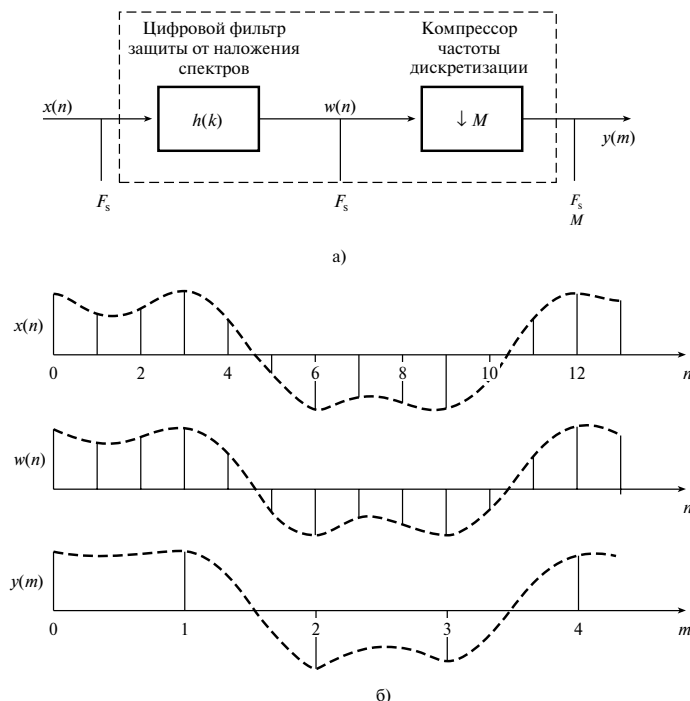


Рис. 9.1. Блок-схема децимации с шагом M (панель *а*). Иллюстрация децимации с шагом $M = 3$ (панель *б*). Обратите внимание на то, что на выход поступает только одна из каждых трех выборок $w(n)$

Спектральное представление процесса децимации приведено на рис. 9.2, где предполагается, что на вход подается широкополосный сигнал $x(n)$. На рис. 9.2, *б* пунктиром изображены зеркальные компоненты сигнала, которые привели бы к наложению, если бы входной сигнал $x(n)$ не был ограничен по полосе перед децимацией.

9.2.2. Увеличение частоты дискретизации: интерполяция с целым шагом

По многим пунктам интерполяция — это цифровой эквивалент процесса цифроаналогового преобразования, когда из цифровых выборок, поданных на цифроаналоговый преобразователь, с помощью интерполяции восстанавливается аналоговый сигнал. При цифровой интерполяции, впрочем, процесс порождает специфические значения.

Для данного сигнала $x(n)$ с частотой дискретизации F_s процесс интерполяции увеличивает частоту дискретизации в L раз, т.е. до LF_s . Схема интерполятора приведена на рис. 9.3, *а*. Он состоит из экспандера частоты дискретизации, обозначенного стрелкой, направленной вверх, и коэффициентом интерполяции L , который показывает, во сколько раз увеличивается частота дискретизации. Для каждой выборки сигнала $x(n)$ экспандер вводит $(L-1)$ нулевую выборку, в результате чего формируется новый сигнал $w(m)$ с ча-

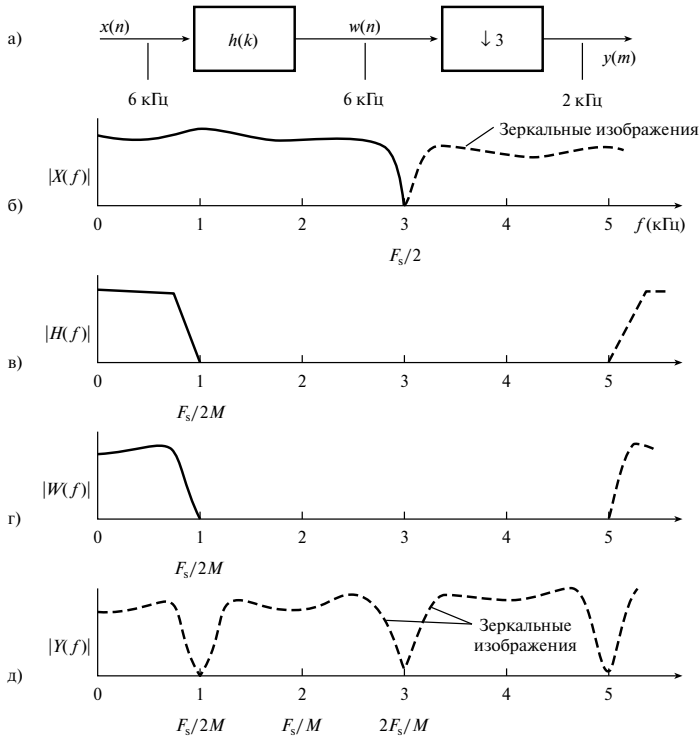


Рис. 9.2. Спектральная интерпретация децимации сигнала с частоты 6 до частоты 2 кГц. Если бы не цифровой фильтр, на сигнал наложился бы зеркальные компоненты

стотой дискретизации LF_s . Далее этот сигнал пропускается через фильтр нижних частот для удаления зеркальных частот, введенных при увеличении частоты, и получается сигнал $y(m)$. Введение $(L - 1)$ нулей приводит к распространению энергии каждой выборки сигнала по L выходным выборкам, т.е. каждая выборка ослабляется в L раз. Для компенсации этого эффекта можно, например, умножить каждую выборку $y(n)$ на L .

Процесс интерполяции характеризуется следующей связью входа и выхода:

$$y(m) = \sum_{k=-\infty}^{\infty} h(k)w(m - k), \quad (9.2, а)$$

где

$$w(m) = \begin{cases} x(m/L), & m = 0, \pm L, \pm 2L, \dots \\ 0 & \text{иначе} \end{cases} \quad (9.2, б)$$

Процесс интерполяции во временной области иллюстрируется на рис. 9.3, б–г для простого случая $L = 3$. Обратите внимание на то, что каждая выборка $x(n)$ порождает три выходные выборки (две нулевые выборки вводит экспандер).

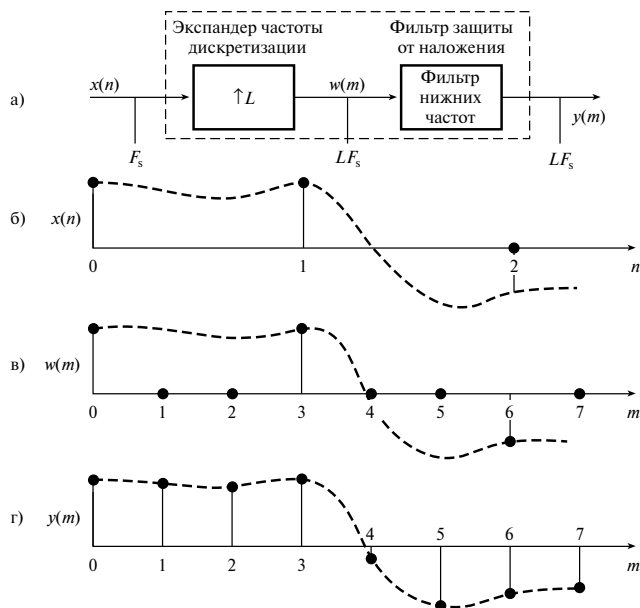


Рис. 9.3. Иллюстрация интерполяции с шагом $L = 3$ во временной области. Обратите внимание на то, что для каждой выборки $x(n)$ получаются три выходные выборки $y(m)$

Интерпретация данного процесса в частотной области приведена на рис. 9.4. Функции $X(f)$, $W(f)$ и $Y(f)$ представляют собой частотные характеристики сигналов $x(n)$, $w(m)$ и $y(m)$ соответственно. $H(f)$ — это амплитудная характеристика фильтра подавления зеркальных частот. Данный фильтр необходим для удаления зеркальных компонентов, обозначенных в $W(f)$ пунктиром. На данном этапе стоит отметить (хотя внимательный читатель уже, возможно, сам это отметил), что процессы децимации и интерполяции дуальны, т.е. взаимно обратны. Данное свойство дуальности означает, что интерполятор можно легко получить из эквивалентного дециматора и наоборот.

9.2.3. Преобразование частоты дискретизации с нецелым шагом

В некоторых ситуациях часто бывает нужно изменить частоту дискретизации в нецелое число раз. Пример — цифровое аудио, где может требоваться передача данных с одного запоминающего устройства на другое, причем системы поддерживают разные частоты дискретизации (возможно, для предотвращения незаконного копирования материалов). В частности, это передача данных с системы воспроизведения компакт-дисков (44,1 кГц) на цифровую аудиопленку (digital audio tape — DAT) (48 кГц). Для выполнения требуемого преобразования частоту дискретизации компакт-диска следует увеличить в 48/44,1 раза.

На практике подобные нецелые множители представляются рациональным числом, т.е. отношением двух целых чисел, скажем L и M , таких что L/M максимально близ-

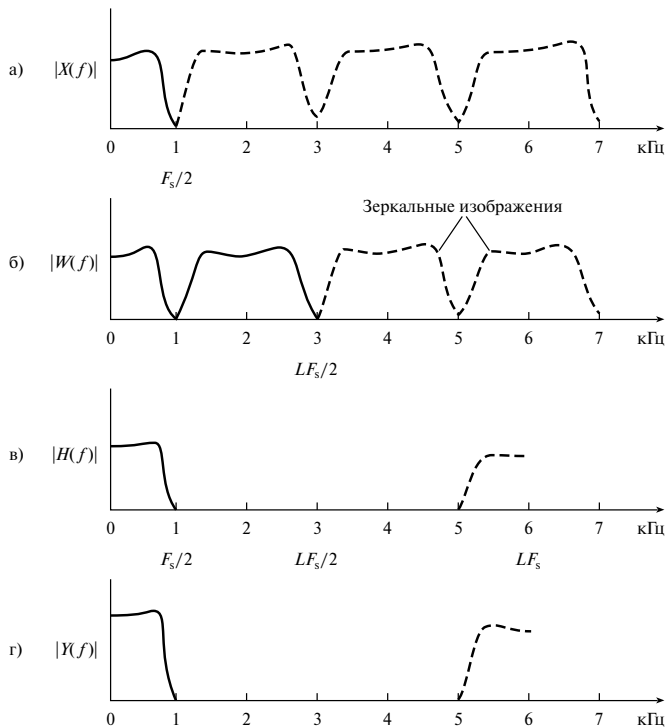


Рис. 9.4. Спектральная интерпретация интерполяции сигнала с 2 до 6 кГц

ко к желаемому множителю. Необходимое же преобразование частоты дискретизации производится в два этапа: 1) интерполяция данных с шагом L и 2) децимация с шагом M (рис. 9.5, а). Необходимо, чтобы процесс интерполяции предшествовал децимации, поскольку в противном случае при децимации исчезнут некоторые необходимые частотные компоненты. В приведенном выше примере “с компакт-диска на цифровую аудиопленку” преобразования частоты в 48/44,1 раза можно достичь следующим образом: интерполировать с шагом $L = 160$, а затем провести децимацию с шагом $M = 147$, т.е. вначале скорость данных компакт-диска увеличивается в $L = 160$ раз до 7056 кГц, а затем уменьшается в $M = 147$ раз до 48 кГц.

Поскольку два фильтра нижних частот на рис. 9.5, а $h_1(k)$ и $h_2(k)$ соединены каскадно и имеют общую частоту дискретизации, их можно объединить, и тогда получится один обобщенный конвертер частоты дискретизации, изображенный на рис. 9.5, б. Если $M > L$, операция, производимая конвертером, называется *децимацией с нецелым шагом*, а если $M < L$ — *интерполяцией*. Если $M = 1$, действие обобщенной системы сводится к простой интерполяции с целым шагом, описанной ранее, а при $L = 1$ оно сводится к децимации с целым шагом.

На рис. 9.5, в иллюстрируется интерполяция с шагом $3/2$. Частота дискретизации вначале увеличивается в 3 раза (к каждой выборке $x(n)$ добавляются две нулевые выборки), а затем сигнал пропускается через фильтр нижних частот, результат — $v(i)$.

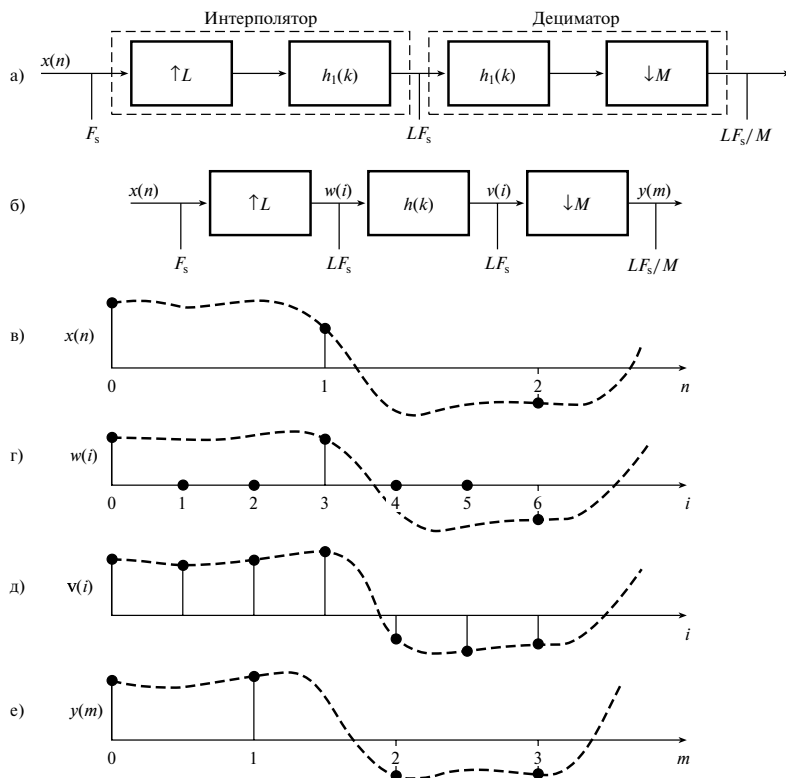


Рис. 9.5. Иллюстрация интерполяции с рациональным шагом ($L = 3$, $M = 2$)

Далее фильтрованные данные сокращаются в два раза, т.е. из каждых двух выборок $v(i)$ остается одна. Иллюстрацией данного процесса в частотной области служит рис. 9.6. Входной сигнал $x(n)$ с частотой дискретизации 2 кГц вначале ускоряется в 3 раза до 6 кГц, фильтруется с целью устранения зеркальных частот, которые иначе вызвали бы наложение, а затем замедляется в 2 раза до частоты 3 кГц.

Пример 9.1

Сигнал $x(n)$ с частотой дискретизации 2,048 кГц нужно подвергнуть децимации с шагом 32 и получить сигнал с частотой дискретизации 64 Гц. Существенными являются частоты сигнала в полосе 0–30 Гц. Цифровой фильтр защиты от наложения спектров должен удовлетворять следующим спецификациям:

отклонение в полосе пропускания	0,01 дБ,
отклонение в полосе подавления	80 дБ,
полоса пропускания	0–30 Гц,
полоса подавления	32–64 Гц.

Компоненты сигнала из диапазона 30–32 Гц должны быть защищены от наложения. Разработайте подходящий однокаскадный дециматор.

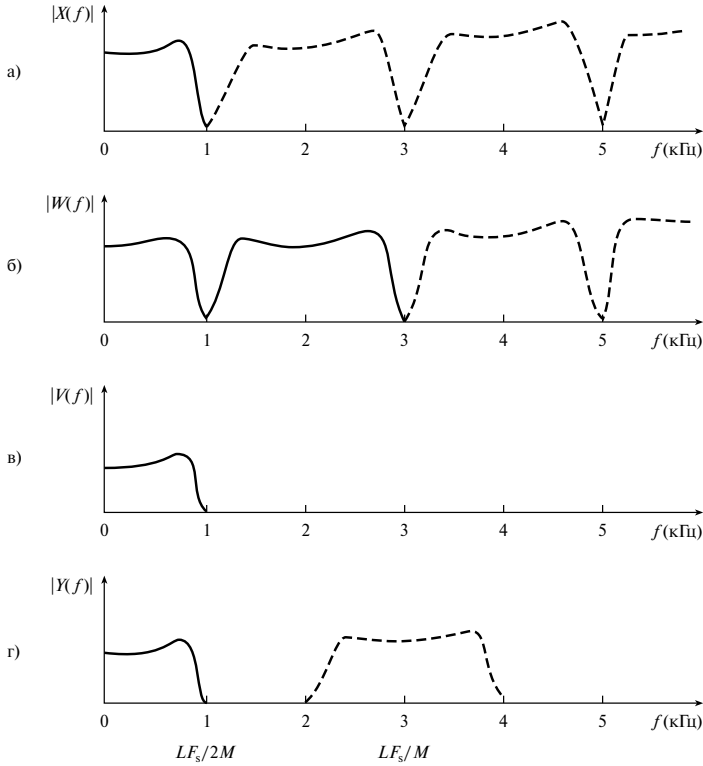


Рис. 9.6. Спектральная интерпретация увеличения частоты дискретизации с 2 кГц в 3/2 раза. Вначале скорость передачи сигнала увеличивается в 3 раза до 6 кГц (панель а); после ограничения по полосе для устранения наложения (панель в) скорость передачи сигнала уменьшается в 2 раза до 3 кГц

Решение

Блок-схема однокаскадного дециматора и спецификации фильтра защиты от наложения спектров приведены на рис. 9.7. По спецификациям и рисунку можно определить следующие параметры:

$$\Delta f = (32 - 30)/2048 = 9,766 \times 10^{-4},$$

$$\delta_p = 0,00115, \text{ так как } 20 \lg(1 + \delta_p) = 0,01 \text{ дБ},$$

$$\delta_s = 0,0001, \text{ так как } -20 \lg(\delta_s) = 80 \text{ дБ}.$$

Оценка числа коэффициентов фильтра для однокаскадного дециматора (см. главу 7):

$$N \approx \frac{D_{\infty}(\delta_p, \delta_s)}{\Delta f} - f(\delta_p, \delta_s)\Delta f + 1, \quad (9.3)$$

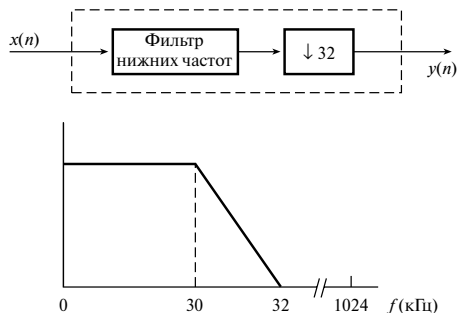


Рис. 9.7. Однокаскадный дециматор из примера 9.1

где Δf — ширина полосы перехода, нормированная на частоту дискретизации,

$$\begin{aligned}
 D_{\infty}(\delta_p, \delta_s) &= (\lg \delta_s) [a_1 (\lg \delta_p)^2 + a_2 (\lg \delta_p) + a_3] + \\
 &\quad + a_4 (\lg \delta_p)^2 + a_2 (\lg \delta_p) + a_6, \\
 f(\delta_p, \delta_s) &= 11,01217 + 0,51244 (\lg \delta_p - \lg \delta_s), \\
 a_1 &= 5,309 \times 10^{-3} & a_2 &= 7,114 \times 10^{-2}, \\
 a_3 &= -4,761 \times 10^{-1} & a_4 &= -2,66 \times 10^{-3}, \\
 a_5 &= -5,941 \times 10^{-1} & a_6 &= -4,278 \times 10^{-1},
 \end{aligned}$$

Здесь δ_p — неравномерность в полосе пропускания, δ_s — неравномерность в полосе подавления.

Подставляя значения δ_p , δ_s и Δf в формулу (9.3), находим, что $N = 3947$. Очевидно, что N слишком велико. Фактически ни один из существующих методов разработки фильтров не подходит для получения коэффициентов, поскольку ошибки аппроксимации будут слишком большими. Вообще, с практической точки зрения разработка фильтра нижних частот для однокаскадного дециматора невозможна. Из данного примера очевидно, что требуется альтернативный, более эффективный метод преобразования частоты дискретизации, особенно если шаг преобразования очень велик. Несколько таких альтернативных подходов рассмотрены в следующих разделах.

9.2.4. Многокаскадное преобразование частоты дискретизации

В предыдущем разделе изменение частоты дискретизации происходило сразу с использованием единственного коэффициента децимации или интерполяции. Если требуется значительное изменение частоты дискретизации, такой подход неэффективен, гораздо лучше преобразовать частоту в несколько этапов (используя несколько каскадов). Фактически многокаскадный подход принят в большинстве практических систем

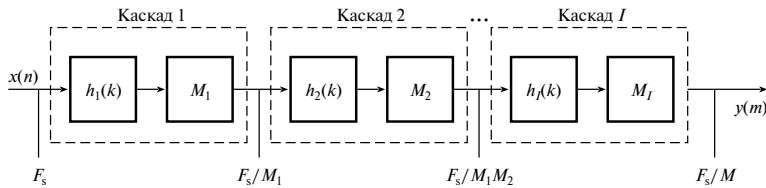


Рис. 9.8. Процесс многокаскадной децимации

обработки при нескольких скоростях. Он постепенно снижает или увеличивает частоту дискретизации, что приводит к существенному снижению требований к фильтрам защиты от наложения спектров или подавления зеркальных частот каждого каскада.

Пример I -каскадного процесса децимации приведен на рис. 9.8. Общий шаг децимации M выражается как произведение меньших шагов:

$$M = M_1 M_2 \dots M_I, \quad (9.4)$$

где целое число M_I — шаг децимации I -го каскада. Каждый каскад — это независимый дециматор, обозначенный пунктирным прямоугольником. Если $M \gg 1$, многокаскадный подход приводит к значительному снижению требований к вычислениям и памяти, ослаблению характеристик фильтров, используемых в дециматорах, а следовательно, к использованию фильтров, менее чувствительных к эффектам конечной разрядности.

Названные преимущества возможны за счет увеличения сложности разработки и реализации системы. Для иллюстрации многокаскадного подхода далее приводится множество примеров, но вначале мы разберем методологию разработки конвертера.

9.3. Разработка практических конвертеров частоты дискретизации

Разработку практического многокаскадного конвертера частоты дискретизации можно разбить на четыре этапа.

1. Задать общие требования к фильтрам защиты от наложения спектров и подавления зеркальных частот.
2. Определить оптимальное число каскадов децимации или интерполяции, дающих наиболее эффективную реализацию.
3. Определить для каждого каскада шаги децимации или интерполяции.
4. Разработать для каждого каскада подходящий фильтр.

9.3.1. Спецификация фильтра

Итак, необходимость в конвертере цифрового фильтра защиты от наложения спектров или подавления зеркальных частот становится очевидной. Фактически производительность системы обработки при нескольких скоростях критично зависит от типа

и качества используемого фильтра. Отметим, что при децимации и интерполяции могут использоваться КИХ- и БИХ-фильтры, хотя более популярны все же КИХ-фильтры.

В сфере обработки сигналов при нескольких скоростях в отличие от обычной цифровой обработки сигналов вычислительная эффективность КИХ-фильтров сравнима (а в некоторых случаях и превышает ее) с производительностью БИХ-фильтров [4, 5, 8]. Кроме того, КИХ-фильтры обладают таким полезными свойствами (подробности см. в главах 6 и 7), как возможность получения линейной фазовой характеристики и малая чувствительность к эффектам конечной разрядности, а также простота реализации. По этим причинам далее в этой главе будут рассматриваться только КИХ-фильтры. При разработке фильтров для систем обработки при нескольких скоростях можно использовать любые методы расчета коэффициентов, описанные в главе 7 для КИХ-фильтров. В частности, широко используются оптимальные и полуполосные фильтры.

Чтобы после уменьшения частоты дискретизации дециматор не приводил к наложению, должны выполняться следующие требования к фильтру:

$$\text{полоса пропускания} \quad 0 \leq f \leq f_p, \quad (9.5, a)$$

$$\text{полоса подавления} \quad F_s/2M \leq f \leq F_s/2, \quad (9.5, б)$$

$$\text{отклонение в полосе пропускания} \quad \delta_p, \quad (9.5, в)$$

$$\text{отклонение в полосе подавления} \quad \delta_s, \quad (9.5, г)$$

где $f_p < F_s/2M$, а F_s — исходная частота дискретизации. Как правило, f_p — это наивысшая значимая частота исходного сигнала.

При интерполяции возникает другая проблема — зеркальные частоты. Для ее решения используется фильтр, пропускающий только полезную информацию и ограничивающий полосу измененных данных до $F_s/2$ (возможно, даже сильнее). Хотя наивысшая рассматриваемая частота после увеличения частоты до LF_s равна $LF_s/2$, согласно теореме о дискретном представлении полосу сигнала необходимо ограничить до $F_s/2$, поскольку это — наибольшая существенная частота $x(n)$. Запишем общие требования к фильтру, используемому при интерполяции:

$$\text{полоса пропускания} \quad 0 \leq f \leq f_p, \quad (9.6, a)$$

$$\text{полоса подавления} \quad F_s/2 \leq f \leq LF_s/2, \quad (9.6, б)$$

$$\text{отклонение в полосе пропускания} \quad \delta_p, \quad (9.6, в)$$

$$\text{отклонение в полосе подавления} \quad \delta_s, \quad (9.6, г)$$

где $f_p < F_s/2$. И еще одно замечание: чтобы компенсировать уменьшение амплитуды в ходе процесса интерполяции, энергию, которая распределяется в полосе пропускания, следует увеличить в L раз.

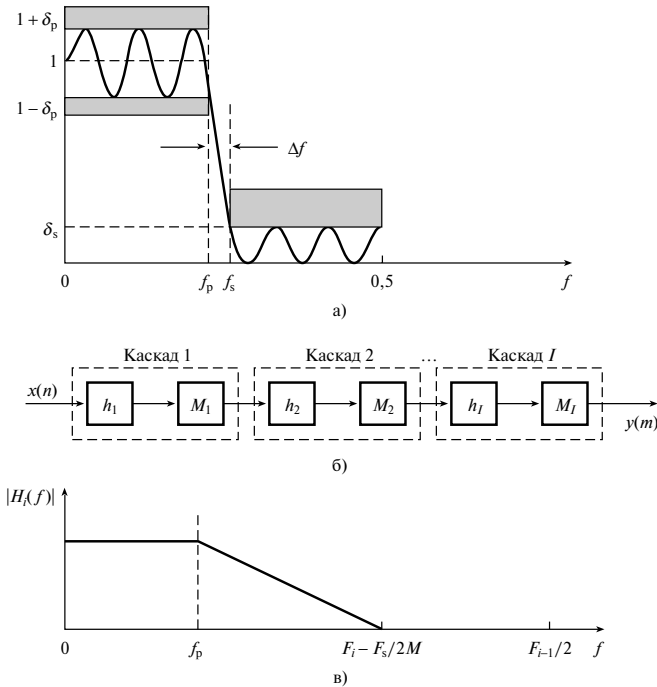


Рис. 9.9. Схема допусков фильтра нижних частот с равными колебаниями характеристики (панель а); многокаскадная структура (панель б); спецификации фильтра i -го каскада, $i = 1, 2, \dots, I$ (панель в)

9.3.2. Требования к фильтрам отдельных каскадов

Для преобразования частоты дискретизации часто используется фильтр с равными колебаниями характеристики в полосе пропускания и полосе подавления (оптимальный), хотя также можно использовать и фильтры, полученные методом взвешивания. Схема допусков для фильтра нижних частот с равными колебаниями характеристики приведена на рис. 9.9, а.

Многокаскадный дециматор (рис. 9.9, б) вводит такие требования к фильтрам каждого каскада (см. также рис. 9.9, в):

$$\text{полоса пропускания} \quad 0 \leq f \leq f_p, \quad (9.7, а)$$

$$\text{полоса подавления} \quad (F_i - F_s/2M) \leq f \leq F_{i-1}/2 \quad i = 1, 2, \dots, I, \quad (9.7, б)$$

$$\text{отклонение в полосе пропускания} \quad \delta_p/I, \quad (9.7, в)$$

$$\text{отклонение в полосе подавления} \quad \delta_s, \quad (9.7, г)$$

$$\text{длина фильтра} \quad N \approx \frac{D_{\infty}(\delta_p, \delta_s)}{\Delta f_i} - f(\delta_p, \delta_s) \Delta f_i + 1, \quad (9.7, \partial)$$

где F_i , N_i и Δf_i — выходная частота дискретизации, длина фильтра и нормированная ширина полосы перехода для i -го каскада дециматора соответственно. Параметры $D_{\infty}(\delta_p, \delta_s)$ и $f(\delta_p, \delta_s)$ имеют то же значение, что и в уравнении (9.3). Частота дискретизации на выходе i -го каскада записывается следующим образом:

$$F_i = F_{i-1}/M_i, \quad i = 1, 2, \dots, I, \quad (9.8)$$

где M_i — коэффициент децимации этого каскада. Исходная и конечная частоты дискретизации — F_0 и F_I соответственно. Эти частоты следующим образом связаны с параметрами, использованными ранее: $F_0 = F_s$ и $F_I = F_s/M$.

Чтобы при многокаскадной децимации общее отклонение в полосе пропускания составило δ_p , отклонение в каждом каскаде должно быть меньшим. Отклонение в полосе подавления каждого каскада совпадает с отклонением всей системы, поскольку при переходе сигнала между каскадами компоненты полосы подавления затухают все сильнее и сильнее. Поэтому общее отклонение в полосе подавления будет гораздо меньшим, чем задано в общих требованиях (если вообще не нулевым). Если дециматор однокаскадный, требования к его фильтру записываются в форме (9.5).

9.3.3. Определение числа каскадов и шагов децимации

Многокаскадная разработка позволяет существенно сэкономить на требованиях к вычислению и памяти по сравнению с однокаскадной структурой. Степень экономии зависит от числа используемых каскадов и выбора шагов децимации для отдельных каскадов. Основная проблема — определить оптимальное число каскадов I и шагов децимации для каждого каскада. Оптимальное число каскадов равно одному, поскольку при этом требуется минимальный объем вычислений, например, если измерять его в числе операций умножения в секунду (УВС) или общей требуемой памяти (ОТП) для хранения коэффициентов:

$$\text{УВС} = \sum_{i=1}^I N_i F_i, \quad (9.9, a)$$

$$\text{ОТП} = \sum_{i=1}^I N_i, \quad (9.9, б)$$

где N_i — число коэффициентов фильтра i -го каскада, причем симметрия коэффициентов фильтра не учитывается.

Выбор числа каскадов I и шагов децимации — это нетривиальная задача. Впрочем, на практике редко выбирается больше 3 или 4 каскадов I . Кроме того, для данного значения M существует только ограниченный набор возможных целых шагов-сомножителей. Следовательно, целесообразно определить все возможные множители M , т.е. весь набор

значений M_i и соответствующие им параметры УВС или ОТП. Затем следует методом перебора выбрать наиболее эффективное или предпочтительное решение. В алгоритмическом виде данный подход представлен в табл. 9.1. Его реализацию в виде программы на языке С можно найти в книге [Ifeachor, 2001], а в виде электронной таблицы — в книге [10].

Таблица 9.1. Алгоритм определения оптимальных значений I и M_i

-
- Задать общие параметры фильтра ($F_s, M, f_p, f_s, \delta_s, \delta_p$).
 - Для каждого значения I ($I = 1, 2, \dots, I_{\max}$) получить все возможные наборы целых коэффициентов децимации M .
 - Для каждого набора коэффициентов децимации определить из уравнений (9.9) требования к фильтру, параметры УВС и ОТП.
 - Для каждого значения I выбрать шаги децимации, дающие наиболее эффективную структуру с точки зрения требований к памяти.
 - Выбрать наиболее эффективное решение.
-

Вообще, чтобы достичь оптимального значения параметра УВС или ОТП, шаги децимации должны удовлетворять следующему соотношению [4, 5]:

$$M_1 > M_2 > \dots > M_I, \quad (9.10)$$

где M_i ($i = 1, \dots, I$) постоянны. В то же время, если множители — это целые числа, при определенных значениях I уравнению (9.10) удовлетворить можно не всегда, например, если $I = 3$ и $M = 32$ (см. комментарии к примеру 9.3).

При $I = 2$ (т.е. для двухкаскадного дециматора) оптимальные значения шагов децимации, минимизирующие параметр ОТП, равны

$$M_{1_{opt}} = \frac{2M}{2 - \Delta f + (2M\Delta f)^{1/2}}, \quad (9.11, a)$$

$$M_{2_{opt}} = \frac{M}{M_{1_{opt}}}. \quad (9.11, б)$$

Если $I > 2$, простого аналитического выражения не существует, поэтому для определения оптимальных шагов децимации M_i приходится использовать компьютерные программы оптимизации или алгоритм, приведенный в табл. 9.1.

9.3.4. Примеры разработки

Пример 9.2

1. На рис. 9.10 приведена блок-схема трехкаскадного дециматора, который используется для уменьшения частоты дискретизации с 3072 до 48 кГц. Предполагая, что шаги дискретизации — 8, 4 и 2, укажите частоту дискретизации на выходе каждого каскада.

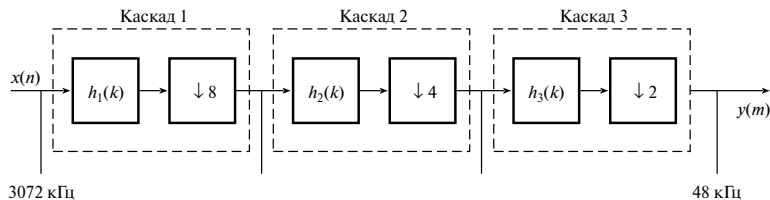


Рис. 9.10. Блок-схема дециматора из примера 9.2

2. Пусть дециматор из п. 1 удовлетворяет следующим спецификациям:

входная частота дискретизации, F_s	3072 кГц,
шаг децимации, M	64,
неравномерность в полосе пропускания	0,01 дБ,
неравномерность в полосе подавления	60 дБ,
важная полоса частот	0–20 кГц.

В каждом каскаде определите граничные частоты фильтра для децимации.

3. Предполагая, что входная и выходная частоты дискретизации дециматора равны соответственно 3072 и 48 кГц,
 - а) запишите общий шаг децимации;
 - б) запишите все возможные наборы целых шагов децимации (только в убывающем порядке), если используется децимация в два этапа;
 - в) повторите п. б, если фильтр реализован как трехкаскадная структура;
 - г) повторите п. б, если фильтр реализован как четырехкаскадная структура.
4. Для дециматора из п. 1 выразите общее число операций умножения в секунду (УВС) и общую требуемую память (ОТП) через длины фильтров N_1 , N_2 и N_3 .

Решение

1. В первом каскаде частота дискретизации уменьшается в 8 раз с 3072 до 384 кГц. Во втором каскаде частота дискретизации уменьшается еще в 4 раза с 384 до 96 кГц. Третий каскад уменьшает данную частоту вдвое с 96 до 48 кГц.
2. Граничные частоты полос пропускания всех трех фильтров для децимации фильтров защиты от наложения спектров равны 24 кГц, что позволяет сохранить полосу частот, представляющих практический интерес. Граничные частоты полос подавления позволяют использовать отличия частот дискретизации. Частота подавления определяется следующим соотношением (см. рис. 9.9, а и уравнение (9.7, б)):

$$f_{si} = F_i - \frac{F_s}{2M},$$

где

F_i — частота дискретизации на выходе каскада,

F_s — базовая частота дискретизации системы,

f_{si} — граничная частота полосы подавления каскада.

Для каскада 1 $f_{s1} = 384 - 3072/(2 \times 64) = 360$ кГц. Следовательно, граничные частоты первого каскада равны 0, 20, 360 и 1536 кГц (частота Найквиста этого каскада, т.е. 3072 кГц/2).

Для каскада 2 $f_{s2} = 96 - 3072/(2 \times 64) = 72$ кГц, набор граничных частот: 0, 20, 72 и 192 кГц.

Для каскада 3 $f_{s3} = 48 - 3072/(2 \times 64) = 24$ кГц, набор граничных частот фильтра защиты от наложения спектров: 0, 20, 24 и 48 кГц.

3, а. Общий шаг децимации $3072/48=64$.

3, б. Возможные наборы целых шагов для децимации в два этапа (в убывающем порядке):

$$32 \times 2,$$

$$16 \times 4,$$

$$8 \times 8.$$

3, в. Возможные наборы целых шагов для децимации в три этапа (в убывающем порядке):

$$16 \times 2 \times 2,$$

$$8 \times 4 \times 2.$$

3, г. Возможные наборы целых шагов для децимации в четыре этапа (в убывающем порядке):

$$4 \times 4 \times 2 \times 2.$$

4. Если выбраны шаги дискретизации $8 \times 4 \times 2$, и длины фильтров каскадов 1, 2 и 3 равны соответственно N_1 , N_2 и N_3 , общее число операций умножения в секунду равно

$$N_1 \times F_1 + N_2 \times F_2 + N_3 \times F_3 = N_1 \times 384 \times 10^3 + N_2 \times 96 \times 10^3 + \\ + N_3 \times 48 \times 10^3.$$

Общая требуемая память —

$$N_1 + N_2 + N_3.$$

Пример 9.3

Частоту дискретизации сигнала $x(n)$ нужно с помощью децимации уменьшить с 96 до 1 кГц. Наивысшая частота, представляющая практический интерес, после децимации равна 450 Гц. Допустим, что должен использоваться оптимальный КИХ-фильтр с общей неравномерностью в полосе пропускания $\delta_p = 0,01$ и отклонением в полосе пропускания $\delta_s = 0,001$. Разработайте эффективный дециматор.

Решение

Начнем с того, что найдем наиболее эффективную структуру для каждого значения I ($I = 1, 2, 3, 4$). Затем сравним эти структуры и выберем из них лучшую.

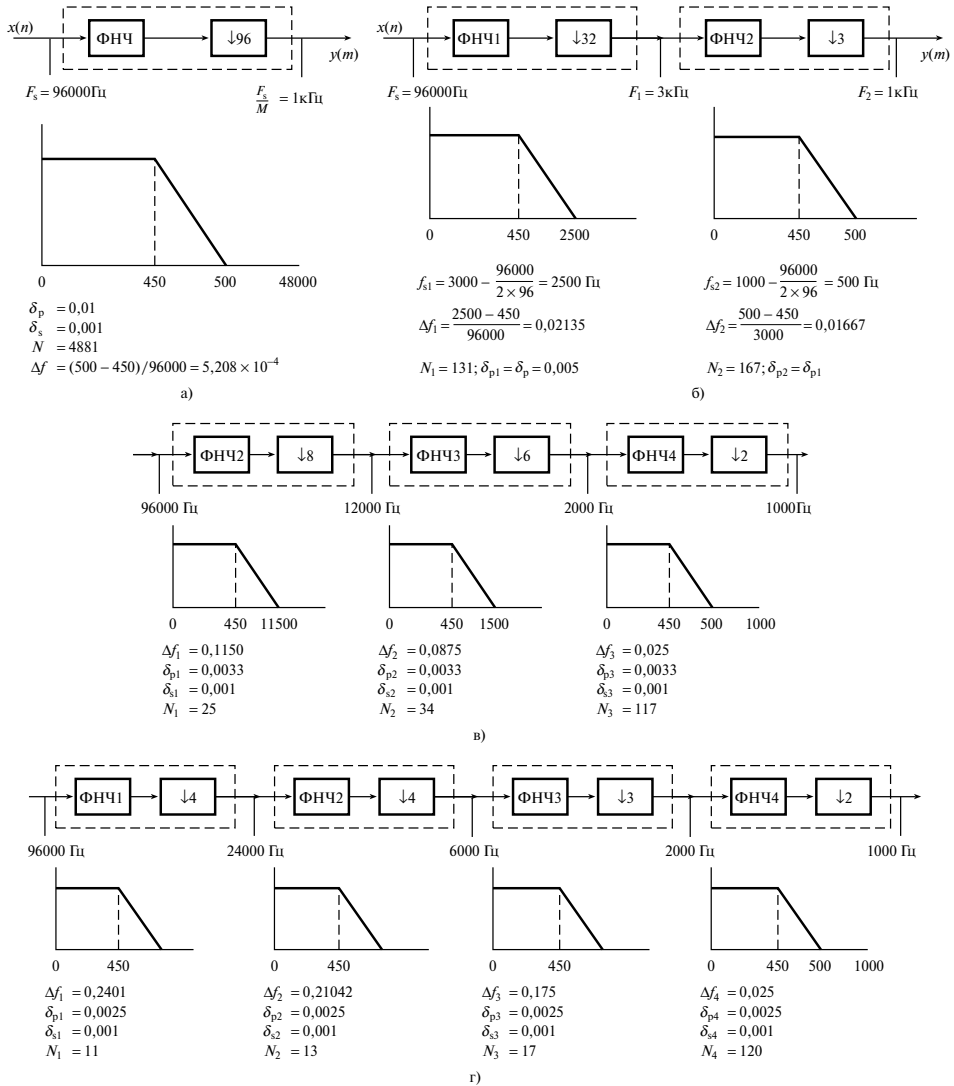


Рис. 9.11. Однокаскадный дециматор (панель а); двухкаскадный дециматор (панель б); трехкаскадный дециматор (панель в); четырехкаскадный дециматор (панель г) (пример 9.3)

1. Рассмотрим вначале однокаскадную структуру ($I = 1$). Блок-схема и спецификация полученного фильтра приведены на рис. 9.11, *а*.
2. Далее рассмотрим двухкаскадную структуру. Используя указанную в тексте программу разработки, находим, что оптимальными целыми шагами при $I = 2$ являются $M_1 = 32$ и $M_2 = 3$. Двухкаскадная система, включая ее спецификации, показана на рис. 9.11, *б*. В первом каскаде частота дискретизации уменьшается в 32 раза до 3 кГц, а во втором каскаде — еще в 3 раза до 1 кГц.
3. В трехкаскадной структуре ($I = 3$) оптимальны (с точки зрения памяти) шаги децимации $M_1 = 8$, $M_2 = 6$ и $M_3 = 2$. Соответствующая система изображена на рис. 9.11, *в*.
4. Оптимальные шаги для четырехкаскадной структуры: $M_1 = 4$, $M_2 = 4$, $M_3 = 3$ и $M_4 = 2$. Система и спецификация фильтров приведена на рис. 9.11, *г*.

Основные результаты представлены ниже.

I	N_1	N_2	N_3	N_4	M_1	M_2	M_3	M_4	УВС	ОТП
1	4881	—	—	—	96	—	—	—	48 881 000	4881
2	131	167	—	—	32	3	—	—	560 000	298
3	25	34	117	—	8	6	2	—	485 000	176
4	11	13	17	120	4	4	3	2	496 000	161

Очевидно, что в общем случае многокаскадные структуры значительно уменьшают вычислительную сложность и смягчают требования к памяти по сравнению с однокаскадными структурами. Это снижение связано с широкими полосами перехода фильтров, использованных в первых каскадах (несмотря на то, что частота дискретизации все еще велика), что дает малые значения N (число коэффициентов фильтра).

Сравнивая эффективность многокаскадных структур, находим, что снижение вычислительной сложности (УВС) и требований к памяти (ОТП) максимально проявятся при переходе от однокаскадной схемы к двухкаскадной. Снижение требований к памяти при переходе от двухкаскадной структуры к трехкаскадной и от трехкаскадной к четырехкаскадной также велико, но не настолько. Снижение вычислительной сложности также значительно при переходе от двухкаскадной структуры к трехкаскадной, но при переходе от трехкаскадной структуры к четырехкаскадной фактически происходит увеличение вычислительной сложности (УВС). В целом похоже, что наиболее эффективная структура — это $I = 3$, при этом стоит помнить, что с ростом I сложность реализации увеличивается, поэтому на практике перед окончательным решением следует учесть сложность программного и аппаратного обеспечения.

Замечания. Как обсуждается в работах [4, 5], если $M_i (i = 1, 2, \dots, I)$ — непрерывные переменные, оптимальные значения M_i удовлетворяют условию $M_1 > M_2 > \dots > M_I$.

Кроме того, значения M_i , минимизирующие требования к памяти, также минимизируют и вычислительную сложность. В то же время, если M_i могут принимать только целые значения, данное условие удовлетворяется не всегда. Поэтому предложенная программа разработки, по сути, находит решения для всех возможных наборов целых шагов. Затем из них выбирается наилучшее решение.

Наиболее эффективное решение с точки зрения вычислений или памяти может соответствовать чрезмерно большому (непрактичному) значению N в одном из каскадов. Чтобы уменьшить значения N в этих каскадах, можно увеличить длину фильтров других каскадов, выбрать иной набор шагов децимации или увеличить число каскадов. Если найдены все возможные решения, на подобные компромиссы идти легко, просто изучая имеющиеся альтернативы.

Пример 9.4

Разработайте на уровне блок-схемы двухкаскадный дециматор, перевыбирающий аудиосигнал с шагом 30 и удовлетворяющий спецификациям, представленным в табл. 9.2.

В ответе следует указать подходящую пару шагов децимации, выбор аргументировать подробным анализом вычислительной сложности и требований к памяти. Для всех фильтров, задействованных в структуре, определите частоты дискретизации на входе и выходе каскадов, а также следующие параметры:

- граничные частоты,
- нормированная ширина полосы перехода,
- неравномерность в полосе пропускания и подавления,
- длина фильтра.

Предполагается использовать КИХ-фильтры с длинами, определенными в табл. 9.2.

Таблица 9.2. Спецификации фильтров двухкаскадного дециматора

входная частота дискретизации, F_s	240 кГц,
наивысшая важная частота	3,4 кГц,
неравномерность в полосе пропускания	0,05,
неравномерность в полосе подавления	0,01,
длина фильтра, $N = \frac{-10 \lg(\delta_p \delta_s) - 13}{14,6 \Delta f} + 1$,	

где

Δf — нормированная ширина полосы перехода.

Решение

Общий вид дециматора представлен на рис. 9.12.

Если использовать целые шаги, то возможны следующие пары (согласно требованиям вычислительной эффективности первый каскад всегда имеет наибольший шаг):

$$\begin{aligned} &15 \times 2, \\ &10 \times 3, \\ &6 \times 5. \end{aligned}$$

Чтобы определить, какую пару использовать, проанализируем вычислительную сложность каждой реализации.

1. При $M_1 = 15$ и $M_2 = 2$ частота дискретизации вначале уменьшается в 15 раз до 16 кГц, а затем еще в 2 раза до 8 кГц. Имеем два фильтра со следующими параметрами:

$$\text{первый каскад: граничные частоты: } 3, 4 \text{ и } 12 \text{ кГц} \left(16 - \frac{240}{2 \times 30} \right),$$

$$\Delta f = \frac{12 - 3,4}{240} = 0,0358,$$

$$\delta_p = \frac{0,05}{2} = 0,025; \quad \delta_s = 0,01; \quad N_1 = 45,$$

второй каскад: граничные частоты: 3, 4 и 4 кГц

$$\Delta f = 0,0375,$$

$$\delta_p = \frac{0,05}{2} = 0,025; \quad \delta_s = 0,01; \quad N_2 = 43.$$

Используется две меры сложности схемы: число операций умножения в секунду (УВС) и общая требуемая память (ОТП). Для получения эффективной реализации структура дециматора выбирается так, чтобы в каждом каскаде операции фильтрации выполнялись с более низкой частотой дискретизации. В итоге получаем:

$$\text{УВС} = (45 \times 16 + 43 \times 8) \times 10^3 = 1064 \times 10^3; \quad \text{память} = 88.$$

2. При $M_1 = 10$, $M_2 = 3$ частота дискретизации вначале уменьшается в 10 раз до 24 кГц, а затем еще в 3 раза до 8 кГц. Параметры фильтров:

$$\text{первый каскад: граничные частоты: } 3, 4 \text{ и } 20 \text{ кГц} \left(24 - \frac{240}{2 \times 30} = 20 \text{ кГц} \right),$$

$$\Delta f = \frac{20 - 3,4}{240} = 0,0691,$$

$$\delta_p = \frac{0,05}{2} = 0,025; \quad \delta_s = 0,01; \quad N_1 = 23,81 \approx 24.$$

второй каскад: граничные частоты: 3, 4 и 4 кГц ($8 - 4 = 4$ кГц),

$$\Delta f = \frac{4 - 3,4}{24} = 0,025; \quad \delta_p = \frac{0,05}{2} = 0,025; \quad \delta_s = 0,01; \quad N_2 = 64.$$

Две меры сложности в данном случае имеют такие значения:

$$\text{УВС} = (24 \times 24 + 64 \times 8) \times 10^3 = 1088 \times 10^3, \quad \text{ОТП} = 88.$$

3. Наконец, рассмотрим третью возможную пару, $M_1 = 6$ и $M_2 = 5$. Проведя тот же анализ, что и выше, получаем такие значения:

первый каскад: граничные частоты: 3, 4 и 36 кГц,

$$\Delta f = 0,1358; \quad \delta_p = \frac{0,05}{2} = 0,025; \quad \delta_s = 0,01; \quad N_1 = 13,$$

второй каскад: граничные частоты: 3, 4 и 4 кГц

$$\Delta f = 0,015; \quad \delta_p = \frac{0,05}{2} = 0,025; \quad \delta_s = 0,01; \quad N_2 = 106,$$

$$\text{УВС} = 1369 \times 10^2; \quad \text{ОТП} = 119.$$

Результаты, полученные для трех рассмотренных ситуаций, сведены в табл. 9.3. Сравнивая вычислительную сложность и требования к памяти, видим, что наиболее подходящая пара шагов децимации — $M_1 = 15$, $M_2 = 2$.

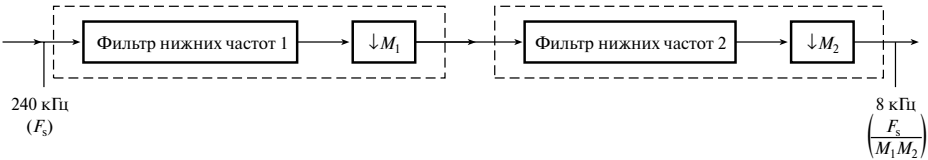


Рис. 9.12. Двухкаскадный дециматор

Таблица 9.3. Вычислительная сложность (УВС) и общая требуемая память (ОТП)

Шаги децимации	УВС	ОТП
$M_1 = 15; M_2 = 2$	1064×10^3	88
$M_1 = 10; M_2 = 3$	1088×10^3	88
$M_1 = 6; M_2 = 5$	1368×10^3	119

9.4. Программная реализация дециматоров

На рис. 9.13, а представлена блок-схема дециматора, где через $h(k)$ обозначен цифровой фильтр защиты от наложения спектров. Если фильтр реализован в прямой форме (т.е. используются линии задержки с отводами), выход фильтра $w(n)$ и его вход $x(n)$ связаны следующим соотношением:

$$w(n) = \sum_{k=0}^{N-1} h(k)x(n-k), \tag{9.12, а}$$

где N — число коэффициентов КИХ-фильтра. Поскольку выход дециматора $y(m) = w(mM)$, уравнение дециматора можно записать следующим образом:

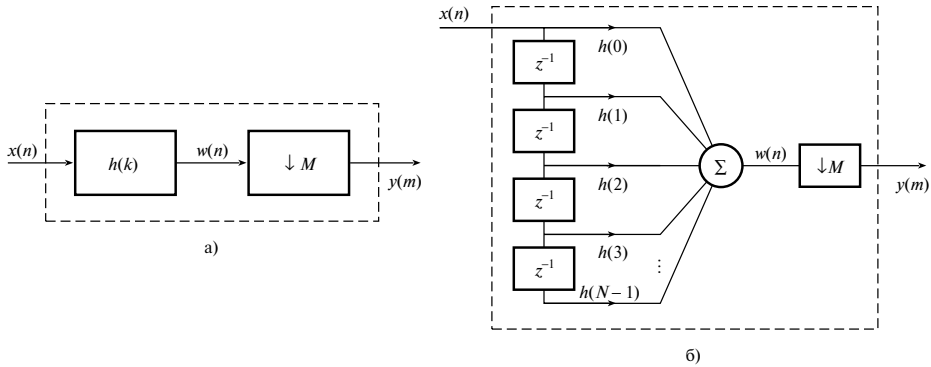


Рис. 9.13. Простая блок-схема дециматора (панель а); функциональная схема прохождения сигнала через дециматор (панель б)

$$y(m) = \sum_{k=0}^{N-1} h(k)x(Mm - k). \quad (9.12, б)$$

Функциональная схема дециматора приведена на рис. 9.13, б. Вход $x(n)$ подается на линию задержки по одной выборке за такт. Для каждой M выборок $x(n)$, поданных на линию задержки, вычисляется одна выходная выборка $y(m)$. При этом берется первая выборка $w(n)$, отбрасываются следующие $(M - 1)$ выборок, запоминается идущая далее выборка, затем снова отбрасываются $(M - 1)$ выборок и т.д. Поскольку для каждой сохраненной выборки следующие $(M - 1)$ выборок $w(n)$ отбрасываются, нет смысла вычислять их значение по формуле (9.12, а). Следовательно, операцию выборки с пониженной частотой (отбрасывание выборок) можно выполнить до умножения выходных выборок на коэффициенты (рис. 9.14, а). Теперь операции умножения и сложения с участием коэффициентов фильтра выполняются при более низкой частоте дискретизации F_s/M , поэтому вычислительные затраты уменьшены в M раз.

Общая схема работы однокаскадного дециматора приведена на рис. 9.14, б. Функциональная схема работы трехкаскадного дециматора показана на рис. 9.15 и является очевидным расширением соответствующей однокаскадной схемы.

9.4.1. Программа многокаскадной децимации

В приложении 9А приводится самодостаточная интерактивная программа на языке С, написанная на основе изложенных выше методов. Программа выполняет децимацию исходных данных с использованием до трех каскадов децимации; см. функциональную схему на рис. 9.15. На каждом этапе децимации требуется целый шаг децимации и набор коэффициентов N -точечного фильтра, представляющих цифровой КИХ-фильтр с линейной фазовой характеристикой.

Входные данные считываются из файла данных на персональном компьютере, а данные после децимации заносятся в заданный пользователем файл выходных данных по

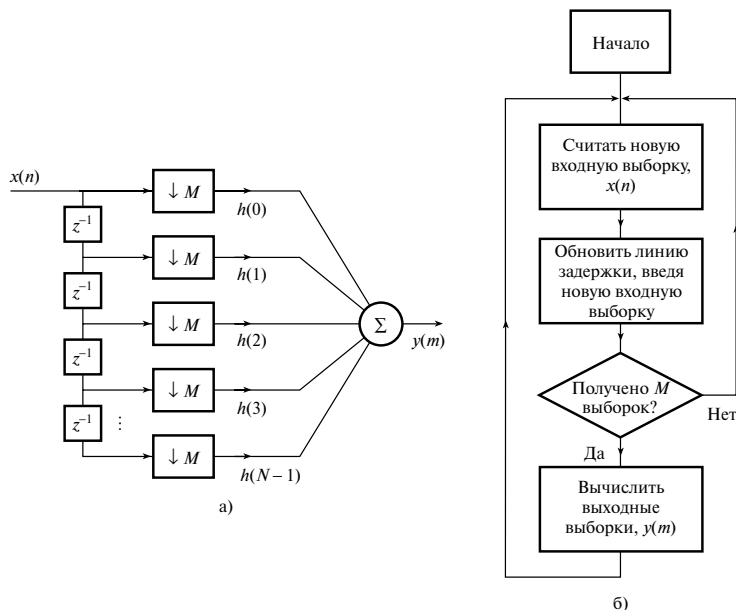


Рис. 9.14. Более эффективная диаграмма прохождения сигнала через дециматор (панель а); функциональная схема процесса децимации (панель б)

одной выборке за такт. Предположим, что используется трехкаскадная децимация (см. рис. 9.15). Для каждой M_1 входных выборок каскада 1 вычисляется одна выходная выборка. Для каждой M_2 выходных выборок каскада 1 вычисляется одна входная выборка каскада 2. Наконец, для каждой M_3 выходных выборок каскада 2 вычисляется одна выходная выборка каскада 3. Итак, в конце цикла децимации для каждой M входных выборок сигнала $x(n)$, где $M = M_1 M_2 M_3$, вычисляется одна выходная выборка, которая и заносится в файл выходных данных. Процесс продолжается, пока не будут обработаны все входные выборки.

Названная программа является самодостаточной. Для ее применения пользователь должен задать число каскадов, общий шаг децимации, кроме того, для каждого каскада нужно задать шаг децимации и набор коэффициентов КИХ-фильтра с линейной фазовой характеристикой, реализованного в прямой форме. Пользователь задает имена файла с данными для децимации, файла с коэффициентами фильтров и файла, в котором будут записаны выходные данные после децимации.

9.4.2. Тестовый пример использования программы децимации

Для проверки процесса децимации используется следующая входная последовательность (широкополосный сигнал) (см. [6]):

$$\begin{aligned} x(n) &= -\alpha, \quad n = 0 \\ &= (1 - \alpha^2)\alpha^{n-1}, \quad n = 1, 2, \dots, \end{aligned} \quad (9.13)$$

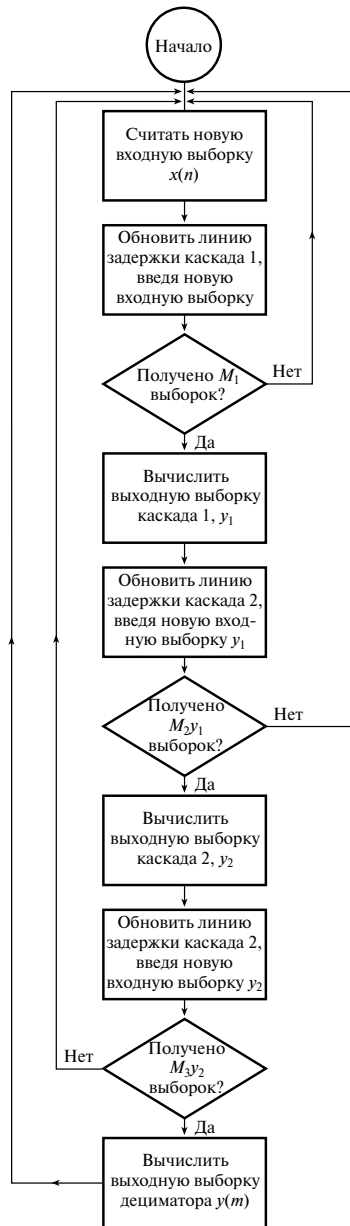


Рис. 9.15. Функциональная схема трехкаскадного дециматора

где $\alpha = 0,9$. Первые 29 выборки последовательности приведены в табл. 9.4.

В данном примере децимация выполняется в два этапа с шагами 5 и 2 (общий шаг децимации — 10). В табл. 9.4 перечислены коэффициенты КИХ-фильтров (длины фильтров — 25 и 28 соответственно) и представлен результат децимации.

9.4.2.1. Выходная задержка

Выход дециматора будет запаздывать относительно входа на определенное число выборок в зависимости от типов фильтров, используемых в каскадах дециматора. Предполагая, что дециматор содержит КИХ-фильтры с линейной фазовой характеристикой, можно получить следующие выражения для задержки одно-, двух- и трехкаскадного дециматора:

$$T(1 \text{ каскад}) = \frac{1}{M}[T_1 - (M - 1)] \text{ выборков,} \quad (9.14, a)$$

$$T(2 \text{ каскада}) = \frac{1}{M_1 M_2}[T_1 + M_1 T_2 - (M_1 M_2 - 1)] \text{ выборков,} \quad (9.14, б)$$

$$T(3 \text{ каскада}) = \frac{1}{M_1 M_2 M_3}[T_1 + M_1 T_2 + M_1 M_2 T_3 - (M_1 M_2 M_3 - 1)] \text{ выборков,} \quad (9.14, в)$$

где T_i , задержка фильтра i -го каскада, вычисляется как

$$T_i = (N_i + 1)/2 \text{ выборков,}$$

а N_i — это число коэффициентов фильтра i -го каскада. В приведенном выше примере двухкаскадного дециматора задержки фильтров равны соответственно 14,5 и 13 выборков, а общая задержка составляет

$$T(2 \text{ каскада}) = [1/(5 \times 2)][14,5 + 5 \times -(5 \times 2 - 1)] = 7,05 \text{ выборков.}$$

Если нужно, чтобы задержка выражалась целым числом выборок, значения N_i нужно определять так, чтобы T , вычисленное по приведенной выше формуле, было целым. Такое требование возникает, например, когда (нужно сравнивать входные и выходные выборки). Например, при передаче данных через фильтр верхних частот в системе обработки при нескольких скоростях в выходную выборку нужно ввести поправку на задержку при проходе через дециматор и интерpolator.

9.5. Программная реализация интерполяторов

На рис. 9.16, *а* приведена блок-схема интерполатора, а его функциональная схема изображена на рис. 9.16, *б*. Для каждой входной выборки $x(n)$, поданной на интерполатор, экспандер (прямоугольник со стрелочкой, направленной вверх) вводит после входной выборки $(L - 1)$ нулевую выборку. Затем все выборки фильтруются, и получается сигнал $y(m)$. Итак, для каждой входной выборки сигнала $x(n)$ имеем L выборок

Таблица 9.4. Данные тестового примера децимации

n	$x(n)$	$y(m)$	$h_1(k)$	$h_2(k)$
0	-0,9000		-0,000 174	-0,000 303
1	0,1900		-0,002 682	0,001 807
2	0,1710		-0,006 346	0,003 120
3	0,1539		-0,011 033	-0,001 169
4	0,1385		-0,014 156	-0,009 267
5	0,1247		-0,012 024	-0,007 792
6	0,1122		-0,000 775	0,011 124
7	0,1010		0,021 904	0,027 651
8	0,0909		0,055 181	0,007 674
9	0,0818	0,000 040	0,094 397	-0,045 444
10	0,0736		0,131 836	-0,064 816
11	0,0662		0,158 866	0,022 946
12	0,0596		0,168 728	0,202 371
13	0,0537		0,158 866	0,352 610
14	0,0483		0,131 836	0,352 610
15	0,0435		0,094 397	0,202 371
16	0,0391		0,055 181	0,022 946
17	0,0352		0,021 904	-0,064 816
18	0,0317		-0,000 775	-0,045 444
19	0,0285	-0,000 286	-0,012 024	0,007 674
20	0,0257		-0,014 156	0,027 651
21	0,0231		-0,011 033	0,011 124
22	0,0208		-0,006 346	-0,007 792
23	0,0187		-0,002 682	-0,009 267
24	0,0168		-0,000 174	-0,001 169
25	0,0152			0,003 120
26	0,0136			0,001 807
27	0,0123			-0,000 303
28	0,0110			
29	0,0099	0,001 116		
30	0,0089			
31	0,0081			
32	0,0072			
33	0,0065			
34	0,0059			
35	0,0053			
36	0,0048			
37	0,0043			
38	0,0039			
39	0,0035	-0,001 659		
40	0,0031			
41	0,0028			
42	0,0025			
43	0,0023			
44	0,0020			
45	0,0018			
46	0,0017			
47	0,0015			
48	0,0013			
49	0,0012	-0,000 402		
50	0,0011			

Здесь $x(n)$ и $y(m)$ — выходные данные и данные после децимации; $h_1(k)$ и $h_2(k)$ — коэффициенты фильтра для децимации.

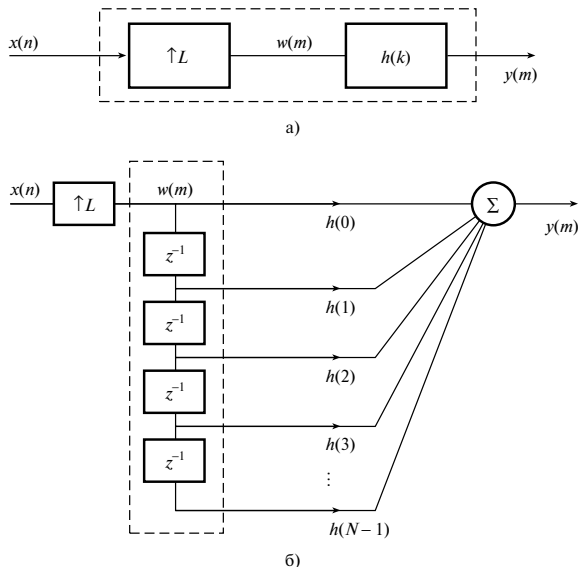


Рис. 9.16. Блок-схема интерполятора (панель а); функциональная схема интерполятора (панель б)

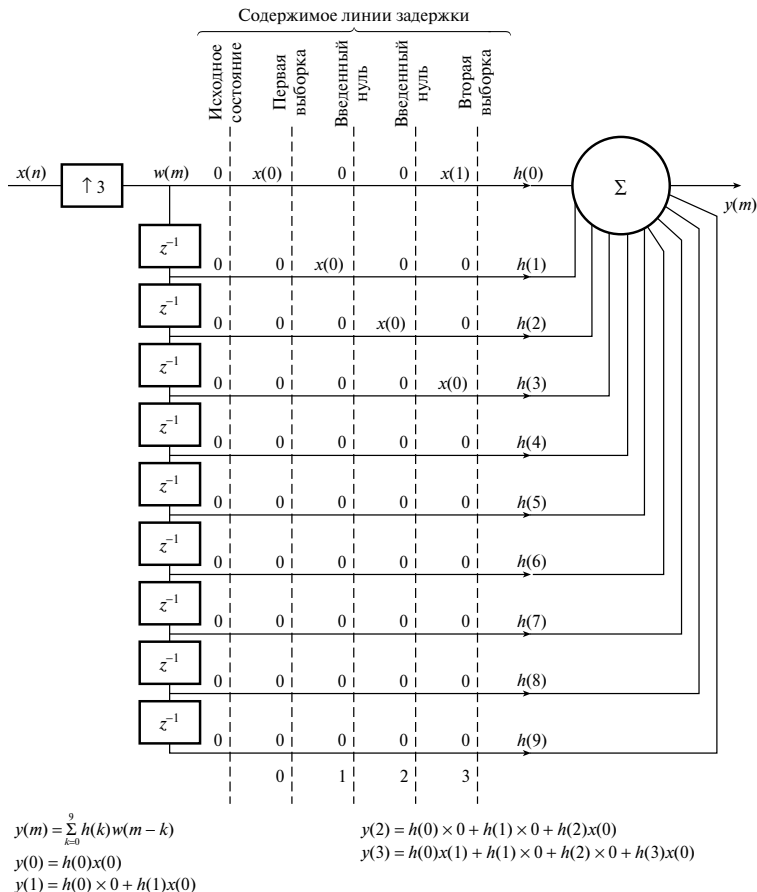
сигнала $y(m)$. Таким образом, интерpolator эффективно увеличивает входную частоту дискретизации с F_s до LF_s . Отметим, что одним из последствий ввода $(L - 1)$ нулей после каждой выборки является рассеивание энергии каждой входной выборки по L выходным выборкам. Следовательно, коэффициент усиления интерполятора равен $1/L$. После интерполяции каждую выходную выборку следует умножить на L , чтобы восстановить ее до прежнего уровня.

Уравнения интерполяции имеют следующий вид:

$$y(m) = \sum_{k=0}^{N-1} h(k)w(m - k) \quad (9.15, a)$$

$$w(m - k) = \begin{cases} x[(m - k)/L], & m - k = 0, L, 2L, \dots \\ 0 & \text{иначе} \end{cases} \quad (9.15, б)$$

Описанный процесс иллюстрируется на рис. 9.17 для простого случая $L = 3$, длина фильтра $= 10$. На линии задержки подается входная выборка с последующими двумя нулями, за ней следует выборка и два нуля и т.д. Для каждой выборки линии задержки (данные или нули) вычисляется выходная выборка. На рисунке показано содержимое линии задержки после подачи в интерpolator двух выборок $x(n)$. Видно, что для каждой поданной входной выборки вычисляются три выходные выборки. Между ненулевыми выборками линии задержки (т.е. выборками собственно $x(n)$) располагается $(L - 1)$ нулей (в данном примере — два). Очевидно, что операции умножения на нулевые выборки выполнять не обязательно.

Рис. 9.17. Иллюстрация процесса интерполяции для простого случая $L = 3$

На рис. 9.18 представлена функциональная схема однокаскадного интерполятора. В данной реализации при вычислении выходных выборок используются только ненулевые выборки. Другая функциональная схема, на этот раз интерполятора, который может иметь до трех каскадов, приведена на рис. 9.19.

В другой эффективной реализации, известной как *многофазная*, или *мультифазная, фильтрация* [8], используется тот факт, что некоторые выборки линии задержки равны нулю. В этом случае экспандер удаляется полностью, чтобы не нужно было хранить нулевые выборки. Далее линия задержки сокращается до N/L позиций (где N — длина фильтра защиты от наложения спектров, а L — шаг интерполяции). При таком подходе для каждой входной выборки, поданной на линию задержки, на основе N/L выборок линии задержки рассчитываются L выходных выборок, причем каждая выходная выборка вычисляется с иным набором коэффициентов фильтра (т.е. коэффициенты фильтра, соответствующие нулевым выборкам, опускаются). Многофазный фильтр имеет такое

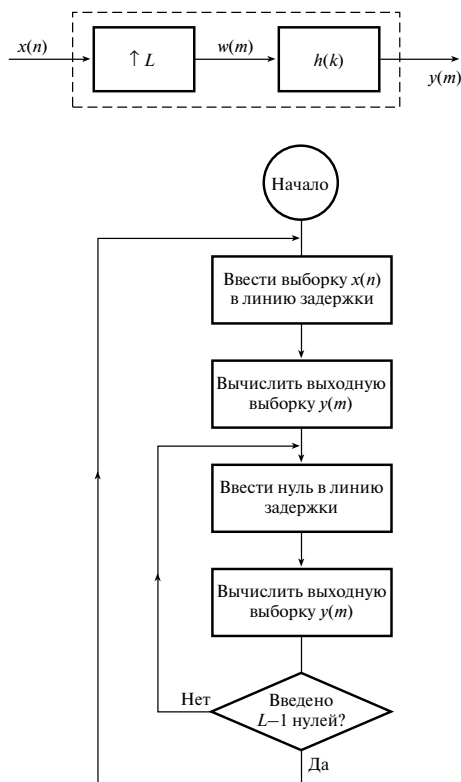


Рис. 9.18. Функциональная схема процесса интерполяции

ограничение: отношение N к L должно быть целым. В следующем разделе рассмотрена программа разработки трехкаскадного интерполятора на языке С.

9.5.1. Программа многокаскадной интерполяции

Программа основана на методе, описанном в предыдущем разделе, и интерполирует входные данные с использованием до трех каскадов; см. функциональную схему на рис. 9.19. Отметим, что вследствие дуальности децимации и интерполяции (дециматор и интерpolator образуют общий конвертер частоты дискретизации) каскады интерполяции нумеруются в обратном порядке. На каждом этапе интерполяции требуется целый шаг интерполяции и набор коэффициентов N -точечного фильтра, представляющих цифровой КИХ-фильтр с линейной фазовой характеристикой. Программа используется на персональном компьютере, хотя ее легко переделать под выполнение на любом другом оборудовании.

Входные данные считываются из файла данных на компьютере, а данные интерполяции заносятся в выходной файл по выборке за раз. Предположим, что используется трехкаскадная интерполяция (см. рис. 9.19). В каскаде 3 вычисляется L_3 выборок. Для

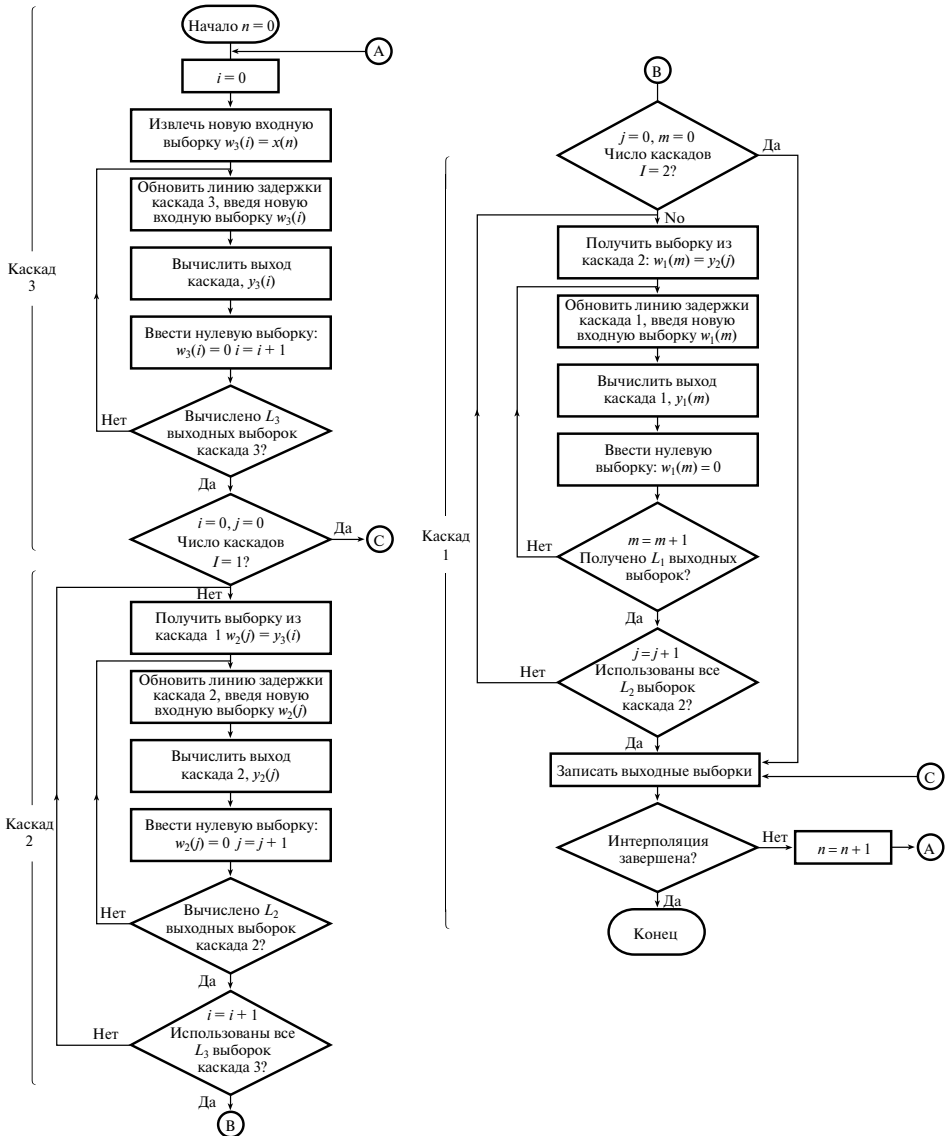


Рис. 9.19. Интерполяция с использованием до трех каскадов. Если, например, $I = 1$, то выполняются только блоки с меткой “каскад 3”

каждой из этих выборок в каскаде 2 вычисляется L_2 выборок. Для каждой выборки, поступившей с каскада 2, в каскаде 1 вычисляется L_1 выборок. В конце цикла для каждой выборки выходного сигнала $x(n)$ вычисляется L выходных выборок (где $L = L_1 L_2 L_3$), которые заносятся в файл выходных данных. Процесс продолжается, пока не будут обработаны все входные выборки. Названную программу интерполяции можно найти на компакт-диске, который прилагается к книге [Ifeachor, 2001] (подробности см. в предисловии).

Указанная программа является самодостаточной. Для ее применения пользователь должен задать число каскадов, общий шаг интерполяции, кроме того, для каждого каскада нужно задать шаг интерполяции и набор коэффициентов КИХ-фильтра. Пользователь задает имена файла с данными для интерполяции, с коэффициентами фильтров и файла, в котором будут записаны выходные данные после интерполяции.

9.5.2. Тестовый пример

Входная последовательность определяется так же, как для дециматора. Пять первых выборок последовательности приведены в табл. 9.5.

В данном примере выполняется двухкаскадная интерполяция. Шаги интерполяции в третьем и втором каскадах равны соответственно 2 и 5, так что общий шаг интерполяции — 10. Результат интерполяции представлен в табл. 9.5. Используемые коэффициенты КИХ-фильтров также приведены в табл. 9.5; длины фильтров — 25 и 28 соответственно.

9.5.2.1. Выходная задержка

Выход интерполятора будет запаздывать относительно входа на определенное число выборок. Ниже приводятся выражения для групповых задержек одно-, двух- и трехкаскадных интерполяторов.

$$T(1 \text{ каскад}) = T_1 \text{ выборка}, \quad (9.16, a)$$

$$T(2 \text{ каскада}) = (T_1 + M_1 T_2) \text{ выборка}, \quad (9.16, б)$$

$$T(3 \text{ каскада}) = (T_1 + M_1 T_2 + M_1 M_2 T_3) \text{ выборка}, \quad (9.16, в)$$

где T_i — задержка фильтра i -го каскада: $T_i = (N_i + 1)/2$ выборка (N_i — число коэффициентов фильтра данного каскада). В приведенном выше примере фильтры имеют задержки соответственно 13 и 14,5, так что суммарная задержка равна $13 + 2 \times 14,5 = 42$ выборки.

Если нужно, чтобы задержка выражалась целым числом выборок, значения N_i следует выбирать так, чтобы T , вычисленное по приведенной выше формуле, было целым. Следовательно, если требуется сравнить входные и выходные выборки (например, в узкополосном фильтре верхних частот, где выполняется операция, обратная к НЧ-фильтрации), в выходную выборку нужно ввести поправку на задержку.

Таблица 9.5. Данные тестового примера интерполяции

n	$x(n)$	$y(m)$	$h_2(k)$	$h_1(k)$
0	-0,9000	$-4,744\,98 \times 10^{-7}$	-0,000 303	-0,000 174
		$-7,313\,815 \times 10^{-6}$	0,001 807	-0,002 682
		$-1,730\,554 \times 10^{-5}$	0,003 120	-0,006 346
		$-3,008\,7 \times 10^{-5}$	-0,001 169	-0,011 033
		$-3,860\,341 \times 10^{-5}$	-0,009 267	-0,014 156
		$-2,995\,969 \times 10^{-5}$	-0,007 792	-0,012 024
		$4,150\,394 \times 10^{-5}$	0,011 124	-0,000 775
		$1,629\,372 \times 10^{-4}$	0,027 651	0,021 904
		$3,299\,083 \times 10^{-4}$	0,007 674	0,055 181
		$4,876\,397 \times 10^{-4}$	-0,045 444	0,094 397
		$5,600\,492 \times 10^{-4}$	-0,064 816	0,131 836
		$5,226\,861 \times 10^{-4}$	0,022 946	0,158 866
		$2,857\,456 \times 10^{-4}$	0,202 371	0,168 728
		$-1,480\,226 \times 10^{-4}$	0,352 610	0,158 866
1	0,1900	$-7,700\,115 \times 10^{-4}$	0,352 610	0,131 836
		-0,001 544 5	0,202 371	0,094 397
		$-2,448\,377 \times 10^{-3}$	0,022 946	0,055 181
		$-3,400\,52 \times 10^{-3}$	-0,064 816	0,021 904
		$-4,320\,959 \times 10^{-3}$	-0,045 444	-0,000 775
		$-5,079\,387 \times 10^{-3}$	0,007 674	-0,012 024
		$-5,534\,875 \times 10^{-3}$	0,027 651	-0,014 156
		$-5,728\,923 \times 10^{-3}$	0,011 124	-0,011 033
		$-5,466\,501 \times 10^{-3}$	-0,007 792	-0,006 346
		$-4,756\,987 \times 10^{-3}$	-0,009 267	-0,002 682
		$-3,522\,772 \times 10^{-3}$	-0,001 169	-0,000 174
		$-1,715\,353 \times 10^{-3}$	0,003 120	
		$5,557\,998 \times 10^{-4}$	0,001 807	
		$3,324\,822 \times 10^{-3}$	-0,000 303	
2	0,1710	$6,400\,164 \times 10^{-3}$		
		$9,565\,701 \times 10^{-3}$		
		0,012 597 6		
		$1,544\,317 \times 10^{-2}$		
		$1,774\,401 \times 10^{-2}$		
		$1,933\,819 \times 10^{-2}$		
		$1,984\,539 \times 10^{-2}$		
		$1,894\,878 \times 10^{-2}$		
		$1,681\,832 \times 10^{-2}$		
		$1,303\,225 \times 10^{-2}$		
		$7,845\,689 \times 10^{-3}$		
		$1,357\,867 \times 10^{-3}$		
		$-6,262\,392 \times 10^{-3}$		
		$-1,462\,789 \times 10^{-2}$		
3	0,1539	$-2,343\,143 \times 10^{-2}$		
		$-3,207\,272 \times 10^{-2}$		
		$-3,972\,186 \times 10^{-2}$		
		$-4,567\,938 \times 10^{-2}$		
		$-4,993\,166 \times 10^{-2}$		
		$-5,142\,782 \times 10^{-2}$		
		$-5,009\,625 \times 10^{-2}$		
		$-4,527\,419 \times 10^{-2}$		
4	0,1385			

Здесь $x(n)$ и $y(m)$ — входные и интерполированные данные, $h_2(k)$ и $h_1(k)$ — интерполирующие фильтры

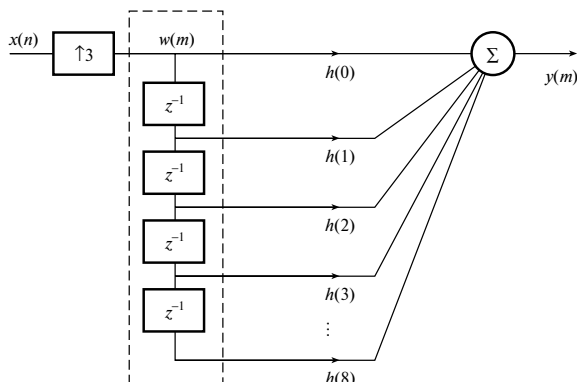


Рис. 9.20. Интерполятор 1:3 с 9-точечным КИХ-фильтром, реализованным в прямой форме

9.6. Преобразование частоты дискретизации с использованием многофазного фильтра

Существует альтернативная эффективная структура, которую можно использовать при реализации дециматоров и интерполяторов, — так называемая *многофазная структура*. Поскольку многофазная фильтрация при интерполяции проще, чем при децимации, рассмотрим вначале ее.

9.6.1. Многофазная реализация интерполяторов

Многофазная реализация интерполяторов построена на основе того факта, что некоторые выборки линии задержки интерполятора равны нулю. В этом случае экспандер удаляется полностью, чтобы не нужно было хранить нулевые выборки. Далее линия задержки сокращается до N/L позиций (где N — длина фильтра защиты от наложения спектров, а L — шаг интерполяции). При таком подходе для каждой входной выборки, поданной на линию задержки, на основе N/L выборок линии задержки рассчитываются L выходных выборок, причем каждая выходная выборка вычисляется с иным набором коэффициентов фильтра (т.е. коэффициенты фильтра, соответствующие нулевым выборкам, опускаются).

Чтобы лучше понять многофазную реализацию интерполяторов, рассмотрим простой интерполятор 1:3, изображенный на рис. 9.20. В этом случае $L = 3$, а число коэффициентов фильтра $N = 8$.

Согласно схеме на рис. 9.20 на линию задержки подается входная выборка, за которой следует два нуля ($L - 1$), затем идет следующая входная выборка, снова два нуля и т.д.

Содержимое линии задержки после подачи первых четырех входных выборок ($x(0)$, $x(1)$, $x(2)$ и $x(3)$), за каждой из которых было подано две нулевые выборки, показано на рис. 9.21, где также указаны моменты взятия выборок при двух частотах дискретизации.

Для каждой выборки, поданной на линию задержки (данные или нули), вычисляется выходная выборка. Следовательно, для каждой реальной выборки (выборки $x(n)$)

вычисляется три выходные выборки. Между каждой парой таких ненулевых выборок располагается $(L - 1)$ нулей (в данном примере — два). Очевидно, что выполнять умножение на нулевые выборки не обязательно.

На рис. 9.21 показаны последовательные выходные выборки интерполятора. В использованной реализации при вычислении выходных выборок применяются только ненулевые входные выборки. Здесь необходимо сделать некоторые замечания.

1. Каждая входная выборка порождает три выходные выборки, причем в расчете каждой выходной выборки фигурируют три подфильтра, каждый из которых имеет три коэффициента:
 - $\{h(0), h(3), h(6)\}$, $\{h(1), h(4), h(7)\}$, $\{h(2), h(5), h(8)\}$;
 - для каждой новой входной выборки процесс повторяется;
 - подфильтры работают с более низкой частотой дискретизации.
2. Общий фильтр представляет собой параллельное соединение подфильтров; (рис. 9.22, а). Подфильтры, называемые также *многофазными фильтрами*, имеют общую линию задержки, что снижает общие требования к памяти в 3 раза. Для каждой новой входной выборки каждый многофазный фильтр выдает выходную выборку с более низкой частотой дискретизации; (рис. 9.22, б).

Общая структура реализации интерполяции на основе многофазных фильтров показана на рис. 9.23. В данном случае для каждой новой входной выборки сигнала $x(n)$ существует L выходных выборок сигнала $y(m)$, по одной на каждый многофазный фильтр. Верхний многофазный фильтр, т.е. $\rho_0(n)$, дает выходные выборки $y_0(n)$, следующий многофазный фильтр, $\phi_1(n)$, дает выход $y_1(n)$, и т.д. На практике многофазные фильтры часто реализуются с использованием коммутативной модели, изображенной на рис. 9.23, б. Коммутатор вращается в направлении против часовой стрелки, начиная с верхнего положения.

Анализ многофазных фильтров показывает, что это — широкополосные фильтры, имеющие разные сдвиги фаз (поэтому фильтры называются многофазными). Вычислительная эффективность структуры на основе многофазных фильтров объясняется тем, что единый N -точечный КИХ-фильтр разбивается на набор небольших подфильтров длины N/L , где N выбирается целым и кратным L , причем каждый фильтр работает с более низкой частотой дискретизации.

Коэффициенты многофазных фильтров, используемых при интерполяции, выражаются следующим образом:

$$\rho_k(n) = h(k + nL), \quad k = 0, 1, \dots, L - 1; \quad 0, 1, \dots, \frac{N}{L} - 1.$$

Интерполятор, изображенный на рис. 9.23, можно преобразовать в дециматор, приведенный на рис. 9.24. Обратите внимание на то, что в этом случае коммутатор вращается по часовой стрелке, начиная с верхнего положения в момент $m = 0$. Коэффициенты многофазных фильтров дециматора следующим образом связаны с коэффициентами фильтров исходного интерполятора:

$$\rho_k(n) = h(k + nM), \quad k = 0, 1, \dots, M - 1; \quad 0, 1, \dots, \frac{N}{M} - 1.$$

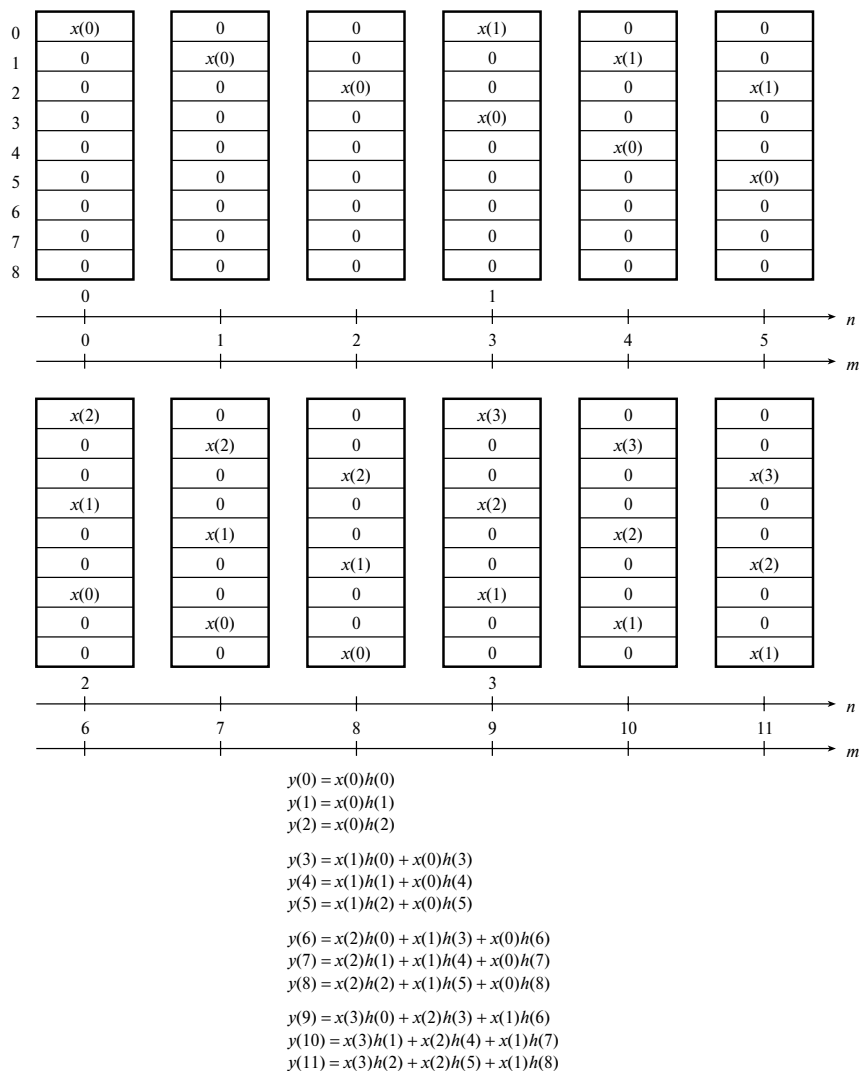
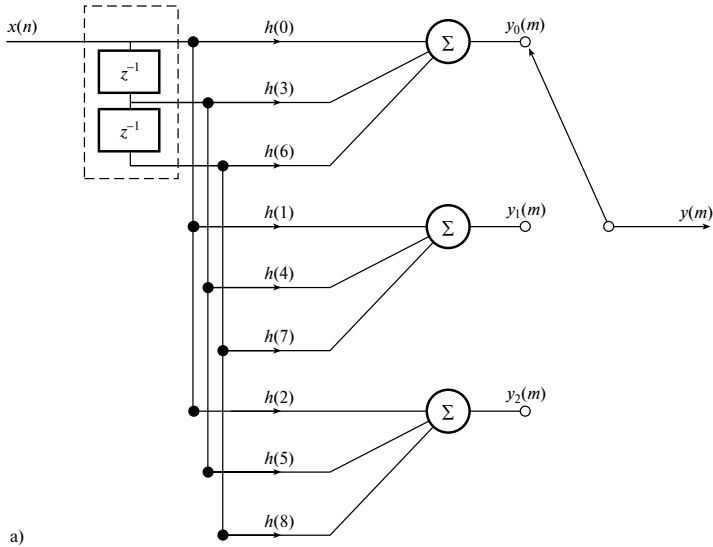
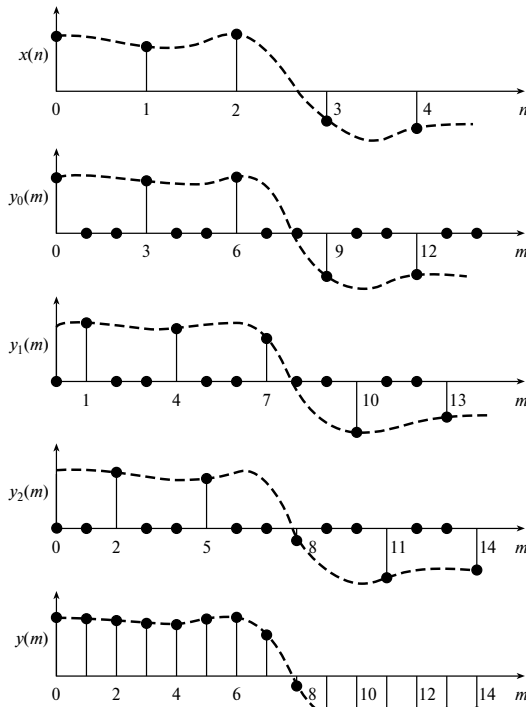


Рис. 9.21. Интерполяция 1:3 с использованием многофазных фильтров: а) содержимое линии задержки в два различных момента взятия выборки n и m ; б) выходные выборки многофазных фильтров



а)



б)

Рис. 9.22. Реализация интерполяции 1:3 на базе многофазных фильтров

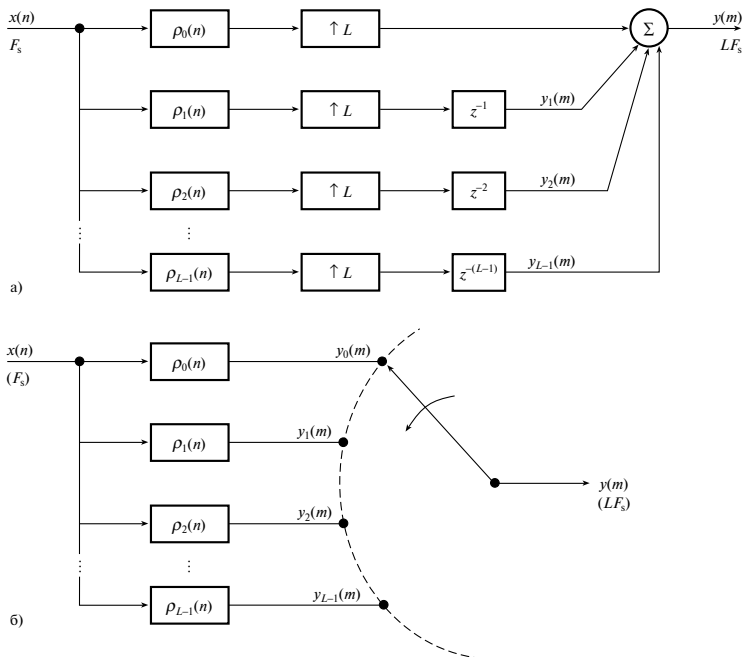


Рис. 9.23. Общая модель интерполяции на базе многофазных фильтров (панель а).
Коммутативная модель реализации (панель б)

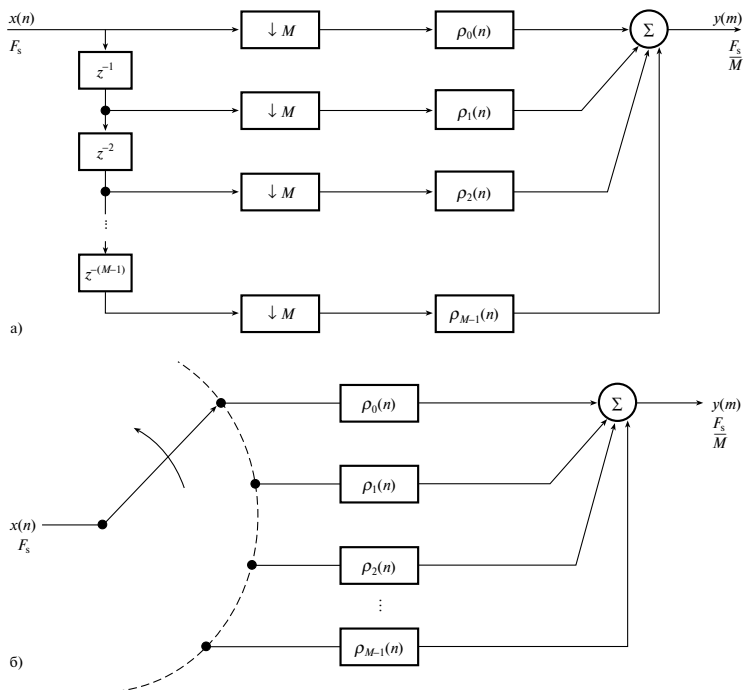


Рис. 9.24. Общая модель децимации на базе многофазных фильтров (панель а).
Коммутативная модель реализации (панель б)

9.7. Примеры применения

Методы работы при нескольких скоростях ускорили развитие цифрового аудио. Например, в проигрывателях компакт-дисков они упрощают процесс цифроаналогового преобразования с поддержанием качества воспроизводимого звука. На передовом крае области цифровых аудиосистем была проделана значительная работа, чтобы объединить методы дельта-модуляции с обработкой при нескольких скоростях, что позволит получать высококачественные цифровые данные из аналоговых аудиосигналов.

В число других областей, в которых используются методы обработки при нескольких скоростях, входит сбор высококачественных цифровых данных, спектральный анализ с высоким разрешением, а также разработка и реализация узкополосной цифровой фильтрации.

Некоторые из подобных сфер применения обработки при нескольких скоростях описаны в следующих разделах.

9.7.1. Высококачественное аналого-цифровое преобразование в цифровом аудио

В сфере цифрового аудио постоянно требуется повышать качество, разрешение и скорость АЦП. Это привело к разработке однобитовых АЦП с использованием методов дельта-сигма-модуляции. В результате появилась возможность полностью отказаться от процесса преобразования на входе цифровых аудиосистем большинства аналоговых схем, в том числе аналоговых фильтров защиты от наложения спектров и схем выборки-хранения.

Упрощенная блок-схема быстрого однобитового процесса аналого-цифрового преобразования показана на рис. 9.25 [1, 12, 14]. Аналоговый аудиосигнал сперва преобразуется в однобитовый поток с помощью дельта-сигма-модуляции с частотой 3,072 МГц. Затем однобитовый поток выбирается с понижением частоты до 48 кГц с помощью многокаскадного дециматора и формируются 16-битовые слова PCM. В настоящее время можно легко приобрести АЦП, в которых используются методы с обработкой при нескольких скоростях, например, 16- и 18-битовые стерео АЦП от Crystal Semiconductor (CS5326, CS5327, CS5328, CS5329) и Motorola Semiconductor (DSP56ADC16).



Рис. 9.25. Упрощенная блок-схема однобитового АЦП

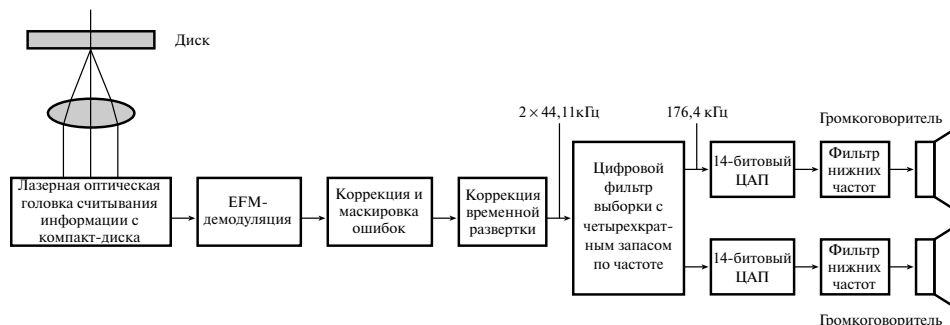


Рис. 9.26. Воспроизведение аудиосигнала в проигрывателе компакт-дисков

9.7.2. Эффективное аналого-цифровое преобразование в высококачественных системах воспроизведения компакт-дисков

Одним из первых серьезных приложений методов с обработкой при нескольких скоростях стало воспроизведение звука и музыки в проигрывателях компакт-дисков.

Рассмотрим рис. 9.26, на котором изображена схема восстановления аналогового аудиосигнала из цифрового сигнала, считываемого с компакт-диска. После декодирования и исправления ошибок цифровой сигнал имеет форму 16-битовых слов, представляющих акустическую информацию с частотой дискретизации 44,1 кГц.

Если эти цифровые коды непосредственно преобразовать в аналоговые, будут получены зеркальные полосы частот, центрированные на частотах, кратных частоте дискретизации 44,1 кГц. Хотя зеркальные частоты будут неслышны (они расположены вне полосы 0–20 кГц), при передаче их на усилитель и громкоговоритель проигрывателя может произойти переполнение или же они могут стать причиной перекрестных помех. Следовательно, частотные составляющие вне полосы частот исходного сигнала следует подавить не менее чем на 50 дБ. Если для такого подавления использовать аналоговые фильтры, придется следовать очень строгим спецификациям, кроме того нужна будет точная настройка, обеспечивающая согласованность двух стереоканалов.

Чтобы избежать указанных проблем, в проигрывателях компакт-дисков используется фильтрация с обработкой при нескольких скоростях. Конкретнее, частота дискретизации данных с помощью интерполяции увеличивается в 4 раза до 176 кГц ($4 \times 44,1 \text{ кГц} = 176,4 \text{ кГц}$) перед подачей данных на ЦАП. Во временной области это выглядит как сигнал с более мелким масштабом. В частотной области зеркальные частоты теперь перенесены в диапазон более высоких частот, так что отфильтровать их стало легче. Следовательно, после цифроаналогового преобразования необходим только относительно простой фильтр нижних частот. При практической реализации цифровой фильтр вводит изменение характеристики по закону $\sin(x)/x$ (см. главу 2) для компенсации влияния схемы хранения, следующей за ЦАП. Коррекция характеристики по закону $\sin(x)/x$ имеет положительный эффект подавления сигналов с обеих сторон частоты 174 кГц более чем на 18 дБ, что еще больше упрощает требования к филь-

трации аналоговых зеркальных компонентов. Для дополнительного подавления после интерполяции вводится простой фильтр Бесселя третьего порядка, имеющий частоту среза по уровню 3 дБ, равную 30 кГц, и линейную (в разумном приближении) фазовую характеристику в полосе пропускания.

Перевыборка данных с повышенной частотой также оказывает свое положительное влияние. Она снижает уровень шума, поскольку шум квантования теперь распределен по более широкой полосе, что позволяет использовать ЦАП с меньшим числом битов и тем же отношением сигнал-шум, что и в 16-битовом ЦАП. Следовательно, на рис. 9.26 интерполированные данные в форме 16-битовых слов после перевыборки и формирования спектра шума перед подачей на 14-битовый ЦАП округляются до 14 бит.

На рынке существуют и другие цифроаналоговые преобразователи, в которых использована представленная выше концепция выборки с повышенной частотой. Примеры: цифроаналоговые преобразователи потока битов от Philips Components (SAA7322, SAA7323, SAA7350).

Пример 9.5

Для ослабления требований к аналоговому фильтру подавления зеркальных частот в цифровой аудиосистеме используется дискретизация с повышенной частотой. Ниже приведена общая спецификация фильтра, используемого в системе:

полоса частот исходного сигнала	0–20 кГц,
входная частота дискретизации F_s	44,1 кГц,
выходная частота дискретизации	176,4 кГц,
затухание в полосе подавления	50 дБ,
неравномерность в полосе пропускания	0,5 дБ,
ширина полосы перехода	2 кГц,
граничная частота полосы подавления	22,05 кГц.

Разработайте подходящий интерполятор.

Решение

Используя программу разработки системы с несколькими скоростями обработки (на компакт-диске к книге [Ifeachor, 2001], подробности см. в предисловии), получаем приведенные ниже шаги интерполяции и характеристики фильтров для возможных интерполяторов с целыми шагами.

Число каскадов	Шаг интерполяции, L_i	Длина фильтра, N_i	Нормированная ширина полосы перехода, Δf_i	Неравномерность в полосе пропускания, δ_p	Неравномерность в полосе подавления, δ_s
1	4	146	0,04535	0,05925	0,00316
2	2	6	0,26162	0,0296	0,00316
	2	83	0,27324	0,0296	0,00316

Вид системы с двухкаскадным интерполятором представлен на рис. 9.27.

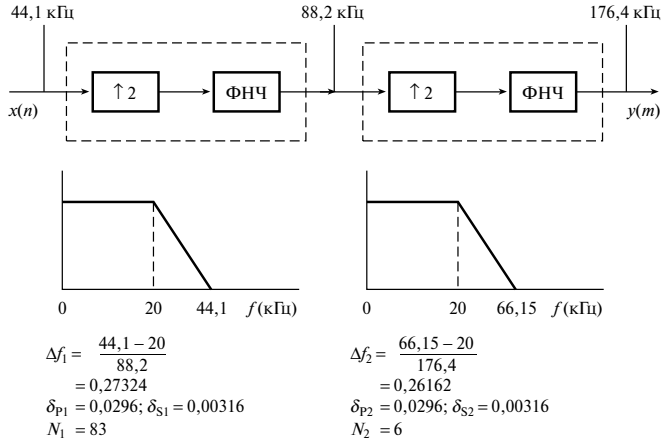


Рис. 9.27. Двухкаскадный интерполятор из примера 9.5



Рис. 9.28. Простая система сбора данных с несколькими скоростями обработки

9.7.3. Сбор высококачественных данных

При сборе почти всех реальных данных требование незначительного наложения часто предписывает использовать относительно сложные аналоговые фильтры защиты от наложения спектров. В многоканальной системе каждый аналоговый канал должен соединяться с отдельным фильтром защиты от наложения спектров, поскольку уплотнить подобные фильтры довольно сложно. В системе с большим числом аналоговых каналов (например, в биомедицине может требоваться до 32 каналов) использование аналоговых фильтров защиты от наложения спектров может быть очень дорогим. Применение более простых цифровых фильтров позволяет существенно снизить стоимость системы. Более того, можно не рассматривать проблему согласования по фазе строго заданных аналоговых фильтров. Также снимается проблема поддержки нескольких частот дискретизации (а каждая частота дискретизации требует иной частоты среза) при использовании аналоговых фильтров защиты от наложения спектров.

На рис. 9.28 приведена блок-схема системы сбора данных с несколькими скоростями обработки (согласно [13]). Желаемый уровень наложения достигается за счет RC-фильтров переднего края, выбирающих входной сигнал с повышенной частотой, а затем понижающих частоту до нужного уровня с помощью обработки при нескольких скоростях. Основная цена такого решения — АЦП должен работать быстрее.

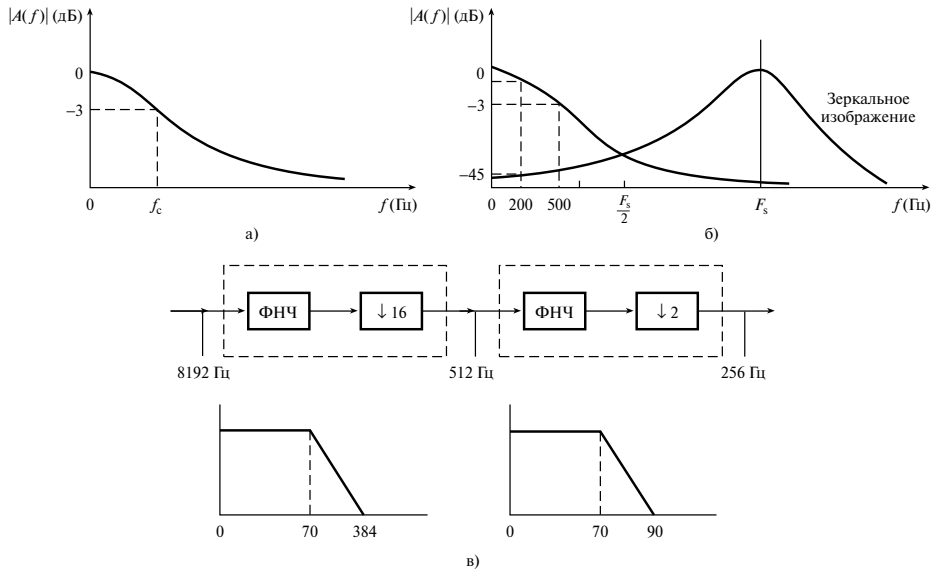


Рис. 9.29. Характеристика фильтра Баттерворта второго порядка (панель а). Спектр широкополосного сигнала после ограничения полосы и дискретизации (панель б). Двухкаскадный дециматор из примера 9.6 (панель в)

Чтобы закрепить материал, представленный в данном разделе, и лучше понять выгоды цифровой реализации фильтра защиты от наложения спектров, рассмотрим на примере реальное приложение описанных концепций.

Пример 9.6

Требуется универсальная многоканальная (до 32 каналов) система сбора физиологических данных. Каждый аналоговый канал пользователь может настраивать отдельно, варьируя частоту среза от 0,5 до 200 Гц, а частоту дискретизации — от 1 до 1000 Гц. Общие требования ко всем каналам:

- неравномерность в полосе пропускания $\leq 0,5$ дБ,
- отношение полезного сигнала к сигналу, который возник из-за наложения ≥ 45 дБ (в полосе пропускания),
- граничная частота полосы пропускания $0,5 \leq f_p \leq 200$ Гц,
- граничная частота полосы подавления $\leq 3f_p$.

Амплитудное и фазовое искажение должно быть минимальным. Для снижения числа и стоимости компонентов, а также размера печатной платы на входе системы должны использоваться только простые аналоговые фильтры.

Решение

Для того чтобы выполнить фильтрацию и защитить от наложения спектры с использованием только аналоговых фильтров, необходимы фильтры очень больших порядков. Альтернативный подход — оснастить все каналы простыми идентичными фильтрами, которые будут выполнять выборку с общей частотой, а затем провести децимацию до желаемой частоты (частот), причем в каждом каскаде разрабатываемого фильтра следует удовлетворять заданным спецификациям.

Во всех каналах можно использовать простой однополосный RC-фильтр, но, чтобы удовлетворять при этом спецификациям, потребуется очень большая частота дискретизации. Используем при разработке фильтр Баттерворта второго порядка, поскольку он хорошо себя зарекомендовал в реальной биомедицинской установке.

Амплитудная характеристика фильтра Баттерворта второго порядка записывается следующим образом:

$$A(f) = \frac{1}{[1 + (f/f_c)^4]^{1/2}}.$$

Сказанное иллюстрируется на рис. 9.29, *а*. Очевидно, что в полосе частот от 0 до f_c существует значительная ошибка по амплитуде. Чтобы довести это значение до определенного спецификациями уровня, наивысшая частота, представляющая практический интерес (в нашем случае — 200 Гц), должна быть гораздо ниже f_c . Чтобы вычислить значение f_c , можно использовать уравнение

$$20 \lg[1 + (200/f_c)^4]^{1/2} \leq 0,5 \text{ дБ}.$$

Решая уравнение относительно f_c , находим, что $f_c \geq 338,39$ Гц. Для удобства, а также, чтобы учесть дополнительные ошибки, которые могут вводиться на последующих этапах обработки, используем значение $f_c = 500$ Гц. При таком f_c характеристика на частоте 200 Гц снижается на 0,11 дБ.

Определим теперь общую частоту дискретизации для всех каналов. Ограничив полосу каждого канала с помощью фильтра Баттерворта второго порядка и дискретизации, получаем сигнал, спектр которого изображен на рис. 9.29, *б* (при широкополосном входном сигнале). Обращаясь к рисунку, видим, что требуется частота дискретизации F_s такая, что на частоте f_p (где $f_p = 200$ Гц) уровень накладывающегося сигнала по крайней мере на 45 дБ ниже уровня сигнала:

$$20 \lg \{1 + [(F_s - 200/500)^4]\}^{1/2} \geq 45 \text{ дБ}.$$

Из данного уравнения находим $F_s \approx 6,67$ кГц. Для удобства в процессе децимации выберем $F_s = 8192$ Гц. Общая спецификация подходящего универсального дециматора должна выглядеть приблизительно так:

входная частота дискретизации	8,192 кГц,
выходная частота дискретизации	$1 \text{ Гц} < F_s < 1000 \text{ Гц}$,
затухание в полосе подавления	50 дБ,
неравномерность в полосе пропускания	0,01 дБ,
границная частота полосы пропускания	$0,5 \text{ Гц} < f_p < 200 \text{ Гц}$,
шаг децимации	$8,192 < M < 8192$,
границная частота полосы подавления	$< 2f_p$.

Для удобства наложим условие на частоту дискретизации F_s , которую может задавать пользователь. Это позволит проводить децимацию только с целыми шагами. Если обрабатывающей способности достаточно для реализации децимации с нецелым шагом, данное ограничение не нужно. Итак, предлагаются следующие возможные частоты дискретизации и соответствующие им шаги децимации:

M	8	16	32	64	128	256	512	1024	2048	4096	8192
F_s (Гц)	1024	512	256	128	64	32	16	8	4	2	1

При решении задачи можно считать, что система состоит из 11 многокаскадных дециматоров, а при заданной спецификации выбирается один из них.

В качестве иллюстрации рассмотрим систему сбора сигналов ЭЭГ. Каждый канал может иметь следующую пользовательскую спецификацию:

частота дискретизации	256 Гц,
затухание в полосе подавления	45 дБ,
неравномерность в полосе пропускания	0,5 дБ,
полоса пропускания	0–70 Гц.

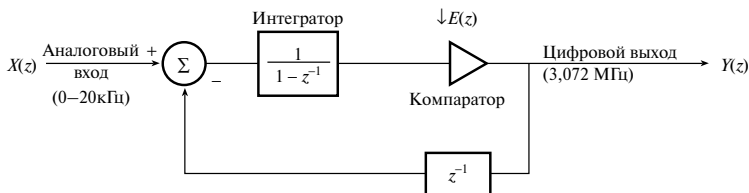
На основе данной спецификации находим спецификацию преобразователя частоты, который согласуется со спецификацией описанного выше универсального дециматора:

входная частота дискретизации	8,192 кГц,
выходная частота дискретизации	256 Гц,
шаг децимации	32,
затухание в полосе подавления	50 дБ,
неравномерность в полосе пропускания	0,01 дБ,
полоса пропускания	0–70 Гц,
полоса подавления	90–128 Гц.

Используя программу разработки (компакт-диск к книге [Ifeachor, 2001]), получаем, что эффективным (с точки зрения вычислительной сложности и сложности системы) дециматором является двухкаскадная система, изображенная на рис. 9.29, в.

Пример 9.7

1. В системе цифровой обработки сигналов для преобразования входного аналогового аудиосигнала с диапазоном 0–20 кГц в цифровой поток битов с частотой 3,072 МГц используется метод выборки с повышенной частотой и дельта-сигма-модулятор первого порядка. Представление дельта-сигма-модулятора на z -плоскости показано на



Примечание. Выходное преобразование, $Y(z)$, дельта-сигма-модулятора первого порядка записывается следующим образом: $Y(z) = X(z) + E(z)(1 - z^{-1})$. Переменные имеют обычное значение.

Рис. 9.30

рис. 9.30. Определите общее улучшение отношения сигнал-шум квантования, которого можно добиться при выборке с повышенной частотой и формировании спектра шумов, оценив таким образом эффективное разрешение конвертера в битах.

- Разработайте на уровне блок-схемы двухкаскадный дециматор, преобразующий выход дельта-сигма-модулятора, изображенного на рис. 9.30, из формы однобитового потока с частотой 3,072 МГц в многобитовый поток с частотой 48 кГц. Неравномерности в полосе пропускания и полосе подавления равны соответственно 0,001 и 0,0001.

Ответ должен включать следующие элементы:

- значения общих шагов децимации и интерполяции плюс (кратко) ход их вычисления;
- значения пар шагов децимации и интерполяции для двухкаскадных преобразователей частоты дискретизации плюс подробный анализ вычислительной сложности и требований к памяти, оправдывающий сделанный выбор;
- спецификацию граничных частот, длин, неравномерности в полосе пропускания и полосе подавления фильтров защиты от наложения спектров и подавления зеркальных частот.

Примечание: можно считать, что используются КИХ-фильтры, реализованные в прямой форме, длина которых определяется следующим образом:

$$\text{длина фильтра, } N = \frac{-10 \lg(\delta_p, \delta_s) - 13}{14,6 \Delta f} + 1,$$

где

Δf – нормированная ширина полосы перехода.

Решение

- Передаточная функция шума равна

$$N(x) = 1 - z^{-1}.$$

Амплитудная характеристика равна

$$\begin{aligned} |N(z)|_{z=e^{i\omega T}} &= |(1 - e^{-i\omega T})| = \\ &= [(1 - \cos \omega T)^2 + \sin^2 \omega T]^{\frac{1}{2}}. \end{aligned}$$

При $f = 20$ кГц $F_s = 3,072$ МГц, $\omega T = 2,3438^\circ$ и $|N(e^{i\omega T})| = 0,0409$. Это эквивалентно уменьшению уровня шума квантования на 27,76 дБ.

2. Общий шаг децимации равен 64 (находится как отношение входной частоты дискретизации к выходной). При таком общем шаге и двухкаскадной реализации есть три возможные комбинации шагов: 8×8 , 16×4 и 32×2 . Мерой вычислительной сложности являются общая требуемая память (ОТП) и число умножений в секунду (УВС). Для дециматора 8×8 частоты дискретизации на выходе поддециматоров равны 384 и 48 кГц. Граничные частоты первого фильтра защиты от наложения спектров: 0, 20 и 360 кГц, откуда находим нормированную ширину перехода — 0,1106. Неравномерность в полое пропускания и заграждения равна $0,001/2=0,0005$ и 0,0001 соответственно. Используя эти значения, получаем $N_1 = 38$. Для второго каскада граничные частоты равны 0, 20 и 24 кГц, а нормированная ширина полосы перехода равна 0,0104 и $N_2 = 396$.

Подобным образом для структуры 16×4 получаем частоты 192 и 48 кГц. Первый фильтр защиты от наложения спектров характеризуется граничными частотами 0, 20 и 168 кГц, шириной перехода 0,048 и $N_2 = 198$. Частоты дискретизации структуры 32×2 равны 96 и 48 кГц. Граничные частоты соответствующего фильтра защиты от наложения спектров — 0, 20 и 72 кГц. Ширина перехода 0,0169, $N_1 = 244$; второй фильтр характеризуется граничными частотами 0, 20 и 24 кГц, шириной перехода 0,04166 и $N_2 = 100$.

В следующей таблице представлена вычислительная сложность названных структур:

$M_1 \times M_2$	Память	УВС
8×8	434	$33,6 \times 10^6$
16×4	284	$26,01 \times 10^6$
32×2	384	$28,82 \times 10^6$

Из приведенного анализа видно, что наиболее эффективной является структура 16×4 . Блок-схема этой структуры представлена на рис. 9.31.

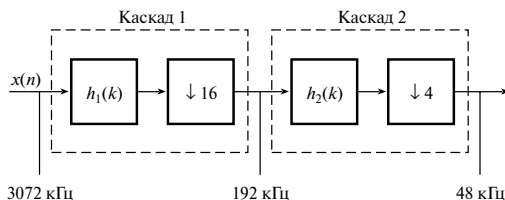


Рис. 9.31

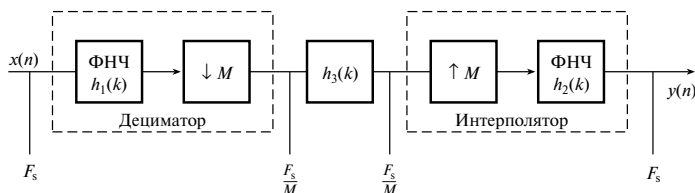


Рис. 9.32. Узкополосная фильтрация с обработкой при нескольких скоростях

9.7.4. Узкополосная цифровая фильтрация с обработкой при нескольких скоростях

Узкополосные цифровые фильтры характеризуются полосами пропускания, очень малыми по сравнению с частотой дискретизации, и резкими переходами между полосами пропускания и подавления. В результате узкополосные КИХ-фильтры описываются очень большим числом коэффициентов, а так как подобные структуры весьма чувствительны к эффектам конечной разрядности (например, шуму округления и ошибкам квантования коэффициентов), возникают проблемы при их разработке и реализации. Более того, требуется больше памяти и вычислительных ресурсов. Названные проблемы можно решить с помощью обработки при нескольких скоростях, в результате будут получены КИХ-фильтры, вычислительная эффективность которых сравнима с эффективностью эллиптических БИХ-фильтров.

На рис. 9.32 приведена простая схема фильтрации с обработкой при нескольких скоростях. Частота дискретизации входной последовательности вначале с помощью децимации предельно уменьшается, затем на низкой частоте выполняется необходимая фильтрация, наконец с помощью интерполяции фильтрованные данные приводятся к исходной частоте дискретизации. Использование одинаковых шагов при децимации и интерполяции гарантирует, что входной сигнал $x(n)$ и выходной сигнал $y(n)$ будут иметь равные частоты дискретизации.

9.7.4.1. Узкополосные полосовые фильтры и фильтры нижних частот

Если требуется реализовать фильтр нижних частот с использованием обработки при нескольких скоростях, то в схеме на рис. 9.32 фильтры $h_1(k)$ и $h_2(k)$ можно сделать фильтрами нижних частот, при этом фильтр $h_3(k)$ становится лишним. Если принять такой подход, то дальнейшая разработка должна быть нацелена на то, чтобы общие характеристики входа-выхода структуры на рис. 9.32 были эквивалентны характеристикам нужного обычного фильтра нижних частот. Практически же полученные характеристики будут отличаться от характеристик обычного фильтра нижних частот вследствие эффектов наложения и появления зеркальных частот. Чтобы общая система удовлетворяла искомым спецификациям, фильтры $h_1(k)$ и $h_2(k)$ берутся идентичными, неравномерность в полосе пропускания каждого равна $\delta_p/2$, неравномерность в полосе подавления — δ_s , причем δ_p и δ_s — неравномерность в полосе пропускания и полосе подавления эквивалентного фильтра нижних частот соответственно.

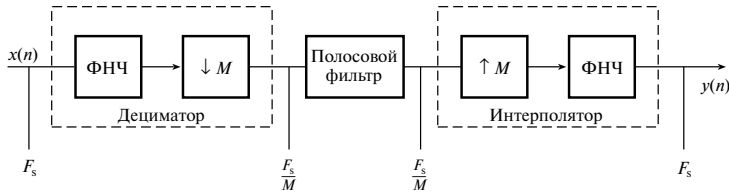


Рис. 9.33. Узкополосная полосовая фильтрация с обработкой при нескольких скоростях

Создать полосовой фильтр с обработкой при нескольких скоростях немного сложнее, особенно если требуется разработать так называемый *полосовой фильтр с целой полосой* (или *целополосной фильтр*) (integer-band bandpass filter), где граничные частоты кратны наименьшей частоте дискретизации системы, т.е. $F_s/2M$. В таких случаях шаг децимации/интерполяции M и граничные частоты фильтра удовлетворяют следующим условиям [8]:

$$M = F_s/2(f_{su} - f_{sl}), \quad (9.17, a)$$

$$f_{sl} = kF_s/M, \quad k - \text{целое}, \quad 0 < k < M - 1, \quad (9.17, б)$$

$$f_{su} = (k + 1)f_s/M, \quad (9.17, в)$$

где f_{sl} и f_{su} — нижняя и верхняя граничные частоты полосы подавления соответственно. В уравнении (9.17, а) выражен максимально возможный шаг децимации, а в уравнениях (9.17, б и в) задаются верхний и нижний края полосы подавления и число полосы k .

Простой, но менее эффективной альтернативой полосовой фильтрации с обработкой при нескольких скоростях является схема, в которой данные вначале проходят децимацию с максимально большим шагом с использованием подходящих фильтров нижних частот (как и выше), затем подаются на полосовой фильтр для понижения частоты сигнала, а после этого восстанавливаются с помощью интерполяции до нужной частоты. Данный подход иллюстрируется на рис. 9.33. Очевидно, что требуется следить, чтобы в процессе децимации и интерполяции желаемая полоса пропускания не подвергалась эффектам наложения и появления зеркальных частот.

9.7.4.2. Узкополосные фильтры верхних частот и режекторные фильтры

Узкополосные фильтры верхних частот и режекторные фильтры можно реализовать как дуальные к фильтрам нижних частот и полосовым соответственно:

$$H_{hp}(w) = 1 - H_{\text{ФНЧ}}(w), \quad (9.18, a)$$

$$H_{bs}(w) = 1 - H_{\text{ПФ}}(w). \quad (9.18, б)$$

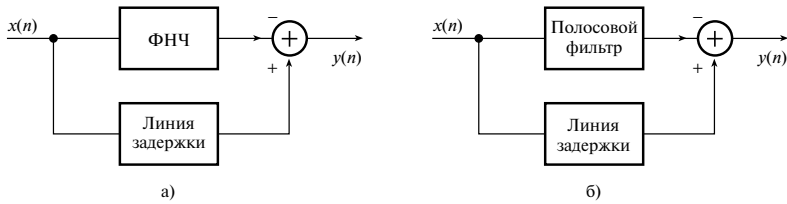


Рис. 9.34. Реализация с обработкой при нескольких скоростях фильтра нижних частот и режекторного фильтра как дуальных к фильтру нижних частот и полосовому фильтру: а) фильтр верхних частот; б) режекторный фильтр

Структура реализации ВЧ- и режекторных фильтров изображена на рис. 9.34. Для фильтра верхних частот, например, сигнал $x(n)$ вначале проходит через фильтр нижних частот. Затем отфильтрованный сигнал вычитается из нефильтованного сигнала, в результате получается нужный сигнал. Отметим, что перед вычитанием сигнал $x(n)$ нужно задержать на время, равное задержке фильтра нижних частот. Очевидно, что для этого задержка при прохождении фильтра нижних частот должна выражаться целым числом выборок. Еще одно замечание: чтобы получить необходимый фильтр верхних частот, при разработке фильтра нижних частот следует использовать скорректированные спецификации полосы пропускания и полосы подавления.

Пример 9.8

В связи с исследовательским проектом в сфере наблюдения за плодом возникла необходимость оценить влияние системы измерения (электрокардиограмма — ЭКГ) на электрическую активность сердца плода [15]. Для этого требуется количественно оценить некоторые элементы ЭКГ, включая основной частотный спектр сигнала. Поскольку сигнал имеет составляющие в окрестности промышленной частоты (50 Гц), нужно использовать фильтр с очень узкой полосой, который подчиняется следующим требованиям:

полоса пропускания	49–51 Гц,
граничные частоты полосы подавления	47 и 53 Гц,
затухание в полосе подавления	30 дБ ($\delta_p = 0,03162$),
неравномерность в полосе пропускания	0,1 дБ ($\delta_s = 0,011579$),
частота дискретизации	500 Гц.

Решение

Если использовать прямую реализацию требуемой системы, то из уравнения (9.3) следует, что нам понадобится 4018 коэффициентов, что слишком много.

При использовании обработки при нескольких скоростях есть несколько альтернатив. Одна из них — это провести децимацию данных до максимально возможной низкой частоты дискретизации (согласующейся с приведенной выше спецификацией) (см. также задачу 9.6). В данном случае самая низкая частота — 125 Гц, поскольку это еще позволяет использовать полосу от 0 до 62,5 Гц. Приведем общую спецификацию дециматора:

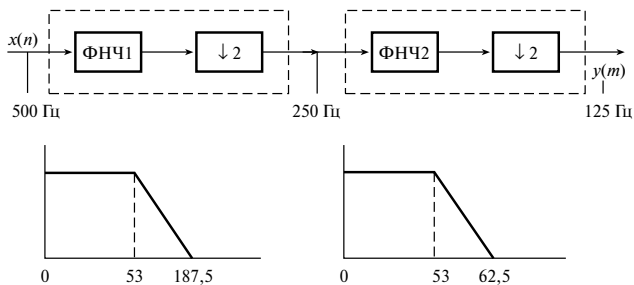


Рис. 9.35. Дециматор для уменьшения частоты ЭКГ-данных

неравномерность в полосе пропускания	0,05 дБ ($\delta_p = 0,0057895$),
затухание в полосе подавления	30 дБ ($\delta_s = 0,03162$),
полоса пропускания	0–53 Гц,
входная частота дискретизации	500 Гц,
выходная частота дискретизации	125 Гц.

Шаг децимации равен 4. С помощью программы разработки, представленной на компакт-диске к книге [Ifeachor, 2001] (подробности см. в предисловии), был построен двухкаскадный дециматор (см. рис. 9.35). Для получения коэффициентов фильтров использовался оптимальный метод (см. главу 7). ЭКГ-данные проходят децимацию с помощью фильтров и программы, представленной в разделе 9.4. Примеры данных до и после децимации изображены на рис. 9.36.

Отметим, что существует реальный промышленный фильтр, удовлетворяющий спецификациям, представленным в задаче (но при иной, пониженной частоте дискретизации). В нашем случае число коэффициентов фильтра равно 113,4 ($\delta_p = 0,0057895$, $\delta_s = 0,03162$). Для восстановления данных после фильтрации до исходной частоты дискретизации можно использовать интерполяцию.

Пример 9.9

Разработайте подходящий фильтр нижних частот с обработкой при нескольких скоростях, предназначенный для удаления смещения базовой линии в ЭКГ плода. Фильтр должен удовлетворять следующим спецификациям:

полоса пропускания	0–0,4 Гц,
полоса подавления	0,5–250 Гц,
неравномерность в полосе пропускания	0,01,
неравномерность в полосе подавления	0,001,
частота дискретизации	500 Гц.

Решение

Проведем вначале децимацию до частоты 1 Гц, а затем интерполируем сигнал до требуемых 500 Гц. При таком подходе общая спецификация фильтра в дециматоре запишется так:

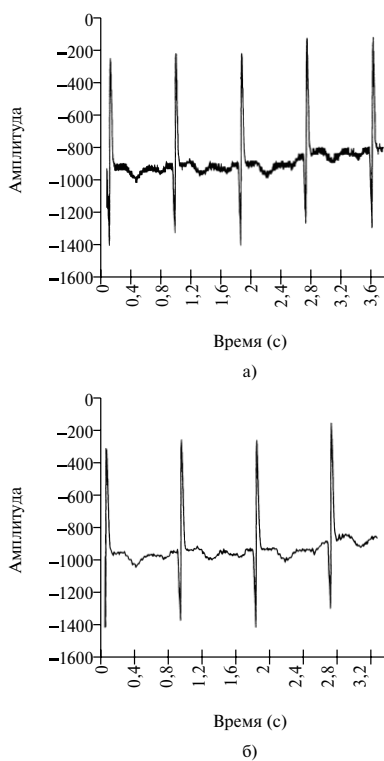


Рис. 9.36. Необработанные ЭКГ-данные (панель *а*); ЭКГ-данные после децимации (выровнены с учетом задержки) (панель *б*)

неравномерность в полосе пропускания	0,01,
неравномерность в полосе подавления	0,001,
граничная частота полосы подавления	0,5 Гц,
полоса пропускания	0–0,4 Гц,
частота дискретизации	500 Гц,
шаг децимации	500.

С помощью программы разработки (книга [Ifeachor, 2001], подробности см. в предисловии) были рассчитаны различные практические дециматоры, включающие до четырех каскадов. Характеристики наиболее интересных из них сведены в табл. 9.6. С учетом вычислительной сложности в качестве лучшего решения был выбран трехкаскадный дециматор, изображенный на рис. 9.37. Как и ранее, при вычислении коэффициентов фильтров использовалась программа оптимальной разработки (см. главу 7).

Таблица 9.6. Резюме по эффективным дециматорам

Чис- ло кас- кадов	УВС	ОТП	M_i	Длина филь- тра N_i	Граница полосы пропускания f_p , Гц	Граница полосы подавления, Гц	Нормиро- ванная ширина полосы перехода	Неравно- мерность в полосе пропускания	Неравно- мерность в полосе подавления
1			500	12 707	0,4	0,5	0,002 2	0,01	0,001
2	1807	430	50	153	0,4	9,5	0,018 20	0,005	0,001
			10	277	0,4	0,5	0,01	0,005	0,001
3	1705	189	25	77	0,4	19,5	0,038 20	0,0033	0,001
			10	53	0,4	1,5	0,055	0,0033	0,001
			2	59	0,4	0,5	0,05	0,0033	0,001
4	1444	172	2	2	0,4	249,5	0,498 2	0,0025	0,001
			25	83	0,4	9,5	0,036 40	0,0025	0,001
			5	27	0,4	1,5	0,110 00	0,0025	0,001
			2	60	0,4	0,5	0,050 0	0,0025	0,001
4	1724	169	25	79	0,4	19,5	0,038 20	0,0025	0,001
			2	3	0,4	9,5	0,455 00	0,0025	0,001
			5	27	0,4	1,5	0,110 0	0,0025	0,001
			2	60	0,4	0,5	0,050 0	0,0025	0,001

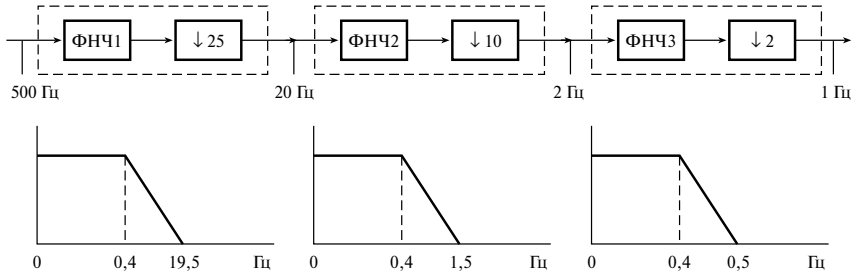


Рис. 9.37. Трехкаскадный дециматор из примера 9.9

9.7.5. Узкополосный спектральный анализ с высоким разрешением

Как говорится в главе 11, важной сферой применения БПФ является оценка спектра сигналов. БПФ выдает спектральные составляющие сигнала, расположенные через равные промежутки в диапазоне от 0 до половины частоты дискретизации. Во многих приложениях, таких как гидролокация, сейсмология, радиолокация, биомедицина и анализ механических колебаний (виброанализ), нужный сигнал может занимать только узкую полосу в спектре полученных данных. В таких случаях прямое использование БПФ потребует значительных и ненужных вычислений. Для выделения и трансляции нужной полосы частот на более низкую частоту перед применением БПФ можно использовать обработку при нескольких скоростях, что позволит значительно снизить объем вычислений, а также использовать компромиссы между разрешением и вычислительной сложностью.

Применение БПФ к данным, выбранным с пониженной частотой, дает эквивалентное разрешение при значительно сокращенных вычислениях или большее разрешение

при примерно таком же объеме вычислений, если сравнивать с прямым применением БПФ к исходной последовательности. Итак, выборка с пониженной частотой позволяет эффективно получать спектр узкополосного сигнала в увеличенном масштабе.

Узкополосный спектральный анализ с использованием обработки при нескольких скоростях — это, по сути, расширение описанной ранее узкополосной полосовой фильтрации, так что он имеет те же ограничения. Сигнал вначале проходит полосовую фильтрацию для выделения нужной полосы частот. Затем частота дискретизации отфильтрованного сигнала уменьшается с помощью децимации до F_s/M , где F_s — частота дискретизации сигнала $x(n)$. После этого с помощью БПФ вычисляется спектр значительно сокращенной последовательности $y(n)$. Для компенсации ошибок, внесенных в спектр вследствие неравномерности полосы сигнала $h(n)$, используется поправочный коэффициент. Если нужная полоса частот не удовлетворяет условиям, то можно использовать несколько более широкую полосу частот, содержащую необходимую полосу. Альтернативный подход — использовать метод, описанный в [11], который включает компьютерный поиск допустимых шагов децимации.

9.8. Резюме

Цифровые системы, поддерживающие несколько частот дискретизации, называются системами с обработкой при нескольких скоростях. Двумя ключевыми элементами таких систем являются дециматор и интерполятор. Дециматор позволяет эффективно уменьшать частоту сигнала в целое число M раз или в рациональное число раз L/M ($L < M$). Интерполятор, наоборот, позволяет увеличивать частоту дискретизации в целое число L раз или в рациональное число раз L/M ($L > M$).

На практике изменения частоты дискретизации реализуются в двух или более каскадах для достижения максимальной вычислительной эффективности или минимальных требований к памяти. Цифровые фильтры, используемые в системах с обработкой при нескольких скоростях, имеют не такие строгие спецификации, что выражается в меньшем числе коэффициентов, а следовательно, меньшей чувствительности к эффектам конечной разрядности. Практический метод разработки простых преобразователей частоты подробно описан в данной главе.

Основное достоинство систем с обработкой при нескольких скоростях — это возможность использования преимуществ цифровой обработки сигналов, в частности, использования ЦОС для ограничения полосы сигнала почти до частоты Найквиста со значительным затуханием и без нарушения условий теоремы о частотном представлении. Преимущества такого подхода проявились во многих областях, включая воспроизведение компакт-дисков, цифровую фильтрацию, сбор данных и сбор данных с высоким разрешением. Многие из таких систем подробно описаны в главе, особое внимание уделено элементам с обработкой при нескольких скоростях.

Компакт-диск к книге [Ifeachor, 2001] (подробности см. в предисловии) содержит набор программ на языке C, позволяющий создавать и воплощать в программной форме системы с обработкой при нескольких скоростях. Использование MATLAB для систем цифровой обработки при нескольких скоростях описано в приложении 9Б.

Задачи

9.1. Однокаскадный дециматор характеризуется следующими параметрами:

шаг децимации 3

коэффициенты фильтра защиты от наложения спектров:

$$h(0) = -0,06 = h(4),$$

$$h(1) = 0,30 = h(3),$$

$$h(2) = 0,62.$$

Существует ряд данных $x(n)$: $\{6, -2, -2, 8, 6, 4, -2\}$. Определите выход фильтра $w(n)$ и выход дециматора $y(m)$.

9.2. 1. На рис. 9.38 приведена блок-схема трехкаскадного дециматора, предназначенного для уменьшения частоты дискретизации с до 1 кГц. Предполагая, что шаги децимации равны 8, 4 и 2, укажите частоту дискретизации на выходе каждого каскада.

2. Пусть дециматор из п. 1 удовлетворяет следующим спецификациям:

входная частота дискретизации, F_s	96 кГц,
шаг децимации, M	96,
неравномерность в полосе пропускания	0,01 дБ,
неравномерность в полосе подавления	60 дБ,
существенная полоса частот	0–450 Гц.

Определите граничные частоты фильтров каждого каскада.

3. Пусть входная и выходная частоты дискретизации дециматора равны 96 и 1 кГц соответственно.

А. Запишите общий шаг децимации.

Б. Запишите все возможные наборы целых шагов децимации (в убывающем порядке), предполагая двухкаскадную реализацию.

В. Повторите п. Б., предполагая трехкаскадную реализацию.

Г. Повторите п. Б., предполагая четырехкаскадную реализацию.

4. Для дециматора из п. 1 вычислите общее число операций умножения в секунду (УВС) и общую требуемую память (ОТП).

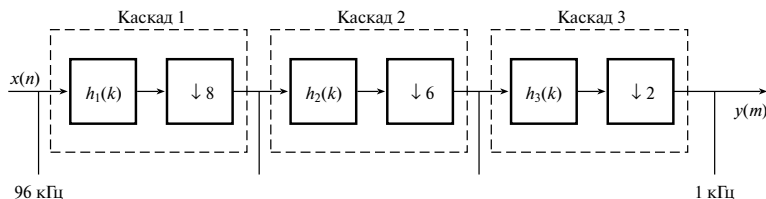


Рис. 9.38

- 9.3. 1. Разработайте на уровне блок-схемы двухкаскадный дециматор, перевыбирающий аудиосигнал с понижением частоты в 32 раза и удовлетворяющий приведенным ниже спецификациям. В ответе следует указать подходящую пару шагов децимации плюс подробный (в разумных пределах) анализ вычислительной сложности и требований к памяти, аргументирующий сделанный выбор. Для каждого фильтра задайте частоты дискретизации на входе и выходе каждого каскада, а также следующие параметры:

граничные частоты,
нормированная ширина полосы перехода,
неравномерность в полосе пропускания и полосе подавления,
длина фильтра.

При решении предполагается использовать реализованные в прямой форме КИХ-фильтры с длиной

$$\text{длина фильтра, } N = \frac{-10 \lg(\delta_p \delta_s) - 13}{14,6 \Delta f} + 1,$$

где

Δf – нормированная ширина полосы перехода.

2. С помощью схем покажите, что дециматор с приведенными ниже спецификациями защищает от наложения существенную полосу частот (0–3,4 кГц).
- | | |
|--|----------|
| Входная частота дискретизации, F_s | 256 кГц. |
| Наивысшая существенная частота данных | 3,4 кГц. |
| Неравномерность в полосе пропускания, δ_p | 0,05. |
| Неравномерность в полосе подавления, δ_s | 0,01. |
- 9.4. Разработайте дециматор для высококачественной системы сбора данных со следующими общими спецификациями фильтра:
- | | |
|--------------------------------------|------------|
| аудиополоса | 0–20 кГц, |
| входная частота дискретизации | 3,072 МГц, |
| выходная частота дискретизации | 48 кГц, |
| неравномерность в полосе пропускания | <0,001 дБ, |
| затухание в полосе подавления | >86 дБ. |
- 9.5. Требуется вычислить спектр узкополосного сигнала, вложенного в широкополосный. Важна полоса частот 49–51 Гц, но составной сигнал занимает полосу 0–100 Гц. При выборке составного сигнала с частотой 1 кГц получена N -точечная последовательность $x(n)$.
1. Проиллюстрируйте получение спектра желаемого сигнала с использованием обработки при нескольких скоростях.
 2. Оцените вычислительные преимущества обработки при нескольких скоростях перед обычным БПФ. Сравните разрешение спектров, полученных обоими методами.
- 9.6. Для извлечения и оценки компонентов сигнала, внесенных питающей сетью, требуется высококачественный эффективный узкополосный фильтр. Фильтр должен удовлетворять следующим спецификациям:

полоса пропускания	49–51 Гц,
граничные частоты полосы подавления	48 и 52 Гц,
затухание в полосе подавления	60 дБ,
неравномерность в полосе пропускания	0,01 дБ,
частота дискретизации	1000 Гц.

Разработайте подходящий фильтр, используя обработку при нескольких скоростях.

- 9.7. Требуется интерпретировать активность в определенном физиологическом сигнале, выбираемом с частотой 256 Гц. Для этого во временной и частотной областях требуется извлечь и проанализировать характерные элементы каждой полосы. На первом этапе, расщеплении сигнала на указанные ниже полосы, разработайте подходящую систему с обработкой при нескольких скоростях.

0,5–4 Гц
4–8 Гц
8–13 Гц
13–16 Гц

Система с обработкой при нескольких скоростях должна вводить в полосы неравномерность не более 0,01 дБ, а внеполосные сигналы должны подавляться минимум на 50 дБ.

- 9.8. В системе цифровой обработки сигналов для преобразования входного аналогового аудиосигнала с диапазоном 0–20 кГц в цифровой поток битов с частотой 3,072 МГц используется метод перевыборки с повышением частоты и дельта-сигма-модулятор первого порядка. Модель дельта-сигма-модулятора в z -области изображена на рис. 9.39.

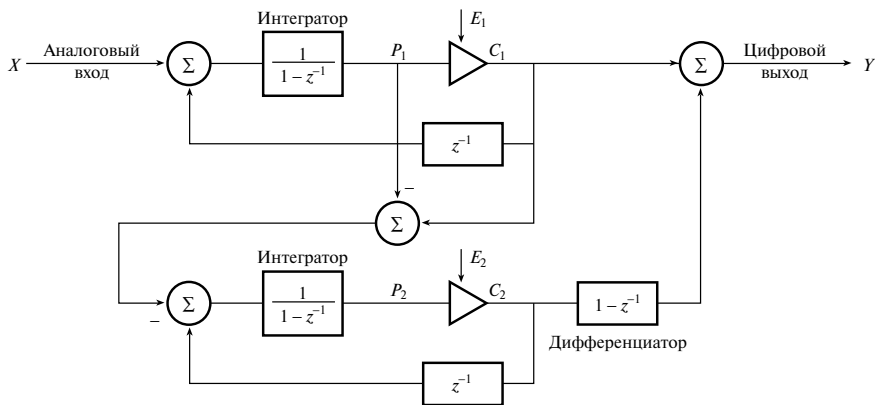


Рис. 9.39

1. Определите общее улучшение отношения сигнал-шум квантования при перевыборке и формировании спектра шума, оценив таким образом эффективное разрешение конвертера в битах.

2. Разработайте на уровне блок-схемы двухкаскадный дециматор для преобразования выхода дельта-сигма-модулятора, изображенного на рис. 9.39, из формы однобитового потока с частотой дискретизации 3,072 МГц в многобитовый поток с частотой 48 кГц. Неравномерности в полосе пропускания и полосе подавления равны 0,001 и 0,0001 соответственно.

Ответ должен включать следующие составляющие:

- значения общих шагов децимации и интерполяции с указанием того, как они были получены;
- значения пар шагов децимации и интерполяции для двухкаскадных преобразователей частоты с подробным анализом вычислительной сложности и требований к памяти, оправдывающих сделанный выбор;
- спецификацию граничных частот, длин, неравномерности в полосе пропускания и подавления для фильтров защиты от наложения спектров и подавления зеркальных частот.

При ответе можно предполагать, что используются КИХ-фильтры, реализованные в прямой форме, а их длина определяется соотношением

$$\text{длина фильтра, } N = \frac{-10 \lg(\delta_p \delta_s) - 13}{14,6 \Delta f} + 1,$$

где

Δf – нормированная ширина полосы перехода.

- 9.9.1, а. Объясните с помощью надлежащей блок-схемы принципы цифровой НЧ-фильтрации с обработкой при нескольких скоростях с использованием структуры дециматор/интерполятор. Назовите основные недостатки цифровой фильтрации с обработкой при нескольких скоростях.

- 1, б. Проанализировав однокаскадный фильтр нижних частот с обработкой при нескольких скоростях, покажите, что, если шаг децимации больше 2, использование обработки при нескольких скоростях вычислительно выгоднее по сравнению с реализациями, которые дают обычные фильтры с фиксированной скоростью обработки.

Продолжите анализ и покажите, что при уменьшении ширины полосы фильтра нижних частот вычислительное преимущество фильтров с обработкой при нескольких скоростях растет. Прокомментируйте практические следствия этого факта.

2. Требуется фильтр нижних частот, удовлетворяющий следующим спецификациям:

граничная частота полосы пропускания	4 Гц,
граничная частота полосы подавления	6,25 Гц,
неравномерность в полосе пропускания	0,001,
неравномерность в полосе подавления	0,0001,
частота дискретизации	500 Гц.

Разработайте на уровне блок-схемы эффективный фильтр с обработкой при нескольких скоростях согласно приведенным выше спецификациям. При разработке используйте двухкаскадный дециматор и двухкаскадный интерполятор. Ответ должен включать следующие элементы:

- значения общих шагов децимации и интерполяции плюс указания по их получению;
 - значения пар шагов децимации и интерполяции для двухкаскадных преобразователей частоты с подробным анализом вычислительной сложности и требований к памяти, оправдывающих сделанный выбор;
 - блок-схемы фильтра нижних частот с обработкой при нескольких скоростях с соответствующими пометками;
 - спецификацию граничных частот, длин, неравномерности в полосе пропускания и подавления для фильтров защиты от наложения спектров и подавления зеркальных частот.
3. Сравните вычислительную сложность фильтра из п. 2 с обычным КИХ-фильтром, реализованным в прямой форме. Ответ прокомментируйте. При ответе можно предполагать, что в разработке используются КИХ-фильтры, реализованные в прямой форме и имеющие длину

$$\text{длина фильтра, } N = \frac{-10 \lg(\delta_p \delta_s) - 13}{14,6 \Delta f} + 1,$$

где

Δf – нормированная ширина полосы перехода.

- 9.10. 1. Объясните роль следующих компонентов системы обработки при нескольких скоростях:
- а) фильтр для децимации;
 - б) компрессор частоты дискретизации.
2. Требуется разработать двухкаскадный дециматор для системы обработки аудиосигналов при нескольких скоростях, удовлетворяющий следующим спецификациям:

входная частота дискретизации, F_s	96 кГц,
шаг децимации, M	96,
наивысшая существенная частота, f_p	450 Гц,
неравномерность в полосе пропускания, δ_p	0,05,
неравномерность в полосе подавления, δ_s	0,01.

В ходе предварительного изучения задачи с помощью программных средств была получена информация, представленная в табл. 9.7. Используя данную информацию и спецификации:

- а) разработайте на уровне блок-схемы двухкаскадный дециматор; в ответе нужно указать подходящую пару шагов децимации (с аргументацией выбора) и частоту дискретизации на входе и выходе каждого каскада;

Таблица 9.7. Данные для разработки двухкаскадного дециматора

#	M_1	M_2	N_1	N_2	УВС	ОТП
0	2	48	2	2651	2 747 000	2653
1	3	32	6	1768	1 960 000	1774
2	4	24	10	1326	1 566 000	1336
3	6	16	17	885	1 157 000	902
4	8	12	24	664	952 000	688
5	12	8	38	443	747 000	481
6	16	6	53	333	651 000	386
7	24	4	88	222	574 000	310
8	32	3	131	167	560 000	298
9	48	2	254	112	620 000	366

Примечания:

#	номер варианта структуры,
M_1, M_2	пара шагов децимации для каскадов 1 и 2,
N_1, N_2	число коэффициентов фильтра для каскадов 1 и 2,
УВС	число умножений в секунду,
ОТП	общая требуемая память.

б) определите для каждого каскада дециматора следующие параметры фильтров:

граничные частоты,
нормированная ширина полосы перехода,
неравномерность в полосе пропускания и подавления,
число коэффициентов фильтра;

в) покажите с помощью соответствующих схем, что важная полоса частот (0–450 Гц) защищена дециматором от наложения.

9.11. Следует разработать двухкаскадный дециматор для системы обработки аудио-сигналов при нескольких скоростях согласно следующим спецификациям:

входная частота дискретизации, F_s	96 кГц,
шаг децимации, M	96,
наивысшая существенная частота, f_p	450 Гц,
неравномерность в полосе пропускания, δ_p	0,05,
неравномерность в полосе подавления, δ_s	0,01.

В ходе предварительного изучения задачи с помощью программных средств была получена информация, представленная в табл. 9.7. Используя данную информацию и спецификации,

а) определите оптимальные шаги децимации каскадов, используя уравнение (9.11);

б) округлите шаги децимации до ближайших целых и получите правильный общий шаг децимации;

в) сравните результат с результатом, полученным в задаче 9.10.2.

Задачи для решения с помощью MATLAB

9.12. Непрерывный сигнал характеризуется следующим уравнением:

$$x(t) = A \cos(2\pi f_1 t) + B \cos(2\pi f_2 t).$$

Эквивалентный сигнал дискретного времени $x(nT)$ получается дискретизацией непрерывного сигнала с частотой

$$F_s = \frac{1}{T}.$$

1. С помощью MATLAB получите 1000 выборок непрерывного сигнала. Предполагайте, что $F_s = 5000$ Гц, $f_1 = 50$ Гц, $f_2 = 100$ Гц, $A = 2$ и $B = 1$.
2. Используйте функцию `decimate` для снижения частоты дискретизации в 10 раз.
3. Используйте функцию `interp` для увеличения частоты дискретизации данных после децимации в 4 раза.

Укажите подходящее число выборок всех дискретных сигналов, полученных в п. 1–3, используя функцию `stem`. Прокомментируйте результаты.

- 9.13. Повторно решите задачу 9.12, используя функцию `resample`. Изобразите амплитудно-частотную характеристику внутреннего фильтра, применяемого при повторной выборке.
- 9.14. Повторно решите задачу 9.12, используя функцию `upfirdn` и подходящий КИХ-фильтр нижних частот, разработанный с помощью оптимального метода. Перечислите коэффициенты фильтра и изобразите его амплитудно-частотную характеристику. Укажите сделанные предположения.
- 9.15. Повторно решите задачу 9.14, используя функцию `resample`.
- 9.16. 1. Разработайте на уровне блок-схемы двухкаскадный дециматор, перевыбирающий аудиосигнал с понижением частоты в 30 раз, согласно следующим спецификациям:

входная частота дискретизации, F_s	240 кГц,
наивысшая важная частота в данных	3,4 кГц,
неравномерность в полосе пропускания, δ_p	0,05,
неравномерность в полосе подавления, δ_s	0,01,

$$\text{длина фильтра, } N = \frac{-10 \lg(\delta_p \delta_s)}{14,6 \Delta f} + 1,$$

где

Δf – нормированная ширина полосы перехода

Шаги децимации в каскадах 1 и 2 считайте равными 15 и 2 соответственно.

2. Используя оптимальный метод, вычислите коэффициенты и изобразите амплитудно-частотную характеристику фильтров защиты от наложения секторов в каскадах 1 и 2.

3. Проверьте дециматор, используя подходящие функции MATLAB и дискретный сигнал, полученный из следующего непрерывного сигнала:

$$x(t) = 2 \sin(2\pi 100t) + 2 \sin(2\pi 1000t) + 2 \sin(2\pi 3400t).$$

Литература

1. Adams R. W. (1986) Design and implementation of an audio 18 bit analog-to-digital converter using oversampling techniques. *J. Audio Engineering Society*, **34**(3), 153–166.
2. Agrawal B. P. and Shenoi K. (1983) Digital methodology for $\Sigma\Delta$ M. *IEEE Trans. Communications*, **31**(3), 360–370.
3. Claasen T. A. C. M., Mecklenbrauker W. F. G., Peek J. B. H. and Van Hurck N. (1980) Signal processing method for improving the dynamic range of A/D and D/A converters. *IEEE Trans. Acoustics, Speech and Signal Processing*, **28**(5), 529–538.
4. Crochiere R. E. and Rabiner L. R. (1975) Optimum FIR digital filter implementations for decimation, interpolation, and narrow-band filtering. *IEEE Trans. Acoustics, Speech and Signal Processing*, **23**(5), 444–456.
5. Crochiere R. E. and Rabiner L. R. (1976) Further considerations in the design of decimators and interpolators. *IEEE Trans. Acoustics, Speech and Signal Processing*, **24**, 296–311.
6. Crochiere R. E. and Rabiner L. R. (1979) A program for multistage decimation, interpolation, and narrow band filtering. In *IEEE Programs for DSP*. Institute of Electrical and Electronics Engineers.
7. Crochiere R. E. and Rabiner L. R. (1981) Interpolation and decimation of digital signals — a tutorial review. *Proc. IEEE*, **69**(3), 300–331.
8. Crochiere R. E. and Rabiner L. R. (1983) *Multirate Digital Signal Processing*. Englewood Cliffs NJ: Prentice-Hall.
9. Crochiere R. E. and Rabiner L. R. (1988) Multirate processing of digital signals. In *Advanced Topics in Signal Processing* (Lim J. S. and Oppenheim A. V. (eds)). Englewood Cliffs NJ: Prentice-Hall.
10. DeFatta D. J., Lucas J. G. and Hodgkiss W. S. (1988) *Digital Signal Processing: A System Design Approach*. New York: Wiley.
11. Liu B. and Mintzer F. (1978) Calculation of narrow-band spectra by direct decimation. *IEEE Trans. Acoustics, Speech and Signal Processing*, **26**(6), 529–534.
12. Matsuya Y., Uchimura K., Iwata A., Kobayashi T., Ishikawa M. and Yoshitome T. (1987) A 16-bit oversampling A-to-D conversion technology using triple integration noise shaping. *IEEE J. Solid State Circuits*, **22**(6), 921–928.
13. Quarmby D. (ed.) (1984) *Signal Processor Chips*, Chapter 5. London: Granada.
14. Welland D. R., Del Signore B. P., Swanson E. J., Tanaka T., Hamashita, K., Hara S. and Takasuka K. (1989) A stereo 16-bit delta-sigma A/D converter for digital audio. *J. Audio Engineering Society*, **37**(6), 476–486.
15. Westgate J. A., Keith R. D. F., Gurnow J. S. K., Ifeakor E. C. and Greene K. R. G. (1990) Suitability of fetal scalp electrodes for fetal electrocardiogram during labour. *J. Clin. Physics & Physiological Measurement*, **11**(4), 297–306.

Дополнительная литература

Analog Devices (1988) *ADSP-2100 Family Applications Handbook*, Volume 2, Chapter 3. Analog Devices, Inc.
 Bellanger M. G. (1977) Computation rate and storage estimation in multirate digital filtering with halfband filters. *IEEE Trans. Acoustics, Speech and Signal Processing*, **25**, 344–346.

- Bellanger M. G., Daquet J. L. and Lepagnol G. P. (1974) Interpolation, extrapolation and reduction of computation speed in digital filters. *IEEE Trans. Acoustics, Speech and Signal Processing*, **22**, 231–235.
- Brown Jr J. L. (1981) Multichannel sampling of lowpass signals. *IEEE Trans. Circuits Systems*, **28**, 101–106.
- Cox R. V., Bock D. E., Bauer K. B., Johnston J. D. and Snyder J. H. (1987) The analogue voice privacy system. *AT&T Technical J.*, **66**, 119–131.
- Elliot D. F. (ed.) (1987) *Handbook of Digital Signal Processing*. New York: Academic Press.
- Goedhart D., van der Plassche R. J. and Stikvoort E. F. (1982) Digital-to-analog conversion in playing a compact disc. *Philips Technical Rev.*, **40**(6), 174–179.
- Goodman D. J. and Carey M. J. (1977) Nine digital filters for decimation and interpolation. *IEEE Trans. Acoustics, Speech and Signal Processing*, **25**(2), 121–126.
- Goodman D. J. and Flanagan J. L. (1971) Direct digital conversion between linear and adaptive delta modulation formats. In *Proc. IEEE Int. Communications Conf.*, Montreal, Canada, June 1971.
- Huber A., De Man E., Schiller E. and Ulbrich W. (1986) FIR lowpass filter for signal decimation with 15 MHz clock frequency. In *IEEE Int. Conf. Acoustics, Speech and Signal Processing*, Tokyo, 7–11 April, pp. 1533–1536.
- Jerri A. J. (1977) The Shannon sampling theorem — its various extensions and applications: a tutorial review. *Proc. IEEE*, **65**(11), 1565–1596.
- Linden D. A. (1959) A discussion of sampling theorems. *Proc. IRE.*, **47**, 1219–1226.
- Mintzer F. (1982) On half-band, third-band and Nth-band FIR filters and their design. *IEEE Trans. Acoustics, Speech and Signal Processing*, **30**, 734–738.
- Mintzer F. and Liu B. (1978) The design of optimal multirate bandpass and bandstop filters. *IEEE Trans. Acoustics, Speech and Signal Processing*, **26**(6), 534–543.
- Mintzer F. and Liu B. (1978) Aliasing error in the design of multirate filters. *IEEE Trans. Acoustics, Speech and Signal Processing*, **26**, 76–88.
- Montijo B. A. (1988) Digital filtering in a high-speed digitizing oscilloscope. *Hewlett Packard J.*, June, 70–76.
- Mou Z. J. and Duhamel P. (1987) Fast FIR filtering: algorithms and implementations. *Signal Processing*, **13**, 377–384.
- Princen J. P. and Bradley A. B. (1986) Analysis/synthesis filter banks design based on time domain aliasing cancellation. *IEEE Trans. Acoustics, Speech and Signal Processing*, **23**, 1153–1161.
- Rabiner L. R. and Crochiere R. E. (1975) A novel implementation for narrow-band FIR digital filters. *IEEE Trans. Acoustics, Speech and Signal Processing*, **23**(5), 457–464.
- Regalia P. A., Fujii N., Mitra S. K. and Neuvo Y. (1987) Active RC crossover networks with adjustable characteristics. *J. Audio Engineering Society*, January–February, **35**(1/2), 24–30.
- Rorabacher D. W. (1975) Efficient FIR filter design for sample rate reduction and interpolation. In *Proc. IEEE International Symposium on Circuits and Systems*, 21–23 April, pp. 396–399.
- Schafer R. W. and Rabiner R. L. (1973) A digital signal processing approach to interpolation. *Proc. IEEE*, **61**, 692–702.
- Scheuermann H. and Gockler H. (1981) A comprehensive survey of digital transmultiplexing methods. *Proc. IEEE*, **69**, 1419–1450.
- Shannon C. E. (1949) Communications in the presence of noise. *Proc. IRE*, **37**, 10–21.
- Thong T. (1989) Practical consideration for a continuous time digital spectrum analyzer. In *Proc. IEEE International Symposium on Circuits and Systems*, Portland OR, May 1989, 1047–1050.
- Tufts D. W., Rorabacher D. W. and Mosier W. E. (1970) Designing simple, effective digital filters. *IEEE Trans. Audio Electro-Acoustics*, **18**, 142–158.
- Vaidyanathan P. P. (1990) Multirate digital filters, filter banks, polyphase networks, and applications: a tutorial. *Proc. IEEE*, **78**(1), 56–93.
- Vaidyanathan P. P. (1993) *Multirate Systems and Filter Banks*. Englewood Cliffs NJ: Prentice-Hall.
- Vaidyanathan P. P. and Nguyen T. Q. (1987) A trick for the design of FIR half-band filters. *IEEE Trans. Circuits and Systems*, **34**, 297–300.
- Van De Plassche R. J. and Dijkmans E. C. (1983) A monolithic 16-bit D/A conversion system for digital audio. In *Digital Audio* (Blessner B. (ed.)), pp. 54–60. Audio Engineering, Inc.
- Zobel R. N. and Tang P. S. (1985) A high performance multichannel decimating FIR digital filter system for microprocessor based data acquisition. *Proc. ISCAS*, 1149–1152.

Приложения

9.A. Программы на С для разработки и реализации систем обработки при нескольких скоростях

На компакт-диске к книге [Ifeachor, 2001] содержатся следующие программы на языке С с примерами их использования (подробности см. в предисловии):

- 1) `decimate.c`, которая производит децимацию данных с использованием до трех каскадов децимации;
- 2) `interp.c`, которая интерполирует данные с использованием до трех каскадов интерполяции;
- 3) `portim.c`, определяющая характеристики I -каскадного дециматора (или интерполятора) при $I = 1, 2, 3, 4$. Вычисляются такие параметры: шаги децимации и характеристики фильтров каждого каскада, а также меры эффективности (такие как число умножений в секунду) при различных конфигурациях.

9.B. Цифровая обработка сигналов с помощью MATLAB

Для выполнения множества операций, связанных с преобразованием частоты дискретизации и обработкой при нескольких скоростях, можно использовать средство MATLAB Signal Processing Toolbox. В данном разделе рассмотрены ключевые функции MATLAB, предназначенные для обработки при нескольких скоростях, и иллюстрируется их использование.

Как обсуждалось в данной главе, создание систем обработки при нескольких скоростях включает следующие этапы:

- 1) спецификация требований к преобразователям частоты дискретизации;
- 2) определение параметров преобразователя (т.е. числа каскадов, шагов изменения частоты дискретизации и характеристики фильтров каскадов);
- 3) разработка цифровых фильтров защиты от наложения спектров и/или подавления зеркальных частот;
- 4) фильтрация с обработкой при нескольких скоростях, выборка с повышением и/или понижением частоты.

Значительно облегчить выполнение этапов 3 и 4 позволяет средство Signal Processing Toolbox. Существует четыре ключевые функции MATLAB для обработки при нескольких скоростях: `decimate`, `interp`, `resample` и `upfirdn`. В дополнение к функциям цифровой обработки при нескольких скоростях имеется много других функций, в частности, связанных с КИХ-фильтрацией.

Функция `decimate` используется для снижения частоты дискретизации последовательности данных в M раз, где M — положительное целое число. Функция выполняет

фильтрацию с целью подавления зеркальных изображений на основе фильтра нижних частот, а затем выбирает сигнал с более низкой частотой, т.е. оставляет одну точку из каждых M точек данных. Команда `decimate` используется со следующим синтаксисом:

```
y=decimate(x,M)
y=decimate(x,M,N)
y=decimate(x,M,'fir')
y=decimate(x,M,N,'fir')
```

Команда `y=decimate(x,M,N,'fir')` фильтрует последовательность данных, собранных в векторе x , с помощью N -точечного КИХ-фильтра нижних частот с нормированной частотой среза $1/M$, а затем уменьшает частоту дискретизации в M раз. Нужный фильтр автоматически генерируется средством Toolbox.

Если параметр `'fir'` опущен (как в первой приведенной выше команде), при децимации используется фильтр Чебышева первого рода n -го порядка с нормированной частотой среза $0,8/M$ и неравномерностью в полосе пропускания 0,05 дБ (по умолчанию значение $N = 8$). Если используется БИХ-фильтр, то автоматически вводится фильтр для компенсации фазового искажения в прямом и обратном по времени направлениях.

Функция `interp` используется для увеличения частоты дискретизации последовательности данных в целое число раз L . Функция вначале расширяет последовательность данных путем введения нулей между выборками, автоматически разрабатывает КИХ-фильтр нижних частот, а затем выполняет фильтрацию с целью подавления зеркальных частот с помощью КИХ-фильтра. Синтаксис команды:

```
y=interp(x,L)
y=interp(x,L,N,alpha)
[y,b]=interp(x,L,N,alpha)
```

Параметр N задает длину фильтра (длина фильтра $= 2^{N+1}$, значение длины по умолчанию — 4), α — это нормированная частота среза (значение по умолчанию 0,5, т.е. $\alpha = f_c/f_{Nyquist}$). Последняя команда позволяет выделить коэффициенты фильтра подавления зеркальных частот в векторе b .

Пример 9Б.1

Непрерывный сигнал характеризуется следующим уравнением:

$$x(t) = A \cos(2\pi f_1 t) + B \cos(2\pi f_2 t).$$

1. Сгенерируйте с помощью MATLAB дискретный эквивалент сигнала. Частота дискретизации равна 1 кГц, $f_1 = 50$ Гц, $f_2 = 100$ Гц, отношение амплитуд частотных компонентов $A/B = 1,5$.
2. Интерполируйте дискретный сигнал с повышением частоты в 4 раза, используя функцию `interp`.
3. К полученному в п. 2 результату примените операцию децимации с шагом 4, используя операцию `decimate`.
4. Изобразите исходный сигнал, сигналы после интерполяции и децимации.

Программа 9Б.1. Иллюстрация простых операций интерполяции и децимации в форме m-файла MATLAB

```
%
% Имя файла: Program EX9B1.m
% Иллюстрация интерполяции с шагом 4
%
Fs=100; % частота дискретизации
A=1.5; % относительные амплитуды
B=1;
f1=50; % частоты сигнала
f2=100;
t=0:1/Fs:1; % вектор времени
x=A*cos(2*pi*f1*t)+B*cos(2*pi*f2*t); % генерируемый сигнал
y=interp(x,4); % сигнал, интерполированный с шагом 4
stem(x(1:25)) % вывод на экран исходного сигнала
xlabel('Discrete time, nT')
ylabel('Input signal level')
figure
stem(y(1:100)) % вывод на экран интерполированного сигнала
xlabel('Discrete time, 4 x nT')
ylabel('Interpolated output signal level')
y1=decimate(y,4);
stem(y1(1:25)) % вывод на экран сигнала после децимации
xlabel('Discrete time, nT')
ylabel('Decimated output signal level')
```

Решение

Реализация решения задачи в форме m-файла MATLAB приведена в программе 9Б.1.

Исходный сигнал и два искоемых сигнала приведены на рис. 9Б.1, *а*, *б*, *в*. Читатель должен отметить разницу сигналов на рис. 9Б.1, *а* и *в*, которая объясняется несовершенством операций преобразования частоты.

Функции `decimate` и `interp` весьма полезны при быстрой реализации изменений частоты дискретизации. Впрочем, они несколько ограничены (например, фильтры защиты от наложения спектров и подавления зеркальных частот вводятся в структуру автоматически). Более гибкими являются функции `upfirdn` и `resample`, которые допускают выбор из большего числа альтернатив.

Для увеличения и/или уменьшения частоты дискретизации последовательности данных в L/M раз (L и M — положительные целые числа) можно использовать функции `upfirdn` и `resample`. Обе команды выполняют аналогичные операции. Синтаксис команды `upfirdn`:

```
y=upfirdn(x,h)
y=upfirdn(x,h,L)
y=upfirdn(x,h,L,M)
```

Команда `y=upfirdn(x,h,L,M)`, например, вначале увеличивает частоту дискретизации в L раз, выполняет КИХ-фильтрацию с использованием коэффициентов, вне-

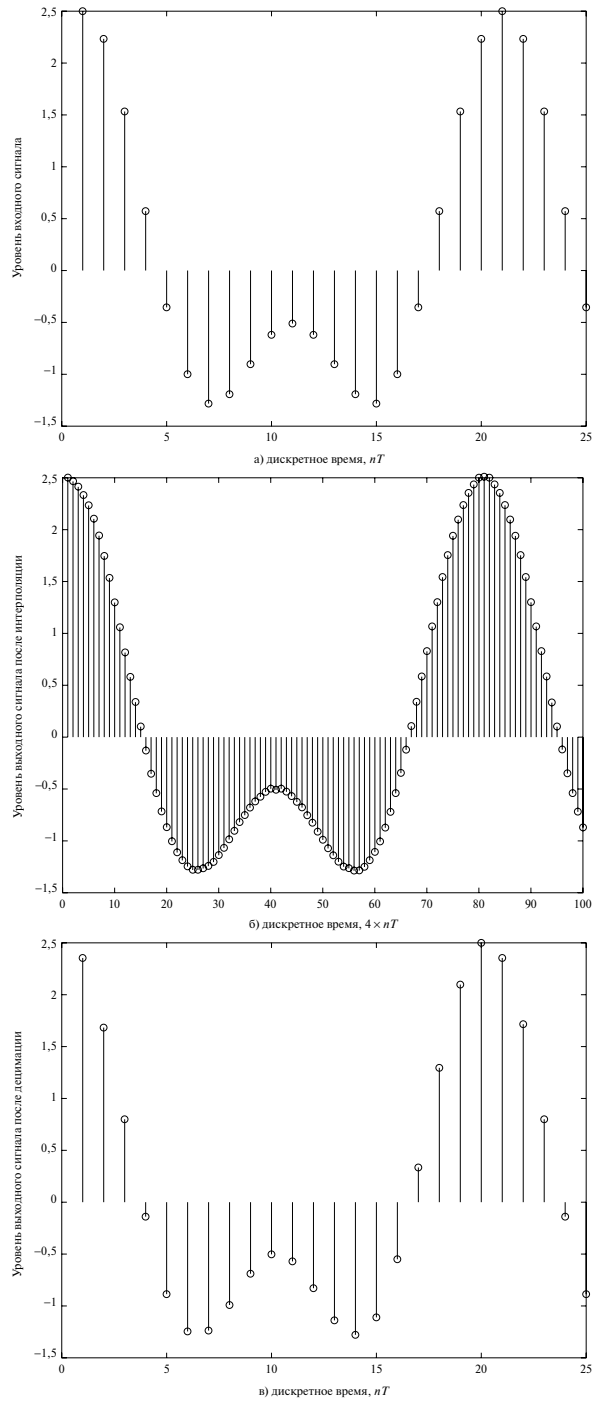


Рис. 9Б.1. Входной сигнал (панель а). Иллюстрация интерполяции с шагом 4 (панель б). Иллюстрация децимации с шагом 4 (панель в)

сенных в вектор h , а затем перебирает данные с понижением частоты в M раз. Для получения КИХ-фильтра можно использовать оптимальный метод, метод вырезания или частотной выборки. Выходной сигнал команда возвращает в вектор y . Если L и M равны 1, производится операция КИХ-фильтрации. Если $L = 1$, данные проходят децимацию (т.е. их частота дискретизации уменьшается) с шагом M , а если $M = 1$, они проходят интерполяцию (т.е. частота дискретизации увеличивается) с шагом L .

Синтаксис команды `resample`:

```
y=resample(x,L,M)
[y,b]=resample(x,L,M)
y=resample(x,L,M,b)
```

Команда `y=resample` ограничивает полосу последовательности данных, внесенных в вектор x , используя КИХ-фильтр, разработанный на основе функции Кайзера с помощью команды `fir1`. Коэффициенты КИХ-фильтра можно получить, используя вторую команду, т.е. `[y,b]=resample(x,L,M)`. Команда `y=resample(x,L,M,b)` предназначена для использования в разработке пользовательского фильтра.

В качестве иллюстрации для решения последней задачи (пример 9Б.1) мы использовали команду `resample`, т.е. с ее помощью частота дискретизации была увеличена, а затем уменьшена в 4 раза. Реализация решения задачи в форме `m`-файла MATLAB приведена в программе 9Б.2. Результаты сходны, но не идентичны из-за различий в реализации.

Программа 9Б.2. Иллюстрация простых операций интерполяции и децимации в форме `m`-файла MATLAB

```
%
% Имя файла: Program EX9B2.m
% Иллюстрация децимации и интерполяции с шагом 4
%
Fs=100; % частота дискретизации
A=1.5; % относительные амплитуды
B=1;
f1=50; % частоты сигнала
f2=100;
t=0:1/Fs:1; % вектор времени
x=A*cos(2*pi*f1*t)+B*cos(2*pi*f2*t); % генерируемый сигнал
y=resample(x,4,1); % сигнал, интерполированный с шагом 4
stem(x(1:25)) % вывод на экран исходного сигнала
xlabel('Discrete time, nT')
ylabel('Input signal level')
figure
stem(y(1:100)) % вывод на экран интерполированного сигнала
xlabel('Discrete time, 4 x nT')
ylabel('Interpolated output signal level')
y1=resample(y,1,4);
stem(y1(1:25)) % вывод на экран сигнала после децимации
xlabel('Discrete time, nT')
ylabel('Decimated output signal level')
```

Адаптивные цифровые фильтры*

10

10.1. Когда и где стоит использовать адаптивные фильтры	699
10.2. Концепции адаптивной фильтрации	701
10.3. Основы теории фильтров Винера	705
10.4. Стандартный адаптивный алгоритм наименьших квадратов	708
10.5. Рекурсивный алгоритм наименьших квадратов	717
10.6. Сфера приложения 1 — адаптивная фильтрация окулярных артефактов на ЭЭГ человека	720
10.7. Сфера приложения 2 — адаптивное телефонное эхоподавление	723
10.8. Другие приложения	725
Задачи	729
Литература	730
Дополнительная литература	730
Приложения	731

*Здесь говорится только о линейных адаптивных фильтрах. Современная адаптивная обработка цифровых сигналов этим не ограничена. — *Прим. ред.*

По сути, адаптивный фильтр — это цифровой фильтр с самонастраивающимися характеристиками. Такой фильтр автоматически адаптируется к изменениям выходных сигналов и является предметом изучения подобласти ЦОС, известной как *адаптивная обработка сигналов*. Ниже описаны ключевые аспекты этой важной темы на основе схемы наименьших квадратов (СНК) и рекурсивной схемы наименьших квадратов (РСНК) — двух аппаратов, наиболее используемых при адаптивной обработке сигналов. Данная глава предназначена для практического применения, теория приводится только в минимально необходимом объеме. Реализацию фильтров, основанных на СНК и РСНК, на языке С можно найти на компакт-диске к книге [Ifeachor, 2001] (подробности см. в предисловии). В главе также рассмотрено несколько реальных областей применения адаптивных фильтров.

10.1. Когда и где стоит использовать адаптивные фильтры

Во многих приложениях часто возникает такая проблема, как загрязнение полезного сигнала нежелательными, часто мощными, сигналами или шумом. Если полезный сигнал и шум занимают строго определенные и разнесенные полосы частот, для извлечения полезного сигнала используются обычные линейные фильтры. Однако существуют ситуации, при которых нужно, чтобы характеристики фильтра были переменными, адаптировались к меняющимся характеристикам сигнала или чередовались интеллект-

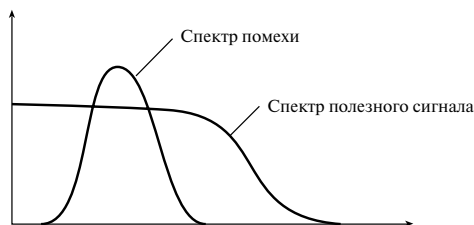


Рис. 10.1. Иллюстрация наложения спектров сигнала и мощной помехи

туально. В подобных случаях, например, когда спектры сигнала и шума перекрываются (рис. 10.1), или когда полоса шумовых частот неизвестна или переменна, коэффициенты фильтра должны меняться, и их нельзя задать заранее. Перечислим типичные приложения, в которых неуместны фильтры с фиксированными коэффициентами.

1. Электроэнцефалография (ЭЭГ), где артефакты, или загрязнение сигнала, порожденное движением или морганием глаза, значительно больше подлинной электрической деятельности мозга и занимают ту же полосу частот, что и сигнал, представляющий медицинский интерес. Чтобы удалить артефакты, сохранив при этом нужные сигналы, обычные линейные фильтры не подходят.
2. В цифровой связи присутствует метод расширения спектра, который используется в условиях, когда сильная активная помеха может интерферировать с полезным сигналом. Интерференция часто происходит в узкой, но неизвестной полосе частот в широкополосном спектре, и единственное эффективное решение этой проблемы предлагают адаптивные фильтры.
3. Цифровая передача данных по телефонным каналам на высокой скорости. Искажение сигнала, вызванное плохими амплитудной и фазовой характеристиками канала, приводит к тому, что импульсы, представляющие различные цифровые коды, интерферируют между собой (межсимвольная интерференция), что затрудняет надежное детектирование кодов в приемнике. Для компенсации внесенного каналом искажения, которое может быть переменным или неизвестным, используется адаптивное выравнивание.

Адаптивный фильтр отличается следующим свойством: его частотная характеристика автоматически регулируется или модифицируется для улучшения производительности фильтра в соответствии с некоторым критерием, что позволяет фильтру адаптироваться к изменениям характеристик входного сигнала. Вследствие самонастройки и присущей адаптивным фильтрам гибкости они используются в таких различных сферах, как телефонное эхоподавление, обработка сигналов с радаров, системы навигации, выравнивание каналов связи и выделение биомедицинских сигналов.

Итак, адаптивные фильтры стоит использовать, если:

- характеристики фильтра должны быть переменными, адаптироваться к меняющимся условиям;
- существует спектральное перекрытие сигнала и шума (рис. 10.1);
- полоса шумовых сигналов неизвестна или меняется со временем.

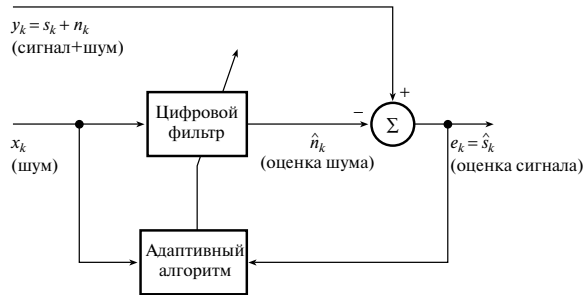


Рис. 10.2. Блок-схема адаптивного фильтра-шумоподавителя

В перечисленных случаях использование обычных фильтров привело бы к неприемлемому искажению полезного сигнала. Ниже показано, что помимо снижения уровня шумов есть и другие ситуации, когда выгодно использовать адаптивные фильтры.

10.2. Концепции адаптивной фильтрации

10.2.1. Адаптивные фильтры как средство подавления шума

Адаптивный фильтр состоит из двух различных частей: цифрового фильтра с регулируемыми коэффициентами и адаптивного алгоритма, который используется для настройки или изменения коэффициентов фильтра (рис. 10.2). На адаптивный фильтр одновременно подаются два входных сигнала, y_k и x_k . Сигнал y_k загрязнен и содержит полезный сигнал s_k и шум n_k , причем предполагается, что эти сигналы не коррелируют. Сигнал x_k — это мера загрязненного сигнала, который коррелирует с n_k . В цифровом фильтре сигнал x_k обрабатывается для получения оценки $\hat{n}_k$ сигнала n_k . Затем полезный сигнал оценивается как разность зашумленного сигнала y_k и выхода цифрового фильтра $\hat{n}_k$:

$$\hat{s}_k = y_k - \hat{n}_k = s_k + n_k - \hat{n}_k. \quad (10.1)$$

Основная цель шумоподавления заключается в получении оптимальной оценки шума в зашумленных сигналах, а следовательно, оптимальной оценки желаемого сигнала. Для этого $\hat{s}_k$ подается в цепь обратной связи, и на основе этого сигнала и подходящего адаптивного алгоритма регулируются коэффициенты цифрового фильтра для минимизации шума в сигнале $\hat{s}_k$. Выходной сигнал $\hat{s}_k$ используется двояко: 1) как оценка желаемого сигнала и 2) как сигнал ошибки, который применяется для регулировки коэффициентов фильтра.

10.2.2. Другие конфигурации адаптивного фильтра

В приведенных выше рассуждениях предполагается, что фильтр используется для адаптивного шумоподавления. Важно помнить, что адаптивные фильтры могут использоваться и в других целях, например, для линейного предсказания, адаптивного выделения сигнала и адаптивного управления. Вообще всегда стоит иметь в виду, что значение сигналов x_k , y_k и e_k или способа их получения зависит от приложения. Некоторые примеры конфигураций адаптивного фильтра приведены на рис. 10.3.

10.2.3. Основные компоненты адаптивного фильтра

В большинстве адаптивных систем цифровой фильтр, показанный на рис. 10.2, реализован с использованием трансверсальной структуры или схемы с конечной импульсной характеристикой (КИХ) (рис. 10.4). Иногда используются и другие формы, например, схема с бесконечной импульсной характеристикой (БИХ) или решетчатая структура, но КИХ-структура применяется наиболее широко из-за своей простоты и гарантированной устойчивости. Выход N -точечного фильтра, изображенного на рис. 10.4, выражается следующим образом:

$$\hat{n}_k = \sum_{i=0}^{N-1} w_k(i) x_{k-i}, \quad (10.2)$$

где $w_k(i)$, $i = 0, 1, \dots$ — регулируемые коэффициенты фильтра (или весовые коэффициенты), а $x_k(i)$ и $\hat{n}_k$ — вход и выход фильтра соответственно. На рис. 10.4 изображена система с одним входом и одним выходом. В системе с несколькими входами и одним выходом x_k может включать одновременные входы N различных источников сигнала.

10.2.4. Адаптивные алгоритмы

Для настройки коэффициентов цифрового фильтра (на рис. 10.2) используются адаптивные алгоритмы, в которых сигнал ошибки e_k минимизируется согласно некоторому критерию, например, по схеме наименьших квадратов. Наибольшее распространение получили алгоритмы, в которых используются схема наименьших квадратов, рекурсивная схема наименьших квадратов и фильтр Калмана. С точки зрения вычислительной сложности и требований к памяти наиболее эффективна схема наименьших квадратов. Более того, для нее несущественна проблема численной неустойчивости, присущая двум другим алгоритмам. По этим причинам во многих ситуациях вначале пробуют схему наименьших квадратов. Впрочем, рекурсивная схема наименьших квадратов имеет лучшую сходимость.

Пример 10.1

Согласно [11] оценка желаемого сигнала на выходе адаптивного шумоподавителя выражается следующим образом:

$$\hat{s}_k = y_k - \hat{n}_k = s_k + n_k - \hat{n}_k.$$

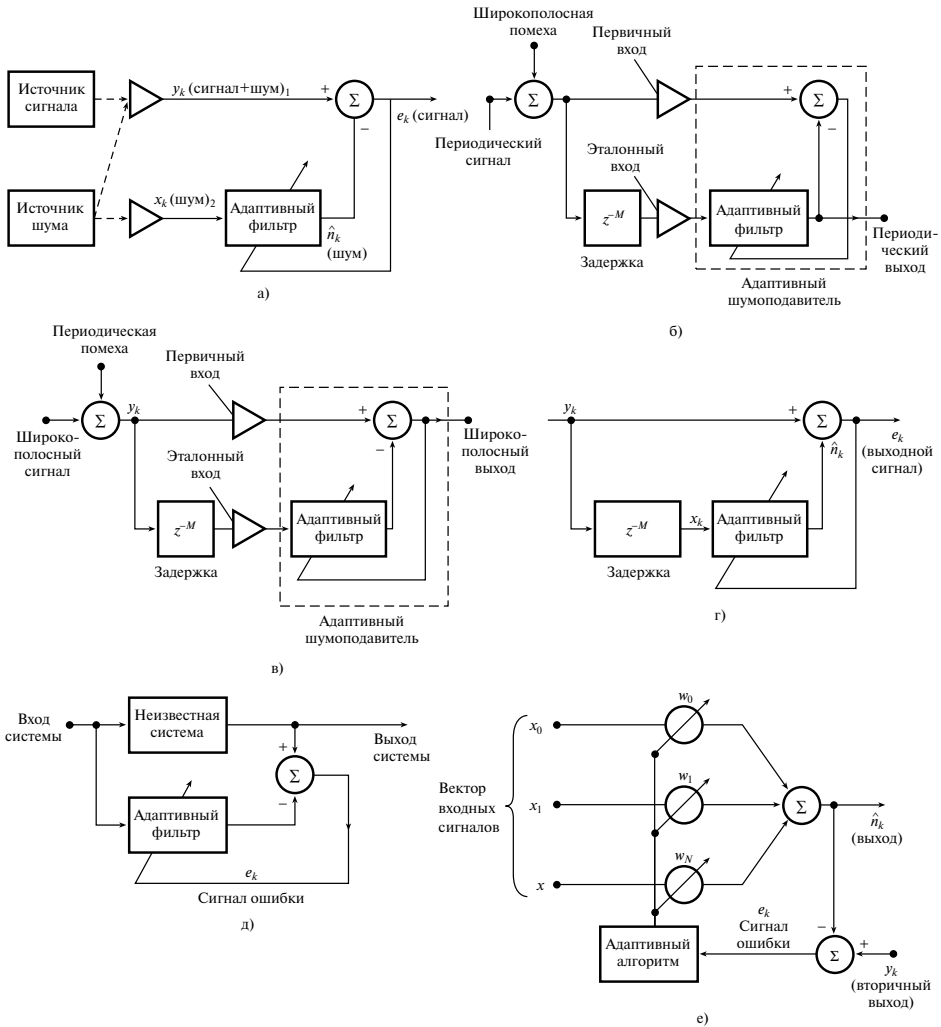


Рис. 10.3. Некоторые конфигурации адаптивного фильтра (согласно [10]): а) адаптивный шумоподавитель; б) адаптивный самонастраивающийся фильтр; в) подавление периодической помехи без внешнего эталонного источника; г) адаптивный селективный фильтр; д) моделирование системы; е) подбор линейной комбинации

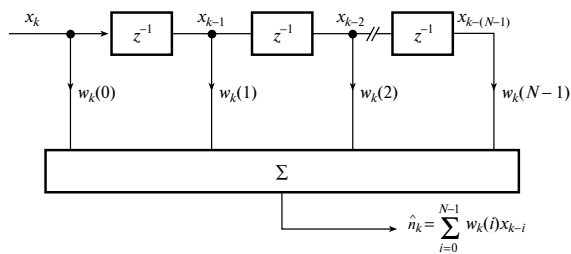


Рис. 10.4. Структура фильтра с конечной импульсной характеристикой

Покажите, что при минимизации общей мощности на выходе шумоподавителя максимизируется выходное отношение сигнал-шум.

Решение

Зашумленный сигнал записывается как

$$y_k = s_k + n_k, \quad (10.3)$$

а оценка полезного сигнала равна

$$\hat{s}_k = y_k - \hat{n}_k = s_k + n_k - \hat{n}_k. \quad (10.4)$$

Возводя уравнение (10.4) в квадрат, получаем

$$\hat{s}_k^2 = s_k^2 + (n_k - \hat{n}_k)^2 + 2s_k(n_k - \hat{n}_k). \quad (10.5)$$

Вычисляя математическое ожидание выражений справа и слева от знака "=", получаем

$$E[\hat{s}_k^2] = E[s_k^2] + E[(n_k - \hat{n}_k)^2] + 2E[s_k(n_k - \hat{n}_k)]. \quad (10.6)$$

Поскольку полезный сигнал s_k не коррелирует с n_k или с $\hat{n}_k$, последнее слагаемое в формуле (10.6) равно нулю, так что получаем

$$E[\hat{s}_k^2] = E[s_k^2] + E[(n_k - \hat{n}_k)^2], \quad (10.7)$$

где $E[s_k^2]$ представляет общую мощность сигнала, $E[\hat{s}_k^2]$ — оценку мощности сигнала (а также общую выходную мощность), а $E[(n_k - \hat{n}_k)^2]$ — остаточную мощность шума, который все еще может содержаться в s_k . Из уравнения (10.7) очевидно, что, если оценка $\hat{n}_k$ — это точная копия n_k , выходная мощность — это мощность только полезного сигнала. При настройке адаптивного фильтра к оптимальному положению минимизируется мощность остаточного шума, а следовательно, общая выходная мощность. На мощность полезного сигнала настройка не влияет, поскольку s_k не коррелирует с n_k . Следовательно,

$$\min E[\hat{s}_k^2] = E[s_k^2] + \min E[(n_k - \hat{n}_k)^2]. \quad (10.8)$$

Из уравнения (10.8) очевидно, что суммарный эффект минимизации общей выходной мощности заключается в максимизации выходного отношения сигнал-шум. Если фильтр настроен так, что $\hat{n}_k = n_k$, то $\hat{s}_k = s_k$. В этом случае выход адаптивного шумоподавителя свободен от шума. Если сигнал y_k не содержит шума, т.е. когда $n_k = 0$, адаптивный фильтр самовыключается (по крайней мере, теоретически) путем присвоения всем весовым коэффициентам нулевых значений.

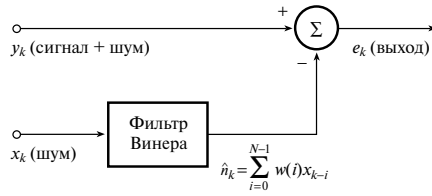


Рис. 10.5. Фильтр Винера (стандартная форма)

10.3. Основы теории фильтров Винера

Многие адаптивные алгоритмы можно рассматривать как аппроксимации дискретного фильтра Винера (рис. 10.5). На фильтр одновременно подаются два сигнала x_k и y_k . Обычно y_k состоит из компонента, который коррелирует с x_k , и компонента, который с x_k не коррелирует. Винеровский фильтр выдает оптимальную оценку той части y_k , которая коррелирует с x_k , затем эта оценка вычитается из y_k и получается e_k .

Предположим, что используется КИХ-фильтр с N коэффициентами (или *весовыми коэффициентами* — весьма популярный термин в литературе). Ошибка e_k между выходом фильтра Винера и исходным сигналом y_k записывается как

$$e_k = y_k - \hat{n}_k = y_k - \mathbf{W}^T \mathbf{X}_k = y_k - \sum_{i=0}^{N-1} w(i)x_{k-i}, \quad (10.9)$$

где $\mathbf{X}_k$ и $\mathbf{W}$ — вектор входного сигнала и вектор весовых коэффициентов соответственно, имеющие вид

$$\mathbf{X}_k = \begin{bmatrix} x_k \\ x_{k-1} \\ \vdots \\ x_{k-(N-1)} \end{bmatrix}, \quad \mathbf{W} = \begin{bmatrix} w(0) \\ w(1) \\ \vdots \\ w(N-1) \end{bmatrix}. \quad (10.10)$$

Квадрат ошибки равен

$$e_k^2 = y_k^2 - 2y_k \mathbf{X}_k^T \mathbf{W} + \mathbf{W}^T \mathbf{X}_k \mathbf{X}_k^T \mathbf{W}. \quad (10.11)$$

Чтобы получить среднеквадратическую ошибку J , нужно найти математическое ожидание обеих частей уравнения (10.11), предполагая, что вектор входа $\mathbf{X}_k$ и сигнал y_k совместно стационарны:

$$\begin{aligned} J = E[e_k^2] &= E[y_k^2] - 2E[y_k \mathbf{X}_k^T \mathbf{W}] + E[\mathbf{W}^T \mathbf{X}_k \mathbf{X}_k^T \mathbf{W}] = \\ &= \sigma^2 + 2\mathbf{P}^T \mathbf{W} + \mathbf{W}^T \mathbf{R} \mathbf{W}, \end{aligned} \quad (10.12)$$

где $E[\cdot]$ обозначает математическое ожидание, $\sigma^2 = E[y_k^2]$ — дисперсия y_k , $\mathbf{P} = E[y_k \mathbf{X}_k]$ — N -компонентный вектор взаимной корреляции, а $\mathbf{R} = E[\mathbf{X}_k \mathbf{X}_k^T]$ — автокорреляционная матрица $N \times N$. График зависимости среднеквадратической ошибки от

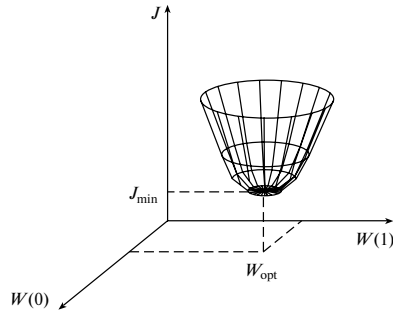


Рис. 10.6. Поверхность производительности

коэффициентов W — это чашеобразная поверхность с однозначно определенной нижней точкой (см. рис. 10.6). Данная структура называется *поверхностью производительности* (performance surface) и она неотрицательна. Градиент поверхности производительности записывается так:

$$\nabla = \frac{dJ}{dW} = -2P + 2RW. \quad (10.13)$$

Каждый набор коэффициентов $w(i)$ ($i = 0, 1, \dots, N - 1$) соответствует точке на поверхности. В точке минимума градиент равен нулю и вектор весовых коэффициентов фильтра достигает оптимального значения W_{opt} (см. пример 10.2):

$$W_{opt} = R^{-1}P. \quad (10.14)$$

Формула (10.14) называется *уравнением Винера–Хопфа*. Задача адаптивной фильтрации заключается в настройке весовых коэффициентов фильтра $w(0), w(1), \dots$ с помощью подходящего алгоритма до достижения оптимальной точки на поверхности производительности.

Практическое использование винеровских фильтров ограничено из-за следующих недостатков:

- требует применения автокорреляционной матрицы R и вектора взаимной корреляции P , причем обе величины *априори* неизвестны;
- фильтры включают обращение матриц, являющееся трудоемким процессом;
- если сигналы нестационарны, то R и P будут меняться со временем, так что W_{opt} придется вычислять многократно.

В приложениях реального времени W_{opt} требуется пересчитывать по мере поступления выборок. Использование адаптивных алгоритмов позволяет не обращаться при этом матрицы и не вычислять явно R и P .

Пример 10.2

Начиная с выражения для среднеквадратической ошибки (10.12), выведите уравнение Винера–Хопфа.

Решение

Среднеквадратическая ошибка записывается следующим образом:

$$J = \sigma^2 + 2\mathbf{P}^T \mathbf{W} + \mathbf{W} \mathbf{R} \mathbf{W}. \quad (10.15)$$

Градиент ∇ ошибки J получается дифференцированием J по вектору весовых коэффициентов $\mathbf{W}$ и приравниванием результата к нулю (см. [4]):

$$\nabla = \frac{dJ}{d\mathbf{W}} = \frac{d\sigma^2}{d\mathbf{W}} + \frac{d(\mathbf{P}^T \mathbf{W})}{d\mathbf{W}} + \frac{d(\mathbf{W}^T \mathbf{R} \mathbf{W})}{d\mathbf{W}}. \quad (10.16)$$

В нашем случае

$$\begin{aligned} \frac{d\sigma^2}{d\mathbf{W}} &= 0, \\ \frac{d(2\mathbf{P}^T \mathbf{W})}{d\mathbf{W}} &= -2\mathbf{P}, \\ \frac{d(\mathbf{W}^T \mathbf{R} \mathbf{W})}{d\mathbf{W}} &= 2\mathbf{R} \mathbf{W}. \end{aligned}$$

Используя эти результаты и положив $\nabla = \mathbf{0}$ в уравнении (10.16), получаем

$$\nabla = \frac{dJ}{d\mathbf{W}} = -2\mathbf{P} + 2\mathbf{R} \mathbf{W}. \quad (10.17)$$

Вектор оптимальных коэффициентов записывается в виде

$$\mathbf{W}_{opt} = \mathbf{R}^{-1} \mathbf{P}. \quad (10.18)$$

10.4. Стандартный адаптивный алгоритм наименьших квадратов

Одним из наиболее удачных адаптивных алгоритмов является алгоритм наименьших квадратов, разработанный группой Уидроу (см. [11]). Взамен вычисления $\mathbf{W}_{opt}$ за один шаг по формуле (10.18) в схеме наименьших квадратов коэффициенты настраиваются при последовательной обработке выборок с минимизацией среднеквадратической ошибки. Результат равносителен спуску по поверхности, изображенной на рис. 10.6, к ее дну.

Схема наименьших квадратов основана на алгоритме быстрейшего спуска, в котором вектор весовых коэффициентов следующим образом обновляется при обработке выборок:

$$\mathbf{W}_{k+1} = \mathbf{W} - \mu \nabla_k, \quad (10.19)$$

где $\mathbf{W}_k$ и ∇_k — векторы весовых коэффициентов и действительного градиента соответственно в k -й момент выборки; параметр μ определяет устойчивость и скорость сходимости.

Алгоритм быстрейшего спуска в формуле (10.19) по-прежнему требует знания $\mathbf{R}$ и $\mathbf{P}$, поскольку эти величины необходимы для получения ∇_k (см. формулу (10.18)). Алгоритм наименьших квадратов — это практический метод оценки весовых коэффициентов фильтра $\mathbf{W}_k$ в реальном времени без обращения матриц (см. формулу (10.18)) или прямого вычисления автокорреляционной матрицы и вектора взаимной корреляции. Алгоритм наименьших квадратов Уидроу–Хопфа последовательного обновления весовых коэффициентов записывается так:

$$\mathbf{W}_{k+1} = \mathbf{W}_k + 2\mu e_k \mathbf{X}_k, \quad (10.20)$$

где

$$e_k = y_k - \mathbf{W}^T \mathbf{X}_k.$$

Очевидно, что алгоритм наименьших квадратов не требует априорных знаний статистик сигнала (т.е. корреляции $\mathbf{R}$ и $\mathbf{P}$), а использует их мгновенные оценки (см. пример 10.3). Весовые коэффициенты, полученные с помощью алгоритма наименьших квадратов, — это только оценки, которые постепенно улучшаются по мере того, как настраиваются коэффициенты и фильтр изучает характеристики сигналов. В конце концов весовые коэффициенты сходятся к каким-то значениям. Условие сходимости:

$$0 < \mu < 1/\lambda_{\max}, \quad (10.21)$$

где $\lambda_{\max}$ — максимальное собственное значение ковариационной матрицы данных. На практике $\mathbf{W}_k$ никогда не достигает теоретического максимума (решение Винера), а флуктуирует около этого значения (рис. 10.7).

10.4.1. Реализация стандартного алгоритма наименьших квадратов

Ниже приводится схема вычислений, принятая в алгоритме наименьших квадратов.

1. Изначально каждому весовому коэффициенту $w_k(i)$, $i = 0, 1, \dots, N-1$ присваивается произвольное фиксированное значение, например, 0.

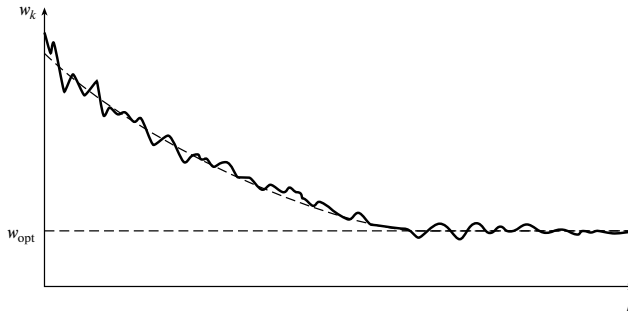


Рис. 10.7. Изменение весовых коэффициентов фильтра

В каждый последующий момент выборки $k = 1, 2, \dots$ выполняются этапы 2–4.

2. Вычислить выход фильтра

$$\hat{n}_k = \sum_{i=0}^{N-1} w_k(i) x_{k-i}.$$

3. Вычислить оценку ошибки

$$e_k = y_k - \hat{n}_k.$$

4. Обновить набор весовых коэффициентов

$$w_{k+1}(i) = w_k(i) + 2\mu e_k x_{k-i}.$$

Простота алгоритма наименьших квадратов и легкость реализации, очевидные из приведенных выше выкладок, объясняют, почему данный метод используется в системах реального времени, когда это возможно. Описанный алгоритм требует примерно $2N + 1$ операций умножения и $2N + 1$ операций сложения для каждого нового набора входных и выходных выборок. Основные операции умножения-накопления, фигурирующие в алгоритме, можно выполнять на большинстве процессоров сигналов, что наводит на мысль использовать прямую реализацию алгоритма.

Функциональная схема алгоритма наименьших квадратов показана на рис. 10.8. На рис. 10.9 и 10.10 соответственно изображены псевдокоды программной и аппаратной реализаций. В приложении к этой главе приведена реализация алгоритма наименьших квадратов в виде программы на языке С.

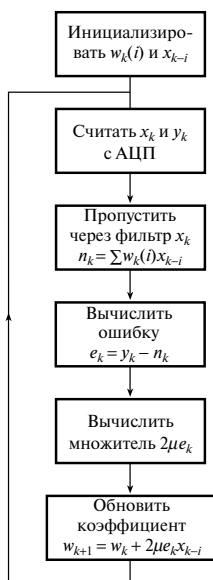


Рис. 10.8. Функциональная схема адаптивной фильтрации по схеме наименьших квадратов

Вход: $x_k(i)$ вектор последних входных выборок
 y_k текущая зашумленная выборка сигнала
 $w_k(i)$ вектор коэффициентов фильтра
 Выход: e_k текущая желаемая выходная выборка (или ошибка)
 $w_k(i)$ вектор обновленных коэффициентов фильтра

/*расчет текущей оценки ошибки*/

```

ek=yk
for i=1 to N do
    ek=ek-xk(i)*wk(i)
end
  
```

/*обновление коэффициентов фильтра*/

```

gk=2u*ek
for i=1 to N do
    wk(i)=wk(i)+xk(i)*gk
end

return
  
```

Рис. 10.9. Код реализации адаптивной фильтрации по схеме наименьших квадратов

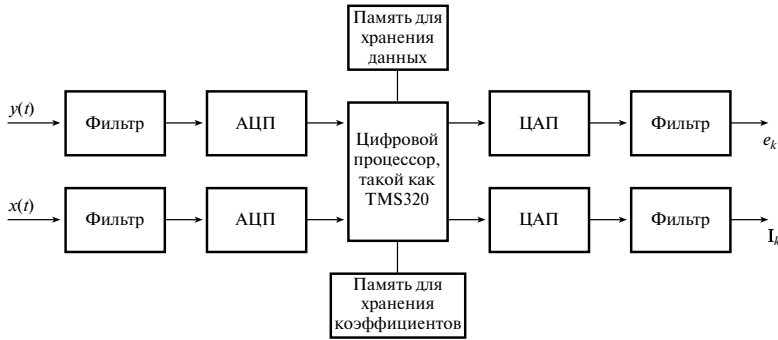


Рис. 10.10. Аппаратная реализация адаптивной фильтрации реального времени по схеме наименьших квадратов

Пример 10.3

Начав с алгоритма быстрейшего спуска

$$\mathbf{W}_{k+1} = \mathbf{W}_k - \mu \nabla_k,$$

где $\mathbf{W}_k$ — вектор весовых коэффициентов фильтра в момент k -й выборки, параметр μ определяет устойчивость и скорость сходимости, ∇_k — действительный градиент на поверхности производительности, выведите алгоритм наименьших квадратов Уидроу–Хопфа для адаптивного шумоподавления, указав сделанные допущения.

Решение

Алгоритм быстрейшего спуска формулируется так:

$$\mathbf{W}_{k+1} = \mathbf{W}_k - \mu \nabla_k. \quad (10.22)$$

Вектор-градиент ∇ , взаимная корреляция первичного и вторичного входов $\mathbf{P}$ и автокорреляция первичного входа $\mathbf{R}$ связаны соотношением

$$\nabla = -2\mathbf{P} + 2\mathbf{R}\mathbf{W}. \quad (10.23)$$

В алгоритме наименьших квадратов при вычислении ∇ используются мгновенные значения. Следовательно,

$$\begin{aligned} \nabla_k &= -2\mathbf{P}_k + 2\mathbf{R}_k\mathbf{W}_k = -2\mathbf{X}_k y_k + 2\mathbf{X}_k \mathbf{X}_k^T \mathbf{W}_k = \\ &= -2\mathbf{X}_k (y_k - \mathbf{X}_k^T \mathbf{W}_k) = -2e_k \mathbf{X}_k, \end{aligned} \quad (10.24)$$

где

$$e_k = y_k - \mathbf{X}_k^T \mathbf{W}_k.$$

Подставляя уравнение (10.24) в выражение для алгоритма быстрой спуска, получаем стандартный алгоритм наименьших квадратов Уидроу–Хопфа:

$$\mathbf{W}_{k+1} = \mathbf{W}_k + 2\mu e_k \mathbf{X}_k, \quad (10.25, a)$$

где

$$e_k = y_k - \mathbf{W}_k^T \mathbf{X}_k. \quad (10.25, б)$$

10.4.2. Практические ограничения стандартного алгоритма наименьших квадратов

Применение стандартного алгоритма наименьших квадратов сопряжено с некоторыми проблемами, что приводит к снижению производительности системы. Часть наиболее существенных проблем рассмотрена в данном разделе.

10.4.2.1. Влияние нестационарности

В стационарной среде поверхность производительности фильтра имеет постоянную форму и ориентацию, а характеристики адаптивного фильтра просто сходятся к оптимальному значению или колеблются около него. Если статистики сигнала меняются после схождения весовых коэффициентов к оптимальному значению, характеристика фильтра должна также перейти в новый набор оптимальных значений путем повторной настройки весовых коэффициентов. При этом считается, что изменение статистик сигнала происходит достаточно медленно, чтобы коэффициенты успевали сходить к оптимальному значению между изменениями. Тем не менее в нестационарной среде точка минимума непрерывно движется, кроме того, могут меняться ориентация и кривизна траектории ее движения (см. рис. 10.11). Значит, в этом случае алгоритм должен не только найти точку минимума поверхности, но и отследить ее меняющееся положение, что существенно снижает производительность схемы. Отметим, что переменная называется *нестационарной*, если ее статистики (такие как среднее, дисперсия, автокорреляция) меняются со временем. Причиной таких изменений могут быть, например, случайные помехи малой продолжительности (рис. 10.12) или неверные данные, причем такие изменения часто приводят к расстройке весовых коэффициентов фильтра¹.

В настоящее время разработано несколько схем, в которых данная проблема исчезает, но обычно это усложняет стандартный алгоритм наименьших квадратов. Одна из таких схем — упорядоченный во времени адаптивный фильтр (согласно [3]).

10.4.2.2. Влияние компонентов сигнала на помехи во входном канале

Производительность алгоритма зависит от измеренного сигнала помехи $x_k(i)$, который сильно коррелирует с реальной помехой, но слабо коррелирует (теоретически

¹ В этом случае особенно ярко проявляются недостатки линейных фильтров. — Прим. ред.

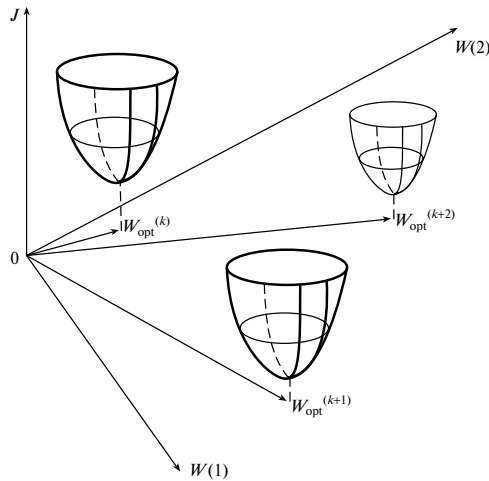


Рис. 10.11. Нестационарная поверхность производительности

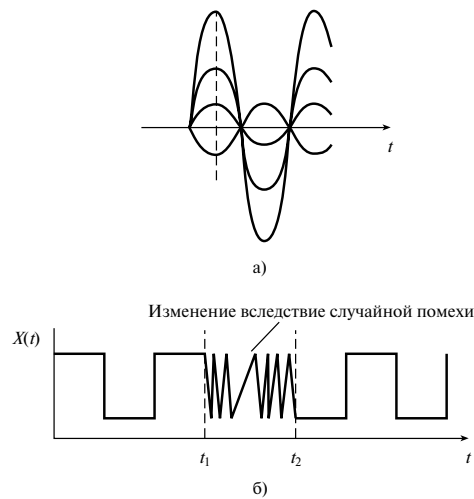


Рис. 10.12. Иллюстрация нестационарных процессов: а) модулированный сигнал; б) спорадическая интерференция

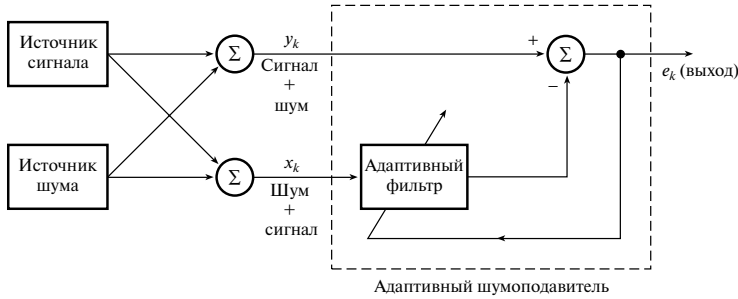


Рис. 10.13. Адаптивное шумоподавление при наличии компонентов сигнала в канале помехи и канале полезного сигнала

совсем не коррелирует) с полезным сигналом. В большинстве случаев данное условие не выполняется. В некоторых приложениях зашумленный входной сигнал может включать и нежелательную помеху, и слабые компоненты сигнала. В результате вместе с шумом подавляются некоторые компоненты полезного сигнала. Подобная ситуация иллюстрируется на рис. 10.13. В работе [11] показано, что в этих случаях процесс адаптивного шумоподавления по-прежнему ведет к значительному улучшению отношения полезный сигнал-шум, но только за счет небольшого искажения сигнала. Впрочем, если x_k содержит только сигналы и совсем не содержит шумовых компонентов, полезный сигнал в y_k может быть полностью уничтожен. Названные результаты подтверждает работа авторов в области биомедицинской обработки сигналов [5].

10.4.2.3. Требования к длине компьютерного представления

Адаптивный КИХ-фильтр, основанный на методе наименьших квадратов, характеризуется следующими уравнениями:

$$\text{цифровой фильтр} \quad \hat{n}_k = \sum_{i=0}^{N-1} w_k(i) x_{k-i}; \quad (10.26, a)$$

$$\text{адаптивный алгоритм} \quad \mathbf{W}_{k+1} = \mathbf{W}_k + 2\mu e_k \mathbf{X}_k, \quad (10.26, б)$$

где

$$e_k = y_k - \mathbf{W}_k^T \mathbf{X}_k.$$

Когда адаптивные фильтры реализуются на практике, весовые коэффициенты фильтра w_k и входные переменные x_k и y_k неизбежно представляются конечным числом битов. Таким образом, необходимые численные операции выполняются с использованием арифметики конечной точности. Рекурсивная природа алгоритма наименьших квадратов означает, что длина слова будет расти без ограничений и что некоторые биты придется отбрасывать перед сохранением каждого обновленного весового коэффициента.

Следовательно, y_k , e_k и $w_k(i)$ могут значительно отличаться от их истинных значений. Использование весовых коэффициентов фильтра и результатов арифметических операций с ограниченной точностью может вводить ошибки в адаптивные фильтры, влияние которых проявляется так: 1) возможная несходимость адаптивного фильтра к оптимальному решению, что сделает его малоэффективным (например, если фильтр используется как шумоподаватель, возможен некоторый остаточный уровень помех); 2) выход фильтра может содержать шум, которые приведет к случайным колебаниям выхода; 3) возможно преждевременное завершение алгоритма. Следовательно, чтобы удерживать эти ошибки в пределах допустимого, нужно использовать достаточное число битов. В большинстве адаптивных систем, описанных в имеющейся литературе, цифровые сигналы x_{k-i} и y_k представляются как числа с фиксированной запятой, имеющие длину от 8 до 16 бит, а квантованные коэффициенты имеют длину от 16 до 24 бит. Размер умножителей меняется в пределах от 8×8 бит до 24×16 бит, а размер накопителей — от 16 до 40 бит. Похоже, что для фильтров небольших порядков (до 100 коэффициентов) достаточно хранить коэффициенты не более чем с 16-битовой точностью и использовать умножитель 16×16 бит с накопителем длиной 32 бит.

10.4.2.4. Уход коэффициентов

При наличии входов определенных типов (например, узкополосных сигналов) коэффициенты фильтра могут постепенно смещаться от оптимального значения и медленно расти, со временем превышая разрешенную длину слова. Данная проблема характерна для алгоритмов наименьших квадратов и в перспективе приводит к ухудшению производительности. Чтобы препятствовать уходу коэффициентов, на практике вводится коэффициент потерь, который слегка смещает коэффициенты к нулю. Два примера подобных схем приведены ниже.

$$w_{k+1}(i) = \delta w_k(i) + 2\mu e_k x_{k-i} \quad 0 < \delta < 1 \quad (10.27, a)$$

$$w_{k+1}(i) = w_k(i) + 2\mu e_k x_{k-i} \pm \delta \quad 0 < \delta < 1 \quad (10.27, б)$$

Небольшой коэффициент потерь δ гарантирует, что уход коэффициентов присутствует, но вводит в ошибку e_k псевдослучайный шум.

Как упоминалось ранее, эффективность стандартного алгоритма наименьших квадратов можно увеличить, несколько его модифицировав. Назовем некоторые улучшенные версии схемы наименьших квадратов.

1. Комплексный алгоритм наименьших квадратов, разрешающий обрабатывать комплексные данные.
2. Блочный алгоритм наименьших квадратов, имеющий значительное вычислительное преимущество и в некоторых случаях более быструю сходимость.
3. Алгоритмы наименьших квадратов с упорядочением во времени, позволяющие решать некоторые нестационарные задачи.

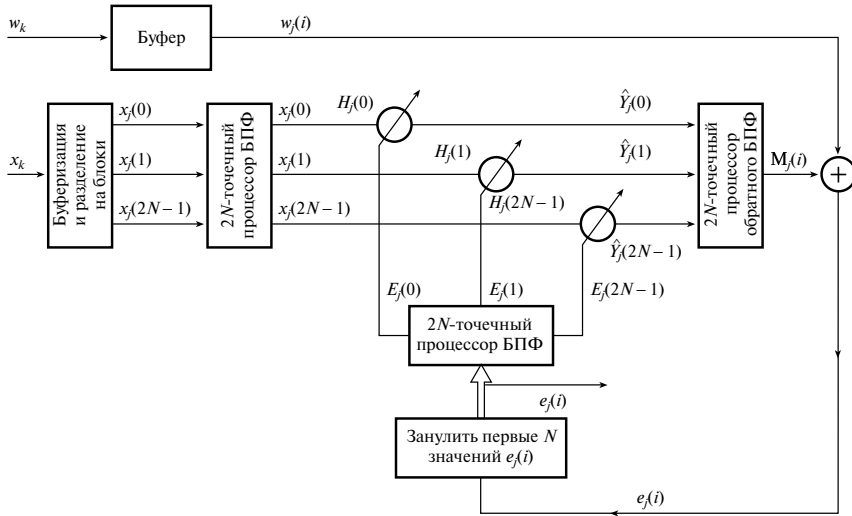


Рис. 10.14. Упрощенная блок-схема фильтра, построенного на основе схемы наименьших квадратов

10.4.3. Другие алгоритмы на основе схемы наименьших квадратов

10.4.3.1. Комплексный алгоритм наименьших квадратов

Обновление весовых коэффициентов фильтра с помощью комплексного алгоритма наименьших квадратов изложено согласно [12]:

$$\tilde{\mathbf{W}}_{k+1} = \tilde{\mathbf{W}}_k + 2\mu \tilde{e}_k \tilde{\mathbf{X}}_{k-i}, \quad (10.28)$$

где символом “тильда” обозначена комплексная переменная. Комплексный алгоритм наименьших квадратов идеально воплощается на базе процессоров PDSP16XXX (Mitel), которые могут выполнять арифметические операции непосредственно с комплексными данными, что выгодно отличает их от обычных процессоров.

10.4.3.2. Быстрые алгоритмы наименьших квадратов

Значительно сэкономить на вычислениях, особенно при большом числе коэффициентов фильтра, позволяют несколько блочных алгоритмов наименьших квадратов. Для уменьшения объема вычислений данные в таких алгоритмах обрабатываются блоками, а не отдельными выборками. Изучив представление в частотной области реализации блочного алгоритма наименьших квадратов, можно отметить, насколько выгодно при выполнении свертки использовать быстрое преобразование Фурье (БПФ) [7]. Соответствующий эффективный фильтр, представленный в частотной области, изображен на рис. 10.14.

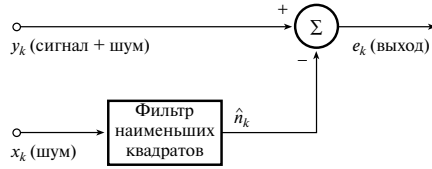


Рис. 10.15. Иллюстрация основной идеи метода наименьших квадратов

10.5. Рекурсивный алгоритм наименьших квадратов

Рекурсивный алгоритм наименьших квадратов основан на хорошо известной схеме наименьших квадратов (рис. 10.15). В ответ на набор входных сигналов $x_k(i)$, $i = 1, 2, \dots, n$ получается выходной сигнал y_k , который измеряется в дискретные моменты времени k . Входные и выходной сигналы связаны следующей регрессионной моделью:

$$y_k = \sum_{i=0}^{n-1} w(i)x_k(i) + e_k, \quad (10.29)$$

где e_k представляет ошибки измерения или другие эффекты, которые нельзя учесть, а $w(i)$ представляет долю i -го входа в первичном сигнале y_k . Задача наименьших квадратов формулируется как получение по данным $x_k(i)$ и y_k оценок величин с $w(0)$ по $w(n-1)$.

Оптимальные оценки (по схеме наименьших квадратов) весовых коэффициентов фильтра $w(i)$ определяются выражением

$$\mathbf{W}_m = [\mathbf{X}_m^T \mathbf{X}_m]^{-1} \mathbf{X}_m^T \mathbf{Y}_m, \quad (10.30)$$

где $\mathbf{Y}_m$, $\mathbf{W}_m$ и $\mathbf{X}_m$ записываются как

$$\mathbf{Y}_m = \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ \vdots \\ y_{m-1} \end{bmatrix} \quad \mathbf{X}_m = \begin{bmatrix} \mathbf{x}^T(0) \\ \mathbf{x}^T(1) \\ \mathbf{x}^T(2) \\ \vdots \\ \mathbf{x}^T(m-1) \end{bmatrix} \quad \mathbf{W}_m = \begin{bmatrix} w(0) \\ w(1) \\ w(2) \\ \vdots \\ w(n-1) \end{bmatrix}$$

$$\mathbf{x}^T(k) = [x_k(0) \quad x_k(1) \quad \dots \quad x_k(n-1)], \quad k = 0, 1, \dots, m-1.$$

Индекс m указывает, что каждая из приведенных выше матриц вычисляется для всех m информационных точек, а через T обозначен транспонированный вектор. Уравнение (10.30) определяет оптимальную оценку $\mathbf{W}_m$ по схеме наименьших квадратов, которую можно получить с помощью любого удобного метода обращения матриц. Наконец, выход фильтра записывается как

$$\hat{n}_k = \sum_{i=0}^{n-1} \hat{w}(i)x_{k-i}, \quad k = 1, 2, \dots, m. \quad (10.31)$$

10.5.1. Рекурсивный алгоритм наименьших квадратов

Вычисление $\mathbf{W}_m$ по формуле (10.30) требует трудоемкого вычисления обратной матрицы. Очевидно, что описанный выше метод наименьших квадратов не подходит для фильтрации в реальном времени. На практике при получении непрерывных данных, когда требуется улучшить оценку $\mathbf{W}_m$ с помощью новых данных, предпочтительны рекурсивные методы. При рекурсивном методе наименьших квадратов оценки $\mathbf{W}_m$ можно обновлять для каждого нового полученного набора данных без прямого повторного трудоемкого обращения матрицы.

Подходящий рекурсивный алгоритм получается, если учитывать данные с экспоненциально затухающими весовыми коэффициентами, чтобы постепенно устранить влияние старых данных на $\mathbf{W}_m$ и позволить отслеживать медленно меняющиеся характеристики сигнала. Итак,

$$\mathbf{W}_k = \mathbf{W}_{k-1} + \mathbf{G}_k e_k, \quad (10.32, a)$$

$$\mathbf{P}_k = \frac{1}{\gamma} [\mathbf{P}_{k-1} - \mathbf{G}_k \mathbf{x}^T(k) \mathbf{P}_{k-1}], \quad (10.32, б)$$

где

$$\begin{aligned} \mathbf{G}_k &= \frac{\mathbf{P}_{k-1} \mathbf{x}(k)}{\alpha_k}, \\ e_k &= y_k - \mathbf{x}^T(k) \mathbf{W}_{k-1}, \\ \alpha_k &= \gamma + \mathbf{x}^T(k) \mathbf{P}_{k-1} \mathbf{x}(k). \end{aligned}$$

По сути, введение $\mathbf{P}_k$ позволяет рекурсивно вычислять обратную матрицу $[\mathbf{X}_k^T \mathbf{X}_k]^{-1}$.

Аргумент k используется, чтобы подчеркнуть тот факт, что величины вычисляются в каждый момент получения выборки; γ называется *коэффициентом забывания*. Если положить $\gamma = 1$, приведенная схема сведется к методу наименьших квадратов. Значение γ обычно выбирается между 0,98 и 1. При меньших значениях наиболее свежим данным присваиваются слишком большие весовые коэффициенты, что приводит к сильной флуктуации оценок. Число предыдущих выборок, вносящих значительный вклад в $\mathbf{W}$ в каждый момент выборки, называется *асимптотической длиной выборки* (asymptotic sample length — ASL) и выражается следующей формулой:

$$\sum_{k=0}^{\infty} \gamma^k = \frac{1}{1 - \gamma}. \quad (10.33)$$

Данное выражение эффективно определяет память фильтра, реализованного с использованием рекурсивной схемы наименьших квадратов. При $\gamma = 1$ фильтр обладает бесконечной памятью.

10.5.2. Ограничения рекурсивного алгоритма наименьших квадратов

Число арифметических операций с выборками в рекурсивной схеме наименьших квадратов равно размерности W_k и P_k в уравнении (10.32). Впрочем, существует две основные проблемы, которые могут встретиться при непосредственной реализации рекурсивного алгоритма наименьших квадратов. Первая, именуемая “раздуванием” (“blow-up”), возникает, если сигнал $x_k(i)$ равен нулю на протяжении значительного времени, вследствие чего матрица P_k будет расти экспоненциально в результате деления на γ ($\gamma < 1$) в каждый момент получения выборки:

$$\lim_{k \rightarrow \infty} P_k = \lim_{k \rightarrow \infty} \left(\frac{P_{k-1}}{\gamma_{k-1}} \right) \quad (10.34)$$

Вторая проблема рекурсивного алгоритма наименьших квадратов — это его чувствительность к ошибкам компьютерного округления, из-за чего матрица P становится отрицательно определенной, а впоследствии — неустойчивой. Для успешной оценки W требуется, чтобы матрица P была неотрицательно определенной, что в методе наименьших квадратов эквивалентно требованию обратимости матрицы $X^T X$. Однако вследствие того, что в уравнении (10.32, б) фигурирует разность величин, положительную определенность P гарантировать нельзя. Данная проблема может еще больше испортить жизнь в многопараметрических моделях, особенно если переменные линейно зависимы и алгоритм реализован в небольшой системе с конечной длиной слова. После того как в алгоритме будет проделано много итераций, два члена в скобках в уравнении (10.32, б) могут стать почти равными. Вычитание таких членов в системе с конечной длиной слова может привести к ошибкам и тому, что матрица P_k станет отрицательно определенной.

Для решения проблемы численной неустойчивости можно соответствующим образом факторизовать матрицу P , чтобы избежать вычитания в уравнении (10.32, б). Подобные алгоритмы факторизации лучше сходятся, а их точность сравнима с точностью рекурсивного алгоритма наименьших квадратов с двойной точностью. Два таких алгоритма — это алгоритм квадратного корня и алгоритм UD-факторизации. С точки зрения памяти и вычислительной сложности UD-алгоритм эффективнее, а следовательно, предпочтительнее. Фактически UD-алгоритм — это формулировка алгоритма квадратного корня без использования квадратного корня, следовательно, свойства обоих алгоритмов одинаковы.

10.5.3. Алгоритмы факторизации

10.5.3.1. Алгоритм квадратного корня

В методе квадратного корня матрица P_k факторизуется следующим образом [8]:

$$P_k = S_k S_k^T, \quad (10.35)$$

где S_k , верхняя треугольная матрица, и S_k^T , транспонированная матрица S_k , равны квадратному корню из P_k . Следовательно, если в ходе итераций обновлять не P_k , а S_k ,

положительная определенность P_k гарантирована, поскольку произведение двух квадратных корней всегда положительно. Величина S_k обновляется следующим образом:

$$S_k = \frac{1}{\gamma^{1/2}} S_{k-1} H_{k-1}, \quad (10.36)$$

где H_k — верхняя треугольная матрица.

10.5.3.2. Алгоритм UD-факторизации

При UD-факторизации P_k факторизуется следующим образом [1]:

$$P_k = U_k D_k U_k^T,$$

где U_k — единичная верхняя треугольная матрица, U_k^T — транспонированная матрица U_k , а D_k — диагональная матрица. Следовательно, вместо обновления P_k , как в рекурсивном алгоритме наименьших квадратов, обновляются ее факторы U и D . Реализацию UD-алгоритма в форме программы на языке C можно найти на компакт-диске, прилагаемом к книге [Ifeachor, 2001] (подробности см. в предисловии).

10.6. Сфера приложения 1 — адаптивная фильтрация окулярных артефактов на ЭЭГ человека

10.6.1. Физиологическая задача

Электроэнцефалограмма (ЭЭГ) человека — это фиксация электрической активности мозга, которая содержит полезную диагностическую информацию по различным неврологическим расстройствам. Стандартные сигналы ЭЭГ поступают с электродов, помещенных на коже черепа, и часто имеют очень маленькую амплитуду порядка 20 мкВ. Сигнал ЭЭГ, подобно всем биомедицинским сигналам, очень уязвим к различным загрязнениям сигнала или артефактам, которые снижают его клиническую полезность. Например, моргание или движение глаз приводит к появлению вокруг глаз значительно-го электрического потенциала, именуемого *электроокулограммой* (electrooculogram — EOG, ЭОГ). Компонент ЭОГ, зашумляющий сигнал ЭЭГ, называется *окулярным артефактом* (ОА). Примеры измеренной ЭОГ и соответствующей зашумленной ЭЭГ приведены на рис. 10.16.

Окулярные артефакты являются главным источником сложностей, когда нужно отличить нормальную активность мозга от патологической. В некоторых случаях, например у младенцев с повреждениями головного мозга и пациентов с опухолями в лобных долях мозга, связанные патологические медленные волны в ЭЭГ и ОА отличить трудно. Сходство ОА с сигналами, представляющими клинический интерес, также затрудняет автоматизацию компьютерного анализа ЭЭГ. Вообще, неврологические расстройства часто проявляют себя в ЭЭГ как медленные волны, которые, к сожалению, не только

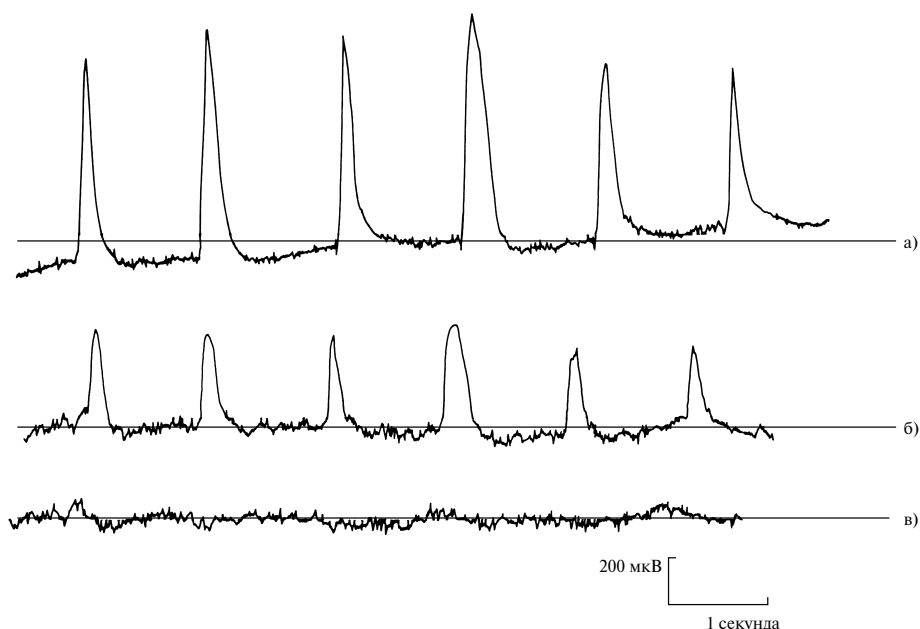


Рис. 10.16. Проблема окулярных артефактов в электроэнцефалографии: а) измеренная ЭОГ; б) соответствующий зашумленный сигнал ЭЭГ; в) сигнал ЭЭГ, скорректированный с учетом артефакта

выглядят подобно ОА, но и занимают ту же полосу частот, что и ОА. Следовательно, проблема заключается в удалении ОА при сохранении сигналов, представляющих клинический интерес.

10.6.2. Алгоритм обработки артефактов

Для обработки ОА было предложено несколько методов. В то же время, такие факторы, как требования клинических лабораторий, ограничения приложений реального времени, стоимость, случайная природа ОА и спектральное перекрытие ОА с некоторых сигналами, исходящими от церебральных источников, требуют, чтобы обработка окулярных артефактов была адаптивной и в реальном времени.

Схема адаптивной фильтрации окулярных артефактов представлена на рис. 10.17. В этом методе для получения оценок окулярных артефактов соответствующим образом масштабируются результаты ЭОГ. После этого оценки артефактов вычитаются из зашумленного сигнала ЭЭГ, и получается чистый (точнее, не содержащий окулярных артефактов) сигнал ЭЭГ. Для иллюстрации сказанного рассмотрим простую задачу ввода в одном канале ЭЭГ поправки на окулярные артефакты с использованием четырех сигналов ЭОГ (рис. 10.17, б). На основе информации, содержащейся в зашумленном сигнале ЭЭГ, y_k , и сигналах ЭОГ, с $x_k(0)$ по $x_k(3)$, вычисляется оценка окулярного артефакта $\sum_{i=0}^3 w_k(i)x_k(i)$. Затем оценка окулярного артефакта вычитается из зашумленного сигнала ЭЭГ и получается сигнал ЭЭГ без артефактов, e_k :

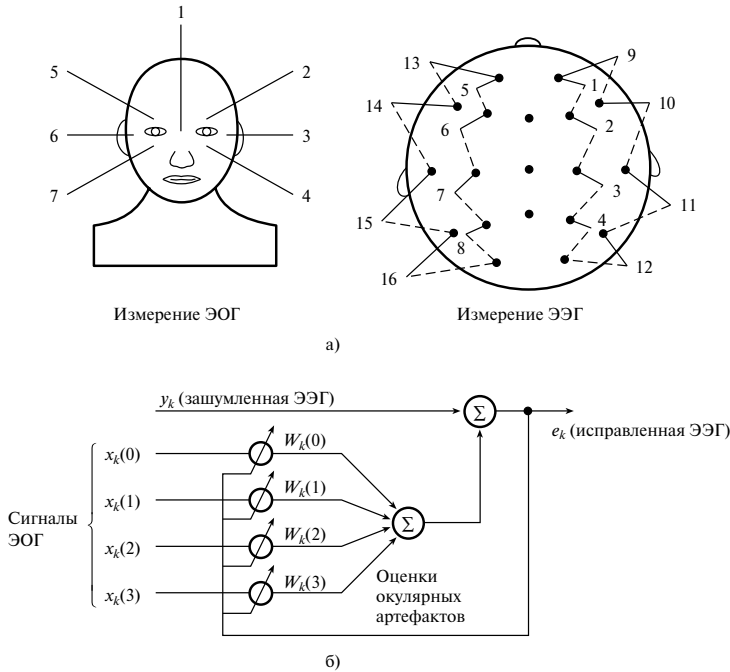


Рис. 10.17. Метод адаптивного устранения окулярных артефактов: а) возможные положения электродов при снятии электроокулограммы и электроэнцефалограммы; б) адаптивный фильтр окулярных артефактов

$$e_k = y_k - \sum_{i=0}^{n-1} w_k(i)x_k(i), \quad (10.37)$$

где $w_k(i)$, $i = 0, 1, \dots, n-1$ — коэффициенты адаптивного фильтра, которые представляют элементы сигналов ЭОГ, входящие в ЭЭГ как артефакты. Величина e_k также используется для регулировки весовых коэффициентов адаптивного фильтра с помощью подходящего рекурсивного алгоритма наименьших квадратов, что позволяет получить оптимальные оценки окулярных артефактов. Чтобы учесть изменения окулярных артефактов, вызванные, например, изменением движения глаза, нужна непрерывная настройка коэффициентов $w_k(i)$.

Алгоритм адаптивной фильтрации, используемый для устранения окулярных артефактов, — это, по сути, описанный выше UD-алгоритм. Данная численно устойчивая формулировка рекурсивного алгоритма наименьших квадратов предпочтительнее алгоритма наименьших квадратов, поскольку имеет меньшее время сходимости. Это позволяет лучше обрабатывать различные окулярные артефакты, для устранения каждого из которых требуется свой оптимальный набор весовых коэффициентов. Пример сигнала ЭЭГ, адаптивно исправленного с учетом артефактов, приведен на рис. 10.16, в.

10.6.3. Реализация в реальном времени

В работе [5] рассмотрена оперативная микропроцессорная система устранения окулярных артефактов, в которой используется описанный выше UD-алгоритм. В данной системе пользователь может выбирать из нескольких реализованных моделей. Система тестировалась на нескольких больных и здоровых людях, причем для различных категорий больных были получены хорошие результаты.

Впрочем, было определено, что если на электроды ЭЭГ и ЭОГ одновременно поступают патологические сигналы, такие как медленные волны, эпилептические пики и сложные сигналы, сигналы в исправленной ЭЭГ имеют сниженную амплитуду. Это объясняется тем, что доля вычитаемого сигнала ЭОГ зависит от степени корреляции между ЭОГ и ее компонентами в ЭЭГ, а присутствие медленных волн, похожих формой на сигналы окулярных артефактов, может привести к вычитанию той части сигнала, которая зависит и от медленных волн, и от ЭОГ. Следовательно, необходимо отличать окулярные артефакты и медленные волны, например, используя для этой цели системы с базами знаний (см. [6]).

10.7. Сфера приложения 2 — адаптивное телефонное эхоподавление

Эхо-сигналы характерны в основном для систем связи, где сигнал рассогласовывается по импедансу. Рассмотрим представленный на рис. 10.18, *а* упрощенный междугородный телефонный канал. Дифференциальная схема на АТС переводит двухпроводную линию от аппарата пользователя в четырехпроводную линию и обеспечивает отдельные маршруты для каждого направления передачи. Такая схема выбрана в основном по экономическим причинам, например, чтобы разрешить уплотнение, т.е. одновременную передачу большого числа звонков.

В идеальном случае речевой сигнал, идущий от пользователя А, проходит по верхнему тракту передачи к правой дифференциальной схеме, а затем отправляется к пользователю В, тогда как сигнал пользователя В передается по нижнему тракту к пользователю А. Гибридная сеть с каждой стороны должна обеспечивать поступление речевого сигнала удаленного пользователя в ее двухпроводный порт, а не в выходной порт. В то же время, из-за несовпадения импедансов гибридная сеть допускает, что часть входных сигналов уходит в выходной тракт и возвращается к собеседнику в виде эха. Когда телефонный звонок производится на большое расстояние (например, с помощью геостационарных спутников), эхо может запаздывать на время до 540 мс и ухудшать связь, раздражая пользователя, причем ухудшение увеличивается с расстоянием. Чтобы обойти эту проблему, в сеть вводится пара эхоподавителей, как показано на рис. 10.18, *б*.

С каждого края системы связи (рис. 10.18, *б*) входной сигнал x_k подается на гибридный и адаптивный фильтры [2]. Подавление производится за счет оценки эха с последующим ее вычитанием из ответного сигнала y_k . Основное сделанное здесь допущение — тракт возврата эхо-сигнала (через гибридный фильтр) предполагается

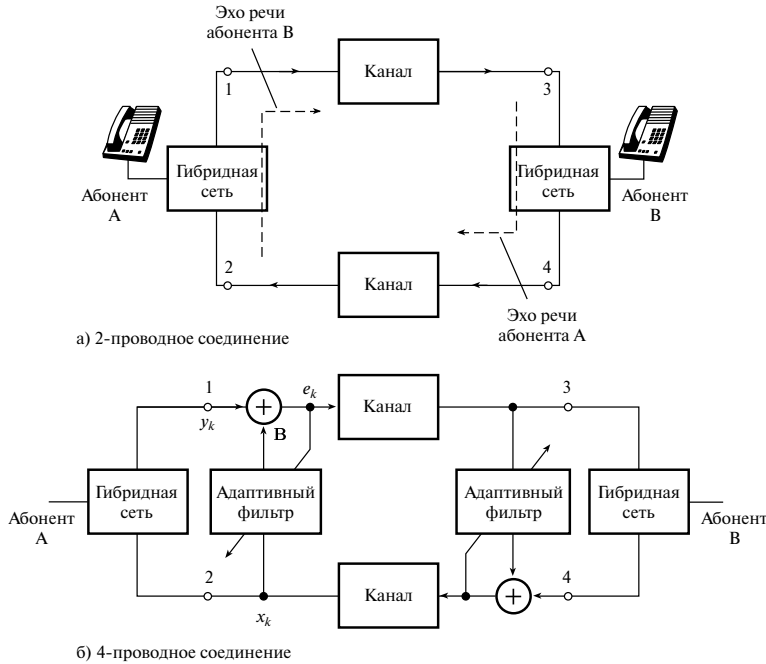


Рис. 10.18. Упрощенная схема междугородного телефонного канала (панель а); эхоподавление в голосовой междугородной телефонной связи (панель б)

линейным и инвариантным во времени. Итак, сигнал в момент k можно выразить следующим образом:

$$y_k = \sum_{i=0}^{N-1} w_k(i)x_{k-i} + s_k, \quad (10.38)$$

где x_k — выборки входного сигнала (от удаленного собеседника), s_k — речь ближнего собеседника плюс любой аддитивный шум, а w_k — импульсная характеристика тракта эхо-сигнала. Эхоподавитель находит оценку данной импульсной характеристики и выдает соответствующую оценку $\hat{y}_k = \sum w_k(i)x_{k-j}$ эхо-сигнала, которая затем вычитается из полезного ответного сигнала y_k . Экономические соображения налагают ограничения на частоту дискретизации, длину коэффициентов фильтра и входные данные, что, в свою очередь, ограничивает производительность эхоподавителя. Кроме того, существуют фундаментальные ограничения, вызванные неточной настройкой адаптивного фильтра и нелинейностями эхо-тракта.

10.8. Другие приложения

10.8.1. Громкоговорящие телефоны

- Для разделения трактов передачи и приема (т.е. отделения громкоговорителя от микрофона) используется гибридная сеть, но, поскольку громкоговоритель и микрофон расположены близко, а гибридная сеть согласована неидеально, между этими двумя устройствами существует значительная акустическая связь [9].
- Обеспечить достаточное усиление в направлении приема и передачи, не вызвав неустойчивости, довольно сложно.
- Обычно для выбора тракта приема и тракта передачи применяется переключатель с речевым управлением, но такое решение проблем неудовлетворительно, поскольку оно не допускает полнодуплексной связи.
- Для оценки и контроля за акустическим и гибридным эхом лучше всего использовать методы адаптивной фильтрации (рис. 10.19, б). В такой задаче число коэффициентов фильтра довольно велико (например, 512), что повышает привлекательность быстрого алгоритма.
- В сетях телеконференций (или системах публичных выступлений) акустическая обратная связь приводит к проблемам, подобным описанным выше. Необходимые для решения этих проблем адаптивные фильтры могут иметь значительное число коэффициентов (250–1000), особенно в помещениях с большими временами реверберации, кроме того, адаптивные алгоритмы должны быстро сходиться.

10.8.2. Компенсация многолучевого распространения

- В системе расширенного спектра каждый бит данных передается как одна из двух ортогональных псевдослучайных последовательностей, имеющих длину M . Передаваемая последовательность зависит от того, какой бит данных обрабатывается — логический 0 или 1. В приемнике находится взаимная корреляция принятой последовательности и двух последовательностей, идентичных использованным в передатчике, на основе сравнения результатов принимается решение, какой бит был передан — 0 или 1.
- Иногда сигнал может распространяться по нескольким трактам. Например, так бывает в гористой или городской местности, где возможно многолучевое распространение сигнала с участием отражения. Принятый сигнал складывается из нескольких компонентов, амплитуды и фазы которых могут отличаться (рис. 10.20). В результате это может привести к снижению производительности приемника.
- Для оценки суммарной характеристики многолучевого распространения и компенсации ее влияния используется адаптивный фильтр.

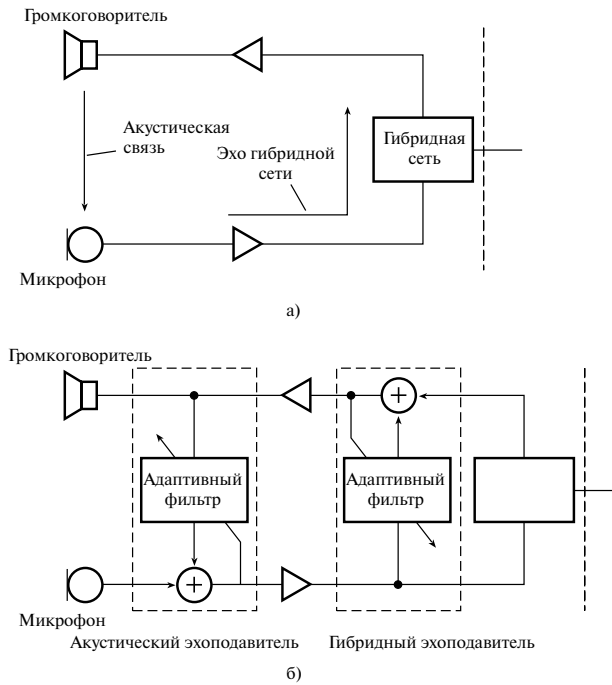


Рис. 10.19. Громкоговорящий телефон (панель а); подавление в громкоговорящем телефоне акустического эха и эха гибридной сети (панель б)

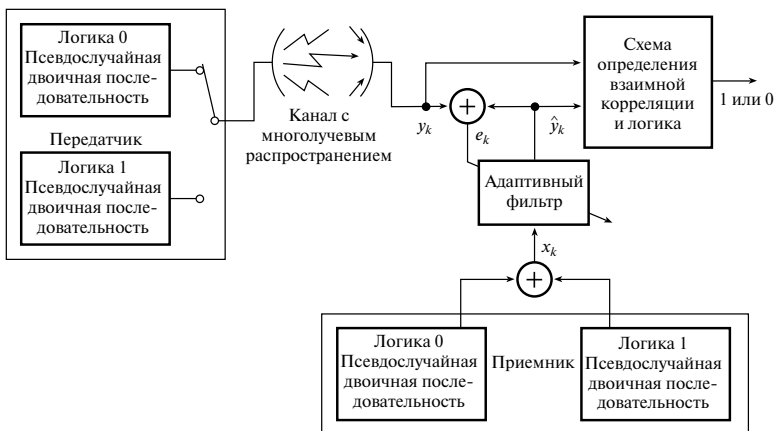


Рис. 10.20. Адаптивная система связи расширенного спектра с компенсацией многолучевого распространения

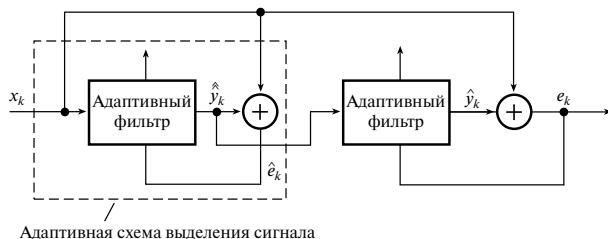


Рис. 10.21. Подавление помехи в приемнике DS/SS

10.8.3. Адаптивное подавление преднамеренной помехи

- Чтобы повысить эффективность приемника в системе расширения спектра методом прямой последовательности (direct sequence spread spectrum — DS/SS) часто требуется подавить преднамеренную помеху. С этой целью можно использовать адаптивную фильтрацию (рис. 10.21). В подобной системе используется тот факт, что станция-постановщик помех передает сильно коррелированный сигнал, тогда как псевдослучайный код имеет слабую корреляцию. Следовательно, выход фильтра y_k — это оценка помехи. Значит, вычитая ее из принятого сигнала x_k , получаем оценку сигнала расширенного спектра.
- Для повышения производительности системы используется двухкаскадный подавитель помех. Влияние конечной корреляции, которое приводит к частичному подавлению полезного сигнала, нейтрализует адаптивный линейный селективный фильтр (по сути, еще один адаптивный фильтр). Для обоих фильтров требуется умеренное число коэффициентов (порядка 16), но частота дискретизации может значительно превышать 400 кГц.

10.8.4. Обработка сигналов с радаров

Методы адаптивной обработки сигналов позволяют решить множество проблем, связанных с раарами. Например, адаптивные фильтры применяются в моностатических РЛС для удаления или подавления шумовых компонентов в полезном сигнале. В высокочастотных поверхностных раарах адаптивные фильтры используются для уменьшения внутриканальных помех, которые являются основной проблемой при работе в высокочастотном диапазоне.

10.8.5. Отделение сигналов речи от фонового шума

При обработке речи серьезную проблему представляет собой фоновый шум. Для улучшения производительности систем передачи речи в зашумленной среде (например, в самолетах-истребителях, танках или автомобилях) можно использовать адаптивный фильтр, что повысит разборчивость речи и облегчит ее распознавание.

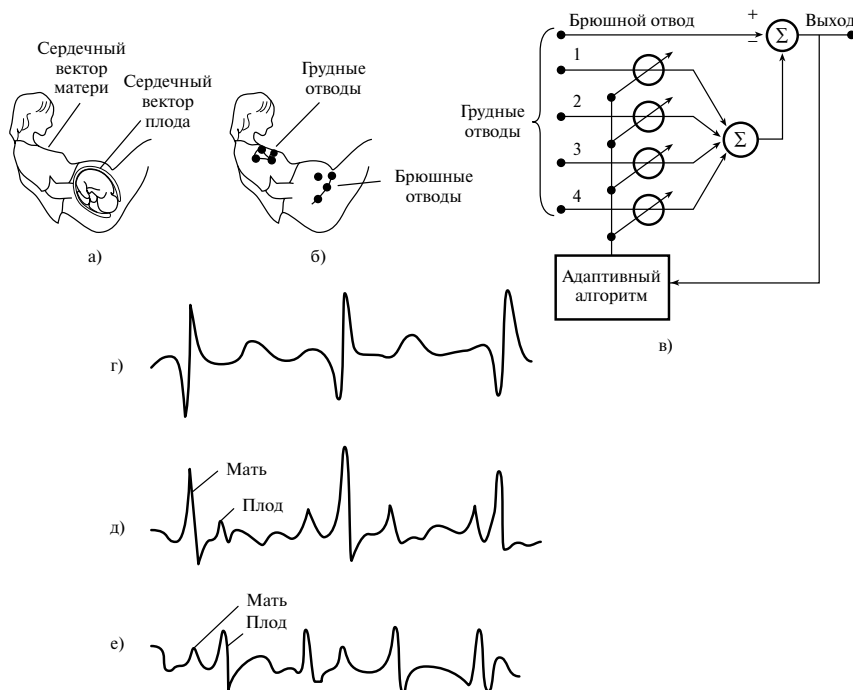


Рис. 10.22. Адаптивное подавление материнской ЭКГ на ЭКГ плода (согласно [11]): а) векторы электрического поля сердца матери и плода; б) расположение отведений электрокардиограммы; в) адаптивная схема; г) идеализированная ЭКГ матери (грудные отведения электрокардиограммы); д) идеализированная зашумленная ЭКГ плода (брюшные отведения); е) выход шумоподавителя с ослабленной ЭКГ матери

10.8.6. Наблюдение за плодом — подавление ЭКГ матери при родах

- При оценке состояния ребенка до и после родов весьма полезна информация, полученная из электрокардиограммы (ЭКГ) плода, например, сердечный ритм плода.
- ЭКГ, полученная с электродов, размещенных на животе матери, загрязнена значительно более сильными фоновыми шумами (например, вызванными мускульной деятельностью и шевелением плода), а также собственной ЭКГ матери.
- Для получения “чистой” ЭКГ плода используются адаптивные фильтры; основная концепция этого процесса иллюстрируется на рис. 10.22.
- Четыре грудных отвода электрокардиограммы используются для детектирования ЭКГ ребенка, а один или несколько отводов для детектирования суммарной ЭКГ матери и ребенка. Как показано на рисунке, для подавления сигналов пульсации сердца матери используется четырехканальный адаптивный фильтр (32 коэффициента на канал).

Задачи

- 10.1. Докажите, что в перечисленных областях выгоднее использовать не обычные фильтры, а адаптивные:

- 1) удаление окулярных артефактов с ЭЭГ человека;
- 2) эхоподавление в междугородной телефонии;
- 3) подавление активной помехи в системах связи расширенного спектра.

Начав с алгоритма быстрейшего спуска,

$$\mathbf{W}_{k+1} = \mathbf{W}_k - \mu \nabla,$$

где $\mathbf{W}_k$ — вектор весовых коэффициентов фильтра в дискретный момент времени k , параметр μ отвечает за устойчивость и скорость сходимости, а ∇_k — вектор истинного градиента поверхности производительности в дискретный момент времени k , выведите алгоритм наименьших квадратов Уидроу–Хопфа для адаптивного шумоподавления. Укажите сделанные предположения. Прокомментируйте практическую значимость алгоритма наименьших квадратов.

Прокомментируйте два главных ограничения алгоритма наименьших квадратов и то, как они снижают эффективность алгоритма. Предложите пути устранения этих недостатков.

- 10.2. Выход адаптивного шумоподавителя записывается следующим образом:

$$e_k = y_k - \mathbf{X}_k^T \mathbf{W},$$

где $\mathbf{W}$ — вектор весовых коэффициентов фильтра, а другие переменные имеют обычное значение. Начав с этого уравнения, выведите

- 1) дискретное уравнение Винера–Хопфа;
- 2) стандартный алгоритм наименьших квадратов.

Разрешается делать оправданные допущения.

- 10.3. Покажите, что адаптивный фильтр самовыключается при отсутствии корреляции между помехой x_k и зашумленным сигналом y_k .
- 10.4. С помощью блок-схемы кратко объясните основные концепции адаптивного шумоподавления. Критично обсудите преимущества и ограничения адаптивного шумоподавления в выбранном приложении реального времени и объясните, как названные ограничения можно обойти.

Литература

1. Bierman G. J. (1976) Measurement updating using the UD factorization. *Automatica*, **12**, 375–382.
2. Duttweiler D. L. (1978) A twelve-channel digital echo canceler. *IEEE Trans. Communications*, **26**, 647–653.
3. Ferrara E. R. and Widrow B. (1981) The time-sequenced adaptive filter. *IEEE Trans. Acoustics, Speech and Signal Processing*, **29**(3), 766–770.
4. Haykin S. (1986) *Adaptive Filter Theory*. Englewood Cliffs NJ: Prentice-Hall.
5. Ifeachor E. C., Jervis B. W., Morris E. L., Allen E. M. and Hudson N. R. (1986) A new microcomputer-based on-line ocular artefact removal (OAR) system. *IEEE Proc.*, **133**, 291–300.
6. Ifeachor E. C., Hellyar M. T., Mapps D. J. and Allen E. M. (1990) Knowledge based enhancement of EEG signals. *Proc. IEEE (Part F)*, **137**(5), 302–310.
7. Mansour D. and Gray A. H. (1982) Unconstrained frequency domain adaptive filter. *IEEE Trans. Acoustics, Speech and Signal Processing*, **30**, 726–734.
8. Peterka P. (1975) A square root filter for real-time multivariate regression. *Kybernetika*, **11**, 53–67.
9. South C. R., Hoppitt C. E. and Lewis A. V. (1979) Adaptive filters to improve loudspeaker telephone. *Electronics Lett*, **15**, 673, 674.
10. Widrow B. and Winter R. (1988) Neural nets for adaptive filtering and adaptive pattern recognition. *IEEE Computer*, 25–30.
11. Widrow B., Glover J. R., McCool J. M., Kaunitz J., Williams C. S., Hearn R. H., Zeidler J. R., Dong E. and Goodlin R. C. (1975a) Adaptive noise cancelling: principles and applications. *Proc. IEEE*, **63**, 1692–1716.
12. Widrow B., McCool J. M. and Ball M. (1975b) The complex LMS algorithm. *Proc. IEEE*, **719**, 720.

Дополнительная литература

- Clark G. A., Mitra S.K. and Parker S. R. (1981) Block implementation of adaptive digital filters. *IEEE Trans. Acoustics, Speech and Signal Processing*, **29**, 744–752.
- Clark G. A., Parker S. R. and Mitra S. K. (1983) A unified approach to time- and frequency-domain realization of FIR adaptive digital filters. *IEEE Trans. Acoustics, Speech and Signal Processing*, **31**, 1073–1083.
- Cowan C. F. N. and Grant P. M. (eds) (1985) *Adaptive Filters*. Englewood Cliffs NJ: Prentice-Hall.
- De Courville M. and Duhamel P. (1995) Adaptive filtering in subbands using a weighted criterion. In *Proc. ICASSP*, Detroit, MI, Vol. 2, pp. 985–988.
- Dentino M., McCool J. M. and Widrow B. (1978) Adaptive filtering in the frequency domain. *Proc. IEEE*, **66**, 1658, 1659.
- Dudek M. T. and Robinson J. M. (1981) A new adaptive circuit for spectrally efficient digital microwave-radio-relay systems. *Electronics and Power*, 397–401.
- Falconer D. D. (1982) Adaptive reference echo cancellation. *IEEE Trans. Communications*, **30**, 2083–2094.
- Ferrara E. R. and Widrow B. (1981) Multichannel adaptive filtering for signal enhancement. *IEEE Trans. Acoustics, Speech and Signal Processing*, **29**, 766–770.
- Gilloire A. and Vetterli M. (1992) Adaptive filtering in subbands with critical sampling: analysis, experiments, and applications to acoustic echo cancellation. *IEEE Trans. Circuits and Systems*, **40**, 1862–1875.
- Harrison W. A., Lim J. S. and Singer E. (1986) A new application of adaptive noise cancellation. *IEEE Trans. Acoustics, Speech and Signal Processing*, **34**, 21–27.
- Holte N. and Stueflotten S. (1981) A new digital echo canceler for two-wire subscriber lines. *IEEE Trans. Communications*, **29**, 1573–1581.
- Lappage R., Clarke J., Palma G. W. R. and Huizing A. G. (1987) The Byson research radar. In *International Conf. Radar 87*, October 1987, London: IEE, 453–461.

- Levin M. D. and Cowan C. F. N. (1994) The performance of eight recursive least squares adaptive filtering algorithms in a limited precision environment. In *Proc. European Signal Processing Conf.*, Edinburgh, pp. 1261–1264.
- Lewis A. (1992) Adaptive filtering applications in telephony. *BT Technology J.*, **10**, 49–63.
- Li Y. and Ding Z. (1995) Convergence analysis of finite length blind adaptive equalizers. *IEEE Trans. Signal Processing*, **43**, 2120–2129.
- Macchi O. (1995) *Adaptive Processing: The LMS Approach with Applications in Transmission*. New York: Wiley.
- Messerschmitt D. G. (1984) Echo cancellation in speech and data transmission. *IEEE J. Selected Areas in Communications*, **2**, 283–297.
- Mikhael W. B. and Wu F. H. (1987) Fast algorithms for block FIR adaptive digital filtering. *IEEE Trans. Circuits and Systems*, **34**, 1152–1160.
- Mueller K. H. (1976) A new digital echo canceler for two-wire full-duplex data transmission. *IEEE Trans. Communications*, **24**, 956–962.
- Ochia K., Araseki T. and Ogihara T. (1977) *IEEE Trans. Communications*, **25**, 589–594.
- Ogue J. C., Saito T. and Hoshiko Y. (1983) A fast convergence frequency domain adaptive filter. *IEEE Trans. Acoustics, Speech and Signal Processing*, **31**, 1312–1314.
- Reed F. A., Feintuch P. L. and Bershad N. J. (1985) The application of the frequency domain LMS adaptive filter to split array bearing estimation with a sinusoidal signal. *IEEE Trans. Acoustics, Speech and Signal Processing*, **33**, 61–69.
- Saulnier G. J., Das P. K. and Milstein L. (1985) An adaptive digital suppression filter for direct sequence spread spectrum communications. *IEEE J. Selected Areas in Communications*, **3**, 676–686.
- Sethares W. A., Lawrence D. A., Johnson C. R. and Bitmead R. R. (1986) Parameter drift in LMS adaptive filters. *IEEE Trans. Acoustics, Speech and Signal Processing*, **34**, 868–879.
- Sondhi M. M. and Berkley D. A. (1980) Silencing echoes on the telephone network. *Proc. IEEE*, **68**, 948–963.
- Tao Y. G., Kolwicz K. D., Gritton C. W. K. and Duttweiler D. L. (1986) A cascable VLSI echo canceller. *IEEE Trans. Acoustics, Speech and Signal Processing*, **34**, 297–303.
- Thornton C. L. and Bierman G. J. (1978) Filtering and error analysis via the UDU covariance factorization. *IEEE Trans. Automatic Control*, **23**, 901–907.
- Widrow B. (1966) *Adaptive Filters 1: Fundamentals*. Report SU-SEL-66-126, Stanford Electronics Laboratory, Stanford University, CA.
- Widrow B. (1971) Adaptive filters. In *Aspects of Network and System Theory* (Kalman R. and DeClaris N. (eds)), pp. 563–587. New York: Holt, Rinehart and Winston.
- Widrow B. (1976) Stationary and nonstationary learning characteristics of the LMS adaptive filter. *Proc. IEEE*, **64**, 1151–1162.
- Widrow B. and Stearns S. D. (1985) *Adaptive Signal Processing*. Englewood Cliffs NJ: Prentice-Hall.
- Widrow B., Mantey P., Griffiths L. and Goode B. (1967) Adaptive antenna systems. *Proc. IEEE*, **55**, 2143–2159.

Приложения

10.A. Программа на С для адаптивной фильтрации

На компакт-диске к книге [Ifeachor, 2001] имеются следующие программы, в которых реализованы адаптивные алгоритмы:

- 1) `lmsflt.c`, алгоритм наименьших квадратов;
- 2) `uduflt.c`, UD-алгоритм;
- 3) `sqrflt.c`, алгоритм квадратного корня;
- 4) `rlsflt.c`, рекурсивный алгоритм наименьших квадратов.

Из-за ограниченного размера данной книги ниже приводится только одна из названных программ.

Программа 10А.1. Реализация на С алгоритма наименьших квадратов (lmsflt.c)

```

/*-----*/
/* реализация алгоритма наименьших квадратов */
/* */
/* manny 6.11.92 */
/* */
/* вход: */
/* x[] вектор входных данных */
/* dk текущий последний элемент входных данных */
/* w[] вектор коэффициентов */
/* */
/* выход: */
/* ek значение ошибки */
/* yk выход цифрового фильтра */
/* w[] вектор обновленных коэффициентов */
/*-----*/
double lmsflt()
{
    int i;
    double uek,yk;
    yk = 0;
    for(i=0; i<N; ++i){          /* цифровая фильтрация */
        yk=yk+w[i]*x[i];
    }
    ek=dk-yk;                    /* вычислить ошибку выхода*/
    uek=2*mu*ek;                /* обновить весовые коэффициенты*/
    for(i=0; i<N; i++){
        w[i]=w[i]+uek*x[i];
    }
    return(yk);
}

```

Для иллюстрации реализации адаптивных фильтров используем программу 10А.1 для детектирования тона в широкополосном шуме.

10.А.1. Адаптивное выделение узкополосных сигналов, спрятанных в шуме

Адаптивные фильтры часто используются для детектирования или выделения узкополосных сигналов, спрятанных в широкополосном шуме. Структура, которая часто используется с этой целью, изображена на рис. 10А.1. Она состоит из элемента задержки, обозначенного через z^{-M} , и адаптивного предсказателя. Элемент задержки удаляет корреляцию, которая может существовать между выборками шумовой составляющей. Адаптивный предсказатель — это, по сути, КИХ-фильтр с настраиваемыми коэффициентами, и его выход y_k — это выделенный узкополосный сигнал. В некоторых случаях требуется получить на выходе фильтра не сигнал y_k , а второй выход адаптивного фильтра e_k . Для оптимизации коэффициентов предсказателя $w_k(i)$ используется подходящий

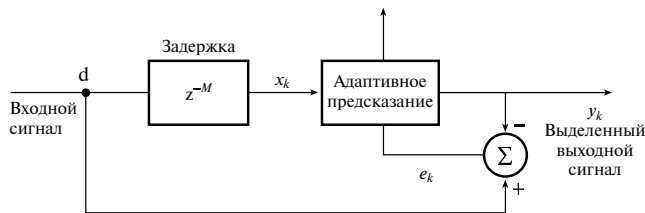


Рис. 10А.1. Адаптивное выделение сигнала

адаптивный алгоритм, в нашем случае это схема наименьших квадратов (подробности см. в разделе 10.4).

При использовании схемы наименьших квадратов адаптивный фильтр характеризуется следующими уравнениями:

$$y_k = \sum_{i=0}^{N-1} w_k(i) x_k(i), \quad (10A.1, a)$$

$$e_k = d_k - y_k, \quad (10A.1, б)$$

$$w_{k+1}(i+1) = w_k(i) + 2\mu e_k x_k(i), \quad i = 0, 1, \dots, N-1, \quad (10A.1, в)$$

где d_k — выборка зашумленного узкополосного сигнала, $x_k(i)$ — вектор входных данных, полученный из задержанных значений d_k , $w_k(i)$ — вектор коэффициентов предсказания в k -й момент выборки, μ — коэффициент устойчивости, y_k — выделенный узкополосный сигнал.

Фигурирующая в программе 10A.1 функция `lmsflt.c` представляет собой реализацию на языке C приведенных выше уравнений. Использование функции `lmsflt.c` для выделения сигнала иллюстрируется в программе 10A.2. Для моделирования задачи к синусоидальному сигналу с частотой 500 Гц был добавлен широкополосный шум, а составной сигнал был записан в формате ASCII в файле `din.dat`. Затем зашумленная синусоида была подана на адаптивный фильтр. Для моделирования адаптивной фильтрации в реальном времени входные данные считывались из файла и подавались на адаптивный фильтр по одной выборке. Отметим, что для любой длинной последовательности данных пользователю из соображений эффективности может потребоваться считывать данные блоками. Результат фильтрации с использованием схемы наименьших квадратов представлен на рис. 10A.2.

Как очевидно следует из рис. 10A.2, для использования адаптивных алгоритмов пользователю потребуется задать параметры адаптивных фильтров, например, длины КИХ-фильтра N , коэффициента задержки M и коэффициента устойчивости μ . Кроме того, следует уделить внимание формату входных данных. Например, в некоторых приложениях входные данные могут поступать от многоканального источника, когда каждый элемент сигнала $x_k(i)$ представляет элемент данных из канала. В этом случае перед применением адаптивного алгоритма массив входных данных следует надлежащим образом модифицировать.

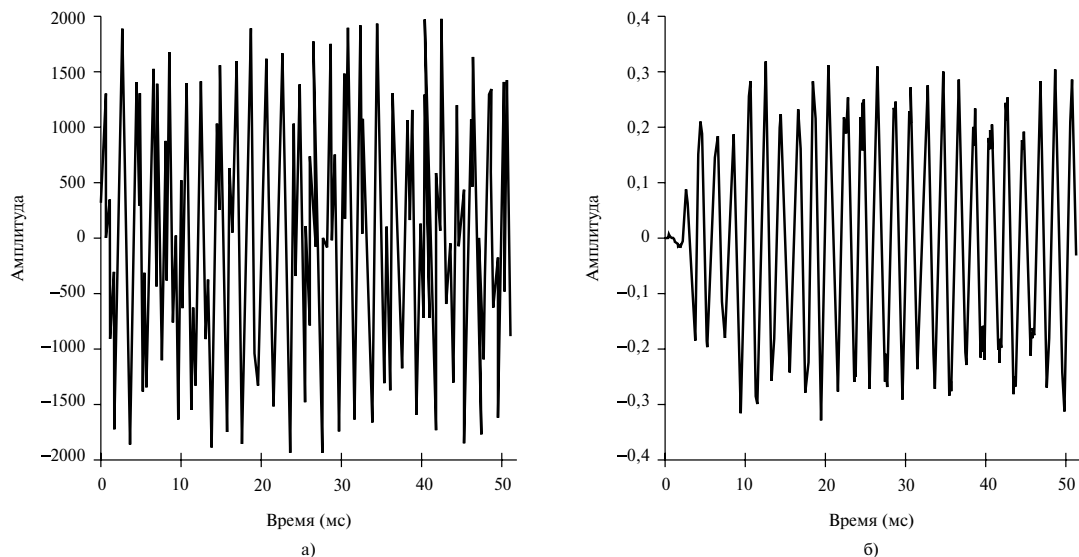


Рис. 10А.2. Адаптивное выделение узкополосного сигнала: а) зашумленный сигнал; б) выделенный сигнал

Программа 10А.2. Программа адаптивного выделения сигнала.

```

/*-----*/
/*
/* Программа иллюстрирует адаптивную фильтрацию
/* с использованием алгоритмов наименьших квадратов
/*
/* имя программы: adfilter.c
/*
/* manny, 7.11.92
/*-----*/
#include <stdio.h>
#include <math.h>
#include <dos.h>

/* определения констант */
#define N 30 /* длина фильтра */
#define M 1 /* задержка */
#define w0 0 /* исходное значение коэффициентов адаптивного */
/* фильтра */

#define npt N+M
#define SF 2048 /* коэффициент уменьшения выборок - предполагается */
/* использование 11-битового АЦП */

#define mu 0.04
double lmsflt();
void initlms();
void update_data_buffers();

```

```

void    initfiles();
float   x[npt], d[npt], dk, ek;
double  w[npt];
FILE    *in,*out,*fopen();
char    din[30];
main()
{
    double yk, yk1;

    initfiles();
    /* адаптивный фильтр, построенный по схеме наименьших квадратов*/
    initlms();
    while(fscanf(in,"%f",&dk)!=EOF){
        dk=dk/SF;
        update__data__buffers();
        yk=lmsflt();
        yk1 =SF*yk;
        fprintf(out,"%lf \n",yk1);
    }
    fcloseall();
}
/* -----*/
void    initfiles()
{
    clrscr();
    printf("введите имя файла с данными для фильтрации \n");
    scanf("%s",din);
    printf("\n");
    printf("отфильтрованные данные записываются в файл dout.dat \n");
    if((in=fopen(din,"r"))= =NULL){
        printf("невозможно открыть файл входных данных \n");
        exit(1);
    }
    if ((out = fopen ("dout.dat", "w"))= =NULL){
        printf("невозможно открыть файл выходных данных \n");
        exit(1);
    }
    return;
}
/* -----*/

void    update__data__buffers()
{
    long  j, k;
    for(j=1; j,N; + +j){    /*обновление буфера x-данных*/
        k=N-j;
        x[k]=x[k-1];
    }
    x[0]=dk;
    if(M>0)
        x[0]=d[M-1];
}

```

```

        for(j=1; j,M; + +j){ /*обновление буфера d-данных*/
            k=M-j;
            d[k]=d[k-1];
        }
        d[0]=dk;
    }

/* -----*/
void    initlms()
{
    long i;
    for(i=0; i,npt; + +i){
        x[i]=0;
        d[i]=0;
        w[i] = w0;
    }
}
/* -----*/

#include    "Imsfit.c";

```

10.Б. Программы MATLAB для адаптивной фильтрации

MATLAB не поддерживает прямо адаптивную обработку сигналов. Однако авторы разработали две программы MATLAB, в которых реализованы основные адаптивные алгоритмы, метод наименьших квадратов и рекурсивный метод наименьших квадратов:

`lmsadf.m` — используется для выполнения стандартной адаптивной фильтрации на основе схемы наименьших квадратов;

`rlsadf.m` — применяется для выполнения стандартной адаптивной фильтрации на основе рекурсивной схемы наименьших квадратов

Обе программы можно найти в Web.

Оценка и анализ спектра

11

11.1. Введение	737
11.2. Принципы оценки спектра	740
11.3. Традиционные методы	742
11.4. Современные параметрические методы оценки	762
11.5. Авторегрессионная оценка спектра	763
11.6. Сравнение методов оценки	770
11.7. Примеры применения	770
11.8. Резюме	774
11.9. Разобранный пример	774
Задачи	776
Литература	779
Приложение	781

В данной главе рассмотрены следующие темы: основные концепции спектрального анализа, подводные камни непараметрических методов, свойства весовых функций, предварительная обработка данных и выбор весовой функции. Описываются и сравниваются непараметрические методы, в том числе метод модифицированной периодограммы, метод Блэкмена–Тьюки, основанный на применении быстрого преобразования Фурье к автокорреляционной функции данных, и метод быстрой корреляции. Описан наиболее распространенный метод параметрической оценки спектра, основанный на авторегрессионном моделировании, а также дан краткий обзор других параметрических методов. Рассмотрены следующие приложения: обработка электрических сигналов мозга для различения здоровых людей и пациентов с болезнями мозга и анализ электроэнцефалограммы человека на основе авторегрессионного моделирования.

11.1. Введение

Преобразование данных из временной области в частотную было описано в разделе 3.1. В данной главе развиваются принципы и навыки оценки и анализа спектра в частотной области. Вообще, графики зависимости амплитуд или фаз гармоник от частоты часто более удачно представляют данные или сигналы, особенно если сигналы имеют случайную природу. Выбирая согласно некоторому критерию определенные гармоники и отбрасывая другие, можно существенно сжать данные. Спектральный анализ оказался полезным в таких областях, как изучение сигналов техники связи или смоделированных откликов человеческой электроэнцефалограммы (ЭЭГ) при диагностике болезней головного мозга [29], других биологических сигналов, метеорологических данных, в управлении технологическим процессом и измерении спектра шума при разработке оптимальных линейных фильтров.

Существующие методы оценки спектра можно разделить на параметрические и непараметрические. Непараметрические методы включают периодограммы, модифицированные периодограммы Барлета (Barlett) и Уэлша (Welsh) и метод Блэкмена–Тьюки (Blackman–Tukey). Плюс: все эти методы можно реализовать с использованием быстрого преобразования Фурье; минус: если данные представляются короткими словами, разрешение по частоте ограничено. Кроме того, чтобы получить значимые результаты, следует быть очень аккуратными. С другой стороны, параметрические методы не только вычислительно эффективнее, но и могут давать большее разрешение. Впрочем, перед оценкой спектра процесса нужно вначале построить достаточно точную модель этого процесса. Если выбран параметрический подход, чаще всего спектр выводится из параметров авторегрессионной модели сигнала (авторегрессионная оценка спектра описана в разделе 11.5).

При выполнении непараметрического спектрального анализа нужно обойти множество подводных камней. Связанные с этим темы наложения, гребешковых искажений, конечной длины данных, просачивания спектральных составляющих и размывания спектра очень важны, и их рассмотрению посвящен раздел 11.3.

Отрицательное воздействие просачивания спектральных составляющих и размывания можно минимизировать, ограничив данные посредством подходящей весовой функции. Дискретные данные по одной выборке множатся на дискретные значения выбранной весовой функции. Тема умножения на финитную взвешивающую функцию рассмотрена в разделе 11.3.2. Основными свойствами весовой функции являются ширина полосы шумового (или прямоугольного) эквивалента, коэффициент расширения спектра, потери (времени) при обработке данных в наихудшем случае и полоса минимального разрешения. Все они рассмотрены в подразделе 11.3.2.1. В разделе по корреляции перекрывающихся областей показано, что усреднение спектра из нескольких взвешенных фрагментов данных вместо непосредственного вычисления спектра всех взвешенных данных дает значительно лучшую оценку спектра. В части подраздела 11.3.2.1, посвященной смещающему эффекту взвешивания, показано, как потери энергии сигнала и смещение постоянной составляющей взвешенных данных можно устранить, обработав данные до взвешивания.

Суждения о качестве спектральных оценок основаны на теории оценок, так что в главе приводятся некоторые базовые концепции этой дисциплины. Статистическая оценка включает определение математического ожидания статистических величин, выведенных из выборок совокупности. Впрочем, при анализе с помощью временных рядов обычно доступны не выборки генеральной совокупности, взятые одновременно, а дискретные данные, полученные как функция времени. Этот момент обычно полагается несущественным, поскольку принимается эргодическая гипотеза: свойства данных временного ряда совпадают со свойствами предполагаемых выборок. Теперь самое время привести несколько статистических определений.

Среднее значение временного ряда, состоящего из элементов данных $x(n)$, $n = 0, 1, \dots, N - 1$, равно его математическому ожиданию $E[x(n)]$:

$$E[x(n)] = \frac{1}{N} \sum_{n=0}^{N-1} x(n), \quad (11.1)$$

где E — оператор математического ожидания. Дисперсия того же ряда выражается как

$$\text{var}[x(n)] = E\{[x(n) - \bar{x}(n)]^2\}. \quad (11.2)$$

Автоковариационная матрица $x(n)$ записывается следующим образом:

$$c_{xx}(m) = E\{[x(n) - \bar{x}(n)][x(n+m) - \bar{x}(n+m)]\}, \quad (11.3)$$

где через m обозначено запаздывание элементов данных, а через $\bar{x}(n)$ записано $E[x(n)]$. Предполагая конечную реализацию, спектральную плотность мощности можно записать как

$$P_E(\omega) = \sum_{m=-\infty}^{\infty} c_{xx}(m) \exp(-i\omega m). \quad (11.4)$$

Отметим, что если в отличие от бесконечно длительных стохастических процессов рассматривать сигналы конечной длительности, более правильным будет использовать спектральную плотность энергии. Спектральная плотность мощности имеет размерность $\text{В}^2\text{Гц}^{-1}$. Если оценивается статистический параметр α , смещение (отклонение) оценки определяется как разность истинного значения (взятого из генеральной совокупности) и оценки:

$$\text{смещение} = \alpha - E[\alpha]. \quad (11.5)$$

Если смещение равно нулю, оценка совпадает с истинным значением; если смещение не равно нулю, оно представляет ошибку определения α и оцененное значение α называется *смещенным*. Хорошие алгоритмы оценки смещения не допускают. Дисперсия α является мерой ширины максимума функции распределения плотности вероятности α . При небольшой дисперсии пики довольно узкие, а при дисперсии, стремящейся к нулю, оцененное значение стремится к значению из генеральной совокупности (истинному), если оценка не смещена. Если дисперсия стремится к нулю при увеличении числа элементов данных N , оценку называют *состоятельной*. Если она несостоятельна, оценки будут флуктуировать с увеличивающейся амплитудой от реализации к реализации с увеличением элементов данных. Следовательно, нужно, чтобы статистические оценки были несмещенными и состоятельными.

Особенности оценки спектра методами периодограмм рассмотрены в разделах 11.3.3 и 11.3.4. Показано, что оценки спектра, выведенные как периодограммы, несостоятельны, т.е. последовательные реализации дают флуктуирующие оценки, несмещенные только для больших наборов данных. Для получения устойчивых и более точных оценок вырезаются блоки данных, спектры которых затем усредняются. Данная идея реализована в методе модифицированных периодограмм Барлетта и Уэлша. В заключение рассмотрен непараметрический метод Блэкмена–Тьюки. Метод заключается в следующем: вначале вычисляется взвешенная автокорреляционная функция данных, и по ее БПФ получается энергетический спектр. Оценка спектра по методу Блэкмена–Тьюки характеризуется большей добротностью, чем другие методы с использованием периодограмм.

Кроме того, следует определить добротность оценок спектральной плотности мощности, что позволит сравнить различные оценки. В качестве выражения для доброт-

ности оценки было предложено (см. [44]) отношение квадрата среднего спектральной плотности мощности к ее дисперсии:

$$Q = \frac{\{E[P_E(f)]\}^2}{\text{var}[P_E(f)]}. \quad (11.6)$$

11.2. Принципы оценки спектра

В начале данного раздела рассматривается форма сигнала напряжения в зависимости от времени. Данная форма сигнала может дать полезную информацию. Например, это может быть синусоида, которую естественно характеризовать ее амплитудой, частотой и фазой. Точнее, такой сигнал можно считать состоящим из одной гармоники известной частоты с определенной амплитудой и фазой. В качестве альтернативы представления сигнала как функции напряжения от времени его можно представить двумя графиками: зависимости амплитуды от частоты и зависимости фазы от частоты. Поскольку синусоида — это единственный сигнал, который имеет одну амплитуду, одну фазу и одну частоту, на графиках амплитуды и фазы синусоиды будет расположена одна точка. С помощью анализа Фурье (см. главу 3) можно показать, что все сигналы можно математически представить как сумму синусоидальных сигналов, каждый из которых имеет свою амплитуду и фазу на заданной частоте. Следовательно, любой сигнал можно альтернативным образом представить зависимостью амплитуды от частоты и фазы от частоты. Данные графики называются *амплитудным* и *фазовым спектрами*. Данные спектры важны, поскольку они предлагают дополняющий способ представления сигнала, из которого яснее видна информация о частотном содержании сигнала. Наблюдаемые формы спектров и изменения в них весьма полезны при понимании и интерпретации сигналов. Амплитудный и фазовый спектры в свою очередь очень часто дают больше полезной информации, чем сигналы. Тема преобразования из временной области в частотную и обратно описана в главе 3. Были изложены концепции преобразования периодических сигналов в частотную область с помощью ряда Фурье и комплексного ряда Фурье. Там же было показано, что частоты синусоидальных компонентов периодического сигнала, известные как Фурье-компоненты, гармонически связаны друг с другом, т.е. каждый кратен первой гармонике f , где

$$f = 1/T_p,$$

где T_p — период повторения сигнала. Известно, что расстояние между любой парой соседних гармоник одинаково и равно $f = 1/T_p$, так что данная величина называется еще *разрешением по частоте* или *частотным разрешением*. Амплитуда амплитудного спектра измеряется в вольтах. Пример периодического сигнала импульсов напряжения представлен на рис. 3.1, *а*, а на рис. 3.1, *б* и *в* изображены соответственно амплитудный и фазовый спектры данного сигнала. О различных способах использования амплитудного и фазового спектров рассказывалось во вступлении к главе 3.

Непериодические, но непрерывные сигналы можно преобразовать из временной в частотную область с использованием преобразования Фурье, описанного в разделе 3.1.2.

Было показано, что “амплитуда” этого преобразования имеет размерность ВГц^{-1} и, если изобразить ее в зависимости от частоты, получим амплитудный спектр. Следовательно, площадь под кривой между двумя частотами характеризует “среднее” напряжение сигнала для частотных компонентов, лежащих между указанными двумя частотами. Возводя в квадрат “амплитуды” вычисленных Фурье-компонентов, получим спектральную плотность энергии сигнала в ДжГц^{-1} . Термином “спектр” часто называют графики зависимости спектральной плотности амплитуды и спектральной плотности энергии от частоты. Данные спектральные плотности для прямоугольного импульса приведены на рис. 3.2, *а* и *б*. В главе 3 также было показано, как спектры дискретных и непериодических импульсов напряжения можно получить с помощью дискретного преобразования Фурье (ДПФ). Было показано, что компоненты ДПФ гармонически связаны с первой гармонической циклической частотой $\Omega = 2\pi/(N - 1)T$, так что первая гармоника равна f , а

$$f = 1/(N - 1)T, \quad (11.7, a)$$

где N — число элементов данных, а T — интервал дискретизации. Поскольку $(N - 1)T$ равно T_p (длительности дискретного сигнала), первая гармоника равна

$$f = 1/T_p. \quad (11.7, б)$$

Как и выше, вследствие гармонической связи разрешение по частоте также равно $1/T_p$. Следовательно, чем длительнее сигнал, тем большим будет разрешение спектра.

Пример вычисления ДПФ последовательности данных $\{1, 0, 0, 1\}$ представлен в разделе 3.2. Показано, что Фурье-образ данной последовательности равен $\{2, 1 + i, 0, 1 - i\}$. Следовательно, амплитуда второй гармоники, $1 + i$, равна $\sqrt{(1^2 + 1^2)} = \sqrt{2}$. Если последовательность данных представляет дискретные уровни напряжения, амплитуда второй гармоники — $\sqrt{2}$ В, а ее энергия — $(\sqrt{2} \text{ В})$, т.е. 2 Дж. Соответствующая фаза равна $\arctg(\text{минимый компонент}/\text{действительный компонент}) = \arctg 1 = 45^\circ$. Последовательность данных представлена на рис. 3.3, *а*, а ее амплитудный и фазовый спектры — на рис. 3.3, *б* и *в* соответственно. Амплитудный спектр имеет размерность “вольт”. В разделе 3.2 показано, что ДПФ и преобразование Фурье связаны соотношением $F(i\omega) = TX(k)$. В разделе 3.5 представлен алгоритм быстрого преобразования Фурье (БПФ), позволяющий ускорить вычисление ДПФ, пример его использования (для той же последовательности $\{1, 0, 0, 1\}$) рассмотрен в разделе 3.5.1.

Если рассматриваемый сигнал длителен по сравнению с интервалом времени, в течение которого можно считать, что сигнал имеет постоянные статистические моменты, оценка спектра с большой вероятностью будет неточной. Сказанное также справедливо в том случае, когда сигнал сильно зашумлен. В таких ситуациях необходимо сгладить оцененный спектр, чтобы получить улучшенную оценку, поскольку сглаживание спектра позволяет устранить элемент случайности. Отношение сигнал-шум в случайных сигналах можно улучшить путем усреднения сигналов; при усреднении K сигналов отношение сигнал-шум улучшается в $\sqrt{K}$ раз. Итак, один метод повышения точности оценки спектра уже есть: разделить данные на K частей равной длины, определить

спектр каждой части, а затем усреднить полученные спектры. Таким образом получают среднюю амплитуду и среднюю фазу каждого гармонического частотного компонента K спектров, графическое представление которых является спектром средних амплитуды и фазы. Точность спектров можно выразить через их дисперсию. Например, чем меньше дисперсия спектральной плотности мощности, тем точнее оценка. Следовательно, важно знать, как влияет метод оценки спектра на его дисперсию. Оценка спектра путем усреднения описана в подразделе 11.3.2.1, где объясняется, что оценка спектра как среднего спектров K частей данных имеет меньшую дисперсию, чем оценка, вычисленная непосредственно, причем дисперсия уменьшается пропорционально числу частей. Даже если шумовая составляющая сигнала мала, т.е. имеем большое отношение сигнал-шум, результат, усредненный по K частям, по-прежнему значительно повышает точность этого метода модифицированных периодограмм. В то же время, разбиение данных на части приводит к тому, что при вычислении БПФ используется меньше выборок, а следовательно, оценка спектра получается более грубой. Этого недостатка можно избежать, введя дополняющие нули (подраздел 11.3.1.2). Итак, всегда нужно помнить, что требования к точности оценки и необходимой спектральной гладкости противоречивы, и добиваться наилучшего компромисса между ними. Другой подход к сглаживанию графиков спектральных функций — это вычислить его по ДПФ взвешенной автокорреляционной функции данных. Этот принцип положен в основу метода Блэкмена–Тьюки (см. раздел 11.3.5). Поскольку автокорреляционная функция данных состоит из среднего сумм произведений данных и запаздывающих копий самих себя (раздел 5.2), отношение сигнал-шум улучшается (раздел 5.2.2). Метод Блэкмена–Тьюки дает спектр с большей добротностью, чем спектр, полученный методами модифицированных периодограмм.

Взвешивающие функции также оказывают сглаживающий эффект на спектр. В частности, функции с небольшими боковыми лепестками в частотной области отфильтровывают шум, который не входит в основной лепесток, и предлагают улучшенное сглаживание. Фактически для реализации этой разновидности спектрального сглаживания нужно свернуть спектр данных со спектром выбранной взвешивающей функции.

Параметрические методы оценки спектра и другие современные методы описаны в разделе 11.4. Данные методы не настолько требуют присутствия человека, как непараметрические методы, и их можно автоматизировать. Ниже описано, как с помощью параметров модели сформировать модель данных, на основе которой по функциям частотных характеристик линейных систем можно получить спектр данных. Описаны также более современные методы спектрального анализа.

11.3. Традиционные методы

В разделе 11.3.1 уточняются различные подводные камни спектрального анализа и объясняется, как их можно избежать. Метод взвешивания и свойства весовых функций описаны в разделе 11.3.2. Некоторые свойства спектров и спектрального сглаживания рассмотрены в разделе 11.3.3.

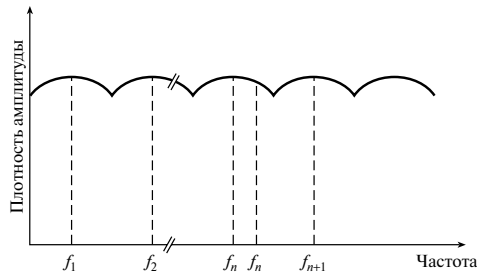


Рис. 11.1. Спектральная плотность амплитуды сигнала с равномерной спектральной плотностью

11.3.1. Подводные камни

11.3.1.1. Частота дискретизации и наложение

Выполнению спектрального анализа должна предшествовать передача аналоговых сигналов через фильтр защиты от наложения спектров, задача которого — предотвратить наложение дискретного сигнала после последующего этапа аналого-цифрового преобразования. *Наложением* называется искажение спектра сигнала при вводе паразитных (побочных) низкочастотных компонентов вследствие объединения недостаточного фильтра защиты от наложения спектров и слишком низкой частоты дискретизации. Данная тема всесторонне рассмотрена в разделе 2.2.

11.3.1.2. “Гребешковое искажение”, или “эффект частотокола”

Как объяснялось в разделе 3.2, дискретное преобразование Фурье (ДПФ) выдает гармонические амплитудные и фазовые компоненты, равномерно размещенные по шкале частот. Густота размещения спектральных линий уменьшается с увеличением длины дискретизованного сигнала. Следовательно, если компонент сигнала расположен между двумя соседними частотными гармониками спектра, представить его надлежащим образом нельзя. Энергия такого компонента будет распределена между соседними гармониками, что приведет к искажению близлежащих спектральных “амплитуд”. Спектральная плотность амплитуды равномерно распределенного сигнала показана на рис. 11.1. Отметим конечную ширину основных лепестков, центрированных на гармонических частотах, и то, что компонент сигнала на такой негармонической частоте, как f_{nh} , нельзя представить надлежащим образом. Для решения этой проблемы гармонические компоненты должны размещаться более тесно и совмещаться с частотами сигнала. Этого можно достичь, введя в реальные данные дополнительные нули. Эти нули называются *дополняющими* и нужны для увеличения точности — приближения оцененного спектра к истинному спектру без использования дополнительной информации. Итак, к N данным требуется добавить такое количество нулей N' , чтобы удовлетворялось условие

$$N + N' = 2^m \quad (11.8)$$

в алгоритме двоичного быстрого преобразования Фурье (БПФ), где m — целое. Кроме того, гармоники частоты $1/(N + N' - 1)T$, где T представляет интервал дискретизации, должны совпадать с частотами сигнала.

Чтобы представить максимальное снижение выигрыша от обработки, которое возникает на частотах, расположенных посередине между гармонически связанными частотами, определяется *гребешковое искажение* (scallop loss) (ГИ) [23]:

$$\text{ГИ} = \frac{|W(\omega_s/2N)|}{W(0)} = \frac{\left| \sum_{n=0}^{N-1} w(nT) \exp(-i\pi n/N) \right|}{\sum_{n=0}^{N-1} w(nT)}, \quad (11.9)$$

где W представляет ДПФ-образ весовой функции (подраздел 11.3.2), $\omega = 2\pi f_s$ — циклическая частота дискретизации, f — частота дискретизации, N — число элементов данных, n — номер элемента данных, $w(nT)$ — весовая функция, дискретизованная во временной области.

Как уже отмечалось, конечная длина данных ограничивает возможное разрешение по частоте до $1/(N - 1)T$ (Гц). В результате получается грубый спектр, который можно сгладить и сделать непрерывным, используя дополнительные нули. Данный процесс является просто интерполяцией спектральной кривой между соседними гармониками. Действительного улучшения разрешения можно добиться только за счет более длительной реализации. После дополнения N' нулями интервал частот между линиями спектра становится равным $1/(N + N' - 1)T$ (Гц).

11.3.1.3. Исключение тренда

Перед вычислением спектра нужно удалить все тренды в данных, поскольку вектор ошибок, из-за которого к данным добавляются тренды, будет интегрироваться и может породить большие ошибки в оцененном спектре.

11.3.1.4. Просачивание спектральных составляющих и размывание спектра

БПФ, которое применяется к набору дискретных данных, — это не совсем истинное БПФ процесса, из которого получены данные. Это объясняется тем, что процесс непрерывен, тогда как данные имеют начало и конец, которые определяются выбранной реализацией. Для эффективного получения данных, которые представляют сигнал длительностью $T_s(s)$, все дискретные значения в интервале T_s умножаются на единицу, а все данные вне этого интервала умножаются на нуль. Это эквивалентно умножению (или взвешиванию) сигнала на прямоугольный импульс (взвешивающую, или весовую, функцию) ширины T_s и высоты 1. В этом случае выборки данных $v(n)$ равны произведению элементов данных $s(n)$ и значений весовой функции $w(N)$:

$$v(n) = w(n)s(n). \quad (11.10)$$

Произведение во временной области эквивалентно свертке в частотной (см. разделы 3.3 и 5.3). Следовательно, БПФ-образ n -й гармоники равен

$$V(\omega_n) = \sum_{k=-N}^N W(\omega_n - \omega_k) S(\omega_k), \quad (11.11)$$

где ω_n — циклическая частота n -й гармоники, $V(\omega_n)$ — комплексный ДПФ-компонент на частоте ω_n , $W(\omega_n)$ — ДПФ-образ функции на частоте ω_n , $S(\omega_k)$ — действительный ДПФ-компонент сигнала на частоте ω_k .

Из уравнения (11.11) следует, что рассчитанный спектр состоит из истинного спектра данных, свернутого со спектром весовой функции. Амплитудный спектр прямоугольного импульса $S_R(\omega_n)$ записывается следующим образом (данное выражение называется также *ядром Дирихле*):

$$S_R(\omega_n) = \frac{T_s(\omega_n T_s/2)}{\omega_n T_s/2} = \text{sinc}\left(\frac{\omega_n T_s}{2}\right). \quad (11.12)$$

Здесь $\text{sinc}(\omega_n T_s/2)$ — функция выборки с частотой $\omega_n/2$ (см. подразделы 3.1.1 и 3.1.2), проиллюстрированная на рис. 3.2, *a*. Она состоит из основного лепестка и бесконечного числа боковых лепестков с максимумами на 0 Гц и $(n + 0,5)/T_s$ Гц соответственно. Теперь амплитудный спектр отдельного синусоидального компонента сигнала на частоте f_n включает в себя два импульса на частотах $\pm f_n$. Свертка с дискретизирующей функцией дает спектр, изображенный на рис. 11.2. Два импульса преобразовались в две перекрывающиеся выборочные функции. Влияние прямоугольной весовой функции проявилось во введении в вычисленный спектр побочных максимумов (из-за влияния боковых лепестков). Такой эффект будет иметь место для каждого частотного компонента сигнала, так что амплитудный спектр сигнала будет искажен из-за перекрестного сложения и вычитания большого числа боковых и главных лепестков весовых функций. Вследствие этого могут появиться ложные или скрываться истинные максимумы спектра. Данный эффект называется *просачиванием спектральных составляющих*. Для того чтобы избежать его, данные следует модифицировать — умножить на весовую функцию, форма которой предназначена для снижения влияния боковых лепестков. Подходящая функция равна 1 в центральной точке и сходит на нуль в точках $n = 0$ и $n = N - 1$. На настоящий момент разработано по крайней мере 23 такие функции, их относительная пригодность рассмотрена в работе [23].

Чтобы минимизировать просачивание спектральных составляющих, форма весовой функции выбирается с минимальными уровнями боковых лепестков. К сожалению, это приводит к увеличению ширины основного лепестка, так что он расширяется на соседние боковые лепестки (происходит *наложение*). Данный эффект имеет место для всех гармоник, а общий результат — наложение спектра сигнала, или *размывание*. Таким образом, весовые функции и их параметры надлежит выбирать тщательно, чтобы добиться оптимального баланса между разрешением по частоте и статистической точностью оценки спектра.

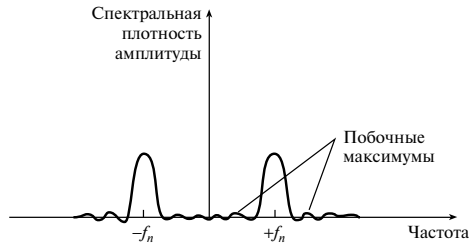


Рис. 11.2. Спектральная плотность амплитуды синусоидального сигнала, свернутого с дискретизирующей функцией

11.3.2. Взвешивание

В данном разделе описаны различные свойства весовых функций, причем в основном рассуждения ведутся во временной области. В то же время, не лишним будет напомнить, что взвешивание можно проводить как во временной (взвешивание данных), так и в частотной области (взвешивание частот), поскольку умножение во временной области эквивалентно свертке в частотной (подраздел 11.3.1.4). Взвешивание в частотной области, следовательно, можно заменить сверткой в частотной области весовой функции со спектром сигнала. Для реализации этой процедуры можно использовать уравнение (5.104).

11.3.2.1. Свойства весовых функций

Ширина полосы шумового (прямоугольного) эквивалента

В подразделе 11.3.1.4 было показано, что из-за явления просачивания спектральных составляющих импульсные характеристики, которые теоретически представляют спектральные плотности амплитуд, становятся выборочными функциями. Таким образом бесконечно узкие спектральные компоненты замещаются дискретизирующими функциями с более широкими полосами. Боковые лепестки этих функций, смещающие компоненты сигнала, можно рассматривать как источники нежелательного шума, а весовую частотную функцию можно считать характеристикой широкополосного фильтра. Следовательно, с такой точки зрения желательно разработать весовую функцию с малой шумовой полосой, уменьшив амплитуды боковых лепестков. *Шумовая полоса* вводится с целью сравнения различных весовых функций по ширине полосы шумового эквивалента. Данная величина определяется как ширина полосы идеального прямоугольного фильтра, который пропускает тот же объем белого шума, что и рассматриваемый спектральный фильтр (см. рис. 11.3) (поэтому данную величину называют еще *шириной полосы прямоугольного эквивалента*). Данное определение позволяет оценивать характеристики боковых лепестков различных функций путем сравнения их шумовых эквивалентов. Ширина полосы шумового эквивалента — очень важный параметр функции, чем он меньше, тем лучше весовая функция.

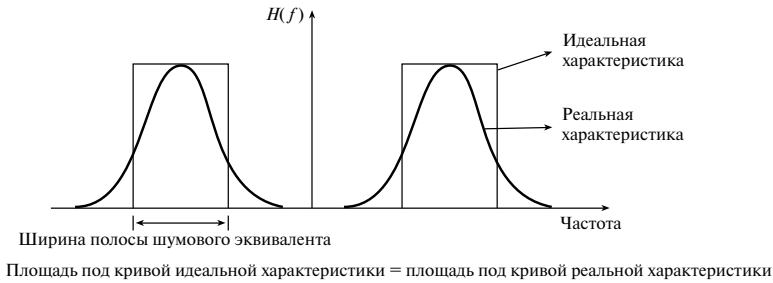


Рис. 11.3. Ширина полосы шумового эквивалента

Ширина полосы шумового эквивалента выражается следующим образом:

$$W_n = \frac{\sum_{n=0}^{n=N-1} w^2(nT)}{\left[\sum_{n=0}^{n=N-1} w(nT) \right]^2}. \quad (11.13)$$

Корреляция перекрывающихся сегментов

При взвешивании данных начало и конец последовательности данных выходят на нуль, и этот эффект представляет потерю информации. В частности, можно пропустить краткие события, происходящие в области спада характеристики. Для того чтобы решить данную проблему, следует разбить последовательность данных на перекрывающиеся сегменты, взвесить и преобразовать каждый сегмент отдельно. Если перекрытие составляет порядка 50–75%, то в сегментах будет отражено большинство особенностей данных. Затем получающиеся спектры усредняются и дают оценку истинного спектра. Разбиение данных иллюстрируется на рис. 11.4. Данная процедура называется *обработкой с избыточностью*, или *перекрытием*. Обычно обработка с перекрытием 50–75% позволяет на 90% использовать потенциал большинства весовых функций, связанный с улучшением производительности [18]. Кроме того, при усреднении спектров сегментов снижается дисперсия спектра. Для K статистически идентичных, но независимых измерений дисперсия среднего составляет $1/K$ дисперсии отдельных спектров. В то же время, сказанное не относится к случаю, когда усредняются спектры перекрывающихся сегментов, поскольку между сегментами существует корреляция. В работе [23] приводится отношение дисперсии усредненного спектра к дисперсии отдельных спектров для 50- и 75-процентного перекрытия. На основе этих данных можно показать, что, например, усреднение четырех спектров снижает дисперсию до уровня 25% от величины исходного спектра. Итак, описанный подход позволяет значительно улучшить оценку спектра.

Выигрыш от обработки

Выигрыш от обработки определяется делением отношения сигнал-шум на выходе процесса взвешивания на отношение сигнал-шум на входе:

$$\text{BO} = \frac{(S/N)_{out}}{(S/N)_{in}}. \quad (11.14)$$

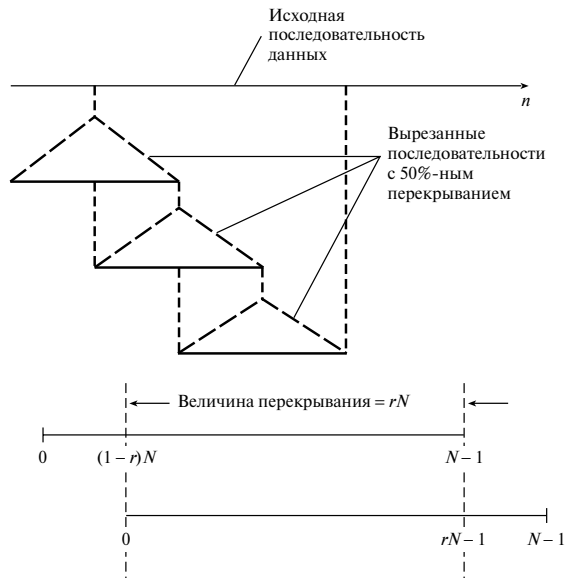


Рис. 11.4. Разбиение данных перед обработкой с избыточностью

Выигрыш от обработки зависит от формы весовой функции, поскольку она определяет ширину полосы ее шумового эквивалента (см. предыдущий раздел). Спад вырезающей функции до нуля уменьшает мощность сигнала, приводя к потерям при обработке, тогда как за счет боковых лепестков улучшается ширина полосы шумового эквивалента.

Потери при обработке в наихудшем случае

Потери при обработке в наихудшем случае определяются как сумма (в децибелах) максимального гребешкового искажения весовой функции и потерь при ее обработке. Эта величина представляет уменьшение выходного отношения сигнал-шум, вызванное взвешиванием и наихудшим (наиболее неудачным) расположением частот. Она всегда составляет от 3,0 до 4,3 дБ. Функций, для которых этот параметр составляет более 3,8 дБ, следует избегать. Это относится к прямоугольной функции, функции Пуассона ($\alpha = 4$), функции Хэмминга-Пуассона ($\alpha = 2, 0$), функции Коши и минимальной четырехточечной функции Блэкмена-Харриса.

Ширина полосы минимального разрешения

Обычно при перекрытии двух идентичных спектральных максимумов их можно разрешить, если точки по уровню 3 дБ не перекрываются (рис. 11.5, а). В то же время, при рассмотрении ДПФ-компонента соседние спектральные составляющие множатся на весовую функцию, а затем когерентно суммируются, т.е. боковые лепестки включаются в сумму. В этом случае усиление каждого компонента в области перекрытия не должно превышать 0,5. Это означает, что спектральное разрешение определяется шириной полосы по уровню 6 дБ, а не шириной полосы по уровню 3 дБ (рис. 11.5, б).



Рис. 11.5. Ширина полосы минимального разрешения: а) разрешение спектральных максимумов определяется ширинами полос по уровню 3 дБ; б) спектральное разрешение максимумов ДПФ определяется ширинами полос по уровню 6 дБ

Смещающий эффект весовых функций

Умножение данных на весовую функцию, выходящую по краям на нуль, уменьшает амплитуду выборок в местах спада, а следовательно, общую мощность сигнала. Можно показать, что все частотные составляющие в равной степени подвержены влиянию весовой функции, и что коэффициент изменения данных пропорционален корню квадратному из коэффициента когерентного усиления мощности, который представляет нормированную мощность данных при рассмотрении как сигнала напряжения. Следовательно, снижение мощности сигнала можно компенсировать без искажения спектра плотности мощности. Весовая функция также выравнивает средний уровень данных, увеличивая таким образом полную энергию низкочастотных компонентов спектра. Данный эффект нужно каким-то образом компенсировать, но прямое вычитание среднего взвешенных данных приводит к более явному проявлению высокочастотных боковых лепестков.

На рис. 11.6 показан спектр плотности энергии 64 точек данных, из которых вначале был вычтен средний уровень данных, после чего данные умножались на соответствующие значения функции Кайзера–Бесселя. Очевидно, что спектр содержит низкочастотные компоненты, введенные в процессе взвешивания. На рис. 11.7 изображен спектр после первого этапа — вычитания среднего из данных. Кажется, что низкочастотные компоненты удалены, но появился набор существенных боковых лепестков на высоких частотах. Ниже приводится пример, демонстрирующий, как можно устранить отрицательное влияние взвешивания.

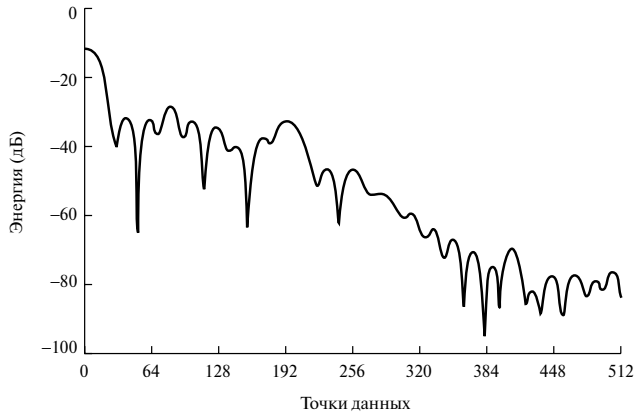


Рис. 11.6. Энергетический спектр при выборке с интервалом 1 с, весовая функция Кайзера–Бесселя, 64 информационные точки, 1024-точечное БПФ

Пример 11.1

Покажите, что влияния взвешивания, проявляющегося в снижении энергии сигнала и введении низкочастотных компонентов в спектр, можно избежать путем взвешивания линейной функции данных, а не самих данных [27].

Решение

Предположим, что исходные данные $s(n)$ имеют нулевое среднее. Пусть средний уровень, введенный в данные при взвешивании, удален путем вычитания из $s(n)$ константы k_1 . Теперь новые взвешенные данные s^1 записываются как

$$s^1(n) = w(n)[s(n) - k_1], \quad (11.15)$$

где $w(n)$ — значения весовой функции, или весовые коэффициенты. Снижение мощности сигнала, вызванное взвешиванием, можно теперь компенсировать, умножив каждое значение $s^1(n)$ на тщательно выбранную константу k_2 . Итак, точки теперь преобразуются к виду

$$S(n) = k_2 w(n)[s(n) - k_1]. \quad (11.16)$$

Требуемое значение k_1 можно найти из условия равенства нулю среднего значения $S(n)$. Следовательно,

$$\sum_{n=0}^{N-1} S(n) = 0.$$

Значит,

$$k_2 \left[\sum_{n=0}^{N-1} w(n)s(n) - \sum_{n=0}^{N-1} w(n)k_1 \right] = 0.$$

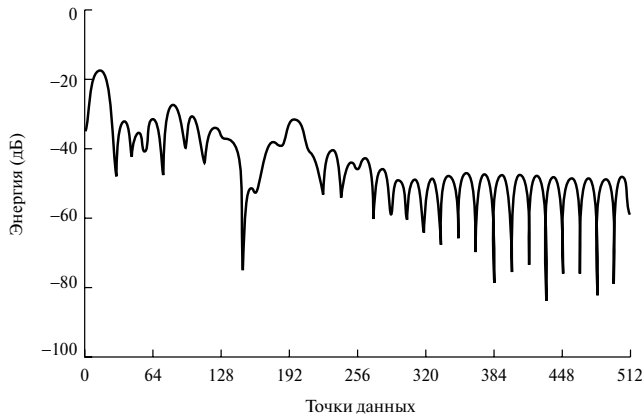


Рис. 11.7. Энергетический спектр при выборке с интервалом 1 с, весовая функция Кайзера-Бесселя, 64 информационные точки, 1024-точечное БПФ. Средний уровень взвешенных данных удален путем вычитания среднего из данных

Таким образом,

$$k_1 = \frac{\sum_{n=0}^{N-1} w(n)s(n)}{\sum_{n=0}^{N-1} w(n)}. \quad (11.17)$$

Нормированная мощность переменного тока данных до взвешивания равна

$$E[(s(n) - k_1)^2] = \sigma_{sN}^2, \quad (11.18)$$

где через E обозначено математическое ожидание, а σ_{sN}^2 — дисперсия $s(n)$ со средним значением k_1 . Нормированная мощность переменного тока взвешенных данных равна

$$E\{k_2^2 w^2(n)[s(n) - k_1]^2\},$$

где $w(n)$ и $s(n)$ взаимно независимы. Далее получаем

$$\begin{aligned} E\{k_2^2 w^2(n)[s(n) - k_1]^2\} &= E[k_2^2 w^2(n)]E\{[s(n) - k_1]^2\} = \\ &= k_2^2 E[w^2(n)]\sigma_{sN}^2. \end{aligned} \quad (11.19)$$

Значение k_2 , требуемое для выравнивания мощности взвешенных и невзвешенных данных, можно получить, приравняв уравнения (11.18) и (11.19):

$$\sigma_{sN}^2 = k_2 E[w^2(n)]\sigma_{sN}^2,$$

откуда

$$k_2^2 = \frac{1}{E[w^2(n)]} = \frac{1}{(1/N) \sum_{n=0}^{N-1} w^2(n)} = \frac{N}{\sum_{n=0}^{N-1} w^2(n)}.$$

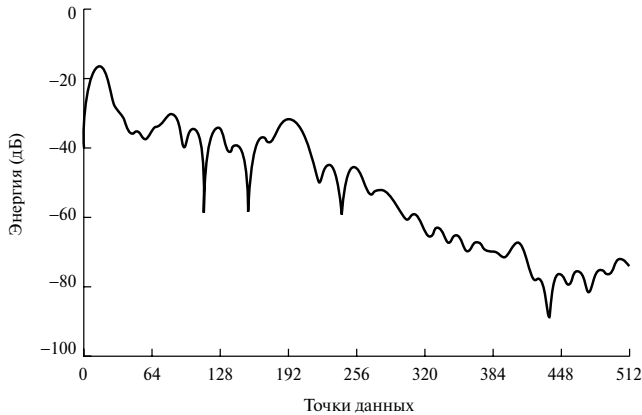


Рис. 11.8. Энергетический спектр при выборке с интервалом 1 с, весовая функция Кайзера–Бесселя, 64 информационные точки, 1024-точечное БПФ. Данные прошли предварительную обработку с целью удаления среднего уровня взвешенных данных и сохранения средней мощности

Следовательно,

$$k_2 = \left[\frac{N}{\sum_{n=0}^{N-1} w^2(n)} \right]^{1/2}. \quad (11.20)$$

Подставляя уравнения (11.17) и (11.20) в уравнение (11.16), окончательно получаем

$$S(n) = w(n) \left[s(n) - \frac{\sum_{n=0}^{N-1} w(n)s(n)}{\sum_{n=0}^{N-1} w(n)} \right] \left[\frac{N}{\sum_{n=0}^{N-1} w^2(n)} \right]^{1/2}. \quad (11.21)$$

На рис. 11.8 показан энергетический спектр данных, получающихся после удаления среднего взвешенных данных и восстановления энергии сигнала согласно уравнению (11.21). Постоянная составляющая, возникшая при взвешивании, и влияние боковых лепестков устранены.

Из приведенного примера видно, что данные $s(n)$ рекомендуется модифицировать согласно уравнению (11.21), а лишь затем применять ДПФ. Данная процедура эквивалентна вычитанию k_1 из данных с последующим умножением разности на k_2 перед взвешиванием.

11.3.2.2. Выбор весовой функции

В работе [23] подробно рассмотрено влияние различных характеристик весовых функций на эффективность метода взвешивания и сделан вывод, что на качество функции влияет в первую очередь наивысший уровень боковых лепестков и наибольший

проигрыш от обработки. В качестве предпочтительных называются весовые функции Блэкмена–Харриса, Дольфа–Чебышева и Кайзера–Бесселя. Менее эффективны функции Тьюки (спад по косинусу), Пуассона, Хеннинга и Хэмминга.

Спад многих весовых функций и их форму во многих случаях можно регулировать, выбирая значение параметра функции α . Это позволяет варьировать ширину основного лепестка и уровень боковых лепестков. Частью искусства взвешивания является выбор методом проб и ошибок значения α , оптимизирующего результаты в конкретной ситуации.

Пример 11.2

Влияние различных весовых функций на амплитудный спектр. На рис. 11.9, *а* показаны ДПФ-компоненты двух синусоид, отличающихся по амплитуде на 40 дБ и имеющих частоты $100f$ и $120f$ (где f — первая гармоника, соответствующая записи длительностью $T_s(s)$), которые были получены без взвешивания, т.е. с прямоугольной весовой функцией. В этом случае сигнал кажется периодическим и бесконечным (из-за гармонической связи сигнала с длиной окна) и точно воспроизводится даже при прямоугольной весовой функции. Чтобы разрушить указанную гармоническую связь, частоту большего сигнала положили равной $102,5f$ (не гармоника), результат представлен на рис. 11.9, *б*. В результате уровень боковых лепестков значительно повышается и в них почти теряется меньший сигнал. Данный эффект можно в значительной степени подавить, используя подходящую весовую функцию. На рис. 11.10, *а* и *б* показаны функции Тьюки (спадающий косинус) с $\alpha = 0,1$ и $\alpha = 0,5$, а на рис. 11.11, *а* и *б* показаны соответствующие спектры. На рис. 11.12, *а* и *б* изображена функция Хэмминга ($\alpha = 0,54$) и получающийся спектр соответственно. На рис. 11.13, *а–в* показаны функции Кайзера–Бесселя с $\alpha = 2, 0,3, 0$ и $4, 0$, а на рис. 11.14 иллюстрируются соответствующие спектры. Наиболее близкий к истине результат дает функция Кайзера–Бесселя с $\alpha = 4, 0$. Впрочем, следует отметить, что если увеличивать α для уменьшения боковых лепестков, одновременно увеличивается ширина основного лепестка, поэтому нужно искать компромиссное значение α . Разумеется, в реальной ситуации все гармоники будут подвергаться воздействию, так что если нужно избежать ложных результатов, очень важно выбрать весовую функцию с точки зрения суммарной эффективности для всех частот.

Весовая функция Дольфа–Чебышева наиболее предпочтительна с точки зрения малых боковых лепестков, хотя она дает наибольшие потери при обработке; впрочем, возможность ее использования для многотонавого детектирования маловероятна из-за когерентного сложения боковых лепестков. Кроме того, структура боковых лепестков этой функции весьма чувствительна к ошибкам коэффициентов. Таким образом, более предпочтительна функция Блэкмена–Харриса или Кайзера–Бесселя. Отметим, что коэффициенты функции Кайзера–Бесселя генерировать проще, а, меняя параметр α , легче найти компромисс между уровнем боковых лепестков и шириной основного лепестка. Данная функция выражается следующим образом (согласно [33]):

$$w(n_{\text{KB}}) = I_0 \left\{ \pi \alpha \left[1, 0 - \left(\frac{n_{\text{KB}}}{N/2} \right)^2 \right]^{1/2} \right\} / I_0(\pi \alpha), \quad 0 \leq |n_{\text{KB}}| \leq N/2, \quad (11.22)$$

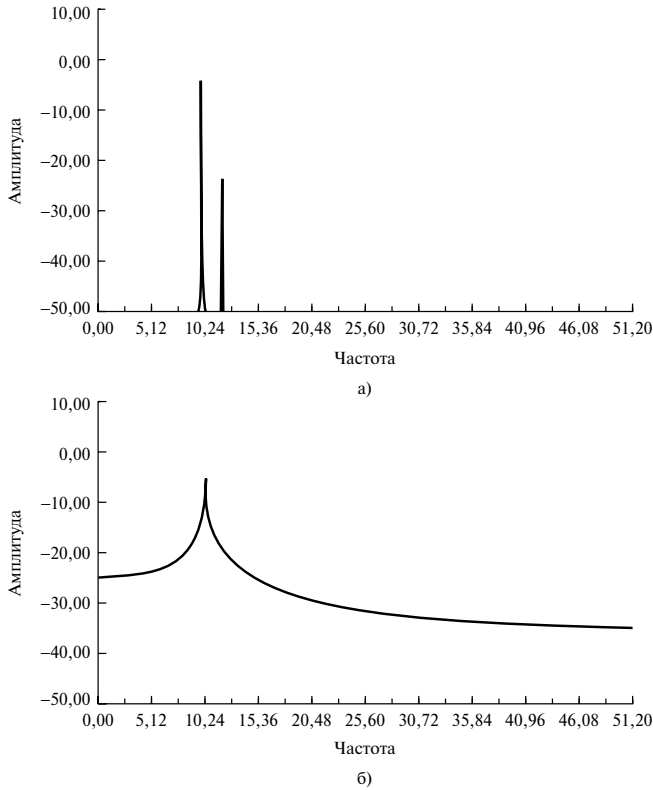


Рис. 11.9. Амплитудный спектр двух синусоид, отличающихся по амплитуде на 40 дБ: а) длина окна кратна обоим периодам; б) длина окна не кратна одному из периодов (большему)

где n — номер выборки весовой функции, α — числовой параметр, регулируя который выбирается наилучший компромисс между уровнем боковых лепестков и шириной основного лепестка, N — число выборок весовой функции.

$$I_0(x) = \sum_{k=0}^K \left[\frac{(x/2)^k}{k!} \right]^2 - \quad (11.23)$$

это модифицированная функция Бесселя первого рода нулевого порядка, а K теоретически равно бесконечности, но, поскольку амплитуда функции Бесселя быстро спадает с k , обычно достаточно положить $K = 32$.

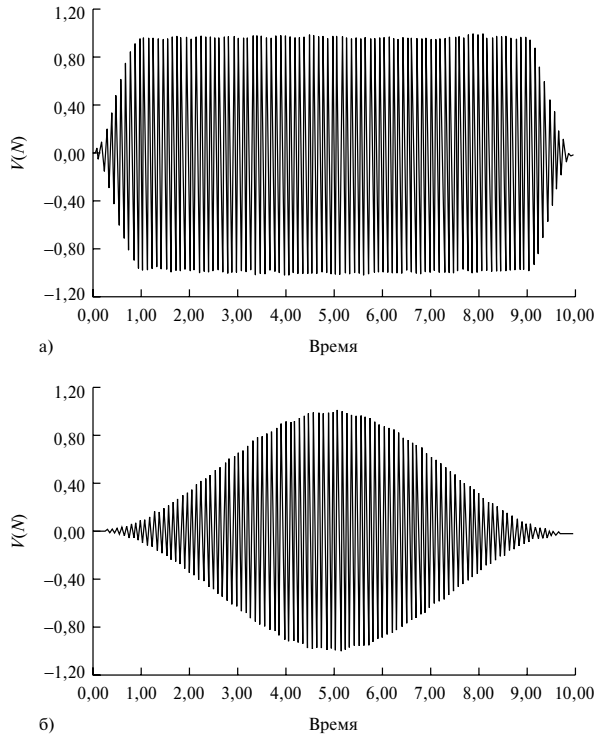


Рис. 11.10. Весовая функция Тьюки: а) $\alpha = 0, 1$; б) $\alpha = 0, 5$

Формула (11.22) определяет весовую функцию между точками $-N/2$ и $N/2 - 1$. Обычно с помощью ДПФ можно распространить эту функцию на промежуток от $n_{\text{ДПФ}} = 0$ до $n_{\text{ДПФ}} = N - 1$, где $n_{\text{ДПФ}}$ — номер элемента данных ДПФ. Следовательно, чтобы ДПФ можно было использовать совместно с весовой функцией Кайзера–Бесселя, эту функцию нужно сместить вправо на $N/2$, в результате чего получится

$$n_{\text{ДПФ}} = n_{KB} + N/2$$

или

$$n_{KB} = n_{\text{ДПФ}} - N/2. \quad (11.24)$$

Окончательно уравнение (11.22) записывается следующим образом:

$$w(n_{\text{ДПФ}}) = I_0 \left\{ \pi \alpha \left[1, 0 - \left(\frac{n_{\text{ДПФ}} - N/2}{N/2} \right)^2 \right]^{1/2} \right\} / I_0(\pi \alpha), \quad 0 \leq |n_{\text{ДПФ}}| \leq N - 1. \quad (11.25)$$

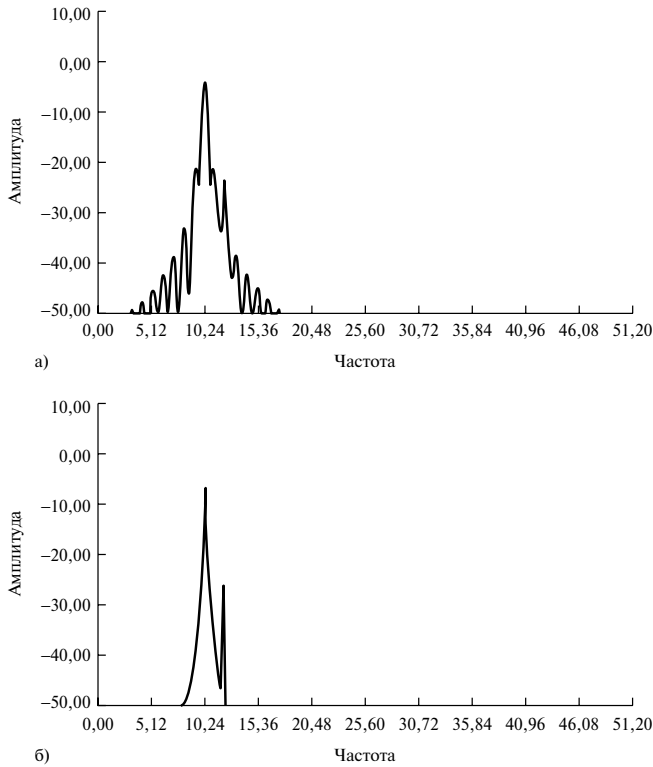


Рис. 11.11. Амплитудный спектр двух синусоид, умноженных на весовую функцию — конический косинус

11.3.3. Метод и свойства периодограмм

Квадрат модуля Фурье-образа $|F(i\omega)|^2$ представляет собой оценку спектральной плотности мощности $E[P(f)]$ и называется *периодограммой*. Можно показать [18, 44], что $E[P(f)]$ равно

$$E[P(f)] = \sum_{m=-\infty}^{\infty} w_B(m) c_{xx} \exp(-2\pi i f m), \quad (11.26)$$

где $c_{xx}(m)$ — автоковариационная матрица $x(n)$, вычисленная с запаздыванием m , f — частота, а w_B — треугольная весовая функция (Барлетта), которая определяется следующим образом:

$$w_B(m) = 1 - \frac{|m|}{N} \quad |m| \leq N - 1. \quad (11.27)$$

Для сравнения отметим, что истинная спектральная плотность мощности $P(f)$ функции $x(x)$ равна

$$P(f) = \sum_{m=-\infty}^{\infty} c_{xx} \exp(-2\pi i f m). \quad (11.28)$$

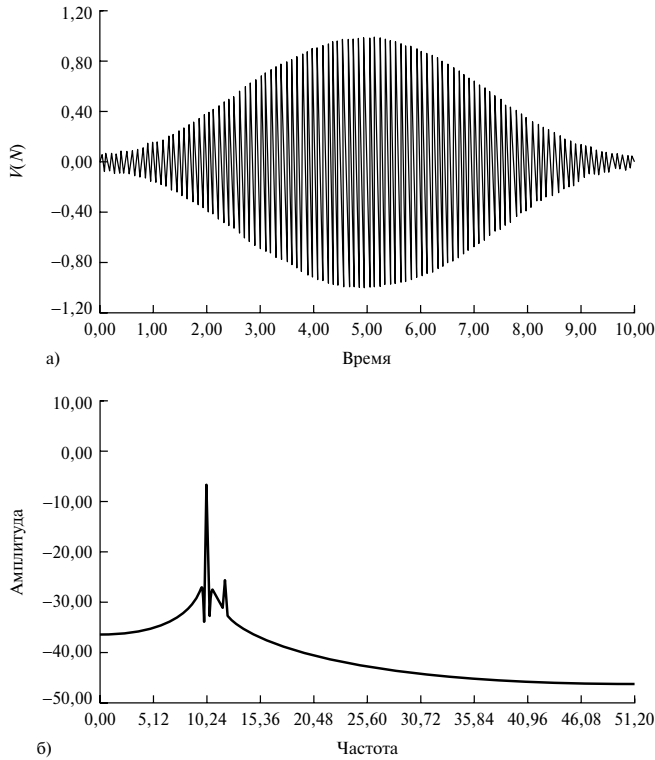


Рис. 11.12. Весовая функция Хэмминга (панель *a*); соответствующий амплитудный спектр двух синусоид (панель *б*)

Следовательно, спектральная плотность мощности, представленная периодограммой, колеблется, а максимальная амплитуда колебаний отклонения определяется следующим образом:

$$\begin{aligned}
 P(f) - E[P(f)] &= \sum_{m=-\infty}^{\infty} [1 - w_B(m)] c_{xx} \exp(-2\pi i f m) = \\
 &= \frac{|m|}{N} P(f).
 \end{aligned}
 \tag{11.29}$$

Для $N \gg |m|$ смещение становится малым и $E[P(f)] \rightarrow P(f)$, т.е. периодограмма асимптотически несмещенная. Кроме того, для больших N дисперсия периодограммы становится равной

$$\text{var}[P(f)] \approx F P^2(f),
 \tag{11.30}$$

где F зависит от используемой весовой функции, т.е. дисперсия зависит от квадрата спектральной плотности мощности и не сходится к нулю с ростом N . Это означает, что оценки спектральной плотности мощности, полученные из периодограммы, несостоятельны и дают колеблющиеся оценки $P(f)$ в последовательных реализациях.

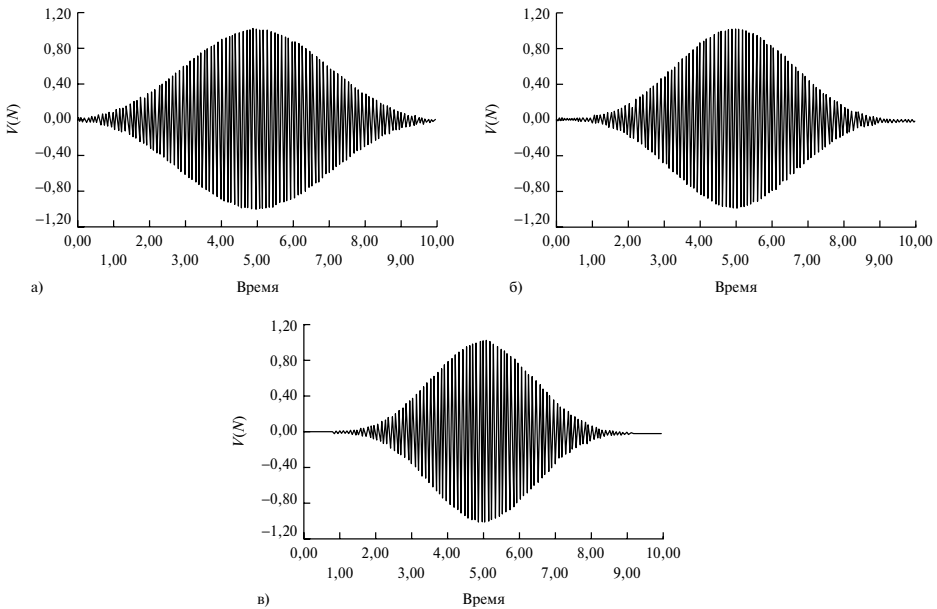


Рис. 11.13. Весовые функции Кайзера–Бесселя: а) $\alpha = 2, 0$; б) $\alpha = 3, 0$; в) $\alpha = 4, 0$

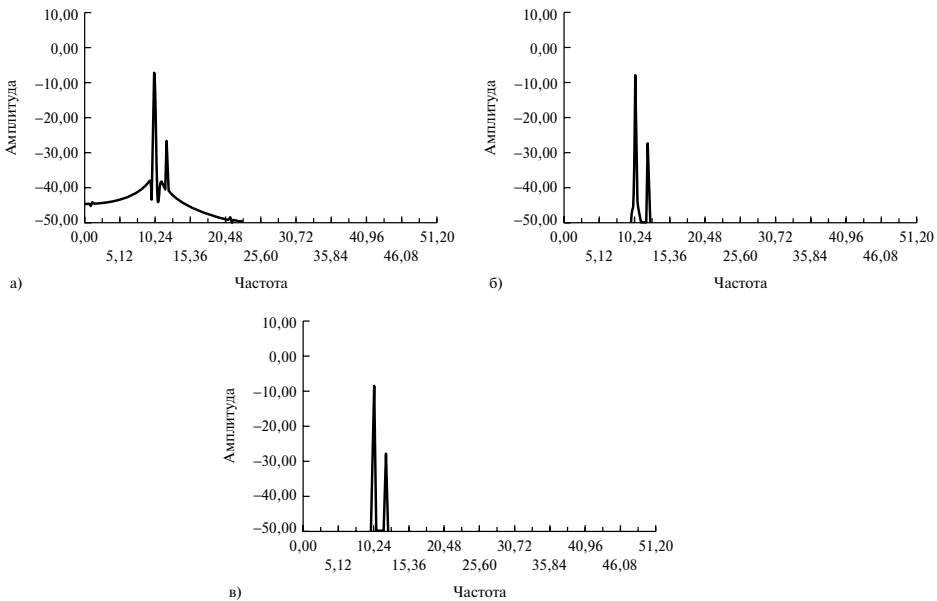


Рис. 11.14. Амплитудный спектр двух синусоид, вычисленный с использованием соответствующих функций Кайзера–Бесселя, представленных на рис. 11.13: а) $\alpha = 2, 0$; б) $\alpha = 3, 0$; в) $\alpha = 4, 0$

Отметим, что автоковариационная матрица обычно определяется усреднением по N членам, хотя ее можно определять и усреднением по $N - |m|$ членам. Обе оценки состоятельны и являются асимптотически несмещенными, но в первом случае получается меньшая дисперсия, поэтому такой способ предпочтительнее.

Более того, при применении для получения спектра ДПФ соответствующая периодограмма определяется как $(1/N)|X(k)|^2$ и имеет размерность нормированной энергии, хотя некоторые авторы называют функцию $(1/N)|X(k)|^2$ *спектральной плотностью мощности*.

11.3.4. Методы модифицированных периодограмм

В работе [50] предлагается для устранения несостоятельности метода периодограмм использовать усреднение нескольких модифицированных периодограмм. Каждая из таких периодограмм состоит из сегмента данных. Данные сегменты могут идти последовательно (метод Барлетта) или перекрываться (метод Уэлша). Данные методы также уменьшают дисперсию оценок спектральной плотности мощности. Метод Барлетта иллюстрируется в примере 11.3.

11.3.4.1. Метод Уэлша

Преимущество метода Уэлша по сравнению с методом Барлетта заключается в дальнейшем уменьшении спектральной плотности мощности. В то же время, это улучшение происходит за счет добавочного уменьшения спектрального разрешения. В методе Уэлша L сегментов данных длины M перекрываются и периодограммы вычисляются по L взвешенным сегментам. Далее периодограммы нормируются на величину U , чтобы компенсировать потери энергии сигнала вследствие процедуры взвешивания. Фактически U приравнивается к $1/k_2^{1/2}$, где k_2 — коэффициент, полученный в подразделе 11.3.2.1 и описывающий эффект смещения весовых функций. Следовательно,

$$U = \frac{1}{M} \sum_{n=0}^{M-1} w^2(n). \quad (11.31)$$

Таким образом, оценка Уэлша спектральной плотности мощности $P_{WE}(f)$ записывается следующим образом:

$$P_{WE}(f) = \frac{1}{L} \sum_{j=0}^{L-1} P_j(f). \quad (11.32)$$

Математическое ожидание оценки Уэлша:

$$E[P_{WE}(f)] = \frac{1}{L} \sum_{j=0}^{L-1} E[P_j(f)] = E[P_j(f)], \quad (11.33)$$

т.е. оно равно математическому ожиданию модифицированной периодограммы. Можно показать [44], что при $N \rightarrow \infty$ и $M \rightarrow \infty$ значения сходятся к истинной спектральной плотности мощности $P(f)$. Итак, для больших N и M оценка Уэлша спектральной

плотности мощности несмещенная. При определенных условиях дисперсия оценок Уэлша сходится к нулю, т.е. оценки состоятельны. Уэлш также показал, что при отсутствии наложения сегментов ($L = K$)

$$\text{var}[P_{\text{WE}}(f)] \approx (1/K)P^2(f),$$

что равно дисперсии оценки Барлетта при тех же условиях. При 50%-ном наложении ($L = 2K$)

$$\text{var}[P_{\text{WE}}(f)] \approx (9/8L)P^2(f),$$

что составляет $9/16=0,56$ дисперсии оценки Барлетта.

11.3.5. Метод Блэкмена–Тьюки

В главе 3 было показано, что спектр плотности мощности определяется ДПФ автокорреляционной функции данных. Зная, что периодограммы можно вычислить прямо по данным как квадрат ДПФ-образа данных, рассмотрим вопрос о полезности данного подхода. В связи с этим, во-первых, заслуживает внимания тот факт, что метод Блэкмена–Тьюки был разработан в 1958 году (см. [6]), тогда как алгоритм БПФ для быстрого вычисления ДПФ не публиковался до 1965 года [17]. Во-вторых, метод Блэкмена–Тьюки имеет некоторые преимущества по сравнению с методом периодограмм. Например, в следующем разделе показано, что метод Блэкмена–Тьюки характеризуется большой добротностью. Кроме того, автокорреляционные функции теперь можно вычислить с помощью ДПФ посредством метода быстрой корреляции (раздел 11.3.6). В результате получаем следующую процедуру Блэкмена–Тьюки для определения спектра плотности мощности:

- 1) вычислить автокорреляционную функцию данных;
- 2) воздействовать на данные подходящей весовой функцией;
- 3) вычислить БПФ получающихся данных и получить спектр плотности мощности.

Сравнивая эту процедуру с методом периодограмм, видим, что сглаживание достигается не за счет усреднения нескольких периодограмм, а за счет усредняющего эффекта процесса автокорреляции.

Взвешивание автокорреляционной функции необходимо для сглаживания ее экстремумов, поскольку при больших значениях параметра запаздывания в вычислении участвует немного точек данных, так что получающиеся оценки менее точны. Спад весовой функции по конусу приводит к тому, что данные оценки учитываются с меньшими весовыми коэффициентами.

Оценка Блэкмена–Тьюки $P_{\text{BTE}}(f)$ равна

$$P_{\text{BTE}}(f) = \sum_{m=-(M-1)}^{M-1} r_{xx}w(m) \exp(-2\pi i f m), \quad (11.34)$$

где $r_{xx}(m)$ — автокорреляционная функция данных, а $w(n)$ — весовая функций длины $2M - 1$, равная нулю при $|m| \geq M$.

Чтобы получить реальные оценки, $w(n)$ должно быть симметричным относительно $m = 0$, а чтобы оценки были положительными, образ $w(n)$ должен быть больше нуля. Не все весовые функции удовлетворяют данным критериям. Например, им не удовлетворяют функции Хеннинга и Хэмминга.

Можно показать, что математическое ожидание оценки Блэкмена–Тьюки равно

$$P_{\text{БТЕ}}(f) = \sum_{m=-(M-1)}^{M-1} c_{xx} w_B(m) \exp(-2\pi i f m), \quad (11.35)$$

где $w_B(m)$ — треугольная весовая функция Барлетта.

Для получения дополнительного сглаживания спектра должно выполняться условие $M < N$. Если $N \gg m$, оценка будет асимптотически несмещенной. Кроме того, если $W(k)$ (ДПФ-образ весовой функции, $w(n)$) уже $P(f)$, истинный спектр плотности мощности равен

$$\text{var}[P_{\text{БТЕ}}(f)] \approx P^2(f) \left[\frac{1}{N} \sum_{m=-(M-1)}^{M-1} w^2(m) \right] \quad (11.36)$$

и при $N/M \rightarrow \infty$ $\text{var}[P_{\text{БТЕ}}(f)] \rightarrow 0$, так что при данных условиях оценка Блэкмена–Тьюки является состоятельной.

11.3.6. Метод быстрой корреляции

В разделе 5.3.7 было отмечено, что если данные превышают 128 элементов и требуется найти их корреляцию, то вычисления будут выполняться быстрее, если использовать теорему о корреляции (уравнение (5.63)) и реализовать вычисления с использованием БПФ. Например, при $N = 1024$ это позволяет десятикратно увеличить скорость. Кроме того, если нужно обработать большие наборы входных данных, например, превышающие объем памяти системы, то можно применить методы наложения-сложения или наложения-хранения (разделы 5.3.8–5.3.10). При вычислении автокорреляции в методе Блэкмена–Тьюки описанным выше способом с использованием БПФ подход называется *методом оценки спектра с помощью быстрой корреляции*.

11.3.7. Сравнение методов оценки спектральной плотности мощности

Выражение для добротности оценок спектральной плотности мощности приводилось в уравнении (11.6). Можно показать (см. [44]), что добротность четырех рассмотренных непараметрических методов спектрального анализа имеет значения, указанные в табл. 11.1, где f — ширина главного лепестка соответствующих весовых функций по уровню 3 дБ. Видно, что метод Блэкмена–Тьюки является наилучшим с точки зрения качества, и что все методы, исключая метод периодограмм, позволяют поддерживать качество при увеличении разрешения по частоте (уменьшения f) за счет увеличения N .

Чтобы добиться удовлетворительных результатов, требуется большая аккуратность и несколько вспомогательных вычислений. С точки зрения баланса наилучшим кажется метод Блэкмена–Тьюки, хотя могут существовать соображения, повышающие привлекательность других методов.

Таблица 11.1. Добротность Q оценок спектральной плотности мощности

Метод оценки	Условия	Q	Комментарии
Периодограмма	$N \rightarrow \infty$	1	Несостоятельная оценка, не зависит от N
Барлетта	$N, M \rightarrow \infty$	$1, 11Nf$	Качество повышается с размером сегмента данных
Уэлша	$N, M \rightarrow \infty$, наложение 50%	$1, 39Nf$	Качество повышается с размером сегмента данных
Блэкмена–Тьюки	$N, M \rightarrow \infty$, треугольная весовая функция	$2, 34Nf$	Качество повышается с размером сегмента данных

11.4. Современные параметрические методы оценки

Для непараметрических методов, описанных в предыдущих разделах данной главы и использующих периодограммы и БПФ, характерны упомянутые выше ограничения: низкое спектральное разрешение при коротких сегментах данных, необходимость взвешивания для предотвращения просачивания спектральных составляющих. Этих сложностей можно избежать, используя параметрические методы [9–11, 20, 22, 31, 38, 41, 49]. Ценой улучшения является необходимость всестороннего исследования подходящей модели каждого процесса, определения необходимого порядка выбранной модели для адекватного представления данных [1–5, 7, 14, 24, 45, 48, 51] и вычисления параметров модели [15, 19, 36, 37, 43, 44, 52]. В число получаемых преимуществ входят повышенное спектральное разрешение, возможность применения к кратким сегментам данных, отсутствие просачивания спектральных составляющих, гребешкового искажения, размывания спектра и смещения весовой функции. Поскольку параметрические методы очень важны, ниже представлен самый распространенный из них — метод авторегрессионного моделирования. Тем не менее, кроме преимуществ, параметрические методы имеют и недостатки, которые можно устранить с помощью альтернативных современных подходов, например, последовательного или адаптивного [20, 30] и метода максимального правдоподобия [13, 34].

Параметрический подход требует параметрического моделирования данных, процесса, хорошо изученного в сфере анализа с помощью временных рядов [7, 24, 43], плюс интерполяции данных, которые рассматриваются как выход линейной системы, возбужденной белым шумом. Данная система представляется полиномиальной передаточной функцией, выраженной через параметры модели. Далее по этой передаточной функции вычисляется спектр данных.

11.5. Авторегрессионная оценка спектра

В данном методе оцифрованный сигнал моделируется как авторегрессионный (АР) временной ряд плюс слагаемое белого шума. После этого спектр находится из параметров АР-модели и дисперсии слагаемого шума. Параметры модели — это решение системы линейных уравнений, полученных минимизацией среднеквадратической ошибки (мощности белого шума) по всем данным. Несколько способов решения данной системы уравнений описано ниже. Важным моментом является выбор числа членов в АР-модели — *порядка* модели. При слишком маленьком порядке оценка плотности мощности будет чрезмерно сглаженной, так что некоторые максимумы могут скрадываться. Если порядок слишком велик, могут вводиться ложные максимумы. Следовательно, для каждого набора данных нужно определить подходящий порядок модели, и ниже эта тема обсуждается подробно. Описанный метод применим к сигналам, спектр плотности мощности которых содержит острые максимумы. Если сигналы не удовлетворяют этому критерию, можно использовать другие модели, такие как модель скользящего среднего или авторегрессионного движения. Поскольку авторегрессионный подход дает решаемые уравнения, его стоит использовать, когда только возможно. Более подробное освещение данной темы можно найти в работах [12, 16, 32, 39, 42, 44].

11.5.1. Авторегрессионная модель и авторегрессионный фильтр

В АР-модели временного ряда текущее значение ряда $x(n)$ представляется линейной функцией предыдущих значений плюс ошибка $e(n)$:

$$x(n) = -a(1)x(n-1) - a(2)x(n-2) - \dots - a(k)x(n-k) - \dots - a(p)x(n-p) + e(n). \quad (11.37)$$

Это уравнение содержит p предыдущих членов и представляет модель порядка p . Его можно записать в более компактной форме:

$$x(n) = -\sum_{k=1}^p a(k)x(n-k) + e(n) = -\sum_{k=1}^p a(k)z^{-k}x(n) + e(n), \quad (11.38)$$

где z^{-k} — оператор запаздывания, которым обозначена задержка на k интервалов выборки. Далее перепишем уравнение (11.38):

$$x(n) + \sum_{k=1}^p a(k)z^{-k}x(n) = \left(1 + \sum_{k=1}^p a(k)z^{-k}\right)x(n) = e(n). \quad (11.39)$$

Отсюда

$$x(n) = \frac{e(n)}{1 + \sum_{k=1}^p a(k)z^{-k}}. \quad (11.40)$$

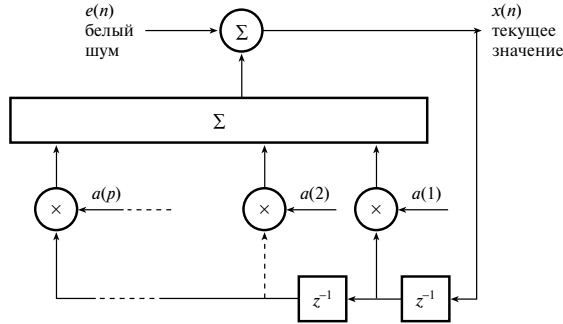


Рис. 11.15. АР-фильтр

Выражая отношение $x(n)/e(n)$, получаем

$$\frac{x(n)}{e(n)} = \frac{1}{1 + \sum_{k=1}^p a(k)z^{-k}} = H(z). \quad (11.41)$$

Здесь $H(z)$ интерпретируется как z -преобразование цифрового БИХ-фильтра с одними полюсами, который имеет коэффициенты $a(k)$. Подобный фильтр называется *авторегрессионным*. Из уравнения (11.41) $x(n)$ можно рассматривать как выходы данного фильтра, порожденные случайными входами $e(n)$. Величина $e(n)$ представляет ошибку между значением, предсказанным моделью, $\hat{x}(n)$ и истинным значением выборки $x(n)$. Как правило, предполагается, что $e(n)$ имеет свойства белого шума, т.е. гауссово распределение плотности вероятностей и равномерный спектр плотности мощности. Следовательно, можно считать, что $x(n)$ генерируется АР-фильтром, на входе которого — источник белого шума. Частотная характеристика фильтра $H(f)$ получается подстановкой $z = e^{i\omega T}$ в уравнение (11.39) (см. раздел 4.4), где ω — угловая частота, а T — период дискретизации. Следовательно,

$$H(f) = \frac{1}{1 + \sum_{k=1}^p a(k)e^{-ik\omega T}}. \quad (11.42)$$

11.5.2. Спектральная плотность мощности авторегрессионного ряда

Требуется найти спектральную плотность мощности $P_x(f)$ АР-ряда $x(n)$. Данная величина связана со спектральной плотностью мощности сигнала белого шума $P_e(f)$, равной ее дисперсии $\sigma_e^2(n)$:

$$P_x(f) = |H(f)|^2 P_e(f) = |H(f)|^2 \sigma_e^2(n) = \frac{\sigma_e^2(n)}{\left| 1 + \sum_{k=1}^p a(k)e^{-ik\omega T} \right|^2}. \quad (11.43)$$

Дисперсия белого шума равна ее среднеквадратическому значению, которое, в свою очередь, равно квадрату среднего значения $e(n)$, а эту величину мы обозначили через E . Итак, значение $\sigma_e^2(n)$ (или E) можно выразить через параметры модели, а поскольку параметры модели определить можно, то спектральная плотность мощности также находится.

11.5.3. Расчет параметров модели — уравнения Юла–Уокера

Оптимальными параметрами модели считаются те, которые минимизируют ошибку $e(n)$ для всех выборок $x(n)$, представленных выражением, подобным уравнению (11.38). Выразим эти ошибки из уравнения (11.38):

$$e(n) = x(n) + \sum_{k=1}^p a(k)x(n-k). \quad (11.44)$$

Далее требуется мера общей ошибки по всем выборкам, N ($1 \leq n \leq N$). Каждая ошибка $e(n)$ может быть положительной или отрицательной, и для больших наборов выборок средняя ошибка обычно мала. Следовательно, средняя ошибка — это не совсем удобная мера точности модели, поэтому обычно используется средний квадрат ошибки (среднеквадратическая ошибка). Среднеквадратическая ошибка определяется следующим образом:

$$E = \frac{1}{N} \sum_{n=1}^N e^2(n) = \frac{1}{N} \sum_{n=1}^N \left(x(n) + \sum_{k=1}^p a(k)x(n-k) \right)^2. \quad (11.45)$$

Выше указывалось, что $E = \sigma_e^2(n)$. Значит, теперь требуется выбрать параметры модели, минимизирующие E . Чтобы получить оптимальное значение каждого параметра в уравнении (11.45), нужно положить частную производную по параметру модели равной нулю. Таким образом для k -го параметра получаем уравнение:

$$\frac{\partial E}{\partial a(k)} = \frac{2}{N} \sum_{n=1}^N \left(x(n) + \sum_{k=1}^p a(k)x(n-k) \right) \frac{\partial}{\partial a(k)} \sum_{k=1}^p a(k)x(n-k) = 0, \quad 1 \leq k \leq p. \quad (11.46)$$

Теперь

$$\frac{\partial}{\partial a(k)} \sum_{k=1}^p a(k)x(n-k) = x(n-k),$$

так что уравнение (11.46) упрощается до вида

$$\frac{\partial E}{\partial a(k)} = \frac{2}{N} \sum_{k=1}^N \left(x(n) + \sum_{k=1}^p a(k)x(n-k) \right) x(n-k) = 0. \quad (11.47)$$

Для k -го параметра получаем следующее уравнение:

$$\frac{1}{N} \sum_{n=1}^N \left(\sum_{k=1}^p a(k)x(n-k) \right) x(n-k) = -\frac{1}{N} \sum_{n=1}^N x(n)x(n-k). \quad (11.48)$$

Проиллюстрируем полученный результат на примере. Записывая левую часть уравнения (11.48) для $k = 1$, получаем

$$\begin{aligned} & \frac{1}{N} \sum_{n=1}^N (a(1)x(n-1) + a(2)x(n-2) + \dots + a(p)x(n-p))x(n-1) = \\ & = \frac{1}{N} (a(1)x(0)x(0) + a(2)x(-1)x(0) + \dots + a(p)x(1-p)x(0)) + \\ & + \frac{1}{N} (a(1)x(1)x(1) + a(2)x(0)x(1) + \dots + a(p)x(2-p)x(1)) + \dots + \\ & + \frac{1}{N} (a(1)x(N-1)x(N-1) + a(2)x(N-2)x(N-1) + \dots + a(p)x(N-p)x(N-1)). \end{aligned}$$

Изучая первые члены в каждой строке, видим, что их сумма дает автокорреляционную функцию с нулевой задержкой $R_{xx}(0)$ ряда $x(n)$, умноженную на $a(1)$. Следовательно, сумма вторых членов дает автокорреляционную функцию с единичной задержкой $R_{xx}(-1)$, умноженную на $a(2)$, а сумма n -х членов строк дает произведение $R_{xx}(-(p-1))$ и $a(p)$. Поскольку для автокорреляционных функций справедливо соотношение $R_{xx}(-j) = R_{xx}(j)$, данное выражение можно записать как

$$R_{xx}(0)a(1) + R_{xx}(1)a(2) + \dots + R_{xx}(k-1)a(k) + \dots + R_{xx}(p-1)a(p).$$

Правая часть уравнения (11.48) равна $-R_{xx}(1)$. Приравняв левую и правую части, получаем

$$\begin{aligned} & R_{xx}(0)a(1) + R_{xx}(1)a(2) + \dots + R_{xx}(k-1)a(k) + \dots + \\ & + R_{xx}(p-1)a(p) = -R_{xx}(1). \end{aligned} \quad (11.49)$$

Подобное уравнение можно записать для каждого значения k , $1 \leq k \leq p$. В матричной форме эти уравнения выглядят так:

$$\begin{pmatrix} R_{xx}(0) & R_{xx}(1) & \dots & R_{xx}(p-1) \\ R_{xx}(1) & R_{xx}(0) & \dots & R_{xx}(p-2) \\ \vdots & \vdots & \ddots & \vdots \\ R_{xx}(p-1) & R_{xx}(p-2) & \dots & R_{xx}(0) \end{pmatrix} \begin{pmatrix} a(1) \\ a(2) \\ \vdots \\ a(p) \end{pmatrix} = - \begin{pmatrix} R_{xx}(1) \\ R_{xx}(2) \\ \vdots \\ R_{xx}(p) \end{pmatrix}. \quad (11.50)$$

Теперь из этой системы уравнений, известных как уравнения Юла–Уокера (Yule–Walker), можно получить параметры модели $a(k)$. В матричной форме уравнение (11.50) записывается следующим образом

$$\mathbf{R}_{xx}(k-j)\mathbf{a}(k) = -\mathbf{R}_{xx}(k). \quad (11.51)$$

Значит, в принципе,

$$\mathbf{a}(k) = -\mathbf{R}_{xx}^T(k-j)\mathbf{R}_{xx}(k). \quad (11.52)$$

Видно, что матрица $\mathbf{R}_{xx}(k-j)$ симметричная, и поскольку все элементы главной диагонали одинаковы (и равны $R_{xx}(0)$), матрица является матрицей Теплица. Она также

положительно определенная, откуда следует, что $x(n)$ не состоит исключительно из синусоид. Далее осталось только найти $a(k)$ из системы уравнений (11.50), используя один из множества существующих способов.

Значение E можно вычислить из уравнения (11.45) или получить другое выражение через автокорреляционные функции и найти $a(k)$ так, как описано ниже. Предполагая, что $a(k)$ и $x(n)$ — действительные и расписывая уравнение (11.45), получаем:

$$\begin{aligned} E &= \frac{1}{N} \sum_{n=1}^N \left\{ x(n) + \sum_{k=1}^p a(k)x(n-k) \right\} \left\{ x(n) + \sum_{k=1}^p a(k)x(n-k) \right\} = \\ &= \frac{1}{N} \left[\left\{ x(n) + \sum_{k=1}^p a(k)x(n-k) \right\} x(n) + \right. \\ &\quad \left. + \left\{ x(n) + \sum_{k=1}^p a(k)x(n-k) \right\} \sum_{k=1}^p a(k)x(n-k) \right]. \end{aligned} \quad (11.53)$$

Из уравнения (11.47), которое справедливо для всех k , находим, что в уравнении (11.53)

$$\frac{1}{N} \sum_{n=1}^N \left\{ x(n) + \sum_{k=1}^p a(k)x(n-k) \right\} \sum_{k=1}^p a(k)x(n-k),$$

таким образом, уравнение (11.53) упрощается до вида

$$\begin{aligned} E &= \frac{1}{N} \sum_{n=1}^N \left\{ x(n) + \sum_{k=1}^p a(k)x(n-k) \right\} x(n) = \\ &= \frac{1}{N} \sum_{n=1}^N x^2(n) + \frac{1}{N} \sum_{n=1}^N \left(\sum_{k=1}^p a(k)x(n-k) \right) x(n) = \\ &= R_{xx}(0) + \sum_{k=1}^p \frac{1}{N} \sum_{n=1}^N a(k)x(n)x(n-k), \end{aligned}$$

так что окончательно получаем

$$E = R_{xx}(0) + \sum_{k=1}^p a(k)R_{xx}(k). \quad (11.54)$$

Уравнение (11.54) или (11.45) и параметры модели из уравнения (11.52) теперь можно подставить в уравнение (11.43) и получить авторегрессионный спектр плотности мощности. Впрочем, опишем вначале методы нахождения $a(k)$ из системы (11.50) и выбор порядка модели p .

11.5.4. Решение уравнений Юла–Уокера

Среднеквадратическая ошибка E , представленная в уравнении (11.45), вычисляется с использованием имеющихся выборок $x(n)$ для n от 1 до N . Предшествующие или будущие значения $x(n)$ полагаются равными нулю. Как объяснялось выше, это

эквивалентно взвешиванию данных и при использовании непараметрических методов оценки спектра приводит к спектральному сглаживанию, вызванному влиянием боковых лепестков и пониженным разрешением. Впрочем, для авторегрессионного метода данное утверждение неверно. Можно показать (см. [32]), что в данном методе неявно оцениваются автокорреляционные функции для задержек, превышающих p , даже если соответствующих значений $x(n)$ не существует. Следовательно, авторегрессионные методы дают улучшенное спектральное разрешение. В то же время можно улучшить и спектральные оценки, например, используя в методе только существующие данные. При таком подходе значения n ($0 \leq n < N$) не требуются, а следовательно их не нужно полагать равными нулю. Ниже приводятся примеры методов, разработанных на описанном принципе. Более полное специализированное изложение можно найти в таких работах, как [12, 32]. Реализации алгоритмов в программном виде представлены в книге [32], также их можно найти в других публикациях и программных пакетах.

11.5.4.1. Автокорреляционный метод

Автокорреляционный метод основан на выражении для среднеквадратической ошибки (11.45). Из системы уравнений (11.50) с помощью алгоритма Левинсона–Дурбина (Levinson–Durbin) (см. [32, 42]) можно вычислительно эффективно найти параметры модели. Данный метод дает худшее разрешение по частоте, чем методы, описанные ниже, а следовательно, он менее подходит для коротких последовательностей выборок.

11.5.4.2. Ковариационный метод

В данном методе пределы суммирования в уравнении (11.45) устанавливаются равными от $n = p$ до $n = N$. Это означает, что для вычисления автокорреляционной функции требуются только доступные значения $x(n)$. Кроме того, среднее вычисляется по $N - p$ слагаемым, а не по N . Таким образом, уравнение (11.45) приводится к виду

$$E = \frac{1}{N-p} \sum_{n=p}^N \left(x(n) + \sum_{k=1}^p a(k)x(n-k) \right)^2. \quad (11.55)$$

Кроме того, используется следующий эквивалент уравнения (11.50):

$$\begin{pmatrix} C_{xx}(1,1) & C_{xx}(1,2) & \cdots & C_{xx}(1,p) \\ C_{xx}(2,1) & C_{xx}(2,2) & \cdots & C_{xx}(2,p) \\ \vdots & \vdots & \ddots & \vdots \\ C_{xx}(p,1) & C_{xx}(p,2) & \cdots & C_{xx}(p,p) \end{pmatrix} \begin{pmatrix} a(1) \\ a(2) \\ \vdots \\ a(p) \end{pmatrix} = - \begin{pmatrix} C_{xx}(1,0) \\ C_{xx}(2,0) \\ \vdots \\ C_{xx}(p,0) \end{pmatrix}, \quad (11.56)$$

где

$$C_{xx}(j,k) = \frac{1}{N-p} \sum_{n=p}^N x(n-j)x(n-k), \quad (11.57)$$

E выражается следующим образом:

$$E = C_{xx}(0,0) + \sum_{k=1}^p a(k)C_{xx}(0,k). \quad (11.58)$$

Матрица $C_{xx}(j, k)$ размером $p \times p$ эрмитова и неотрицательно определенная. Для решения уравнения (11.56) можно использовать метод разложения Холецкого [35]. Суммируется только $N - p$ запаздывающих компонентов, так что для кратких последовательностей данных могут иметь место некоторые краевые эффекты. Отметим, что ковариационный метод дает лучшее спектральное разрешение, чем автокорреляционный.

11.5.4.3. Модифицированный ковариационный метод

В модифицированном ковариационном методе минимизируется среднее оцененных ошибок предсказания с опережением и запаздыванием [12, 32]. По-прежнему используются уравнения (11.56) и (11.58), но уравнение (11.57) модифицируется до вида

$$C_{xx}(j, k) = \frac{1}{2(N - p)} \left\{ \sum_{n=p}^N x(n - j)x(n - k) + \sum_{n=1}^{N-p} x(n + j)x(n + k) \right\}. \quad (11.59)$$

Матрица $C_{xx}(j, k)$ размером $p \times p$ снова эрмитова и неотрицательно определенная, а для решения уравнения (11.56) используется метод разложения Холецкого [35]. Данный подход не гарантирует строгой устойчивости фильтра с одними полюсами, но на практике такие фильтры обычно устойчивы. Кроме того, метод дает статистически устойчивые спектральные оценки с высоким разрешением.

11.5.4.4. Метод Бэга

Данный метод основан на аспектах, которые не рассматриваются в данной книге. Он дает точные спектральные оценки для авторегрессионных данных.

11.5.5. Порядок модели

Порядок авторегрессионной модели, которая наилучшим образом аппроксимирует данные, нужно выбирать аккуратно для каждого набора данных, поскольку порядок зависит от статистических свойств данных. Данный момент можно проиллюстрировать на данных ЭЭГ, где различные сегменты данных требуют моделей различных порядков [42]. Модели низкого порядка предпочтительнее, поскольку нужно подбирать меньше параметров. Впрочем, если порядок слишком низок, оценка спектра будет слишком сглаженной. С другой стороны, слишком высокий порядок приведет к появлению ложных максимумов и спектральной неустойчивости. Для оценки порядка Экейком (Akaike) было предложено два параметра. Первый — это ошибка окончательного предсказания (Final Prediction Error — FPE), которая определяется следующим образом [1]:

$$FPE(p) = \frac{N + p}{N - p} E(p). \quad (11.60)$$

Второй параметр — это информационный критерий Экейка (Akaike Information Criterion), который определяется так [3]:

$$AIC(p) = N \ln E(p) + 2p. \quad (11.61)$$

Параметр $AIC(p)$ применяется для небольших наборов данных, тогда как $FPE(p)$ рекомендуется использовать для более крупных наборов. На практике обычно пытаются так выбрать p , чтобы одновременно минимизировать $FPE(p)$ и $AIC(p)$.

11.6. Сравнение методов оценки

Из непараметрических методов наибольшую добротность имеет метод Блэкмена–Тьюки, следовательно, он наиболее предпочтителен, хотя для удобства может выбираться и другой метод.

Параметрические методы предлагают большее разрешение по частоте и не требуют использования весовых функций. Метод максимального правдоподобия Кейпена дает несмещенную оценку с минимальной дисперсией, при которой спектральное разрешение находится между разрешением метода Бэга (метода наименьших квадратов без ограничений) и непараметрических методов. Для нестационарных данных лучше всего подходят методы адаптивной фильтрации, делающие акцент на наиболее свежих данных.

11.7. Примеры применения

11.7.1. Использование спектрального анализа с помощью ДПФ для различения болезней мозга

В работе [29] описано использование амплитудного и фазового спектров, полученных с помощью БПФ, для различения пациентов с болезнью Хантингтона, шизофренией, болезнью Паркинсона и здоровых пациентов путем анализа выбранных гармоник спектра возможной отрицательной вариации (contingent negative variation — CNV) на электроэнцефалограмме (ЭЭГ) исследуемых.

CNV — это зависящий от событий потенциал (Event-Related Potential — ERP), который проявляется как отрицательное смещение электрического потенциала на скальпе при вызове подходящего слухового раздражителя. Схематическое изображение сигнала CNV приведено на рис. 11.16. CNV — это отрицательный пик, находящийся между точками начал слуховых раздражителей S_1 и S_2 . Считается, что у людей, подверженных перечисленным выше заболеваниям, сигнал CNV искажен. Определяя спектр подходящих частей сигнала и статистически их обрабатывая, людей можно относить к одной из указанных выше категорий.

С помощью специализированной контрольно-измерительной системы обработки сигналов [25, 46] было записано несколько CNV для больных каждой группы и здоровых людей. Затем данные обрабатывались для уменьшения эффекта фоновой ЭЭГ и окулярных артефактов на сигнал CNV. Чтобы можно было провести сравнение во временной области и обеспечить надлежащую работу алгоритма удаления окулярных артефактов, был исключен средний уровень сигналов. Это привело к положительному смещению базовых линий до и после раздражения. Следовательно, для коррекции базовой линии



Рис. 11.16. Схематический сигнал CNV

потребовалось вычистить средние характеристики различных участков из соответствующих отрезков. Затем сигнал пропустили через цифровой фильтр нижних частот для удаления нежелательных высокочастотных компонентов ЭЭГ. При этом использовался не искажающий сигнал КИХ-фильтр. Затем применялся алгоритм удаления окулярных артефактов с помощью метода пропорциональных вычитаний (см. [28]). В каждой категории было обработано восемь сигналов CNV. Усредненные результаты показаны на рис. 11.17–11.20 (группа здоровых пациентов, группа с болезнью Хантингтона, группа шизофреников и группа с болезнью Паркинсона соответственно).

Затем два сегмента каждого сигнала CNV, имеющие длительность 512 мс (64 выборки), взвешивались с использованием функции Кайзера–Бесселя. Эксперименты показали, что при значении весового параметра $\alpha = 0,75$ достигается приемлемый компромисс между уровнем боковых лепестков и шириной основного лепестка. Для снижения гребешкового искажения к 64 выборкам было добавлено 960 дополняющих нулей. Затем находились ДПФ полученных наборов из 1024 элементов данных. К первым 96 гармоническим составляющим полученных спектров применялось четыре статистических теста (описаны в [26]). Данные тесты были названы *тестом на ближайшую и наиболее удаленную среднюю амплитуду*, *тестом разности средних амплитуд до и после раздражения*, *тестом Релея круговой дисперсии* и *модифицированным тестом Релея круговой дисперсии*. В результате были собраны необходимые тестовые статистики.

Чтобы уменьшить число тестовых статистик, выбрав из них наиболее селективные, использовали одномерный критерий, проверку по критерию Стьюдента и ступенчатый дискриминантный анализ. Все эти процедуры описаны в [29], кроме того, компания SAS выпустила пакет статистических программ, где реализованы все названные тесты [47].

После получения статистических результатов была выполнена классификация отдельных пар пациентов с помощью дискриминантного анализа [40]. Для реализации использовался пакет SAS, а более подробно этот вопрос освещен в [29]. Краткие результаты эксперимента приведены в табл. 11.2.

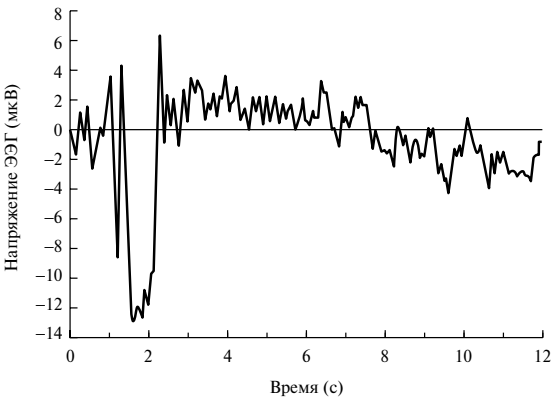


Рис. 11.17. Сигнал CNV здорового пациента после предварительной обработки и усреднения

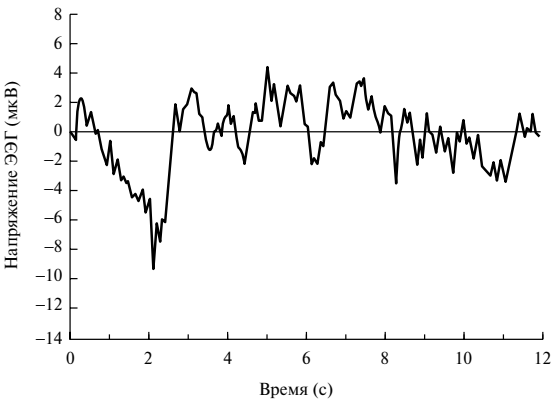


Рис. 11.18. Сигнал CNV пациента с болезнью Хантингтона после предварительной обработки и усреднения

Таблица 11.2. Результаты разделения по группам пациентов с заболеваниями головного мозга

Типы объектов		Процент правильной классификации	
Тип 1	Тип 2	Тип 1	Тип 2
БХ	контрольный	100	100
шизофреник	контрольный	95	100
БП	контрольный	93,8	87,5
БХ	шизофреник	100	90,9
БХ	БП	90,9	81,8
шизофреник	БП	81,3	93,8

БХ — пациент с болезнью Хантингтона, БП — пациент с болезнью Паркинсона, контрольный — контрольный пациент того же пола и возраста, что и тестируемый.

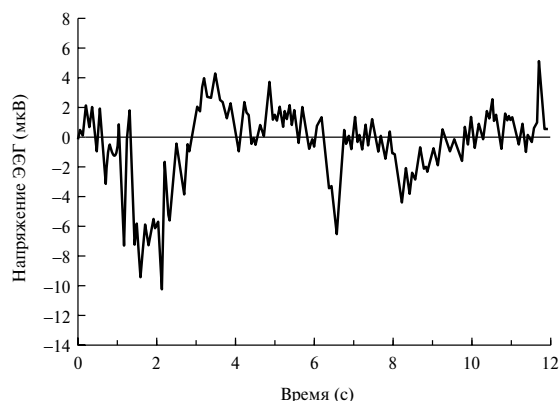


Рис. 11.19. Сигнал CNV пациента с шизофренией после предварительной обработки и усреднения

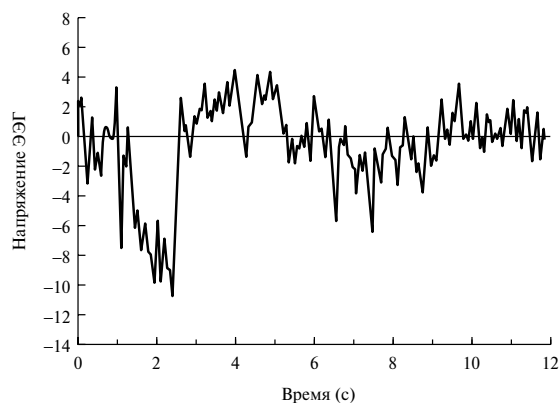


Рис. 11.20. Сигнал CNV пациента с болезнью Паркинсона после предварительной обработки и усреднения

Из результатов видно, что для сигналов CNV спектральный анализ весьма полезен, и что объединение этой методики со статистическими методами позволяет с высокой степенью точности различать здоровых пациентов, шизофреников и пациентов с болезнями Хантингтона и Паркинсона.

11.7.2. Спектральный анализ ЭЭГ с использованием авторегрессионного моделирования

Записанное изменение потенциала событий моделировалось как сумма измеренного сигнала электроэнцефалограммы (ЭЭГ) со смоделированным сигналом ERP. Смоделированный записанный потенциал ERP и смоделированный потенциал ERP изображены на рис. 11.21, *а*. На рис. 11.21, *б* иллюстрируется запись спектра плотности мощности смоделированного сигнала ERP с помощью метода быстрого преобразования Фу-

рье. Затем для получения спектра сигналов использовался метод авторегрессионного определения спектра, описанный в разделе 11.5. Для расчета параметров модели был использован ковариационный метод (см. раздел 11.5.4.2). Спектр амплитудной характеристики смоделированного записанного потенциала ERP показан на рис. 11.21, *в* для авторегрессионной модели порядка 50. Данный спектр более гладкий, чем спектр, полученный с помощью БПФ. Для сравнения на рис. 11.21, *г* показан спектр амплитудной характеристики измеренной ЭЭГ. Этот спектр содержит меньше энергии ниже частоты 2 Гц, чем спектр смоделированного сигнала ERP, чего следовало ожидать, поскольку энергия ERP сконцентрирована на низких частотах. Спектр амплитудной характеристики ERP показан на рис. 11.21, *д*. Похоже, что основная энергия сосредоточена в полосе до 2 Гц, что подтверждает предыдущие наблюдения. Сравнивая рис. 11.21, *г* и *д*, видим, что большая часть энергии ниже 2 Гц связана в основном с ERP, а не с ЭЭГ. Наконец, на рис. 11.21, *е* приведен спектр амплитудной характеристики ERP, но для модели порядка 6, а не 50. Данные спектры идентичны, следовательно, модели низкого порядка достаточно для определения спектра узкополосного ERP, тогда как модель большого порядка необходима для нахождения спектра ЭЭГ, который имеет более широкую полосу.

11.8. Резюме

В данной главе утверждается, что параметрические методы оценки спектра дают более достоверные результаты. Поскольку эти методы можно автоматизировать, они, скорее всего, предпочтительнее непараметрических методов, основанных на понятии периодограммы. Непараметрические методы менее надежны и более трудоемки, так как требуют для получения значимых результатов значительных исследований. Впрочем, их преимущества проявляются при использовании для различения болезней головного мозга.

11.9. Разобранный пример

Пример 11.3

Покажите, что оценка Барлетта спектральной плотности мощности асимптотически несмещенная, дисперсия оценки уменьшается с увеличением числа сегментов данных, и оценки спектра состоятельны. Какое влияние оказывает модифицированный метод периодограмм на разрешение по частоте?

Решение

В рассматриваемом методе вычисляется периодограмма K неперекрывающихся сегментов из M элементов данных. Если общее число элементов данных равно N , тогда $K = N/M$. Среднее K периодограмм равно оценке Барлетта спектральной плотности мощности $P_{BE}(f)$, где

$$P_{BE}(f) = \frac{1}{K} \sum_{j=0}^{K-1} P_j(f) \quad (11.62)$$

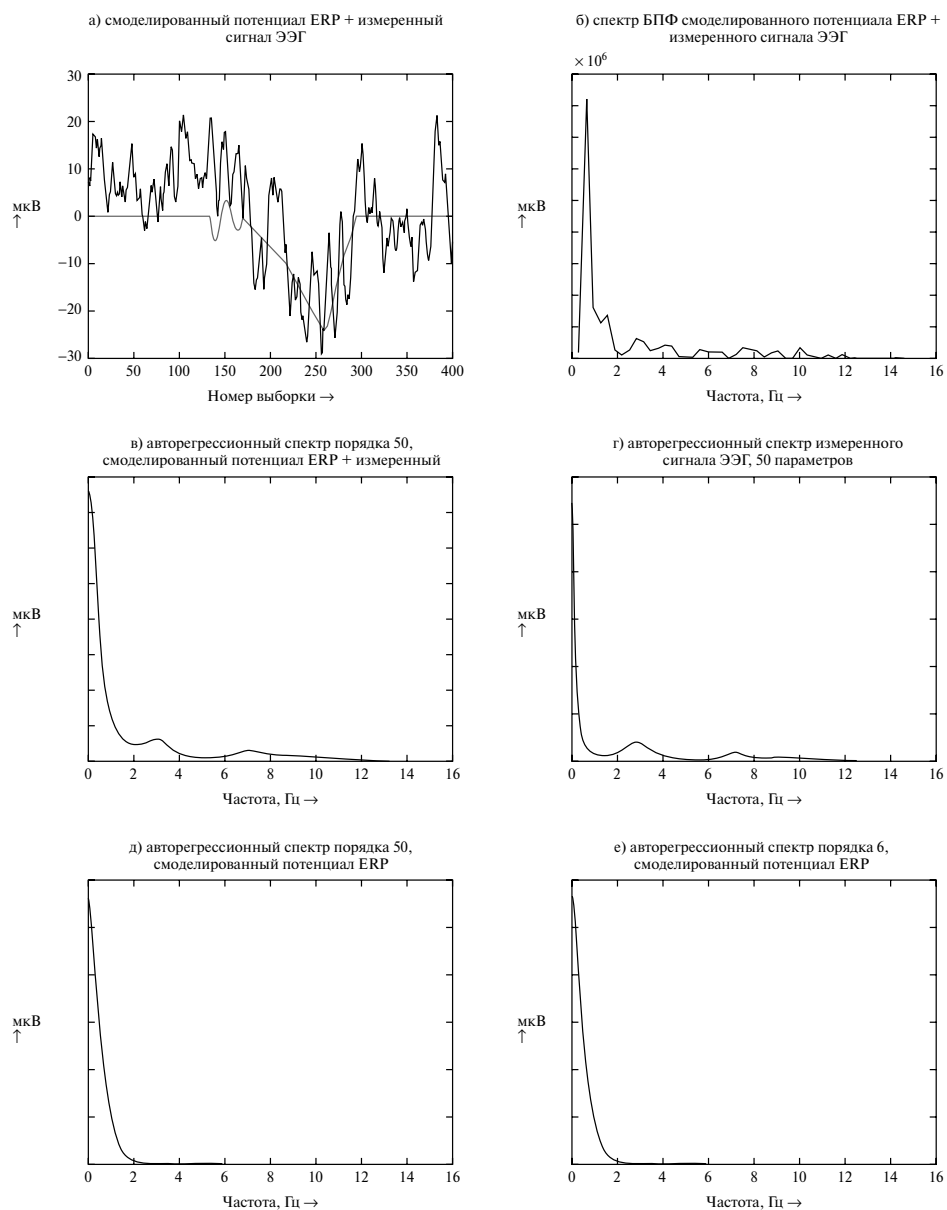


Рис. 11.21. Иллюстрация получения спектра авторегрессионным методом

и через j обозначен порядковый номер, а $P_j(f)$ — соответствующая i -я оценка спектральной плотности мощности.

Изучая математическое ожидание $P_{BE}(f)$, получаем

$$\begin{aligned} E[P_{BE}(f)] &= \frac{1}{K} \sum_{j=0}^{K-1} E[P_j(f)] = \\ &= E[P_j(f)] = \\ &= \sum_{m=-(M-1)}^{M-1} \left(1 - \frac{|m|}{M}\right) c_{xx}(m) \exp(-2\pi i f m). \end{aligned} \quad (11.63)$$

Для $M \gg |m|$ элемент весовой функции Барлетта $1 - |m|/M$ стремится к единице и $E[P_{BE}(f)]$ переходит в истинную спектральную плотность мощности $P(f)$. Таким образом, оценка Барлетта функции $P(f)$ является асимптотически несмещенной.

Теперь рассмотрим дисперсию:

$$\text{var}[P_{BE}(f)] = \frac{1}{K^2} \sum_{j=0}^{K-1} \text{var}[P_j(f)] = \frac{1}{K} \text{var}[P_j(f)]. \quad (11.64)$$

Следовательно, дисперсия уменьшается обратно пропорционально числу сегментов K , на которые разбита последовательность из N элементов данных. Поскольку дисперсия уменьшается с увеличением K , оценка Барлетта функции $P(f)$ состоятельна. Впрочем, поскольку число элементов данных, по которым вычисляются периодограммы, уменьшается с N в K раз до $M = N/K$, и спектральное разрешение также уменьшается в K раз. Следовательно, ширина основного лепестка при использовании метода Барлетта в K раз больше, чем при применении полного набора из N элементов данных.

Задачи

- 11.1. 1. Сигнал выбирается с частотой 30 кГц. Первые 524 288 выборок подвергаются быстрому преобразованию Фурье. Вычислите частоту первой гармоники и частотное разрешение спектра.
2. Если известно, что истинный спектр сигнала содержит синусоидальный компонент с частотой 5,7505 Гц, каким образом следует модифицировать данные, чтобы обеспечить надлежащее представление этого компонента в оцененном спектре?
- 11.2. Сигнал выбирается с частотой 8 кГц. Получены следующие уровни напряжения: 1,0; 1,0; 1,0; 1,0; 1,0; 1,0; 1,0; 1,0 В. Затем данные умножаются на весовую функцию, соответствующие дискретные значения которой — 0; 0,5; 1,0; 1,0; 1,0; 1,0; 0,5; 0,0. Определите ДПФ-образ, а следовательно, и Фурье-образ взвешенных данных.

- 11.3. Вычислите гребешковое искажение и ширину полосы шумового эквивалента данных, представленных в задаче 11.2.
- 11.4. Используя рис. 11.10, 11.12 и 11.13, оцените ширину полосы шумового эквивалента и выигрыш от обработки следующих весовых функций (в каждом случае используйте восемь выборов):
- прямоугольная;
 - Тьюки (спад по косинусу), $\alpha = 0, 1$;
 - Тьюки, $\alpha = 0, 5$;
 - Хэмминга, $\alpha = 0, 54$;
 - Кайзера–Бесселя, $\alpha = 2, 0$;
 - Кайзера–Бесселя, $\alpha = 4, 0$.
- 11.5. Импульсный телекоммуникационный сигнал выбирается каждые 0,167 мкс, в результате получаются такие выборки: 0, 0, 1, 1, 1, 1, 0 В. Ненулевые значения умножаются на весовую функцию Кайзера–Бесселя с $\alpha = 4, 0$. Вычислите энергетический спектр взвешенного импульса с использованием восьмиточечного БПФ (или ДПФ). (Весовая функция представлена на рис. 11.13, в.)
- 11.6. Определите гребешковое искажение, потери при обработке и наихудшие потери от обработки весовой функцией из задачи 11.5.
- 11.7. По данным задачи 11.5 определите наихудшие потери от обработки и амплитуду первого бокового лепестка весовой функции Кайзера–Бесселя с $\alpha = 4, 0$, функции Хэмминга с $\alpha = 0, 54$, функции Тьюки с $\alpha = 0, 5$ и прямоугольной весовой функции. Результаты сведите в таблицу и выберите наиболее подходящую весовую функцию. (Для получения оценок весовых функций используйте рис. 11.13, в, 11.12, а и 11.10, б.)
- 11.8. Сигнал, который выбирается с частотой 8 кГц, дает следующие выборки: 0; 4, 0; 2, 4; 1, 0; –1, 0; –3, 8; –1, 3; 0 В. Вычислите и изобразите энергетический спектр:
- воздействовав на данные весовой функцией Кайзера–Бесселя с $\alpha = 2, 0$;
 - после модификации данных согласно формуле (11.21);

$$S(n) = w(n) \left[s(n) - \frac{\sum_{n=0}^{N-1} w(n)s(n)}{\sum_{n=0}^{N-1} w(n)} \right] \times \left[\frac{N}{\sum_{n=0}^{N-1} w^2(n)} \right]^{1/2},$$

где $w(n)$ — дискретные значения весовой функции Кайзера–Бесселя с $\alpha = 2, 0$;

- объясните причины отличий результатов, полученных в п. а и б.
- 11.9. Получите энергетический спектр дискретной последовательности данных $\{0, 1, 0, 1, 0, 1, 0, 1\}$ и сравните его со спектром сигнала прямоугольной формы (мандра).

- 11.10. С помощью метода модифицированных периодограмм Барлетта получите энергетический спектр данных из задачи 11.9, разбив данные на две неперекрывающиеся последовательности: $\{0, 1, 0, 1\}$ и $\{0, 1, 0, 1\}$.
- 11.11. С помощью метода модифицированных периодограмм Уэлша получите энергетический спектр данных из задачи 11.9, разбив данные на три последовательности равной длины с перекрытием 50%.
- 11.12. Пусть последовательность данных из задачи 11.9 искажается случайным шумовым компонентом и становится равной $\{0, 763; 1, 656; 0, 424; 1, 939; 0, 133; 1, 881; 0, 328; 1, 348\}$. Вычислите энергетический спектр зашумленных данных и сравните его со спектром исходных данных. Оцените по выборкам отношение сигнал-шум.
- 11.13. Повторно решите предыдущую задачу, используя модифицированный метод периодограмм Барлетта и две неперекрывающиеся последовательности данных $\{0, 763; 1, 656; 0, 424; 1, 939\}$ и $\{0, 133; 1, 881; 0, 328; 1, 348\}$.
- 11.14. Повторно решите задачу 11.11, используя зашумленные данные из задачи 11.12.
- 11.15. Пусть исходная последовательность данных из задачи 11.9 значительно искажена шумом и перешла в последовательность $\{6, 03; 6, 18; 3, 35; 8, 42; 1, 05; 7, 96; 2, 59; 3, 75\}$. Вычислите энергетический спектр этих данных и оцените по выборкам отношение сигнал-шум.
- 11.16. Получите улучшенную оценку энергетического спектра данных из задачи 11.15, используя метод Барлетта с двумя последовательностями равной длины.
- 11.17. Улучшите качество оценки энергетического спектра из задачи 11.15, используя метод модифицированных периодограмм Уэлша с тремя последовательностями равной длины с 50%-ным перекрытием.
- 11.18. Составьте таблицу для сравнения различных методов оценки спектра и отношений сигнал-шум (результаты задач 11.9–11.17). Обсудите влияние отношения сигнал-шум на различные методы и выберите наиболее понравившийся метод.
- 11.19. Получите спектр плотности мощности данных из задачи 11.9, т.е. $\{0, 1, 0, 1, 0, 1, 0, 1\}$, используя метод Блэкмена–Тьюки и сравните результат с полученным в задаче 11.9. Используйте весовую функцию $\{0; 0, 5; 1; 1; 1; 1; 0, 5; 0\}$.
- 11.20. Повторно решите задачу 11.19, используя зашумленные данные из задачи 11.12.
- 11.21. Повторно решите задачу 11.19, используя зашумленные данные из задачи 11.15.
- 11.22. Сравните результаты, полученные при решении задач 11.12–11.17, и выясните, какой метод лучше при наличии шума — периодограмм или Блэкмена–Тьюки.
- 11.23. Определите энергетический спектр меандра с периодом 1,0 мкс, который попеременно принимает значения 0 и 5 В, используя метод Блэкмена–Тьюки. Сравните ваш результат с теоретическим.
- 11.24. Напишите компьютерную программу генерации желаемого сигнала и разработайте подходящие процедуры анализа спектра, используя описанные в данной главе методы оценки энергетического и фазового спектров.
- 11.25. Придумайте несколько амплитудных и фазовых спектров, демонстрирующих некоторые интересные особенности (например, близлежащие максимумы), на

которых можно проверять методы оценки спектра. Преобразуйте их во временную область, а затем добавьте шум, чтобы получить а) малое отношение сигнал-шум; б) единичное отношение сигнал-шум; в) большое отношение сигнал-шум. Теперь рассчитайте и обсудите амплитудные и фазовые спектры.

Литература

1. Akaike H. (1969) Fitting autoregressive models for prediction. *Ann. Institute of Statistical Mathematics*, **21**, 243–247.
2. Akaike H. (1973) Information theory and an extension of the maximum likelihood principle. In *2nd International Symposium on Information Theory* (Petrov B. N. and Csaki F. (eds)), pp. 267–281. Budapest: Akademiai Kiade.
3. Akaike H. (1974) A new look at the statistical model identification. *IEEE Trans. Automatic Control*, **19**, 716–722.
4. Akaike H. (1978) A Bayesian analysis of the minimum AIC procedure. *Ann. Institute of Statistical Mathematics*, **30A**, 9–14.
5. Akaike H. (1979) A Bayesian extension of the minimum AIC procedure of autoregressive model fitting. *Biometrika*, **66**, 237–242.
6. Blackman R. B. and Tukey J. W. (1958) *The Measurement of Power Spectra*. New York: Dover.
7. Box G. E. P. and Jenkins G. M. (1976) *Time-Series Analysis, Forecasting, and Control*. San Francisco CA: Holden-Day.
8. Burg J. P. (1967) Maximum entropy spectral analysis. In *Proc. 37th Meeting Society Exploration Geophysicists*, Oklahoma City, October. Reprinted in Childers D. G. (ed.) (1968) *Modern Spectrum Analysis*. New York: IEEE Press.
9. Burg J. P. (1968) A new analysis technique for time series data. In *NATO Advanced Study Institute on Signal Processing with Emphasis on Underwater Acoustics*, 12–23 August. Reprinted in Childers D. G. (ed.) (1968) *Modern Spectrum Analysis*. New York: IEEE Press.
10. Cadzow J. A. (1979) ARMA spectral estimation: an efficient closed-form procedure. In *Proc. RADC Spectrum Estimation Workshop*, Rome NY, October 1979, pp. 81–97.
11. Cadzow J. A. (1982) Spectral estimation: an overdetermined rational model equation approach. *Proc. IEEE*, **70**, 907–938.
12. Candy J. V. (1989) *Signal Processing: The Modern Approach*. New York: McGraw-Hill.
13. Capon J. (1969) High-resolution frequency-wavenumber spectrum analysis. *Proc. IEEE*, **57**, 1408–1418.
14. Chatfield C. (1979) Inverse autocorrelation. *J. Royal Statistical Society A*, **142**, 363–377.
15. Chatfield C. (1984) *The Analysis of Time Series*, 3rd edn. London: Chapman and Hall.
16. Clarkson P. M. (1993) *Optimal and Adaptive Signal Processing*. Boca Raton FL: CRC Press.
17. Cooley J. W. and Tukey J. W. (1965) An algorithm for the machine calculation of complex Fourier series. *Mathematics of Computation*, **19**, 297–301.
18. DeFatta D. J., Lucas J. G. and Hodgkiss W. S. (1988) *Digital Signal Processing: A System Design Approach*, Section 6.6.5, p. 263. New York: Wiley.
19. Durbin J. (1959) Efficient estimation of parameters in moving-average models. *Biometrika*, **46**, 306–316.
20. Friedlander B. (1982) Lattice methods for spectral estimation. *Proc. IEEE*, **70**, 990–1017.
21. Gersch W. (1970) Spectral analysis of EEGs by autoregressive decomposition of time series. *Mathematical Biosciences*, **7**, 205–222.
22. Graupe D., Krause D. J. and Moore J. B. (1975) Identification of autoregressive-moving average parameters of time series. *IEEE Trans. Automatic Control*, **20**, 104–107.
23. Harris F. J. (1978) On the use of windows for harmonic analysis with the discrete Fourier transform. *Proc. IEEE*, **66**(1), 51–84.
24. Jenkins G. M. and Watts D. G. (1968) *Spectral Analysis and its Applications*. San Francisco CA: Holden-Day.

25. Jervis B. W. and Saatchi M. R. (1990) An integrated system for process control and the acquisition, storage and processing of data. In *IEE Colloq. on PC-Based Instrumentation*, 31 January, IEE Digest No. 1990/025.
26. Jervis B. W., Nichols M. J., Johnson T. E., Allen E. M. and Hudson N. R. (1983) A fundamental investigation of the composition of auditory evoked potentials. *IEEE Trans. Biomedical Engineering*, **30**(1), 43–50.
27. Jervis B. W., Coelho M. and Morgan G. W. (1989a) Spectral analysis of EEG responses. *Medical and Biological Engineering and Computing*, **27**, 230–238.
28. Jervis B. W., Coelho M. and Morgan G. W. (1989b) Effect on EEG responses of removing ocular artefacts by proportional EOG subtraction. *Medical and Biological Engineering and Computing*, **27**, 484–490.
29. Jervis B. W., Saatchi M. R., Allen E. M., Hudson N. R., Oke S. and Grimsley M. (1993) A pilot study of computerised differentiation of Huntington's disease, schizophrenia, and Parkinson's disease patients using the contingent negative variation. *Medical and Biological Engineering and Computing*, 31 (January), 31–38.
30. Kalouptsidis N. and Theodoridis S. (1987) Fast adaptive least-squares algorithms for power spectral estimation. *IEEE Trans. Acoustics, Speech and Signal Processing*, **35**, 661–670.
31. Kay S. M. (1980) A new ARMA spectral estimator. *IEEE Trans. Acoustics, Speech and Signal Processing*, **28**, 585–588.
32. Kay S. M. (1988) *Modern Spectral Estimation: Theory and Application*. Englewood Cliffs NJ: Prentice-Hall.
33. Kuo F. F. and Kaiser J. F. (1966) *System Analysis by Digital Computer*, Chapter 7, pp. 232–238. New York: Wiley.
34. Lacoss R. T. (1971) Data adaptive spectral analysis methods. *Geophysics*, **36**, 661–675.
35. Lawson C. L. and Hanson R. J. (1974) *Solving Least Squares Problems*. Englewood Cliffs NJ: Prentice-Hall.
36. Levinson N. (1947) The Weiner RMS error criterion in filter design and prediction. *J. Mathematical Physics*, **25**, 261–278.
37. Makhoul J. (1975) Linear prediction: a tutorial review. *Proc. IEEE*, **63**, 561–580.
38. Marple S. L. (1980) A new autoregressive spectrum analysis algorithm. *IEEE Trans. Acoustics, Speech and Signal Processing*, **28**, 441–454.
39. Marple S. L. (1987) *Digital Spectral Analysis*. Englewood Cliffs NJ: Prentice-Hall.
40. Morrison D. F. (1976) *Multivariate Statistical Methods*, 2nd edn. New York: McGraw-Hill.
41. Nuttall A. H. (1976) *Spectral Analysis of a Univariate Process with Bad Data Points via Maximum Entropy and Linear Predictive Techniques*. NUSC Technical Report TR-5303, New London CN.
42. Pardey J., Roberts S. and Tarassenko L. (1996) A review of parametric modelling techniques for EEG analysis. *Medical Engineering Physics*, **18**, 2–11.
43. Priestley M. B. (1981) *Spectral Analysis and Time Series*, Volume 1, *Univariate Series*, Chapter 6 and 7. New York: Academic Press.
44. Proakis J. G. and Manolakis D. G. (1989) *Introduction to Digital Signal Processing*, Sections 1.3.2, 11.2.4, 11.3 and 11.3.4 and Appendix 6A. Basingstoke: Macmillan.
45. Rissanen J. (1983) A universal prior for the integers and estimation by minimum description length. *Ann. Statistics*, **11**, 417–431.
46. Saatchi M. R. and Jervis B. W. (1991) PC-based integrated system developed to diagnose specific brain disorders. *Computing and Control Engineering J.*, **2**(2), 61–68.
47. SAS (1982) *SAS User Guide*. SAS Institute.
48. Shibata R. (1976) Selection of the order of an autoregressive model by Akaike's information criterion. *Biometrika*, **63**, 117–126.
49. Ulrych T. J. and Clayton R. W. (1976) Time series modelling and maximum entropy. *Physics Earth and Planetary Interiors*, **12**, 188–200.
50. Welch P. D. (1967) The use of fast Fourier transform for the estimation of power spectra. *IEEE Trans. Audio and Electroacoustics*, **15**, 70–73.
51. Whittle P. (1965) *Prediction and Regulation*. London: English Universities Press.
52. Wold H. (1954) *A Study of the Analysis of Stationary Time Series*. Stockholm: Almqvist and Wichsells.

Приложение

11.A. Программы MATLAB для оценки и анализа спектра

Средство MATLAB под названием Signal Processing Toolbox содержит массу полезных функций для оценки спектра и спектрального анализа с использованием параметрических и непараметрических методов. Для иллюстрации использования этих функций на практике мы разработали следующие полезные программы:

welchm.m	программа, иллюстрирующая оценку и анализ спектральной плотности мощности с помощью метода Уэлша;
burgm.m	программа, иллюстрирующая оценку и анализ спектральной плотности мощности с помощью метода Бэга;
yulewalkm.m	программа, иллюстрирующая оценку и анализ спектральной плотности мощности с помощью метода Юла–Уокера.

Все программы можно найти в Web.

Универсальные и специализированные процессоры ЦОС

12

12.1. Вступление	784
12.2. Компьютерные архитектуры обработки сигналов	784
12.3. Универсальные процессоры ЦОС	804
12.4. Выбор цифрового процессора сигналов	818
12.5. Реализация алгоритмов ЦОС на универсальных процессорах ЦОС	820
12.6. Специализированная аппаратура ЦОС	847
12.7 Резюме	852
Задачи	852
Литература	857
Дополнительная литература	857
Приложение	859

В данной главе объясняются ключевые фундаментальные вопросы, связанные с универсальными и специализированными процессорами для цифровой обработки сигналов (ЦОС), влиянием алгоритмов ЦОС на аппаратную и программную архитектуру данных процессоров, а также с реализацией ключевых алгоритмов ЦОС на блоке специализированной аппаратуры или на универсальных процессорах ЦОС для работы в реальном времени.

Обработка в реальном времени обычно означает “как можно скорее”, но в заданных временных рамках. Обработку в реальном времени можно разбить на две обширные категории (хотя данные категории также можно разбить на более мелкие классы): потоковая обработка, например, цифровая фильтрация, где данные обрабатываются по выборке за такт, и блочная обработка, например, нахождение БПФ и корреляции, где за такт обрабатывается блок данных фиксированной длины. Отметим, что для реализации алгоритмов ЦОС в реальном времени требуется и программное, и аппаратное обеспечение. В качестве аппаратной поддержки можно использовать матрицы процессоров, стандартные микропроцессоры, чипы ЦОС или микропрограммируемые специализированные устройства. Программной поддержкой служат коды или микрокоды на языке ассемблера низкого уровня, “родном” для аппаратуры ЦОС, и/или коды на эффективном языке высокого уровня, таком как С или С++. В настоящее время обычно используют именно языки высокого уровня, особенно с новейшими процессорами ЦОС, имеющими сложную и запутанную структуру.

Процессоры ЦОС, появившиеся в начале 1980-х, стали в настоящее время значительно сложнее, повысили свои возможности и расширили число сфер применения. Все это также привело к значительному увеличению числа доступных процессоров ЦОС. Чтобы помочь читателю сориентироваться в выборе необходимого процессора ЦОС, в главу включен обзор основных отличий поколений процессоров ЦОС с фиксированной или плавающей запятой.

12.1. Вступление

Процессоры ЦОС можно для удобства разделить на две обширные категории: универсальные и специализированные. В число распространенных процессоров ЦОС входят такие устройства с фиксированной запятой, как TMS320C54x от Texas Instruments и DSP563x от Motorola, и такие устройства с плавающей запятой, как TMS320C4x (Texas Instruments) и ADSP21xxx SHARC (Analog Devices).

Существует два типа специализированных устройств.

1. Аппаратное обеспечение, разработанное для эффективного выполнения специальных алгоритмов ЦОС, таких как цифровая фильтрация, быстрое преобразование Фурье. Устройства данного типа иногда называют *алгоритмическими процессорами ЦОС*.
2. Аппаратное обеспечение, разработанное для специального приложения, например, в сфере контроля, телекоммуникаций или цифрового аудио. Устройства данного типа иногда называют *процессорами ЦОС специального назначения (специализированными)*.

В большинстве случаев специализированные процессоры выполняют такие алгоритмы ЦОС, как кодирование-декодирование РСМ. Кроме того, они должны выполнять другие операции, отражающие специфику приложения. В качестве примеров специализированных процессоров ЦОС можно привести процессор Cirrus CS8420 для конвертеров частоты дискретизации в цифровой аудиоаппаратуре, подавитель речевого эха в многоканальной телефонии Mitel MT9300, процессор БПФ PDSP16515A и программируемый КИХ-фильтр VPDSP16256.

Все универсальные и специализированные процессоры можно построить с помощью отдельных чипов или блоков умножителей, АЛУ, ячеек памяти и т.д.

Рассмотрим архитектурные особенности процессоров ЦОС, которые позволили применить цифровую обработку в реальном времени во многих областях.

12.2. Компьютерные архитектуры обработки сигналов

Большинство доступных сейчас универсальных процессоров основаны на архитектуре фон Неймана, при которой операции выполняются последовательно¹. Упрощенная архитектура стандартного процессора фон Неймана показана на рис. 12.1. При обработке инструкции в таком процессоре блоки процессора, не задействованные в каждой фазе инструкции, находятся в холостом состоянии до передачи им управления. Повышение скорости процессора достигается за счет ускорения работы отдельных блоков, хотя существует определенный предел, ограничивающий их возможную скорость работы.

Если устройство должно работать в реальном времени, архитектуру процессора ЦОС нужно оптимизировать под выполнение функций ЦОС. Например, на рис. 12.2 показана общая аппаратная архитектура, подходящая для цифровой обработки сигналов в реальном времени. Она характеризуется следующими особенностями.

¹ Уже нет — в большинство (чтобы не сказать — во все) универсальные процессоры встроено как минимум нефоннеймановское ядро (MMX, 3DNow и другие технологии, ставшие аппаратным ширпотребом). — Прим. ред.

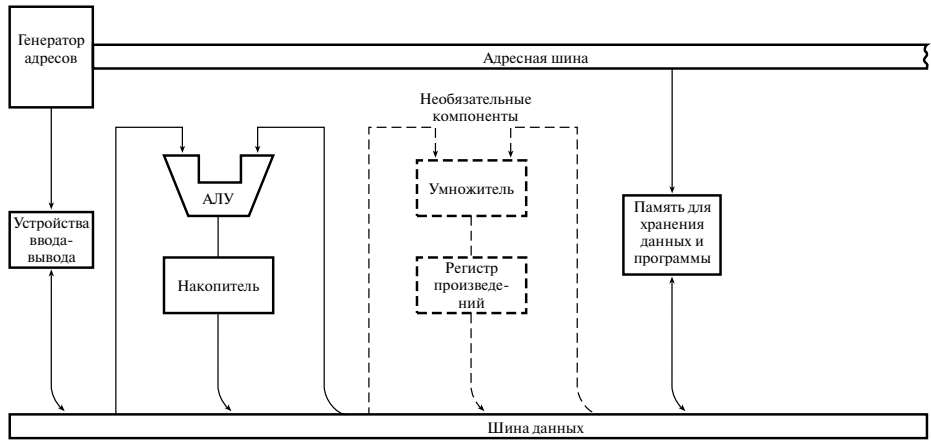


Рис. 12.1. Упрощенная архитектура стандартных микропроцессоров

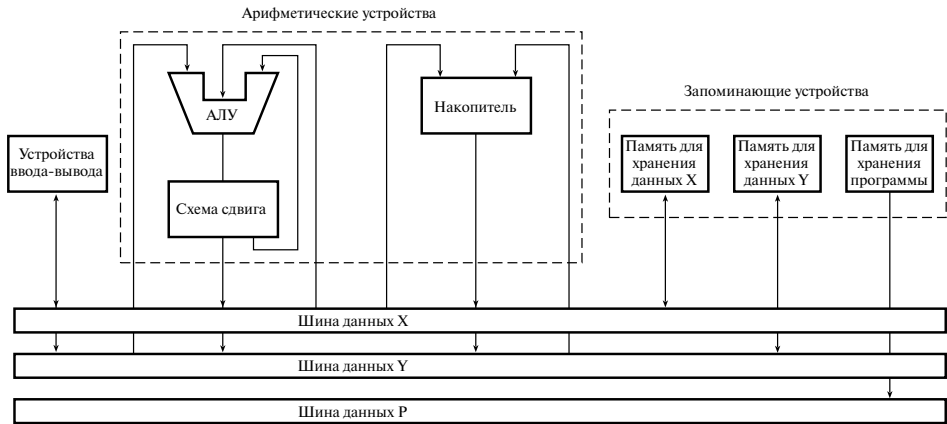


Рис. 12.2. Стандартная универсальная аппаратная архитектура обработки сигналов

- Многошинная структура с отдельной памятью для данных и инструкций программы. Обычно память для хранения данных содержит входные данные, промежуточные значения и выходные выборки, а также фиксированные коэффициенты, например, для цифровой фильтрации или БПФ. Команды программы хранятся в специально отведенных ячейках памяти.
- Порт ввода-вывода позволяет обмениваться данными со внешними устройствами (АЦП, ЦАП) или передавать цифровые данные другим процессорам. Прямой доступ к памяти (Direct Memory Access — DMA) (если он есть) позволяет быстро обмениваться блоками данных с памятью (ОЗУ) для хранения данных, причем обычно это происходит под внешним управлением.
- Арифметические устройства для логических и арифметических операций, в число которых выходят АЛУ, аппаратные умножители и схемы сдвига (или умножители-накопители).

Почему необходима именно такая архитектура? Большинство алгоритмов ЦОС (фильтрация, корреляция, преобразование Фурье) включают повторяющиеся арифметические операции, такие как умножение, сложение, обращение к памяти и интенсивная передача данных через центральный процессор. Архитектура стандартных микропроцессоров не предназначена для такого типа деятельности. При разработке аппаратуры ЦОС важно оптимизировать под операции ЦОС и аппаратную архитектуру, и систему команд. В процессорах ЦОС для этого широко используется концепция параллелизма. В частности, применяются следующие средства:

- гарвардская архитектура;
- конвейерная обработка;
- быстрые специализированные аппаратные умножители-накопители;
- специальные команды, предназначенные для ЦОС;
- копирование;
- встроенная память/кэш;
- расширенный параллелизм — векторная архитектура (SIMD), архитектура с командными словами сверхбольшой длины (VLIW) и статическая суперскалярная обработка.

Для успешной разработки структуры ЦОС важно понимать и правильно использовать все названные ключевые архитектурные особенности.

12.2.1. Гарвардская архитектура

Принципиальная особенность гарвардской (двухшинной) архитектуры заключается в том, что память для хранения данных и память для хранения программы располагаются в разных местах, допуская полное совмещение во времени операций вызова команды из памяти и ее выполнения. Стандартные микропроцессоры, такие как Intel 6502, характеризуются одношинной структурой, через которую передаются и данные, и команды, как показано на рис. 12.1.

Предположим, что при использовании стандартного микропроцессора требуется считать значение `op1`, которое находится по адресу памяти `ADR1`, в аккумулятор, а затем

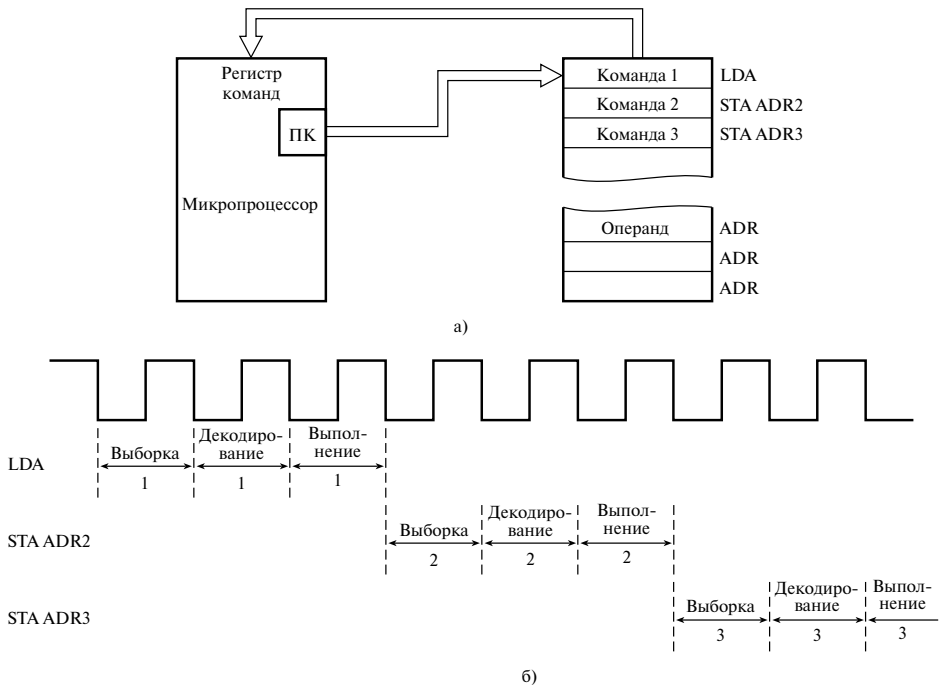


Рис. 12.3. Иллюстрация выборки, декодирования и выполнения команды в негарвардской архитектуре с единой областью памяти: а) вызов команды из памяти; б) временная диаграмма процесса

сохранить его по двум другим адресам, ADR2 и ADR3. В этом случае могут использоваться такие команды:

LDA	ADR1	загрузить операнд op1 в аккумулятор из ячейки ADR1
STA	ADR2	сохранить op1 по адресу ADR2
STA	ADR3	сохранить op1 по адресу ADR3

Обычно каждая из приведенных команд включает три различных этапа:

- выборка команды из памяти;
- декодирование команды;
- выполнение команды.

В нашем случае выборка команды включает вызов следующей команды из памяти, а выполнение команды предполагает чтение данных из памяти либо запись данных в память. В стандартном процессоре без гарвардской архитектуры команды программы (т.е. программный код) и данные (операнды) содержатся в одной области памяти (см. рис. 12.3). Следовательно, вызов следующей команды при выполнении текущей невозможен, поскольку обе операции требуют обращения к памяти.

В гарвардской архитектуре (рис. 12.4), где команды программы и данные хранятся в различных областях памяти, вызов следующей команды может совпадать с выполнением текущей команды (см. рис. 12.5). Обычно память программы содержит программный

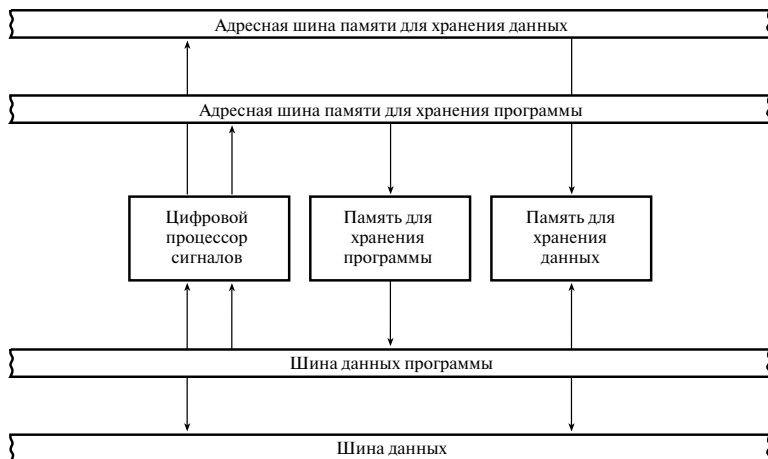


Рис. 12.4. Стандартная гарвардская архитектура с отдельными областями памяти для хранения данных и программы. Вызов команд данных и команд программы может выполняться одновременно, поскольку используются две независимые памяти

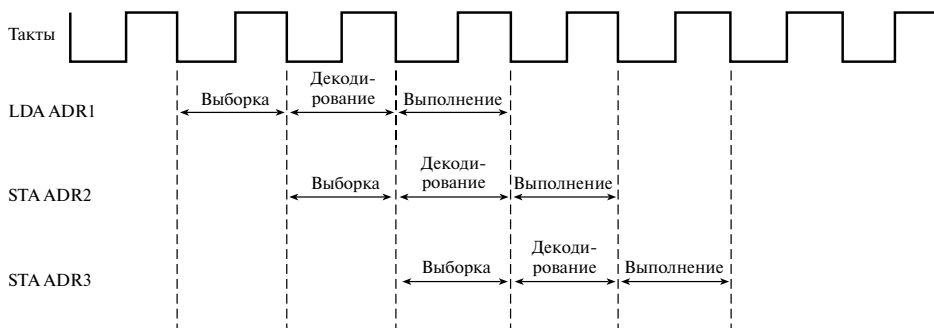


Рис. 12.5. Иллюстрация совмещения команд, возможного при использовании гарвардской архитектуры

код, тогда как память для хранения данных включает переменные, например, выборки входных данных.

Строгая гарвардская архитектура используется лишь в нескольких процессорах ЦОС (например, Motorola DSP56000), в большинстве устройств применяется модифицированная гарвардская архитектура (например, семейство процессоров TMS320). В модифицированной архитектуре также выделяются отдельные области памяти для хранения программы и данных, но в отличие от строгой гарвардской архитектуры здесь разрешена связь между двумя областями памяти.

12.2.2. Конвейерная обработка

Конвейерная обработка — это метод, который позволяет совмещать несколько операций в процессе выполнения. При конвейерной обработке задача разбивается на несколь-

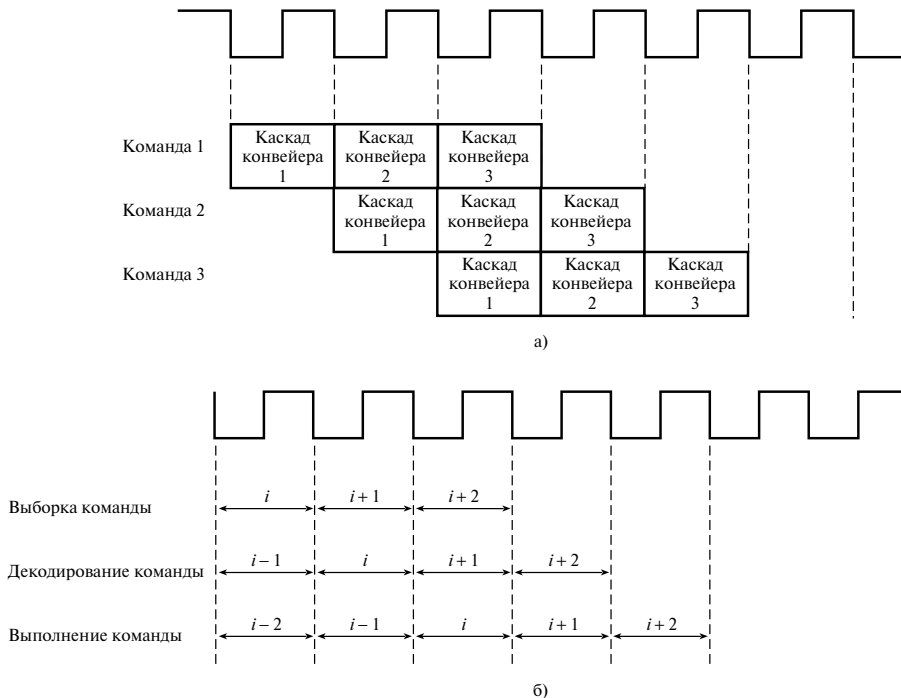


Рис. 12.6. Иллюстрация концепции конвейерной обработки

ко различных подзадач, которые затем совмещаются при выполнении. Данный метод широко используется в цифровой обработке сигналов для повышения скорости. Конвейер в цифровой обработке подобен обычной производственной линии на фабрике, например, предприятию по сборке телевизоров или автомобилей. Как и на производственной линии, задача разбивается на небольшие независимые подзадачи, именуемые *каскадами конвейера*. Последовательное соединение каскадов формирует конвейер.

Как было показано в предыдущем разделе, команду можно разбить на три этапа. Каждый этап команды рассматривается как каскад конвейера. Таким образом можно организовать наложение команд, при котором новая команда будет начинать выполняться в первый момент каждого такта (рис. 12.6, а).

На рис. 12.6, б приведена временная диаграмма трехкаскадного конвейера, на которой выделены этапы команд. Обычно каждый этап в конвейере занимает один машинный цикл. Следовательно, в течение данного цикла могут быть одновременно активны до трех различных команд, хотя все они будут на разных стадиях завершения. Основным момент при использовании конвейерной обработки команд заключается в том, что три части команды (т.е. выборка из памяти, декодирование и выполнение) независимы, так что выполнение нескольких команд может совмещаться. На рис. 12.6, б показано, что в ходе i -го цикла процессор может одновременно извлекать из памяти i -ю команду, декодировать $(i - 1)$ -ю команду и в то же время выполнять $(i - 2)$ -ю команду.

Трехкаскадный конвейер, рассмотренный выше, основан на технике, которая применяется в процессорах TMS320 (Texas Instruments). В том же процессоре TMS320 конвейерная обработка используется и по-другому, когда для получения конвейера применяется несколько регистров. Счетчик упреждающей выборки содержит адрес следующей команды, которая будет вызываться из памяти, регистр команд содержит команду, которая будет выполняться, и регистр команды в очереди хранит команды, которые будут выполняться при выполнении текущей команды. Адрес следующей выполняемой команды фиксирует программный счетчик.

Благодаря использованию внутреннего параллелизма потока команд конвейерная обработка позволяет существенно снизить в среднем время выполнения одной команды. Пропускная способность машины с конвейерной обработкой определяется числом команд, пропущенных через конвейер за единицу времени. Как и в производственной линии, все каскады конвейера должны быть синхронизированы. Время перехода команды на следующий этап (см. рис. 12.6, *а*) равно одному циклу и зависит от наиболее медленного каскада конвейера. В идеальном конвейере среднее время выполнения одной команды равно [6]

$$\frac{\text{время на команду (без конвейерной обработки)}}{\text{число каскадов конвейера}}. \quad (12.1)$$

В идеальном случае увеличение скорости равно числу каскадов конвейера (частей, на которые разбита команда). На практике увеличение скорости будет меньше из-за служебных издержек на организацию конвейера, задержек в регистрах конвейера и т.п.

Пример 12.1

В машине без конвейерной обработки вызов, декодирование и выполнение команды занимает 35, 25 и 40 нс соответственно. Определите увеличение пропускной способности, если этапы команд обрабатываются конвейером. Предполагайте, что служебные издержки составляют 5 нс на каждом этапе, и что другие задержки несущественны.

Решение

В машине без конвейерной обработки среднее время выполнения команды просто равно сумме времен, необходимых на реализацию отдельных ее этапов: $35 + 25 + 40 = 100$ нс. В то же время, если предположить, что процессор имеет фиксированный машинный цикл, и что этапы команды синхронизированы по системным часам, завершение каждой команды потребует трех машинных циклов: $40 \times 3 = 120$ нс. Эта величина соответствует пропускной способности $8,3 \times 10^6$ команд с^{-1} .

В машине с конвейерной обработкой тактовая частота определяется скоростью самого медленного каскада плюс служебные издержки. В нашем случае машинный цикл равен $40+5=45$ нс, что вводит ограничение на среднее время выполнения команды. Пропускная способность (при заполненном конвейере) равна $22,2 \times 10^6$ команд с^{-1} . Следовательно,

$$\text{ускорение вычислений} = \frac{\text{среднее время выполнения команды (без конвейера)}}{\text{среднее время выполнения команды (с конвейером)}} = \quad (12.2)$$

$$= \frac{120}{45} = 2,67 \text{ раза (предполагается, что без конвейера команда выполняется за три цикла)}$$

В машине с конвейерной обработкой каждая команда по-прежнему требует трех тактов, но в каждом такте процессор занимается выполнением до трех различных команд. Конвейерная обработка увеличивает пропускную способность системы, но не время выполнения каждой команды. Как правило, из-за служебных издержек на организацию конвейера время выполнения каждой команды даже немного увеличивается.

Конвейерная обработка существенно влияет на память системы. Число обращений к памяти в машине с конвейерной обработкой увеличивается, особенно с увеличением числа этапов. При цифровой обработке сигналов конвейерная обработка сигналов облегчается при использовании гарвардской архитектуры, где данные и команды размещаются в различных областях памяти.

При последовательном соединении такого медленного устройства, как память для хранения данных, и арифметического элемента, последний часто простаивает довольно долго. В таких случаях можно применять конвейерную обработку, что позволит полнее использовать возможности арифметического устройства. Ниже данная концепция иллюстрируется на примере.

Пример 12.2

Большинство алгоритмов ЦОС характеризуются операциями умножения-накопления, например:

$$a_0x(n) + a_1x(n-1) + a_2x(n-2) + \dots + a_{N-1}x(n-(N-1)).$$

На рис. 12.7 представлена конфигурация системы без конвейерной обработки, в которой приведенное выше уравнение реализовано с помощью арифметического элемента. Транспортные задержки для памяти, умножителя и накопителя равны соответственно 200, 100 и 100 нс.

1. Чему равна пропускная способность системы?
2. Измените конфигурацию системы для конвейерной обработки с двукратным увеличением скорости. Проиллюстрируйте операции в новой конфигурации с помощью временной диаграммы.

Решение

1. Коэффициенты a_k и массивы данных хранятся в памяти, как показано на рис. 12.7, а. В режиме без конвейерной обработки коэффициенты и данные вызываются последовательно и подаются на умножитель. Произведения суммируются в накопителе (накапливающем сумматоре). Последовательные операции умножения-накопления будут завершаться каждые 400 нс (200+100+100), так что пропускная способность равна $2,5 \times 10^6$ операций с^{-1} .

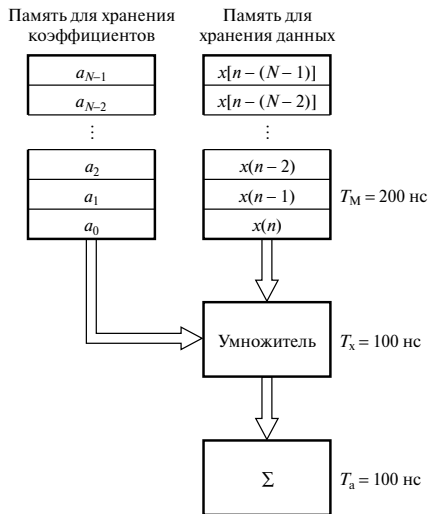


Рис. 12.7. Конфигурация умножителя-накопителя без конвейерной обработки. Произведения подаются в накопитель каждые 400 нс

2. Фигурирующие в вычислениях арифметические операции можно разбить на три различных этапа: считывание из памяти, умножение и накопление. Для увеличения скорости данные шаги разных операций можно совместить. Для двукратного увеличения скорости следует ввести конвейерные регистры между памятью и умножителем и между умножителем и накопителем, как показано на рис. 12.8. Временная диаграмма полученного блока с конвейерной обработкой представлена на рис. 12.9. Как очевидно из диаграммы, операция умножения-накопления завершается каждые 200 нс. Поскольку по условию задачи издержки на организацию конвейера можно не учитывать, ограничивающим фактором является стандартная задержка передачи через самый низкоскоростной элемент, в данном случае — память.

Алгоритмы ЦОС часто имеют итерационную, но сложную структуру, что делает их идеально подходящими для многоуровневой конвейерной обработки. Например, БПФ требует постоянного вычисления по схеме “бабочка”. Хотя каждое такое вычисление требует иных данных и коэффициентов, основная арифметическая операция в схеме “бабочка” остается без изменений. Следовательно, такие арифметические устройства, как процессоры БПФ, можно модернизировать, сыграв на этой особенности вычислений. Конвейерная обработка гарантирует постоянный поток команд на центральный процессор и в общем случае приводит к значительному увеличению пропускной способности системы. В то же время, иногда конвейерная обработка может порождать проблемы. Например, в некоторых процессорах ЦОС конвейерная обработка может привести к выполнению нежелательной команды, особенно при наличии команд ветвления, и разработчику следует помнить, что такая возможность существует.

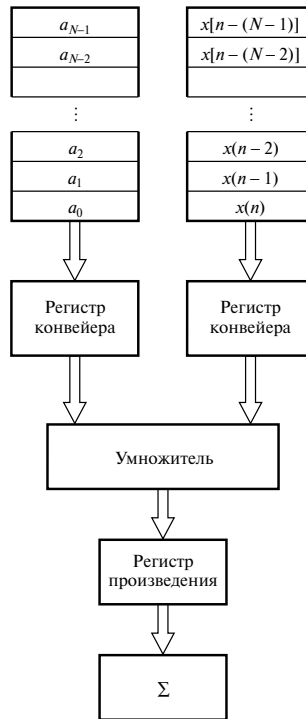


Рис. 12.8. Конфигурация умножителя-накопителя с конвейерной обработкой. Регистры конвейера используются как временное хранилище для коэффициентов и пар выборок данных. Регистр произведения также служит временным хранилищем произведения

12.2.3. Аппаратный умножитель-накопитель

Основными численными операциями в цифровой обработке сигналов являются умножение и сложение. Умножение в программной форме печально знаменито своей трудоемкостью, а если используется арифметика с плавающей запятой, сложение оказывается даже еще более трудоемкой операцией. Чтобы максимально ускорить цифровую обработку сигналов в реальном времени, рекомендуется использовать специализированные аппаратные умножители-накопители с арифметикой с плавающей или фиксированной запятой. Такое аппаратное обеспечение теперь стандартно используется во всех цифровых процессорах сигналов. В процессоре с фиксированной запятой аппаратный умножитель за один такт (обычно 25 нс) принимает два 16-битовых дробных числа, представленных в форме дополнения до двух, и вычисляет их 32-битовое произведение. Среднее время выполнения команды умножения-накопления можно значительно уменьшить, если использовать специальные команды повторения.

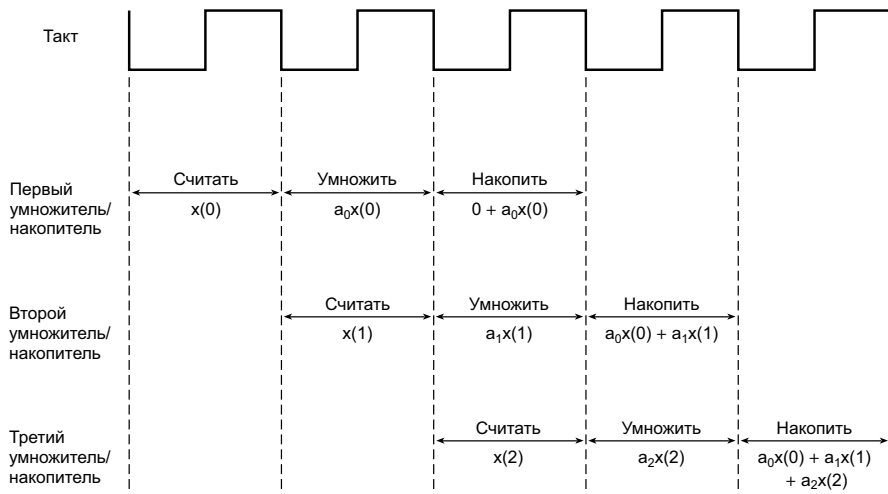


Рис. 12.9. Временная диаграмма блока умножителя-накопителя с конвейерной обработкой. При заполненном конвейере операция в блоке завершается каждый такт (200 нс)

Типичная конфигурация аппаратного умножителя-накопителя ЦОС изображена на рис. 12.10. В такой конфигурации умножитель имеет пару входных регистров, которые содержат входы умножителя, и 32-битовый регистр произведения, который содержит результат умножения. Выход регистра Р (product — произведение) соединяется с накопителем двойной точности, в котором накапливаются произведения.

Принцип весьма сходен с тем, что используется в аппаратных умножителях-накопителях с плавающей запятой, за исключением того, что входы и произведения — это нормированные числа, представленные в форме с плавающей запятой. Умножители-накопители с плавающей запятой позволяют быстро вычислять результаты алгоритмов ЦОС с минимальными ошибками. Как обсуждалось в главах 7 и 8, такие алгоритмы ЦОС, как КИХ- и БИХ-фильтрация подвержены эффектам конечной разрядности (квантование коэффициентов и арифметические ошибки). Операции с плавающей запятой позволяют использовать большой динамический диапазон и снижают арифметические ошибки, хотя во многих приложениях достаточно динамического диапазона, который предлагает представление с фиксированной запятой.

12.2.4. Специальные команды

Процессоры ЦОС предлагают некоторые специальные команды, оптимизированные для цифровой обработки сигналов. Эти команды обладают двойным преимуществом. Во-первых, они позволяют создавать более компактный код, который занимает меньше места в памяти (получается практически такой же компактный код, как и программа, написанная на языке высокого уровня, например, С). Во-вторых, они увеличивают скорость выполнения алгоритмов ЦОС. Специальные команды, внедренные в микросхемы ЦОС, включают: 1) команды, поддерживающие базовые операции ЦОС; 2) команды,

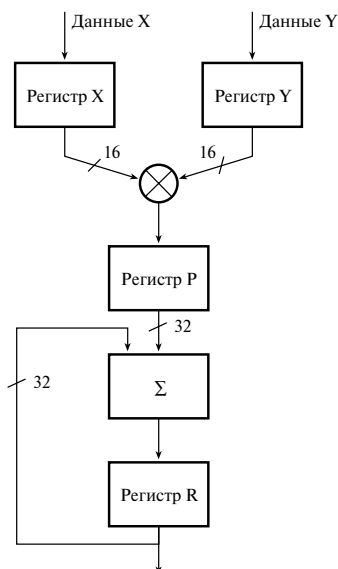


Рис. 12.10. Типичная для ЦОС конфигурация умножителя-накопителя

которые снижают служебные издержки для организации циклов команд; 3) команды, ориентированные на конкретное приложение.

Многие ключевые алгоритмы ЦОС, такие как алгоритмы цифровой фильтрации и поиска корреляции, требуют смещения или задержки данных, чтобы освободить место для новых выборок данных. Процессоры ЦОС предлагают специальные команды, которые позволяют копировать выборку, находящуюся в ячейке памяти со следующим порядковым адресом, так же, как будто выборка извлечена из памяти или обрабатывается в данный момент, причем все это за один такт. Например, во втором поколении семейства TMS320 процессоров ЦОС пара команд LTD и MPY позволяет одновременно загружать данные во временный регистр для умножителя, смещать данные (для реализации единичной задержки, которая обозначается z^{-1}) и накапливать произведения. Специальные команды, которые поддерживают операции умножения-накопления со сдвигом данных, в современных процессорах ЦОС стали стандартными.

Специальные команды также позволяют ускорить часто повторяющиеся операции ЦОС. Например, во втором поколении процессоров семейства TMS320 есть команда, которая позволяет повторить следующую за ней команду заданное число раз. Поскольку команда повтора требует только одной операции извлечения из памяти, фрагмент кода, который обычно требует цикла в несколько тактов, эффективно превращается в одноктактовую команду. Команды повтора особенно полезны в цифровой обработке сигналов для расчета внутренних циклов, например, в КИХ- и адаптивной фильтрации, и записи данных в порядке, определяемом обращенными битами, при вычислении БПФ.

Напомним, что КИХ-фильтры характеризуются следующим уравнением:

$$y(n) = \sum_{k=0}^{N-1} h(k)x(n-k),$$

где N — длина фильтра.

В процессоре TMS320C50, например, уравнение КИХ-фильтрации можно эффективно реализовать с использованием пары команд:

```
RPT    NM1
MACD   HNM1, XNM1
```

Первая команда RPT NM1 загружает длину фильтра минус 1 (т.е. $N - 1$) в счетчик повторения команды и приводит к N -кратному повторению следующей за ней команды умножения-накопления с перемещением данных (MACD). Команда MACD выполняет за один такт несколько команд:

- 1) умножает выборку данных $x(n-k)$, которая находится в памяти для хранения данных, на коэффициент $h(k)$, который находится в памяти для хранения программы;
- 2) добавляет предыдущее произведение к накапливающему сумматору;
- 3) реализует единичную задержку, которая обозначается как z^{-1} , путем смещения выборки данных $x(n-k)$ в ячейку со следующим по возрастанию адресом для обновления линии задержки с отводами.

В семействе процессоров ЦОС Motorola DSP56000, как и в семействе TMS320, для эффективной реализации КИХ-фильтра можно использовать команду MAC вместе с командой повторения (REP):

```
RPT    #N-1
MAC     X0, Y0, A      X: (R0)+, X0      Y: (R4)+, Y0
```

Здесь команда повторения используется вместе с командой MAC для выполнения длительных умножений и суммирования произведений. Снова отметим возможность осуществить умножение с помощью одной команды, что позволяет иметь несколько трактов обработки данных. Содержимое регистров X0 и Y0 перемножается, а результат добавляется к накопителю. В это же время следующая выборка данных и соответствующий коэффициент извлекаются для умножения из ячеек X и Y.

В наиболее современных процессорах ЦОС была развита концепция повторения команды и введена команда, которая позволяет повторять заданное число раз не только одну команду, но и блок кода. В семействе TMS320 (например, TMS320C50, TMS320C54 и TMS320C30) используется такой формат вычисления повторяющего блока команд (цикл с нулевыми служебными издержками):

```
RPTB loop
:
:
loop (последняя команда)
```


Команды повторения, предлагаемые некоторыми процессорами ЦОС, имеют функции, характерные для языков высокого уровня. В семействах Motorola DSP56000 и DSP56300 есть циклы DO с нулевыми служебными издержками, которые также могут быть вложенными. Приведенный ниже пример иллюстрирует вложенный цикл DO, в котором внешний цикл выполняется N раз, а внутренний — NM раз.

```
DO #N, LOOP1
:
DO #M, LOOP2
:
:
LOOP2 (здесь располагается последняя команда)
:
LOOP1 (здесь располагается последняя команда внешнего цикла)
```

Вложенные циклы полезны при эффективной реализации таких функций ЦОС, как алгоритмы БПФ и двумерная обработка сигналов.

Процессоры ЦОС, производимые Analog Devices (например, ADSP-2115 и SHARC), также позволяют организовать вложенные циклы. Процессор ADSP-2115 поддерживает до четырех уровней вложения циклов. При вложении используется следующий формат:

```
CNTR=N
DO LOOP UNTIL CE
:
:
LOOP: (последняя команда цикла)
```

Цикл повторяется до обнуления счетчика. Цикл может содержать большой блок команд. Формат вложения циклов, по сути, идентичен формату, который используется в процессорах семейства DSP56000.

Адаптивная фильтрация — это еще одна ключевая характерная функция современной обработки сигналов, и для ее поддержки также были реализованы специальные команды. В адаптивной фильтрации важным шагом является обновление коэффициентов (см. главу 10) и вычисление нового набора коэффициентов на основе старого. Например, в адаптивных фильтрах, основанных на схеме наименьших квадратов, задача обновления коэффициентов характеризуется следующим уравнением:

$$h_{k+1}(i) = h_k(i) + 2\mu e_k x(k-i), \quad (12.3, a)$$

где

$$e_k = y_k - \hat{y}_k, \quad (12.3, б)$$

а $\hat{y}_k$, выход адаптивного фильтра, выражается следующим образом:

$$\hat{y}_k = \sum_{i=0}^{N-1} h_k(i)x(k-i). \quad (12.3, в)$$

Обычно часть задачи обновления коэффициентов, которая относится к КИХ-фильтрации (уравнение 12.3, *в*), реализуется с использованием специальных команд умножения-накопления, как описывалось ранее. Задачу обновления коэффициентов (уравнение (12.3, *а*)) можно выполнить с помощью команды повторения блока с нулевыми служебными издержками на организацию цикла. В процессоре TMS320C54 рассчитывать выход адаптивного фильтра и обновлять коэффициенты фильтра можно с помощью команды адаптивной фильтрации на базе схемы наименьших квадратов LMS, команды умножения ST//MPY и команды повтора блока RPTBD. Указанные команды позволяют снизить время выполнения и размер кода адаптивных фильтров.

В быстром преобразовании Фурье (БПФ), другой ключевой операции ЦОС, всегда требуется скремблировать последовательность входных данных до БПФ и восстанавливать обычный порядок после БПФ, чтобы гарантировать, что точки данных идут в правильной последовательности. Все высококачественные универсальные процессоры ЦОС предлагают специальные команды для адресации с инвертированием битов, позволяющие выполнять требуемое скремблирование/во восстановление одновременно с извлечением выборок данных из памяти или перемещением выборок.

Например, адресацию с инвертированием битов в процессорах TMS320 можно использовать для обращения битов при занесении в память N -точечной комплексной последовательности данных:

```
RPT N2  
BLDD #XN, *BR0+
```

В данном случае используются комплексные входные данные, так что каждая выборка состоит из действительной и мнимой частей. Следовательно, каждая выборка требует двух ячеек памяти.

Современные процессоры ЦОС также предлагают ориентированные на приложения команды для таких областей, как кодирование речи (например, при поиске в кодовой книге), цифровое аудио (например, в системах объемного звучания) и телекоммуникации (например, в декодировании Витерби).

12.2.5. Средства дублирования

В цифровой обработке сигналов дублирование подразумевает использование нескольких стандартных блоков, например, более одного АЛУ, умножителя или ячейки памяти. Часто блоки располагаются так, чтобы их можно было использовать параллельно. В ЦОС нормой является наличие одного центрального процессора с дублированными одним или несколькими арифметическими устройствами.

В то же время, в области ЦОС уже используется концепция полнофункциональной параллельной обработки, когда, например, над одной задачей работают несколько независимых процессоров или несколько процессоров под общим управлением. На настоящий момент на рынке представлено несколько процессоров ЦОС с параллельной обработкой, например, TMS320C40 и ADSP-21060 SHARC.

12.2.6. Встроенная память/кэш

В большинстве случаев чипы ЦОС работают настолько быстро, что медленная недорогая память не может поддерживать такой темп. Обычно при этом процессор замедляется — вводятся состояния ожидания. В некоторых процессорах состояния ожидания имеют программное управление, хотя иногда для замедления процессора требуются внешние аппаратные блоки. Разумеется, наличие состояний ожидания означает, что процессор не может работать с полной скоростью.

Чтобы решить описанную проблему, во многие чипы ЦОС встраивают скоростные внутренние ОЗУ и/или ПЗУ данных. В таких процессорах низкоскоростная внешняя память может использоваться для хранения программного кода. При инициализации код может передаваться в быструю внутреннюю память для выполнения операций на полной скорости. Для реализации в реальном времени и создания окончательных прототипов полезны скоростные встроенные стираемые программируемые ПЗУ (СППЗУ). Некоторые чипы имеют также встроенный программный кэш, который может использоваться часто повторяемыми блоками программы. Если код выполняется в кэше, можно не вызывать лишний раз команды из памяти, ускоряя выполнение программы.

В настоящее время встроенная память используется повсеместно.

12.2.7. Расширенный параллелизм — SIMD, VLIW и статическая суперскалярная обработка

В настоящее время в архитектуре процессоров ЦОС наблюдается тенденция к повышению производительности за счет увеличения числа команд, выполняемых в каждом такте, и числа операций, выполняемых при вызове одной команды [1, 2, 4, 8, 10].

В новейших архитектурах процессоров ЦОС для повышения вычислительной эффективности интенсивно используются методы параллельной обработки. Наиболее часто применяются (иногда совместно) следующие три техники: архитектура SIMD (single instruction, multiple data — с одним потоком команд и многими потоками данных, также ОКМД — “одна команда, много данных”), архитектура VLIW (very-large-instruction-word — архитектура с командными словами сверхбольшой длины) и суперскалярная обработка [4, 5, 10].

Обработка с архитектурой *SIMD* используется для увеличения числа операций, выполняемых при вызове одной команды. Обычно в процессорах ЦОС с архитектурой *SIMD* процессор имеет несколько трактов передачи данных и несколько операционных блоков. Следовательно, одна команда может передаваться нескольким операционным блокам для обработки блоков данных одновременно, увеличивая число операций, выполняемых за один такт (см., например, рис. 12.11). В контексте рис. 12.11 процессор ЦОС имеет два операционных блока, каждый со своим АЛУ, умножителем-накопителем и схемой сдвига. Процессор может выполнять две различных арифметических операции (например, сложение и умножение-накопление) одновременно в течение одной команды. В число процессоров ЦОС с архитектурой *SIMD* входят DSP16000 (Lu-

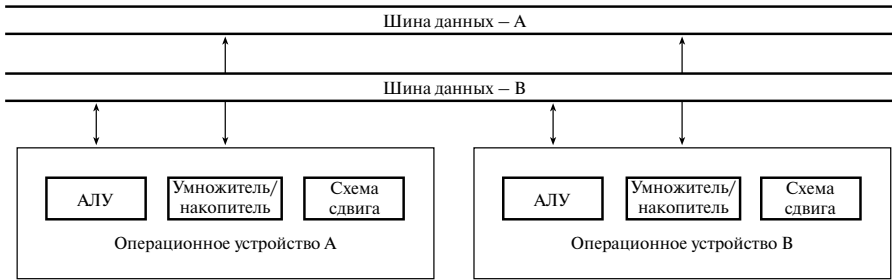


Рис. 12.11. Двойные арифметические устройства и двойные тракты передачи данных для обработки по схеме SIMD

cent), TMS320C62x (Texas Instruments) и TigerSHARC (Analog Devices) ADSP-TS001 (Analog Devices)².

Привлекательной особенностью архитектур SIMD, особенно архитектур, которые поддерживают данные нескольких размеров, является возможность эффективного увеличения числа доступных операционных блоков, а следовательно, числа операций, выполняемых за такт, путем разделения блоков. Например, в процессоре TigerSHARC два доступных умножения можно разделить для одновременного выполнения четырех операций умножения-накопления 16×16 бит вместо одной операции умножения-накопления 32×32 бит (рис. 12.12).

Очевидно, что в приложениях, где данные обрабатываются параллельно, архитектура SIMD может значительно повысить эффективность работы процессора. Впрочем, в приложениях с последовательными данными возможность увеличения вычислительной эффективности за счет обработки по схеме SIMD незначительна. По этой причине в следующем поколении процессоров ЦОС, нацеленных на многоканальные приложения (такие как системы мобильной связи третьего поколения), используются возможности обработки с одним потоком команд и многими потоками данных (SIMD) [4, 8, 10].

Обработка *с командными словами сверхбольшой длины* (Very Long Instruction Word — VLIW) позволяет существенно увеличить число команд, обрабатываемых за такт [10]. По сути, *командное слово сверхбольшой длины* — это конкатенация нескольких *коротких* команд, и для выполнения нескольких коротких команд за один такт требуется несколько операционных блоков, функционирующих параллельно.

Принципы архитектуры VLIW и схема потока данных улучшенных процессоров ЦОС с фиксированной запятой TMS320C62x иллюстрируются на рис. 12.13. Центральный процессор содержит два тракта передачи данных и восемь независимых операционных блоков, организованных в два набора — (L1, S1, M1 и D1) и (L2, S2, M2 и D2). В данном случае каждая *короткая* команда имеет длину 32 бит. Восемь таких команд вместе образуют пакет *командного слова сверхбольшой длины*, который может обрабатываться параллельно несколькими устройствами.

Обработка по схеме VLIW начинается с извлечения центральным процессором из встроенной программной памяти пакета команд (восемь 32-битовых команд). Восемь команд в *вызванном* пакете формируют *исполняемый* пакет, если они могут выпол-

²Также MMX — ядро процессора Pentium и его клонов. — Прим. ред.

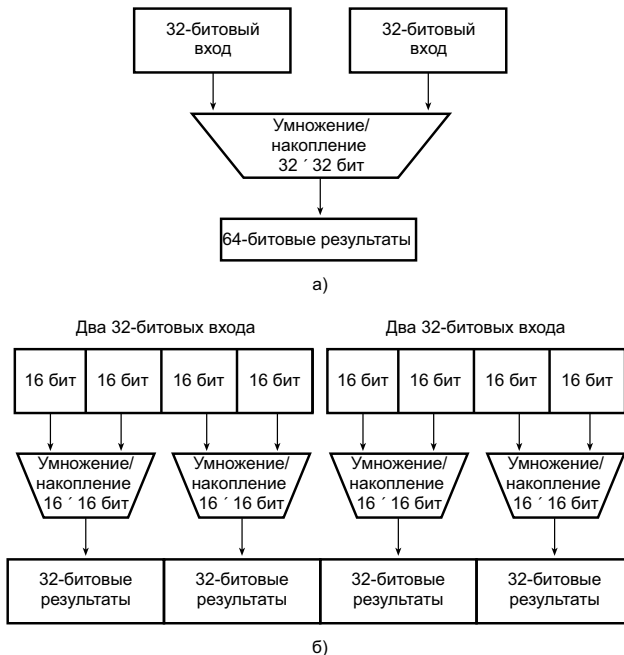


Рис. 12.12. Иллюстрация использования обработки по схеме SIMD и поддержки данных нескольких размеров для увеличения в процессоре TigerSHARC числа умножителей-накопителей с одного до четырех

няться параллельно, а затем (по обстановке) передается восьми операционным блокам. Следующий 256-битовый пакет команд вызывается из программной памяти во время декодирования и выполнения исполняемого пакета. Если восемь команд в вызванном пакете нельзя выполнять параллельно (например, если все восемь команд — команды умножения/накопления, в течение одного такта можно выполнять только две из них, так как существует всего два умножителя), тогда формируется несколько исполняемых пакетов, которые передаются операционным блокам по одному за раз. Вызываемый пакет всегда имеет длину 256 бит (восемь команд), но размер исполняемого пакета может составлять от одной до восьми команд.

Архитектура VLIW предназначена для поддержки параллелизма на уровне команд. Если объединить данную архитектуру с высокой тактовой частотой (обычно 200 МГц), обычно получаются весьма эффективные процессоры ЦОС. Следует, правда, помнить, что в семействе TMS320C62х, например, параллельная обработка команд планируется во время компиляции. Впрочем, если команды нельзя выполнять параллельно, вычислительная эффективность подобных процессоров не используется.

Другой способ увеличить скорость выполнения команд процессора ЦОС (число команд, обработанных за такт) состоит в *суперскалярной* обработке, в которой используется параллелизм на уровне команд. Традиционно термином *суперскалярный* называют архитектуру компьютера, позволяющую выполнять несколько команд за один такт [5].

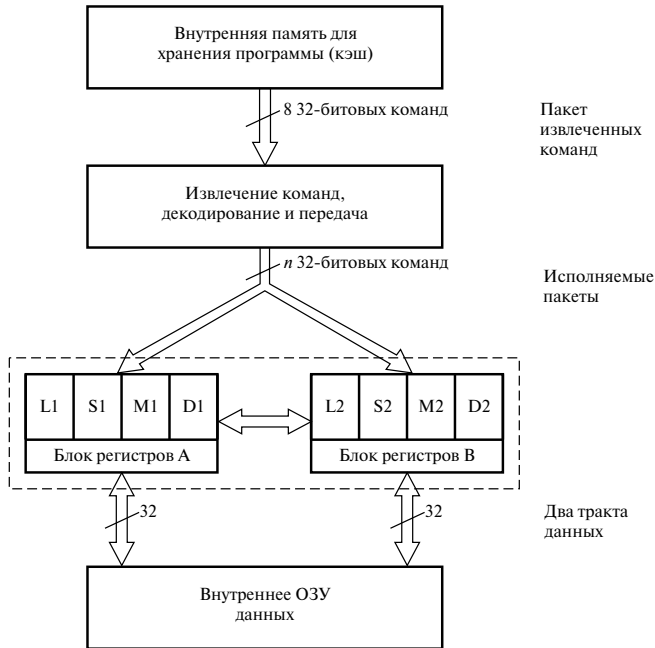


Рис. 12.13. Принципы архитектуры с командными словами сверхбольшой длины (Very Long Instruction Word — VLIW) и схема потока данных в процессорах TMS320C62x. *Примечание:* L1, L2 — логические элементы; S1, S2 — схемы сдвига/логические элементы; M1, M2 — умножители; D1, D2 — адресные элементы

Подобные архитектуры широко используются в таких универсальных процессорах, как PowerPC и Pentium. *Суперскалярные* процессоры ЦОС содержат несколько операционных блоков, и несколько команд могут передаваться этим блокам для параллельной обработки. Кроме того, для дальнейшего увеличения производительности интенсивно эксплуатируется метод параллельной обработки.

Наиболее известным суперскалярным процессором ЦОС является TigerSHARC (Analog Devices) (см. [4] и рис. 12.14). Процессор TigerSHARC описан в спецификации как статический суперскалярный процессор ЦОС, поскольку параллелизм команд определяется до их выполнения. Фактически процессор TigerSHARC объединяет концепции SIMD, VLIW и суперскалярной обработки. Данный улучшенный процессор ЦОС имеет несколько трактов обработки данных и два набора независимых операционных блоков, в каждом из которых имеется умножитель, АЛУ, 64-битовая схема сдвига и блок регистров (см. рис. 12.14). TigerSHARC — это процессор с плавающей запятой, но он также поддерживает арифметику с различными типами данных (8-, 16- и 32-битовые числа).

Длина команды в процессоре TigerSHARC не фиксирована. В каждом такте из внутренней памяти извлекается до четырех 32-битовых команд, которые передаются двум параллельным наборам операционных блоков. Команды могут передаваться и параллельно двум блокам (архитектура SIMD — Single Instruction, Multiple Data — с одним

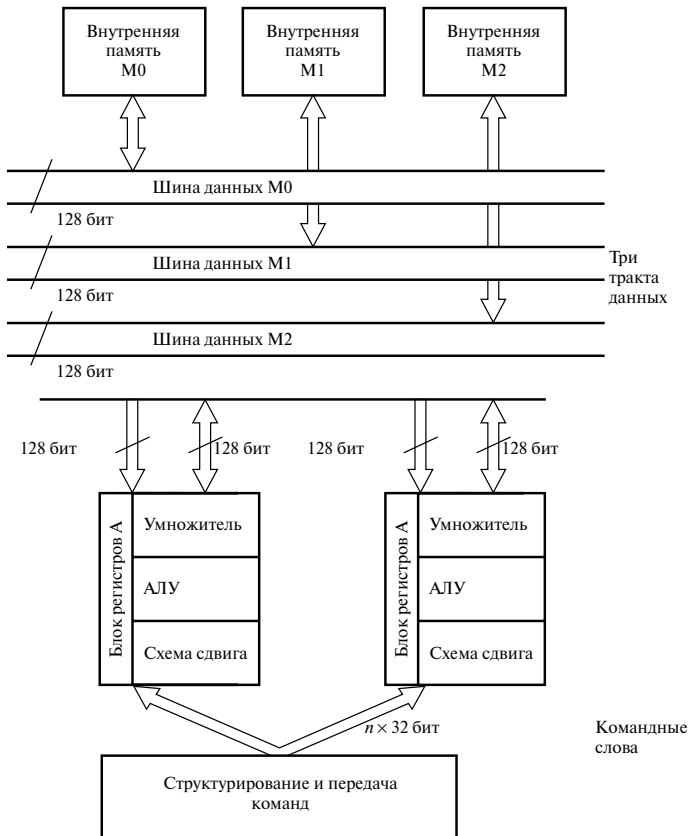


Рис. 12.14. Принципы суперскалярной архитектуры и схема потока данных в процессоре ЦОС TigerSHARC

потоком команд и многими потоками данных) или каждому операционному блоку независимо. Каждый операционный блок (АЛУ, умножитель или схема сдвига) получает свой вход от блока регистров и возвращает результаты ему же. Блоки регистров соединены с тремя трактами передачи данных, так что они могут одновременно в одном такте считывать два входа и записывать выход в память. Данная архитектура загрузки-хранения подходит для стандартных операций ЦОС, у которых часто бывает два входных значения и одно выходное.

Как обсуждалось ранее, поскольку процессор может работать с несколькими размерами данных (8-, 16-, 32- и 64-битовые числа), возможна организация параллельной обработки на следующих уровнях. Следовательно, в каждом такте процессор TigerSHARC может выполнять до восьми операций сложения-вычитания и восемь операций умножения-накопления с 16-битовыми входами вместо двух операций умножения-накопления с 32-битовыми входами. Возможность обработки смешанных типов данных и разбиения больших командных слов на отдельные команды для операционных блоков позволяет процессору интенсивно использовать параллелизм на уровне команд.

Стоит также отметить, что в процессорах ЦОС, использующих такие улучшенные архитектуры, как VLIW и суперскалярная обработка, для эффективного использования параллельных операционных блоков необходима некоторая форма статического планирования команд перед выполнением программы. Кроме того, следует учитывать возможную зависимость по данным (например, результаты требуются до того, как они готовы) и зависимость по управлению (например, при использовании команд ветвления), что может представлять проблемы при параллельной обработке.

12.3. Универсальные процессоры ЦОС

Универсальные процессоры ЦОС — это, по сути, высокоскоростные микропроцессоры с гарвардской архитектурой и наборами команд, оптимизированных под операции ЦОС. В данных процессорах везде, где только возможно, выполнение трудоемких операций (таких как сдвиг/масштабирование, умножение и т.д.) облегчается за счет интенсивного использования гарвардской архитектуры, конвейерной обработки и специализированного аппаратного обеспечения.

Универсальные процессоры ЦОС значительно развились за последнее десятилетие в результате непрекращающихся поисков лучших способов выполнения операций ЦОС с точки зрения вычислительной эффективности, легкости использования, стоимости, расхода энергии, размера и требований приложения [1, 7, 8]. Ненасытный спрос на улучшенную вычислительную эффективность привел к существенному снижению времени выполнения команд, увеличению тактовой частоты и, что более важно, усложнению аппаратных и программных архитектур. В настоящее время обычным является наличие специализированных встроенных арифметических аппаратных устройств (например, для поддержки быстрых операций умножения-накопления), большой встроенной памяти со множественным доступом и специальных команд для эффективного выполнения операций ЦОС во внутреннем ядре. Авторы также отмечают тенденцию к увеличению размера слова (например, для поддержания качества сигнала) и более интенсивному использованию параллелизма (для увеличения числа команд, выполняемых в одном такте, и числа операций, выполняемых при вызове одной команды). Таким образом, по мнению авторов, в новых универсальных процессорах ЦОС увеличивается число трактов обработки данных и используются арифметики, поддерживающие параллельные операции. Вводятся процессоры ЦОС, основанные на архитектурах SIMD (Single Instruction, Multiple Data — архитектура с одним потоком команд и многими потоками данных), VLIW (Very Long Instruction Word — архитектура с командными словами сверхбольшой длины) и суперскалярной архитектуре для поддержки эффективной параллельной обработки. В некоторых процессорах ЦОС, чтобы дополнительно повысить производительность, используют специализированные встроенные сопроцессоры для ускорения таких специфических алгоритмов ЦОС, как КИХ-фильтрация и декодирование Витерби. Вообще, на развитие процессоров ЦОС значительное влияние оказало взрывоподобное развитие технологий в сфере связи и цифровых аудиосистемах, а также рост числа приложений, в которых используются встроенные процессоры ЦОС.

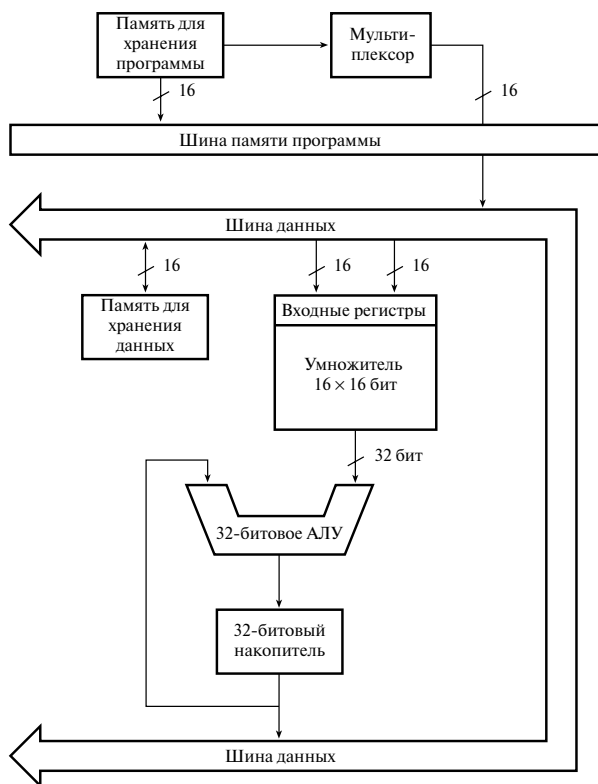


Рис. 12.15. Упрощенная архитектура процессора ЦОС с фиксированной запятой первого поколения (Texas Instruments TMS320C10)

В следующих двух разделах кратко описаны архитектурные особенности нескольких поколений процессоров ЦОС с фиксированной и плавающей запятой.

12.3.1. Процессоры ЦОС с фиксированной запятой

Доступные на сегодня процессоры ЦОС с фиксированной запятой отличаются друг от друга деталями архитектуры и встроенными ресурсами. Резюме по ключевым особенностям четырех поколений процессоров ЦОС с фиксированной запятой, созданных основными производителями полупроводниковых технологий, представлено в табл. 12.1. Условная классификация процессоров ЦОС на четыре поколения основана на исторических причинах, архитектурных особенностях и вычислительной эффективности.

Таблица 12.1.1. Особенности универсальных процессоров ЦОС с фиксированной запятой производства Texas Instruments, Motorola и Analog Devices

Поло- жение	ЦОС с фиксированной запятой	Ширина тракта данных (биты)	Число трактов данных (биты)	Длина слова накопителя (биты)	Длина слова (биты)	Длина команды (биты)	Размер встроенного ОЗУ(слова)	Размер кэша команд (число команд)	Число умножителей	Коэффициент производительности*
1	TMS320C10 (Texas Instruments)	16	1	16	32	16	144		1	
2	TMS320C50 (Texas Instruments)	16	2	16	32	16	10 K		1	10 @ 50 МГц
	DSP56002 (Motorola)	24	2	24	56	24	1 K		1	13 @ 50 МГц
	DSP-2100 (Analog Devices)	16	2	16	40	24	32 K	16	1	13 @ 52 МГц
	1600 (Lucent Technologies)	16	2	16	36	16		15	1	22 @ 120 МГц
3	TMS320C54 (Texas Instruments)	16	3	16	40	16	32 K		1	25 @ 100 МГц
	DSP56300 (Motorola)	24	3	24	56	24		3 K	1	25 @ 100 МГц
	16000 (Lucent Technologies)	32	2	32	40	32	127 K	31	2	36 @ 100 МГц
4	TMS320C6200 (Texas Instruments)		2		40	256	17 K	64 K	2	86 @ 133,6 МГц

* Коэффициент производительности рассчитывается на основе скорости расчета контрольных ядер/вычисления контрольных алгоритмов ЦОС (см. [1, 7]).

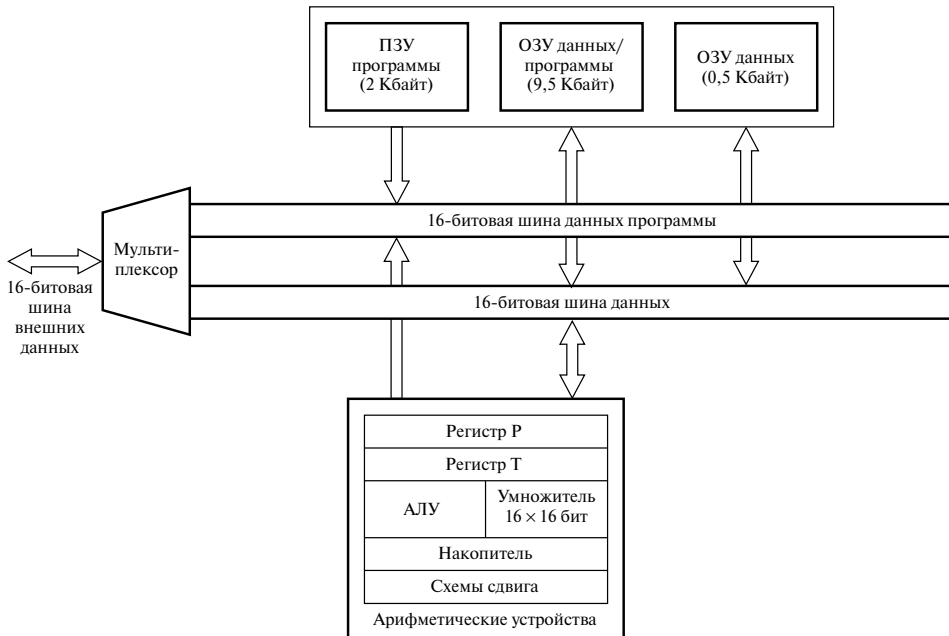


Рис. 12.16. Упрощенная архитектура процессора ЦОС с фиксированной запятой второго поколения (Texas Instruments TMS320C50)

Стандартная архитектура семейства процессоров ЦОС с фиксированной запятой *первого поколения* TMS320C1x, впервые использованная в 1982 году Texas Instruments, изображена на рис. 12.15. Ключевые особенности TMS320C1x — выделенные арифметические устройства, которые включают умножитель и накопитель. Процессоры семейства имеют модифицированную гарвардскую архитектуру с двумя отдельными областями памяти для данных и программы. Кроме того, процессоры имеют встроенную память и несколько специальных команд для выполнения основных алгоритмов ЦОС.

Процессоры ЦОС с фиксированной запятой *второго поколения* имеют значительно улучшенные архитектуры по сравнению с процессорами первого поколения. В большинстве случаев в число этих особенностей входили большая встроенная память и много специальных команд для поддержки эффективного выполнения алгоритмов ЦОС. В результате вычислительная производительность процессоров ЦОС второго поколения в 4–6 раз превышает производительность процессоров первого поколения.

Типичными представителями процессоров ЦОС второго поколения являются семейства TMS320C5x (Texas Instruments), DSP5600x (Motorola), ADSP21xx (Analog Devices) и DSP16xx (Lucent Technologies). Процессоры ЦОС первого и второго поколений производства Texas Instruments имеют много общего архитектурно, но процессоры второго поколения имеют больше особенностей и более высокую скорость (см. табл. 12.1). Внутренняя архитектура процессоров семейства TMS320C5x показана на рис. 12.16 (в упрощенном виде, чтобы акцентировать внимание на двойной области памяти, характерной для гарвардской архитектуры). В число специальных команд для выполнения

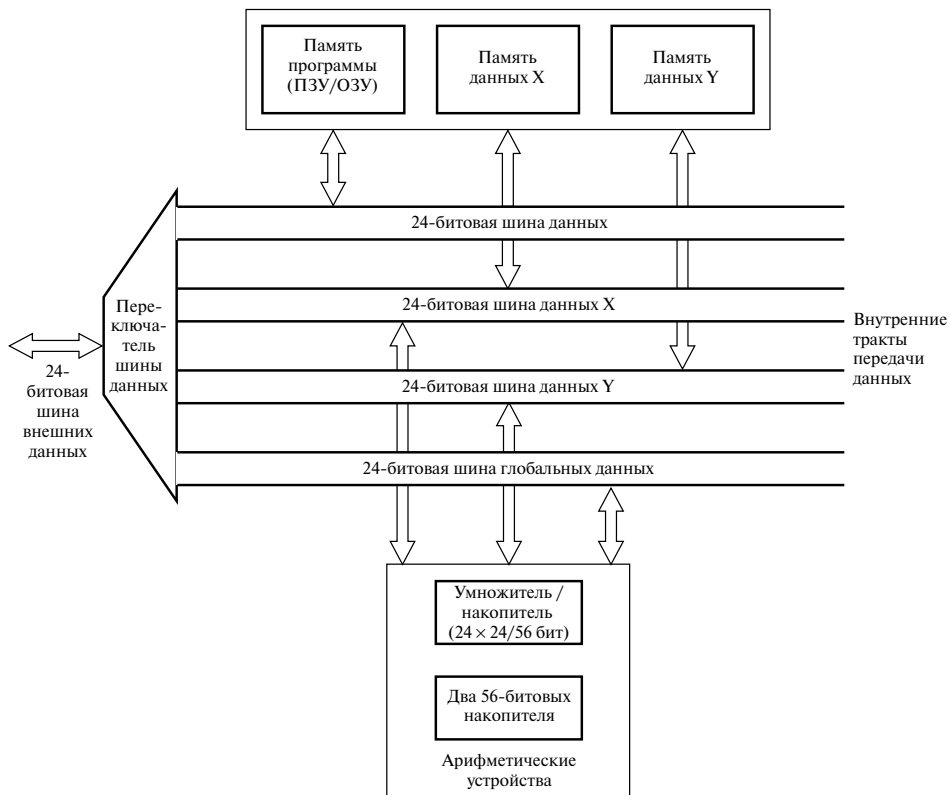


Рис. 12.17. Упрощенная архитектура процессора ЦОС с фиксированной запятой второго поколения (Motorola DSP56002)

операций ЦОС входит команда умножения и накопления со смещением данных, которую, например, можно объединить с командой повтора и реализовать КИХ-фильтр со значительной экономией времени. Возможность адресации с инвертированием битов полезна при БПФ. Кроме того, в отличие от процессоров первого поколения процессоры, семейства C5x имеют большую встроенную память.

Процессор DSP5600x (Motorola) является высокоточным цифровым процессором сигналов с фиксированной запятой. Его архитектура изображена на рис. 12.17. Он имеет две независимые области памяти для хранения данных (области X и Y) и одну область для хранения программ. Наличие двух отдельных областей для хранения данных позволяет естественно разбить данные для операций ЦОС и облегчает выполнение алгоритмов ЦОС. Например, в графических приложениях данные могут записываться как данные, относящиеся к координате X, и данные, относящиеся к координате Y, при КИХ-фильтрации — как коэффициенты и данные, в БПФ — как действительная и мнимая части числа. В ходе выполнения программы за один такт можно одновременно извлечь пару выборок из внутренней памяти или занести пару выборок во внутреннюю память. Внешне две области данных уплотняются в единую шину данных, что несколько сни-

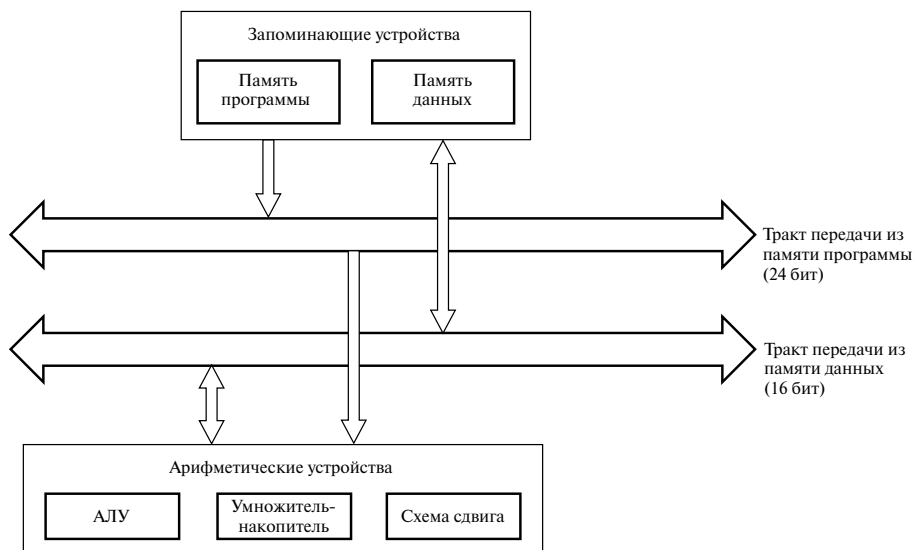


Рис. 12.18. Упрощенная архитектура процессора ЦОС с фиксированной запятой второго поколения (Analog Devices ADSP2100)

жает преимущества двойной внутренней памяти. Кроме того, процессоры семейства имеют следующие арифметические устройства: два 56-битовых накопителя и одно-тактный аппаратный умножитель-накопитель с фиксированной запятой. Умножитель-накопитель принимает 24-битовые входы и выдает 56-битовое произведение. Размер 24 бит обеспечивает достаточную точность для представления большинства переменных ЦОС, тогда как 56-битовый накопитель (включая 8 защитных бит) предотвращает арифметическое переполнение. Указанных длин слов достаточно для большинства приложений, включая цифровое аудио с его весьма строгими требованиями. Процессоры TMS5600x предлагают также специальные команды, позволяющие организовывать циклы с нулевыми служебными издержками, кроме того, существует возможность адресации с инвертированием битов, чтобы скремблировать входные данные перед быстрым преобразованием Фурье или восстанавливать правильный порядок данных после БПФ.

Другим семейством процессоров ЦОС с фиксированной запятой второго поколения является ADSP21xx (Analog Devices), имеющий две отдельные области памяти — в одной хранятся только данные, а в другой — данные и программный код. Упрощенная блок-схема внутренней архитектуры процессоров ADSP21xx изображена на рис. 12.18. Основными компонентами являются АЛУ, умножитель-накопитель и схемы сдвига. Умножитель-накопитель за один такт принимает вход 16×16 бит и выдает 32-битовое произведение. Накопитель процессора ADSP21xx имеет 8 защитных бит, которые могут использоваться для расчетов с повышенной точностью. При создании процессора ADSP21xx разработчики отошли от стандартной гарвардской архитектуры, поскольку процессор позволяет хранить в программной памяти данные и команды программы. Чтобы указать, когда из программной памяти извлекаются данные, а не

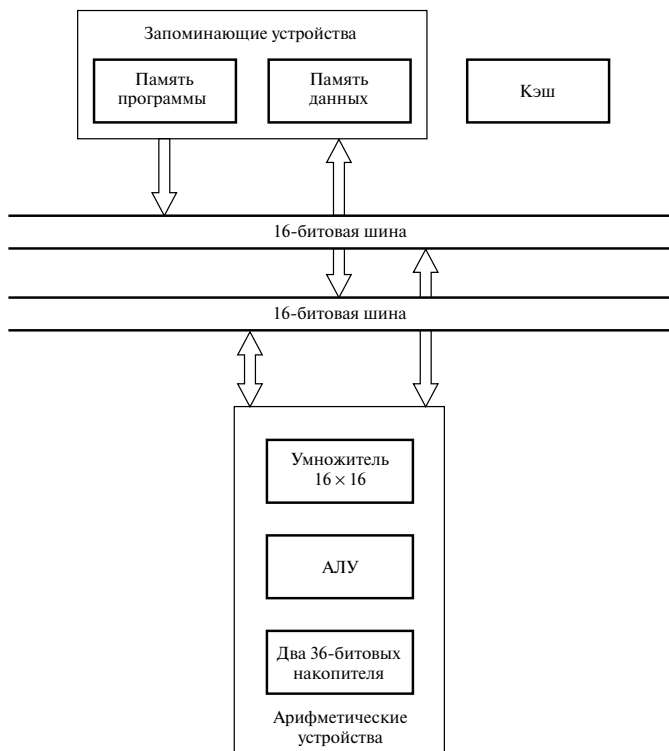


Рис. 12.19. Упрощенная архитектура процессора ЦОС с фиксированной запятой DSP16xx (Lucent Technologies)

команды программы, используется сигнальная шина. Хранение данных в программной памяти препятствует равномерному потоку данных через центральный процессор, поскольку извлечение данных и команд не может происходить одновременно. Чтобы избежать в этой связи значительного снижения производительности, в семействе ADSP21xx используется встроенный кэш программной памяти, который содержит 16 последних выполненных команд. Это устраняет потребность в извлечении из программной памяти повторяющихся команд (данная особенность весьма существенна при выполнении циклов программы). Процессоры ADSP21xx предлагают также специальные команды для организации циклов с нулевыми служебными издержками и поддерживают возможность адресации с инвертированием разрядов битов (требуется при БПФ). Процессоры семейства имеют большую встроенную память (для более интенсивной передачи данных предоставляется до 64 Кбайт внутреннего ОЗУ). Стоит также отметить, что данные процессоры поддерживают прямой доступ к памяти, и внешние устройства могут обмениваться данными и командами с ОЗУ процессора ЦОС без участия самого процессора.

Семейство процессоров ЦОС с фиксированной запятой DSP16xx (Lucent Technologies) (см. рис. 12.19) нацелено на рынок телекоммуникаций и модемов. С точки зрения вычислительной производительности это один из самых мощных процессоров второго

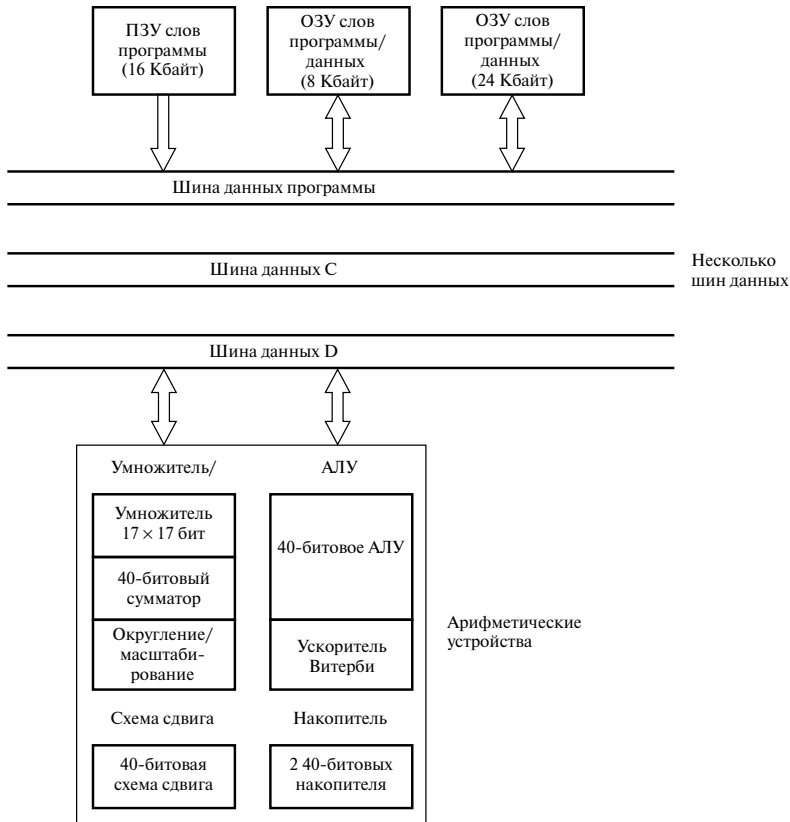


Рис. 12.20. Упрощенная архитектура процессора ЦОС с фиксированной запятой третьего поколения (Texas Instruments TMS320C54x)

поколения. Процессор имеет гарвардскую архитектуру и подобно другим процессорам второго поколения имеет два тракта передачи данных, тракты X и Y . В набор его арифметических устройств входят специализированный умножитель 16×16 бит, 36-битовое АЛУ/схема сдвига (включает четыре защитных бит) и двойные накопители. Предлагаются специальные команды, например для организации циклов команд и выполнения блоков команд с нулевыми служебными издержками.

Процессоры ЦОС с фиксированной запятой *третьего поколения* являются, по сути, улучшением процессоров предыдущего поколения. В общем случае повышение производительности достигается за счет более интенсивного и/или эффективного использования доступных встроенных ресурсов. По сравнению с процессорами ЦОС второго поколения, процессоры третьего поколения отличаются большим числом трактов передачи данных (обычно три по сравнению со старыми двумя), более широкими трактами передачи данных, большими встроенной памятью и кэшем команд и в некоторых случаях двойными умножителями-накопителями. В результате производительность процессоров ЦОС третьего поколения обычно в 2–3 раза превышает производительность процес-

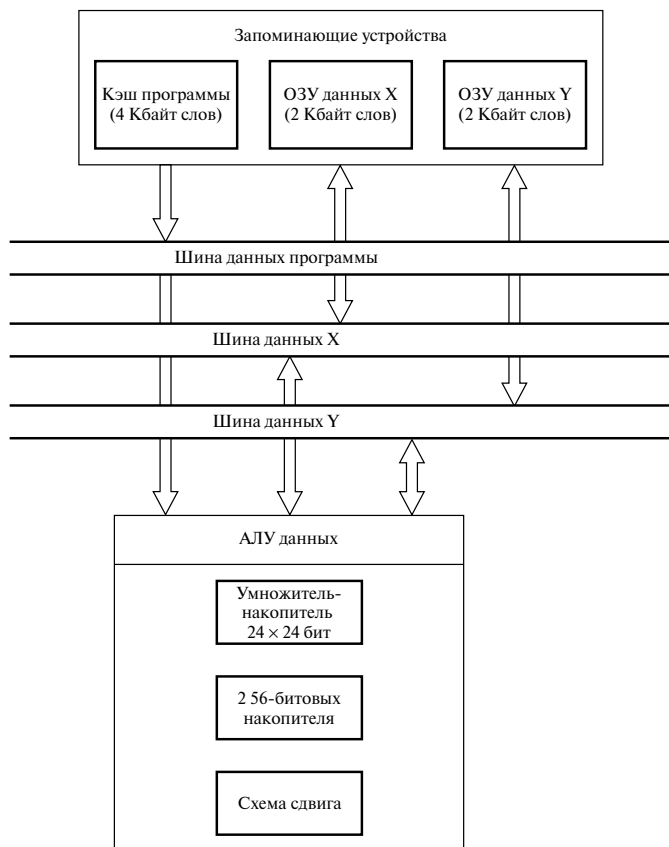


Рис. 12.21. Упрощенная архитектура процессора ЦОС с фиксированной запятой третьего поколения (Motorola DSP56300)

соров второго поколения того же семейства (согласно [1, 7]). Упрощенные архитектуры процессоров ЦОС третьего поколения TMS320C54x, DSP563x и DSP16000 изображены на рис. 12.20, 12.21 и 12.22 соответственно. Большинство процессоров ЦОС с фиксированной запятой третьего поколения нацелены на применение в цифровой связи и цифровом аудио (вообще, развитие этих приложений оказало огромное влияние на развитие процессоров ЦОС). Таким образом, авторы обнаружили в некоторых процессорах особенности, облегчающие использование процессоров в указанных сферах. Процессор TMS320C54x, например, включает специальные команды для адаптивной фильтрации (которая часто используется для эхоподавления и адаптивного выравнивания в сфере телекоммуникаций) и для поддержки декодирования Витерби. В процессорах третьего поколения производители полупроводниковых технологий также серьезно отнеслись к вопросу потребляемой мощности (вследствие его важности в таких портативных устройствах, как мобильные телефоны). Большинство процессоров ЦОС третьего поколения имеют низкий расход энергии и средство управления режимом электропитания.

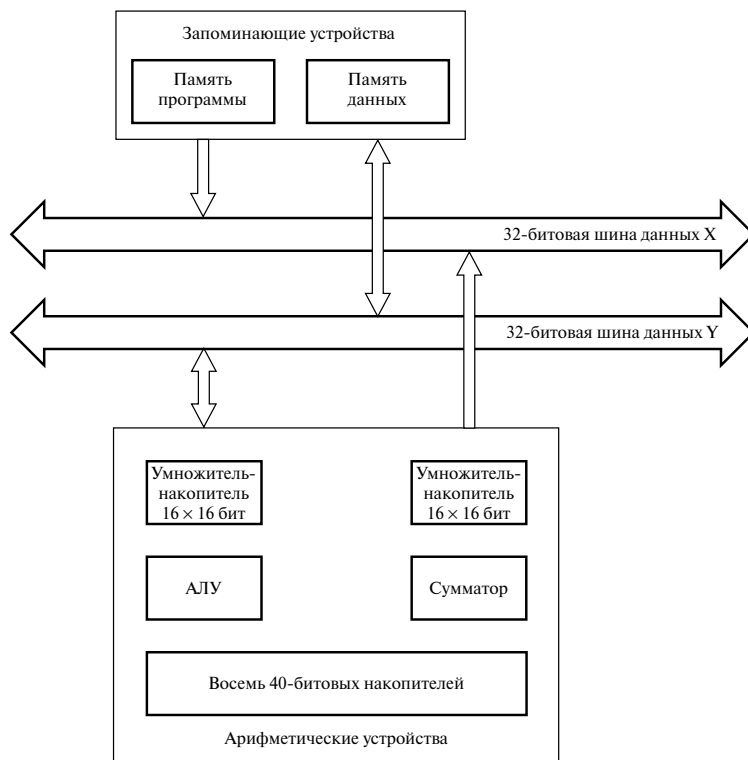


Рис. 12.22. Упрощенная архитектура процессора ЦОС с фиксированной запятой третьего поколения (Lucent Technologies DSP16000)

Процессоры ЦОС с фиксированной запятой *четвертого поколения* с их новыми архитектурами нацелены в первую очередь на большие и/или развивающиеся многоканальные приложения, такие как построение медицинских изображений, цифровые абонентские линии, модемы удаленного доступа, беспроводные базовые станции и мобильные системы третьего поколения. Новой архитектурой с фиксированной запятой, которая привлекла значительную долю внимания в обществе ЦОС, стала *архитектура с командными словами сверхбольшой длины (VLIW)* (подробности представлены в разделе 12.2). В архитектуре новых процессоров интенсивно используется параллелизм, при этом сохранены некоторые преимущества предыдущих процессоров ЦОС. По сравнению с предыдущими поколениями, четвертое поколение процессоров с фиксированной запятой отличается более длинными командными словами, более широкими трактами передачи данных, большим числом регистров, большим кэшем команд и наличием нескольких арифметических устройств, что позволяет выполнять значительно больше команд и операций за такт.

Семейство процессоров ЦОС с фиксированной запятой TMS320C62x (Texas Instruments) основано на архитектуре VLIW (рис. 12.23). Процессор ядра имеет два независимых арифметических тракта, в каждом из которых четыре операционных блока — логическое устройство (Li), схема сдвига/логическое устройство (Si), умножитель (Mi) и устройство адреса данных (Di). Обычно процессор ядра извлекает из памяти восемь

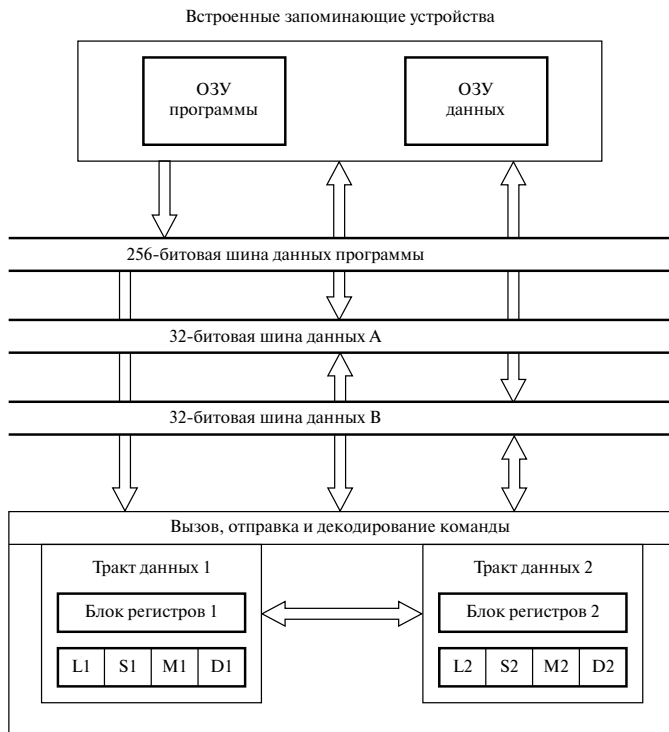


Рис. 12.23. Упрощенная архитектура процессора ЦОС с фиксированной запятой четвертого поколения (Texas Instruments TMS320C62x). Обратите внимание на два независимых арифметических тракта передачи данных, каждый из которых имеет четыре операционных блока — L1, S1, M1 и D1; L2, S2, M2 и D2

32-битовых команд за раз, так что длина получающегося слова равна 256 бит (отсюда и название — *с командными словами сверхбольшой длины*). При наличии в сумме восьми операционных блоков, по четыре в каждом тракте, процессор TMS320C62x может выполнять до восьми команд параллельно в одном такте. Процессор имеет значительную программную память и большой кэш данных (обычно 4 Кбайт программного кэша/кэша данных уровня 1 и 64 Кбайт программного кэша/кэша данных уровня 2). Каждый тракт данных имеет собственный блок регистров (шестнадцать 32-битовых регистров), но может также обращаться к регистрам другого тракта данных. Преимущества архитектуры VLIW включают простоту и значительную вычислительную эффективность. Недостатки — повышенное использование программной памяти (организация кодов для согласования внутреннего параллелизма процессора может привести к неэффективному использованию памяти). Более того, оптимальной производительности процессоров можно достичь, только когда все операционные блоки заняты, что не всегда возможно вследствие зависимости по данным, задержек команд и ограничений при использовании операционных блоков. Впрочем, существуют сложные программные средства для сжатия кода, планирования команд, распределения ресурсов и использования богатого потенциала процессора вообще.

12.3.2. Процессоры ЦОС с плавающей запятой

Возможность разработки процессоров для выполнения операций ЦОС с высокой скоростью, высокой точностью и использованием арифметики с плавающей запятой стала долгожданным открытием. Такие операции минимизируют влияние конечной разрядности, которое проявляется как переполнение, ошибки округления и ошибки квантования коэффициентов. Кроме того, такой подход облегчает разработку алгоритмов, поскольку разработчик может создать алгоритм на большом компьютере на языке высокого уровня, а затем перенести его на устройство ЦОС проще, чем он бы это сделал при арифметике с фиксированной запятой.

Ключевые особенности процессоров ЦОС с плавающей запятой аналогичны свойствам процессоров с фиксированной запятой — существуют специальные команды для операций ЦОС и множественные тракты передачи данных для выполнения нескольких операций. При этом, как и процессоры ЦОС с фиксированной запятой, доступные процессоры ЦОС с плавающей запятой существенно отличаются архитектурно. Некоторые ключевые особенности трех поколений процессоров ЦОС с плавающей запятой производства Texas Instruments и Analog Devices сведены в табл. 12.2.

Пожалуй, самым известным семейством универсальных процессоров ЦОС с плавающей запятой первого поколения является TMS320C3x, которое представлено 32-битовыми однокиповыми цифровыми процессорами сигналов и поддерживает как арифметические операции с целыми числами, так и действия с плавающей запятой. Процессоры имеют большую память и множество встроенных периферийных средств, упрощающих разработку систем. В число этих средств входят программный кэш (позволяет ускорить выполнение часто вызываемых кодов) и встроенная память двойного доступа. Размеры областей памяти удовлетворяют запросам приложений с большими требованиями к памяти, как, например, графические приложения и задачи из сферы обработки изображений. В процессоре TMS320C30 умножение с плавающей запятой требует 32-битовых операндов и выдает 40-битовое нормированное произведение, представленное в форме с плавающей запятой. Целочисленное умножение требует 24-битовых входов и дает 32-битовый результат. Всего поддерживается три формата с плавающей запятой. Первый — короткий 16-битовый формат с 4-битовой экспонентой, одним бит знака и 11 бит мантиссы. Данный формат используется в тех операциях с плавающей запятой, где требуется немедленный результат. Второй формат — формат обычной точности с 8-битовой экспонентой, одним бит знака и 23-битовой мантиссой (всего 32 бит). Третий — это 40-битовый формат повышенной точности, который имеет 8-битовую экспоненту, один бит знака и 31-битовую мантиссу. Отметим, что описанное представление в форме с плавающей запятой отличается от стандарта IEEE, но процессор имеет средства конвертации между двумя форматами. Процессор TMS320C3x объединяет особенности гарвардской архитектуры (отдельные шины для команд программы, данных и ввода-вывода) и фон-неймановской архитектуры (унифицированное адресное пространство).

Во втором поколении универсальных процессоров ЦОС с плавающей запятой акцент был сделан на многопроцессорной обработке и поддержке нескольких процессоров. Ключевыми вопросами здесь являются передача данных между процессорами, передача данных с прямым доступом к памяти и совместное использование глобальной памяти.

Таблица 12.2. Особенности универсальных процессоров ЦОС с фиксированной запятой производства Texas Instruments, Motorola и Analog Devices

Поко- ление	ЦОС с фиксированной запятой	Ширина тракта данных (биты)	Число трактов данных	Длина слова накопителя (биты)	Длина слова команды (биты)	Размер встроенного ОЗУ(слова)	Размер кэша команд (число команд)	Число умножителей	Коэффициент производительности*
1	TMS320C10 (Texas Instruments)	16	1	16	32	16	144	1	
2	TMS320C50 (Texas Instruments)	16	2	16	32	16	10 K	1	10 @ 50 МГц
	DSP56002 (Motorola)	24	2	24	56	24	1 K	1	13 @ 50 МГц
	DSP-2100 (Analog Devices)	16	2	16	40	24	32 K	1	13 @ 52 МГц
	1600 (Lucent Technologies)	16	2	16	36	16	15	1	22 @ 120 МГц
3	TMS320C54 (Texas Instruments)	16	3	16	40	16	32 K	1	25 @ 100 МГц
	DSP56300 (Motorola)	24	3	24	56	24		1	25 @ 100 МГц
	16000 (Lucent Technologies)	32	2	32	40	32	127 K	2	36 @ 100 МГц
4	TMS320C6200 (Texas Instruments)		2		40	256	17 K	2	86 @ 133,6 МГц

* Коэффициент производительности рассчитывается на основе скорости расчета контрольных ядер/вычисления контрольных алгоритмов ЦОС (см. [1, 7]).

Наиболее известными семействами процессоров ЦОС с плавающей запятой второго поколения являются TMS320C4x (Texas Instruments) и ADSP-2106x SHARC (Analog Devices). Семейство C4x имеет некоторые архитектурные особенности семейства C3x, но разработано для многопроцессорной обработки. Семейство C40x имеет хорошие возможности ввода-вывода — шесть последовательных портов для передачи данных между процессорами и шесть 32-битовых каналов прямого доступа к памяти для быстрой передачи данных. Данная архитектура позволяет выполнять несколько операций параллельно в одном такте. Семейство C4x поддерживает арифметику с плавающей запятой и фиксированной запятой. Родной формат данных с плавающей запятой в процессоре C40 отличается от стандарта IEEE 754/854, хотя преобразовать формат довольно легко.

Процессоры ЦОС ADSP-2106x SHARC (Analog Devices) также являются 32-битовыми устройствами с плавающей запятой. Они имеют большую внутреннюю память и впечатляющие возможности ввода-вывода — 10 каналов прямого доступа к памяти, позволяющих обращаться к внутренней памяти без вмешательства процессора, и шесть портов связи для высокоскоростной передачи данных между процессорами. Архитектура допускает совместное использование глобальной памяти, так что до шести процессоров SHARC могут обращаться к внутренним ОЗУ друг друга на скорости передачи данных, достигающей максимальной. Семейство ADSP-2106x поддерживает арифметику с фиксированной запятой и плавающей запятой. Формат обычной точности согласуется со стандартом IEEE 754/854 (24-битовая мантисса и 8-битовая экспонента). Архитектура также позволяет выполнять за такт несколько операций.

В процессорах ЦОС *третьего поколения* еще больше была развита концепция параллелизма, увеличено число команд и число операций за такт, чтобы процессоры удовлетворяли требованиям многоканальных и вычислительно интенсивных приложений. Это стало возможным за счет применения новых архитектур, в частности, VLIW (Very Long Instruction Word — архитектура с командными словами сверхбольшой длины) и суперскалярной архитектуры. Двумя лидирующими семействами процессоров ЦОС с плавающей запятой третьего поколения являются TMS320C67x (Texas Instruments) и ADSP-TS001 (Analog Devices). Семейство TMS320C67x имеет ту же архитектуру VLIW, что и улучшенные процессоры ЦОС с фиксированной запятой четвертого поколения TMS320C62x (детальная информация представлена в разделе 12.3.1 и на рис. 12.23).

Семейство TigerSHARC поддерживает смешанные арифметические типы (арифметику с плавающей и фиксированной запятой) и типы данных (8-, 16- и 32-битовые числа). Данная гибкость позволяет повышать эффективность за счет выбора типа арифметики и формата данных, наиболее подходящих для данного приложения. Как и семейство TMS320C67x, процессоры TigerSHARC нацелены на масштабные многоканальные приложения, такие как мобильные системы третьего поколения (беспроводные системы третьего поколения), цифровые абонентские линии (xDSL) и модемы удаленного множественного доступа для служб Internet. Процессор TigerSHARC с его статической суперскалярной архитектурой (см. рис. 12.14 и раздел 12.3.1) объединяет хорошие особенности архитектуры VLIW, обычной архитектуры ЦОС и компьютеров RISC (reduced instruction set computer — компьютер с сокращенным набором команд). Процессор имеет два операционных блока, в каждом из которых есть умножитель, АЛУ и 64-битовая схема сдвига. Процессор может выполнять до восьми операций

умножения-накопления за такт с 16-битовыми входами и 40-битовым накоплением, две 40-битовые операции умножения-накопления с 16-битовыми комплексными данными или две 80-битовые операции умножения-накопления с 32-битовыми данными. При 8-битовых данных TigerSHARC может выполнять до 16 операций в такте. TigerSHARC имеет значительную пропускную способность памяти, а сама память организована в три 128-битовых блока. При работе с данными могут использоваться разные форматы — обычные 32-битовые слова, длинные 64-битовые слова и учетверенные 128-битовые слова. В одном такте может обрабатываться до четырех 32-битовых команд. Чтобы избежать использования больших служебных издержек (недостаток структуры VLIW), длинные командные слова могут разбиваться на отдельные короткие команды, которые затем передаются независимо каждому блоку.

12.4. Выбор цифрового процессора сигналов

В последние годы вопрос выбора процессора ЦОС для конкретного приложения становится важным, поскольку число доступных процессоров продолжает расти [1, 8]. В число специфических факторов, которые следует рассмотреть при выборе процессора ЦОС для данного приложения, входят архитектурные особенности, скорость выполнения, тип арифметики и длина слова.

1. *Архитектурные особенности.* Большинство доступных сейчас процессоров ЦОС имеют хорошие архитектуры, ресурсов которых, впрочем, может не хватать для определенного приложения. Ключевыми характеристиками процессоров являются размер встроенной памяти, наличие специальных команд и возможности ввода-вывода. Наличие встроенной памяти — необходимое требование в большинстве приложений ЦОС реального времени, поскольку это означает быстрый доступ к данным и быстрое выполнение программы. Для приложений со строгими требованиями к памяти (цифровое аудио, система Dolby, факс-модем, кодирование-декодирование MPEG) размер внутренней памяти ОЗУ может стать решающим фактором при выборе процессора. Если внутренней памяти недостаточно, ее можно дополнить высокоскоростной внешней памятью, хотя это может привести к увеличению стоимости системы. Для приложений, требующих быстрых и эффективных вычислений или обмена потоками данных с внешним миром, весьма важны такие средства ввода-вывода, как интерфейсы к АЦП и ЦАП, возможность прямого доступа к памяти и поддержка многопроцессорной обработки. В зависимости от приложения важен богатый набор специальных команд поддержки операций ЦОС, например, возможность организации циклов с нулевыми служебными издержками, специализированные команды ЦОС и круговая адресация.
2. *Скорость выполнения.* Поскольку большинство задач ЦОС требует срочного решения, важной мерой производительности является скорость процессоров ЦОС. Традиционно двумя основными единицами измерения этой величины являются тактовая частота процессора в мегагерцах и число выполняемых команд в миллионах команд за секунду (Million Instructions Per Second — MIPS) или, если используются процессоры ЦОС с плавающей запятой, в миллионах операций с плавающей запятой

в секунду (million floating-point operations per second — MFLOPS). Впрочем, подобные меры могут в некоторых случаях не подходить из-за значительных отличий в принципах работы различных процессоров ЦОС, большинство из которых может выполнять несколько операций в одной машинной команде. Например, в семействе процессоров C62x возможно выполнение восьми команд за такт. Для разных процессоров также отличается число операций, выполняемых в каждом такте. Следовательно, сравнивать скорости работы процессоров на основе названных выше мер бессмысленно. Альтернативная мера основана на скорости выполнения контрольных алгоритмов [1, 7] — например, центральных алгоритмов ЦОС, таких как БПФ, КИХ- и БИХ-фильтрация. В табл. 12.1 и 12.2 представлены коэффициенты производительности, рассчитанные на основе подобных контрольных алгоритмов; изучая эти данные, можно получить представление об относительной производительности различных популярных процессоров ЦОС.

3. *Тип арифметики.* Двумя наиболее распространенными типами арифметики, используемыми в современных процессорах ЦОС, являются арифметики с фиксированной и плавающей запятой. Арифметика с плавающей запятой — это естественный выбор в приложениях с широкими и переменными требованиями к динамическому диапазону (динамический диапазон можно определить как разность между наибольшим и наименьшим уровнем сигнала, который можно представить, или как разность между наибольшим сигналом и минимальным уровнем шума, измеренными в децибелах). Процессоры с фиксированной запятой предпочтительны с точки зрения низкой стоимости, они подходят для массового производства (например, сотовые телефоны и компьютерные дисководы). При использовании арифметики с фиксированной запятой возникают вопросы, связанные с ограничениями динамического диапазона, которые обязательно должен рассмотреть разработчик (см. главу 13). Вообще процессоры с плавающей запятой дороже процессоров с фиксированной запятой, хотя в последние годы разница в цене существенно уменьшилась. Отметим также, что большинство существующих процессоров ЦОС с плавающей запятой также поддерживают арифметику с фиксированной запятой.
4. *Длина слова.* Еще одним важным параметром в ЦОС является длина слова данных процессора, поскольку она может существенно влиять на качество сигнала. Этот параметр определяет, насколько точно можно представить параметры и результаты операций ЦОС (см. главу 13). Вообще, чем длиннее слово данных, тем меньше ошибки при цифровой обработке сигнала. В аудиообработке с фиксированной запятой, например, для поддержания CD-качества длина слова процессора должна быть не меньше 24 бит, что позволит поддерживать наименьший уровень сигнала достаточно выше минимального уровня шума, генерируемого обработкой сигнала. В процессорах ЦОС с фиксированной запятой используются разнообразные длины слов процессоров, в зависимости от приложения (см. табл. 12.1). В процессорах ЦОС с фиксированной запятой, нацеленных на рынок телекоммуникаций, обычно используются слова 16-битовой длины (например, TMS320C54x), тогда как в процессорах, нацеленных на аудиоприложения высокого качества, используются 24 бит (например, DSP56300). В последние годы авторы отмечают тенденцию к использованию большего числа битов для АЦП и ЦАП (например, 24-битовый аудиокодек

Cirrus, CS4228), поскольку стоимость подобных устройств постоянно снижается. Таким образом, авторы предвидят повышение спроса на процессоры для аудиообработки с большими длинами слов. В накопителях процессоров с фиксированной запятой также могут требоваться защитные биты (обычно 1–8 бит) для предотвращения арифметического переполнения в процессе операций умножения и накопления повышенной точности. Дополнительные биты эффективно расширяют динамический диапазон, доступный для процессоров ЦОС. В большинстве процессоров ЦОС арифметика обычной точности подразумевает использование 32-битовых данных (24-битовая мантисса и 8-битовая экспонента). Данный размер также совместим с форматом IEEE представления в форме с плавающей запятой (IEEE 754). Большинство процессоров ЦОС с плавающей запятой также позволяют выполнять операции с фиксированной запятой и часто поддерживают арифметику с фиксированной запятой с переменным размером данных.

На практике при выборе процессора в расчет могут приниматься и такие факторы, как опыт работы (знакомство) с конкретным семейством процессоров ЦОС, легкость использования, срок присутствия на рынке и стоимость.

12.5. Реализация алгоритмов ЦОС на универсальных процессорах ЦОС

12.5.1. Цифровая КИХ-фильтрация

Нерекурсивные N -точечные КИХ-фильтры со структурой, представленной на рис. 12.24, *а*, характеризуются следующим разностным уравнением (см. главу 7):

$$y(n) = \sum_{k=0}^{N-1} h(k)x(n-k). \quad (12.4)$$

В программе 12.1 представлен фрагмент реализации на языке С общего КИХ-фильтра. При фильтрации в реальном времени предполагается, что данные и коэффициенты хранятся в памяти, как показано на рис. 12.24, *б*. Чтобы понять, как работает КИХ-фильтр, рассмотрим простой случай $N = 3$, который описывается следующим разностным уравнением:

$$y(n) = h(0)x(n) + h(1)x(n-1) + h(2)x(n-2), \quad (12.5)$$

где $x(n)$ представляет последнюю входную выборку, $x(n-1)$ — предпоследнюю входную выборку, а $x(n-2)$ — предпредпоследнюю входную выборку.

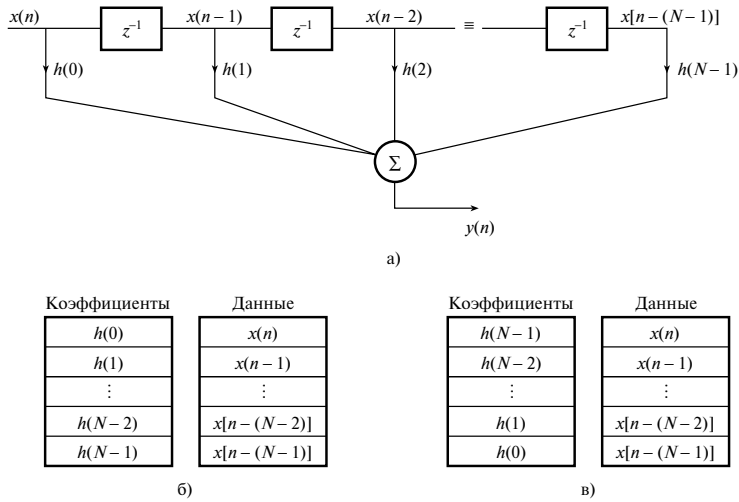


Рис. 12.24. Реализация КИХ-фильтра: а) структура фильтра; б) карты памяти коэффициентов и данных; в) альтернативная карта памяти

Программа 12.1. Псевдокод С для выполнения КИХ-фильтрации

```

nm1=N-1;
yn=0;
for(k=0; k,nm1;++k){      /* данные смещаются, чтобы освободить место
                           для новой выборки */
    x[nm1-k]=x[nm1-k-1];
    x[0]=xn;
}
for(k=0; k,N;++k){
    yn=yn+h[k]*x[k];      /* данные фильтруются и вычисляется
                           выходная выборка*/
}
return(yn);              /* выходная выборка фильтра */
    
```

Предположим, что данные на трехкоэффициентный КИХ-фильтр поступают с АЦП. Первое, что нужно сделать — выделить два набора соседствующих ячеек памяти (в ОЗУ), один для хранения входных данных ($x(n), x(n-1), x(n-2)$), а другой для хранения коэффициентов фильтра, ($h(0), h(1), h(2)$), как показано ниже:

ОЗУ данных	Память коэффициентов
0	$h(0)$
0	$h(1)$
0	$h(2)$

Во время инициализации значения ячеек ОЗУ, в которых планируется хранить выборки данных, обнуляются, поскольку фильтр всегда начинает работу, не имея данных. Затем выполняются следующие операции.

1. *Первый момент выборки.* Считывается выборка данных с АЦП, ОЗУ данных смещается на одну позицию (чтобы освободить место для новых данных), сохраняется новая входная выборка, вычисляется выходная выборка согласно формуле (12.5), а затем выходная выборка посылается ЦАП:

ОЗУ данных	Память коэффициентов	
$\rightarrow x(1)$	$h(0)$	$y(1) = h(0)x(1) + h(1)x(0) + h(2)x(-1)$
0	$h(1)$	
0	$h(2)$	

2. *Второй момент выборки.* Повторяются описанные выше действия, и новая выходная выборка посылается ЦАП:

ОЗУ данных	Память коэффициентов	
$\rightarrow x(2)$	$h(0)$	$y(2) = h(0)x(2) + h(1)x(1) + h(2)x(0)$
$x(1)$	$h(1)$	
0	$h(2)$	

3. *Третий момент выборки.* Повторяются описанные выше действия, и новая выходная выборка посылается ЦАП:

ОЗУ данных	Память коэффициентов	
$\rightarrow x(3)$	$h(0)$	$y(3) = h(0)x(3) + h(1)x(2) + h(2)x(1)$
$x(2)$	$h(1)$	
$x(1)$	$h(2)$	

4. *Четвертый момент выборки.* Повторяются описанные выше действия, и новая выходная выборка посылается ЦАП:

ОЗУ данных	Память коэффициентов	
$\rightarrow x(4)$	$h(0)$	$y(4) = h(0)x(4) + h(1)x(3) + h(2)x(2)$
$x(3)$	$h(1)$	
$x(2)$	$h(2)$	

Обратите внимание на то, что старейшая входная выборка удалена из памяти.

5. *Момент выборки n .* Повторяются описанные выше действия, и новая выходная выборка посылается ЦАП:

ОЗУ данных	Память коэффициентов	
$\rightarrow x(n)$	$h(0)$	$y(n) = h(0)x(n) + h(1)x(n-1) + h(2)x(n-1)$
$x(n-1)$	$h(1)$	
$x(n-2)$	$h(2)$	

Реализация трехточечного КИХ-фильтра на процессоре ЦОС с фиксированной точкой первого поколения (TMS320C10) представлена в программе 12.2. В данном случае расчет произведений начинается с верхнего адреса памяти данных и коэффициентов, что позволяет использовать команду TMS320C10 для передвижения данных. Для реализаций КИХ-фильтров на базе TMS320C10 основными являются пара команд LTD и MPR. Например, приведенная ниже пара команд выполняет сдвиг, подразумеваемый в уравнении (12.4) или обозначенный на рис. 12.24, *a* через z^{-1} , добавляет предыдущее произведение в накопитель и рассчитывает следующее произведение $h(k)x(n-k)$.

Программа 12.2. Линейный код трехточечного КИХ-фильтра

NXTPT	IN	XN, ADC	
	ZAC		
	LT	XNM2	
	MPY	H2	; $h(2)x(n-2)$
	LTD	XNM1	; $0+h(2)x(n-2)$; $x(n-2)=x(n-1)$
	MPY	H1	; $h(1)x(n-1)$; $x(n-1)=x(n-2)$
	LTD	XN	; $h(2)x(n-2)+h(1)x(n-1)$; $x(n-1)=x(n)$
	MPY	H0	; $h(0)x(n)$
	APAC		; $h(2)x(n-2)+h(1)x(n-1)+h(0)x(n)$
	SACH	YN, 1	; сохранить выходную выборку
	OUT	YN, DAC	; передать выходную выборку ЦАП
	B	NXTPT	

LTD XNM1
MPY H1

В частности, команда LTD XNM1 загружает в регистр Т (временный) выборку данных $x(n-1)$ (хранилась в ОЗУ данных по адресу XNM1), добавляет предыдущее произведение $h(2)x(n-2)$, которое все еще находится в регистре Р (произведение), к накопителю и смещает $x(n-1)$ в ячейку со следующим адресом, т.е. $x(n-2) = x(n-1)$. Вторая команда MPY множит содержимое регистра Т на $h(1)$ и оставляет результат в регистре произведения. Схема сдвигов гарантирует, что, когда потребуется вычислять следующую выборку, входные выборки данных будут находиться в правильных ячейках.

Линейный код КИХ-фильтра, подобный программе 12.2, — это быстрая реализация, которая, впрочем, не является универсальной и для больших N -точечных фильтров не позволяет получить компактную программу. В частности, для реализации универсального КИХ-фильтра требуется организовать внутренний цикл, реализующий уравнение КИХ-фильтра и вычисляющий выход фильтра, как задано в уравнении (12.4).

Функциональная схема N -точечной КИХ-фильтрации, на которой показан внутренний цикл, представлена на рис. 12.25. В процессоре ЦОС первого поколения внутренний цикл КИХ-фильтра можно реализовать с помощью следующих команд:

```

LOOP   LTD    *,AR0 ;сдвиг/обновление линии задержки и
                ;накопление произведений
        MPY    *-,AR1 ;умножение следующего коэффициента и элемента данных
        BANZ   LOOP

```

В рассматриваемом случае вспомогательные регистры AR0 и AR1 применяются для указания на элемент данных и коэффициент, которые нужно перемножить. Вспомогательный регистр AR1 содержит длину фильтра и используется как счетчик цикла. Для управления циклом используется команда ветвления BANZ и проверка содержимого регистра AR1. Отметим, что реализация КИХ-фильтра на процессоре ЦОС первого поколения неэффективна из-за служебных издержек, связанных с управлением циклом.

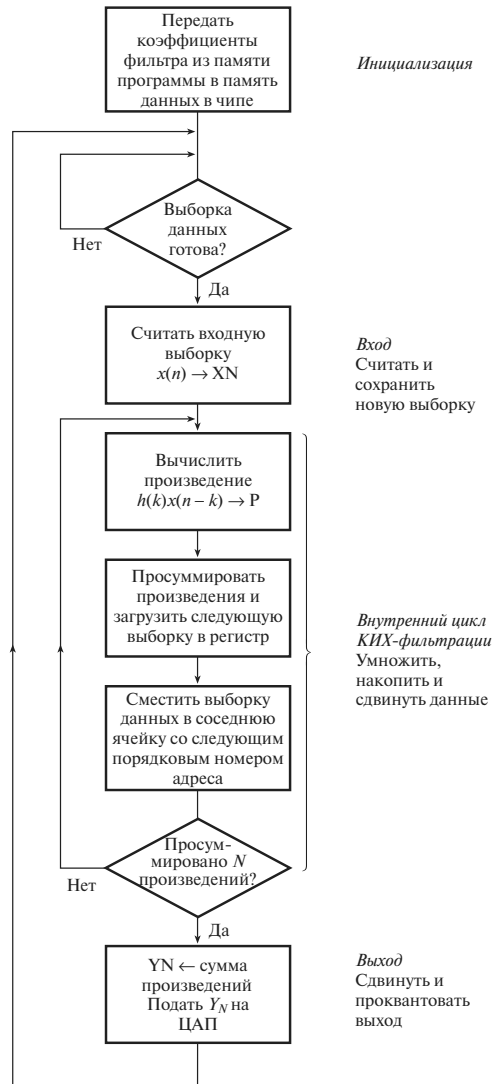


Рис. 12.25. Функциональная схема КИХ-фильтра. Внутренний цикл КИХ-фильтра вычисляет сверточную сумму, представленную в уравнении (12.25)

Процессоры ЦОС с фиксированной запятой второго поколения, такие как TMS320C50 и DSP56000 (Motorola), позволяют создавать циклы с нулевыми служебными издержками и содержат специальные команды умножения и накопления, которые помогают сократить время выполнения внутреннего цикла КИХ-фильтрации. В процессоре TMS320C50, например, внутренний цикл N -точечного КИХ-фильтра, показанный на рис. 12.25, можно эффективно выполнить с помощью следующих команд:

```
RPT      NM1
MACD     HNM1, XNM1
```

Команда RPT NM1 загружает длину фильтра минус 1 (т.е. $(N - 1)$) в регистр повтора и инициирует $(N - 1)$ -кратное выполнение с нулевыми служебными издержками команды умножения и накопления со сдвигом данных MACD. Команда MACD объединяет пару команд LTD MPY в одну, что позволяет быстрее их выполнить. Вообще, пара команд RPT и MACD — это хороший пример экономных специальных команд, доступных в процессорах ЦОС.

Альтернативным подходом к реализации N -точечных КИХ-фильтров на процессорах ЦОС второго и последующих поколений является использование круговых буферов. Очевидно, что в КИХ-фильтрации содержимое памяти коэффициентов статично, а вот память данных меняется с поступлением каждой новой входной выборки. С точки зрения эффективности выгоднее подавать последующие выборки данных в раздвижное окно, удаляя старейшие выборки из памяти. Для обработки изменений в блоке входных данных, используемых при КИХ-фильтрации, можно применять круговой буфер, при этом данные не придется сдвигать, как в линейных буферах.

Концептуально круговой буфер идентичен линейному, если рассматривать два конца линейного буфера как соседние ячейки памяти, т.е. самая новая и самая старая выборки данных $x(n)$ и $x[n - (N - 1)]$ находятся в соседних ячейках памяти; (рис. 12.26, а). В круговом буфере на рис. 12.26 указатель данных (обозначен стрелочкой) ссылается на ячейку памяти с новейшей входной выборкой $x(n)$, а предыдущие входные выборки данных $x(n - 1)$, $x(n - 2)$, ..., $x(n - 7)$ хранятся в последующих ячейках памяти по часовой стрелке. В каждый момент получения выборки, как и ранее, выполняется внутренний цикл КИХ-фильтра — каждая выборка данных множится на соответствующий коэффициент фильтра $h(n)$, и произведения накапливаются. Единственное отличие состоит в том, что выборки данных не смещаются. После вычисления внутреннего цикла указатель находится возле ячейки $x(n - 7)$ со старейшей выборкой данных, на место которой затем записывается следующая входная выборка $x(n)$. Работа кругового буфера для трех последовательных выборок данных иллюстрируется на рис. 12.26, а–в.

На практике круговая адресация реализуется за счет использования для автоматического циклического возврата, когда указатели адреса выходят за границы буфера, модульной арифметики. Обычно в таких случаях требуется задать начальный адрес кругового буфера и размер буфера (или размер модуля). Ниже приводится реализация внутреннего цикла N -точечного КИХ-фильтра с использованием круговой адресации на базе процессора DSP56000.

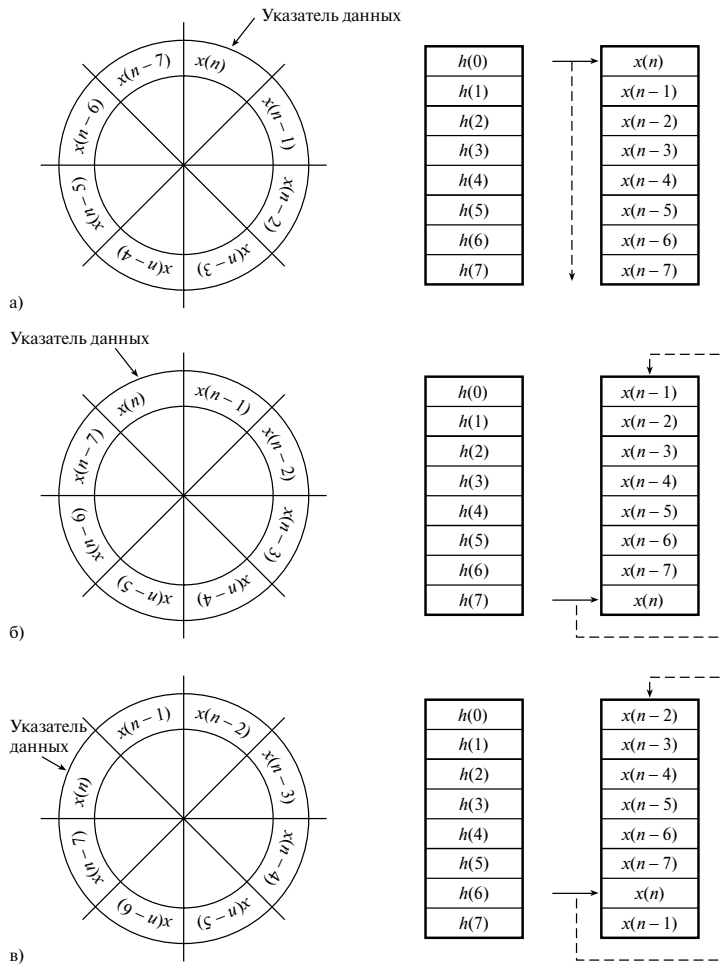


Рис. 12.26. Иллюстрация принципов реализации КИХ-фильтра на основе кругового буфера

```

MOVE    #XDATA, R0
MOVE    #COEFF, R4
MOVE    #N-1, M0      ;размер буфера/модуля
MOVEP   X: INPUT, X: (R0) ;считывание и запись входной выборки
CLR     A              ;очистка накопителя
REP     #N-1          ;выполнение внутреннего цикла КИХ-фильтра
MAC     X0, Y0, A X: (R0)+, X0    Y: (R4)+, Y0
MACR    X0, X0, A (R0)-

```

В представленном случае круговые буферы используются для хранения данных и коэффициентов. Круговой буфер данных выполняет неявный временной сдвиг, как описано выше. В то же время, для удобства автоматического циклического возврата указателя коэффициентов используется круговой буфер коэффициентов. Первые четыре команды в приведенной выше программе устанавливают указатели адреса R0 и R4. Пара команд REP и MAC выполняет внутренний цикл КИХ-фильтра. Команда повтора REP повторяет следующую за ней команду $N - 1$ раз. В следующей строке для выполнения набора операций умножения используется архитектура и параллелизм DSP56000 — здесь умножаются элементы данных и коэффициенты, расположенные в ячейках X0 и Y0, произведение добавляется к накопителю, из ячеек X и Y извлекается следующая пара “элемент данных — коэффициент” и обновляются указатели.

Кроме области КИХ-фильтрации, круговая адресация применяется при эффективной реализации нескольких других функций ЦОС, требующих временных сдвигов или простых очередей, например, корреляции, фильтров с обработкой при нескольких скоростях (фильтров децимации и интерполяции) и генерации периодических сигналов. Использование круговой адресации делает ненужным смещение данных или постоянную проверку (обновление) указателей адреса. Все процессоры ЦОС старших поколений имеют улучшенные возможности круговой адресации.

Пример 12.3

Цифровой фильтр, удовлетворяющий приведенным ниже спецификациям, нужно реализовать на базе процессора ЦОС с фиксированной запятой второго поколения TMS320C50.

Частота режекции	1,875 кГц.
Подавление на частоте режекции	60 дБ.
Граничные частоты полосы пропускания	1,575 и 2,175 кГц.
Неравномерность в полосе пропускания	0,01 дБ.
Частота дискретизации	7,5 кГц.

Решение

Приведенным спецификациям удовлетворяет 61-точечный оптимальный КИХ-фильтр. Разработка этого фильтра подробно рассмотрена в разделе 7.6.5. Здесь мы остановимся только на его реализации. Коэффициенты фильтра квантуются до 16 бит (формат Q15) путем умножения каждого коэффициента на 2^{15} с последующим округлением результата до ближайшего целого. Квантованные и неквантованные коэффициенты фильтра перечислены в табл. 12.3. Как показано на функциональной схеме на рис. 12.25, полный КИХ-фильтр имеет по крайней мере четыре необходимых элемента.

1. Система инициализации, которая может включать настройку таблицы коэффициентов.
2. Вводная часть программы, которая может включать считывание входной выборки $x(n)$, например, с АЦП через последовательный порт.
3. Вычисление внутреннего цикла. Реализация уравнения КИХ-фильтрации для получения $y(n)$.
4. Выходная часть программы, включающая сдвиг (округление) результата, полученного при расчете внутреннего цикла, и передачу его, например, ЦАП через последовательный порт.

Таблица 12.3. Коэффициенты фильтра из примера 12.3

<i>Квантованные коэффициенты</i>	
Длина фильтра = 61	
***** Импульсная характеристика *****	
$H(1) = 0,12743640E - 02 = H(61)$	42
$H(2) = 0,26730640E - 05 = H(60)$	0
$H(3) = -0,23681110E - 02 = H(59)$	-78
$H(4) = -0,17416350E - 05 = H(58)$	0
$H(5) = 0,43428480E - 02 = H(57)$	142
$H(6) = 0,53579250E - 05 = H(56)$	0
$H(7) = -0,71570240E - 02 = H(55)$	-235
$H(8) = -0,49028620E - 05 = H(54)$	0
$H(9) = 0,10897540E - 01 = H(53)$	357
$H(10) = 0,89629280E - 05 = H(52)$	0
$H(11) = -0,15605960E - 01 = H(51)$	-511
$H(12) = -0,85508990E - 05 = H(50)$	0
$H(13) = 0,21226410E - 01 = H(49)$	695
$H(14) = 0,12250150E - 04 = H(48)$	0
$H(15) = 0,27630130E - 01 = H(47)$	-905
$H(16) = -0,11091200E - 04 = H(46)$	0
$H(17) = 0,34579770E - 01 = H(45)$	1133
$H(18) = 0,13800660E - 04 = H(44)$	0
$H(19) = -0,41774130E - 01 = H(43)$	-1369
$H(20) = -0,11560390E - 04 = H(42)$	0
$H(21) = 0,48832790E - 01 = H(41)$	1600
$H(22) = 0,12787590E - 04 = H(40)$	0
$H(23) = -0,55359840E - 01 = H(39)$	-1814
$H(24) = -0,90065860E - 05 = H(38)$	0
$H(25) = 0,60944450E - 01 = H(37)$	1997
$H(26) = 0,88997300E - 05 = H(36)$	0
$H(27) = 0,65232190E - 01 = H(35)$	-2137
$H(28) = 0,38167120E - 05 = H(34)$	0
$H(29) = 0,67925720E - 01 = H(33)$	2226
$H(30) = 0,27041150E - 05 = H(32)$	0
$H(31) = 0,93115220E + 00 = H(31)$	30512

Поскольку этапы 1, 2 и 4 зависят от системы, сконцентрируемся на вычислении внутреннего цикла. Внутренний цикл КИХ-фильтра можно реализовать с помощью следующих команд TMS320C50:

```

SACL XN          ;записать новейшую выборку x(n) в память данных
LAR  AR1, #XNM1  ;указать на положение старейшей выборки данных x[n-(N-1)]
ZAP              ;очистить накопитель и регистр произведения
MAR *, AR1       ;сделать AR1 текущим вспомогательным регистром
RPT #60          ;выполнить внутренний цикл КИХ-фильтрации
MACD #COEFF, *-  ;умножить и накопить со сдвигом данных
APAC             ;прибавить последнее произведение

```

В данном случае памяти данных и коэффициентов организованы так, как показано на рис. 12.24, в. Вспомогательный регистр AR1 используется для организации непрямого адресации при вычислении внутреннего цикла (команда MACD) и изначально указывает

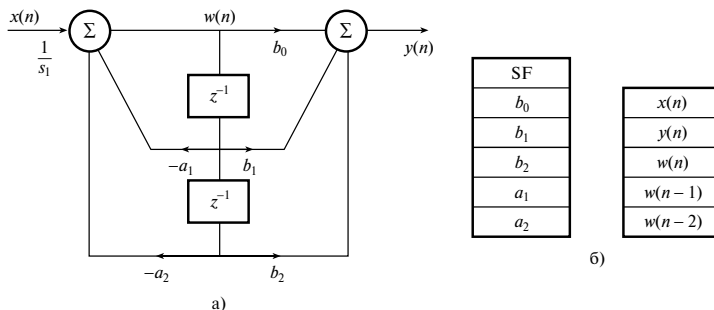


Рис. 12.27. Каноническое звено второго порядка: а) диаграмма реализации; б) память для хранения данных и коэффициентов

на старейшую выборку данных, XNM1, в памяти для хранения данных. Во внутреннем цикле команда MACD делает следующее:

- добавляет предыдущее произведение к накопителю — изначально произведение равно нулю;
- умножает коэффициент $h(k)$ на данные, на которые ссылается указатель AR1 — изначально $h(k) = h(N-1)$, и устанавливает во вспомогательном регистре ссылку на $x[n - (N-1)]$;
- копирует данные, на которые указывает AR1, в ячейку памяти со следующим адресом — изначально $x[n - (N-1)]$ копируется в $x(n-N)$; т.е. старейшая выборка данных отбрасывается. Последняя команда MACD копирует $x(n)$ в $x(n-1)$, чтобы освободить место для следующей входной выборки;
- уменьшает AR1 на 1 (т.е. указатель теперь ссылается на следующую выборку в памяти для хранения данных) — изначально AR1 указывает на $x[n - (N-1)]$, а затем последовательно указывает на $x[n - (N-2)]$, $x[n - (N-3)]$, ..., $x(n)$ по мере кругового прохождения цикла;
- увеличивает адрес COEFF на 1; последовательно получают следующие адреса: $h(N-1)$, $h(N-2)$, ..., $h(0)$.

12.5.2. Цифровая БИХ-фильтрация

12.5.2.1. Стандартные компоновочные блоки БИХ-фильтров

В качестве стандартных компоновочных блоков при создании цифровых БИХ-фильтров используются звенья БИХ-фильтров второго порядка. Двумя наиболее используемыми структурами второго порядка являются каноническое звено (рис. 12.27) и реализация в прямой форме (рис. 12.28). Каноническое звено второго порядка характеризуется следующими уравнениями:

$$w(n) = SF_1 x(n) - a_1 w(n-1) - a_2 w(n-2), \quad (12.6, a)$$

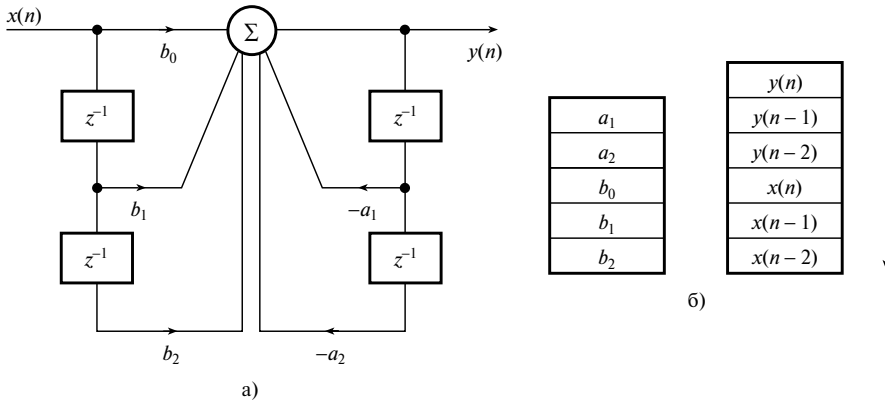


Рис. 12.28. Реализация звена второго порядка в прямой форме: а) диаграмма реализации; б) память для хранения данных и коэффициентов

$$y(n) = b_0 w(n) + b_1 w(n-1) + b_2 w(n-2), \quad (12.6, б)$$

где $x(n)$ представляет входные данные, а $w(n)$ — внутренний узел, $y(n)$ — выходная выборка фильтра, SF_1 — масштабный множитель, равный $1/s_1$. Разностное уравнение звена БИХ-фильтра, реализованного в прямой форме (рис. 12.28, б), записывается как

$$y(n) = b_0 x(n) + b_1 x(n-1) + b_2 x(n-2) - a_1 y(n-1) - a_2 y(n-2), \quad (12.7)$$

где $x(n-k)$ — последовательность входных, а $y(n-k)$ — выходных данных. Память для хранения данных и коэффициентов для реализации в прямой форме изображена на рис. 12.28, б.

Фильтр в прямой форме проще программировать, и такая реализация может быть несколько быстрее, чем реализация в канонической форме, поскольку в ней используется более простая система индексации: сравните, например, уравнения (12.6) и (12.7).

БИХ-фильтры больших порядков реализуются как каскадная либо параллельная комбинация звеньев фильтров второго порядка (см. главу 8).

Каскадная реализация

Передаточная функция $H(z)$ БИХ-фильтра N -го порядка, полученного как каскад звеньев второго порядка, записывается следующим образом:

$$H(z) = \prod_{k=1}^{N/2} \frac{b_{0k} + b_{1k}z^{-1} + b_{2k}z^{-2}}{1 - a_{1k}z^{-1} - a_{2k}z^{-2}}. \quad (12.8)$$

Каскадная реализация БИХ-фильтра четвертого порядка ($N = 4$) с использованием канонических звеньев второго порядка показана на рис. 12.29, а. Память для хранения переменных фильтра (данных и коэффициентов) показана на рис. 12.29, б. Набор

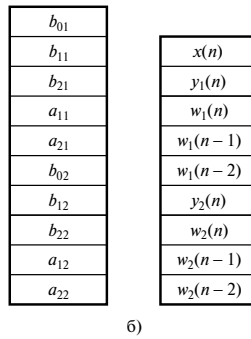
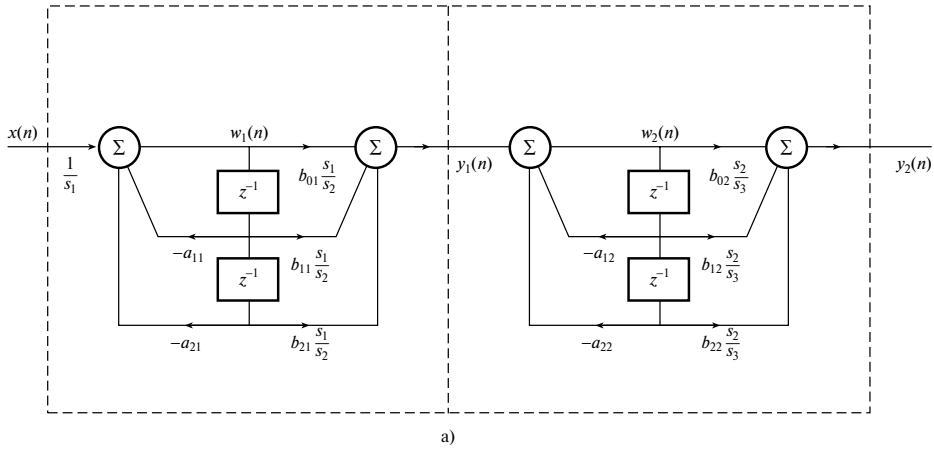


Рис. 12.29. Каскадная реализация БИХ-фильтра: а) диаграмма реализации; б) память для хранения данных и коэффициентов

разностных уравнений БИХ-фильтра четвертого порядка, полученного с помощью канонических звеньев, записывается следующим образом:

$$w_1(n) = \text{SF}_1 x(n) - a_{11} w_1(n-1) - a_{21} w_1(n-2), \quad (12.9, a)$$

$$y_1(n) = b_{01} w_1(n-1) + b_{21} w_1(n-2), \quad (12.9, б)$$

$$w_2(n) = y_1(n) - a_{12} w_1(n-1) - a_{22} w_2(n-2), \quad (12.9, в)$$

$$y_2(n) = b_{02} w_2(n) + b_{12} w_2(n-1) + b_{22} w_2(n-2). \quad (12.9, г)$$

Псевдокод С для реализации БИХ-фильтра в форме каскада канонических звеньев второго порядка представлен в программе 12.3.

В программах 12.4 и 12.5 соответственно представлены фрагменты реализаций БИХ-фильтра N -го порядка, который состоит из каскадного соединения M биквадратных звеньев (где $M = N/2$), на базе процессоров TMS320C50 и DSP56000.

Программа 12.3. Псевдокод С для реализации каскадного БИХ-фильтра

```

for(n=0; n, (Nsamples-1); ++n){      /* Nsamples - число выборок данных */
    xn = x[n];
    for(k=1; k, N; ++k){
        wk=sk[k]*xn-a1[k]*w1[k]-a2[k]*w2[k];
        yk=(b0[k]*wk+b1[k]*w1[k]+b2[k]*w2[k]); /* выход первого звена */
        w2[k] = w1[k]; /* сдвиг и запись данных элемента задержки */
        w1[k] = wk;
        xn = yk; /* выход k-го звена подается на вход следующего звена */
    }
    y[n] = yk;                          /* n-я выходная выборка */
}

```

Программа 12.4. Реализация M каскадно соединенных канонических звеньев фильтров второго порядка на базе процессора TMS320C50

SPLK	#M-1, BRCR	; число биквадратных звеньев	
RPTB	M_IIR	; расчет M биквадратных звеньев, соединенных каскадом	
LT	*-, AR2	; загрузка wk(n-2)	
MPYA	+, AR1	; расчет wk(n-2)*ak(2)	
MPY	+	; wk(n-1)* ak(1)	
LTA	*, AR1	; расчет и запись в память wk(n)=x(n)+	
		; wk(n-1)*ak(1)+wk(n-2)*ak(2)	
SACH	*0+, 1		
MPY	*-	; расчет wk(n-2) * bk(2)	
LACL	#0	;	
LTD	*-, AR2	; сдвиг данных wk(n-2) = wk(n-1)	
MPY	*, AR1	; yk=yk+wk(n-2) * bk(2), wk(n-1)*bk(1)	
LTD	*-, AR2	; сдвиг данных wk(n-1) = wk(n)	
MPY	*, AR1	; расчет wk(n-2) * bk(2) +	
		; wk(n-1) * bk(1), wk(n) * bk(0)	
M_IIR :			
LTA	*, AR4	; прибавить последнее произведение	
SACH	*, 1	; квантование и запись выходной выборки	

Программа 12.5. Реализация M каскадно соединенных канонических звеньев фильтров второго порядка на базе процессора DSP56000

DO	#M, M_IIR	; расчет M биквадратных звеньев	
MAC	-X0, Y0, A	X: (R0)-, X1	Y: (R4)+, Y0 ;
MACR	-X1, Y0, A	X1, X: (R0)+	Y: (R4)+, Y0 ; сдвиг данных
			; w(n-2)= w(n-1)
MAC	X0, Y0, A	A, X: (R0)+	Y: (R4)+, Y0
MAC	X1, Y0, A	X: (R0) +, X0	Y: (R4)+, Y0
M_IIR			
RND			
MOVEP A, Y: OUTPUT			

Пример 12.4

1. Разработайте и реализуйте БИХ-фильтр нижних частот, используя процессор ЦОС с фиксированной запятой TMS320C50. Фильтр должен удовлетворять следующим спецификациям:

частота дискретизации	15 кГц,
полоса пропускания	0–3 кГц,
ширина полосы перехода	450 Гц,
неравномерность в полосе пропускания	0,5 дБ,
затухание в полосе подавления	45 дБ.
2. Повторите п. 1, используя процессор ЦОС с фиксированной запятой TMS320C54.
3. Повторите п. 1, используя процессор ЦОС с фиксированной запятой DSP56000.
4. Повторите п. 1, используя процессор ЦОС с фиксированной запятой DSP56300.

Решение

Подробно разработка фильтра рассмотрена в главе 8 (раздел 8.8). Показано, что спецификации удовлетворяет эллиптический фильтр четвертого порядка со следующей передаточной функцией:

$$H(z) = \frac{1 + 0,675\,718z^{-1} + z^{-2}}{1 - 0,495\,935z^{-1} + 0,761\,864z^{-2}} \times \frac{1 + 1,649\,656z^{-1} + z^{-2}}{1 - 0,829\,328z^{-1} + 0,307\,046z^{-2}}.$$

Разностные уравнения каскадной реализации с использованием канонических звеньев совпадают с приведенными в формуле (12.9).

Коэффициенты, масштабированные, чтобы избежать переполнения, и квантованные до 16 бит, перечислены в табл. 12.4. Коды реализации фильтра четвертого порядка с использованием фильтров в канонической форме на базе процессоров TMS320C50 и TMS320C54 не приводятся из-за недостатка места, но их можно найти в книге [Ifeachor, 2001] (см. предисловие). Для реализаций на базе DSP56000 и DSP56300 коэффициенты квантуются до 24 бит (длина коэффициентов).

Таблица 12.4. Коэффициенты фильтра до и после квантования до 16 бит

	<i>Коэффициент</i>	<i>Масштабированный коэффициент</i>	<i>Квантованный коэффициент</i>
b_{02}	1	0,9999695	32767
b_{12}	0,675718	0,675718	22142
b_{22}	1	0,9999695	32767
a_{12}	–0,495935	–0,495935	–16251
a_{22}	0,761864	0,761864	24965
b_{01}	1	0,1311136	4296
b_{11}	1,649656	0,2162924	7087
b_{21}	1	0,1311136	4296
a_{11}	–0,829328	–0,829328	–27175
a_{21}	0,307046	0,307046	10061

$$s_1 = 2,479\,158\,(L_1); \text{ SF} = 0,403\,362\,7; s_2 = 18,908\,47\,(L_1).$$

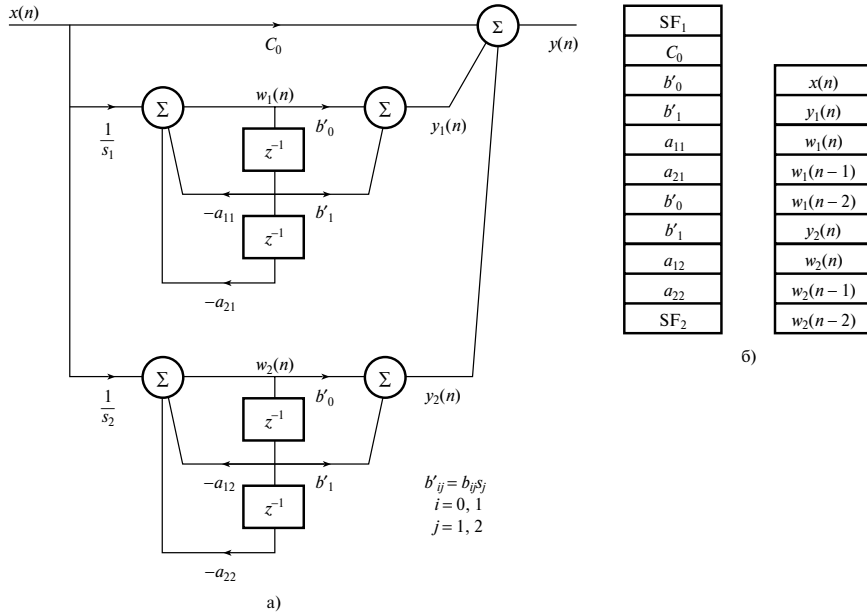


Рис. 12.30. Реализация БИХ-фильтра четвертого порядка: а) диаграмма реализации; б) схема памяти для хранения данных и коэффициентов

Параллельная реализация

Передаточная функция БИХ-фильтра N -го порядка в параллельной реализации записывается следующим образом:

$$H(z) = \prod_{k=1}^{N/2} \frac{b_{0k} + b_{1k}z^{-1}}{1 + a_{1k}z^{-1} + a_{2k}z^{-2}} + C. \quad (12.10)$$

Диаграмма реализации с использованием канонических звеньев второго порядка с $N = 4$ представлена на рис. 12.30. Разностные уравнения канонического звена записываются следующим образом:

$$\begin{aligned} w_1(n) &= \text{SF}_1 x(n) - a_{11}w_1(n-1) - a_{21}w_1(n-2), \\ y_1(n) &= b_{01}w_1(n) + b_{11}w_1(n-1), \\ w_2(n) &= \text{SF}_2 x(n) - a_{12}w_2(n-1) - a_{22}w_2(n-2), \\ y_2(n) &= b_{02}w_2(n) + b_{12}w_2(n-1), \\ y(n) &= c_0x(n) + y_1(n) + y_2(n). \end{aligned}$$

Простой код С для БИХ-фильтра, реализованного как параллельная комбинация канонических звеньев второго порядка, представлен в программе 12.6.

Программа 12.6. Псевдокод С параллельной реализации фильтра четвертого порядка

```

for(n=0; n, (Nsamples-1); ++n){      /* Nsamples - число выборок данных */
    y[n] = c*x[n];                    /* выход через постоянный тракт */
    for(k=1; k,N; ++k){
        wk = sk[k]*x[n]-a1[k]*w1[k]-a2[k]*w2[k];
        yk = (b0[k]*wk+b1[k]*w1[k])/sk[k];      /* выход первого звена */
        w2[k]=w1[k];                      /* сдвиг и запись данных элемента задержки*/
        w1[k]=wk;
        y[n]=yk+y[n];
    }
}

```

Пример 12.5

Представьте передаточную функцию, приведенную в примере 12.4, в параллельной форме, используя в качестве элементарных составляющих канонические звенья второго порядка. Реализуйте фильтр на базе аппаратного обеспечения, указанного в предыдущем примере.

Решение

Используя рассмотренную в главе 4 программу разложения на элементарные дроби, по коэффициентам каскадной реализации получим коэффициенты параллельной реализации. В результате передаточная функция преобразуется к следующему виду:

$$\begin{aligned}
 H(z) = & \frac{-0,132\,922\,5 - 0,180\,523z^{-1}}{1 - 0,028\,994z^{-1} + 0,044\,541\,6z^{-2}} + \\
 & + \frac{-0,058\,534 + 0,508\,420z^{-1}}{1 - 0,048\,489\,9z^{-1} + 0,017\,951\,1z^{-2}} + 0,249\,923\,79 \\
 & s_1 = 5,524\,484\,4, \quad s_2 = 2,4794.
 \end{aligned}$$

Значения коэффициентов до и после квантования до 16 бит приведены в табл. 12.5. Коды реализации фильтра на базе процессоров TMS320C50, TMS320C54, DSP56000 и DSP56300 можно найти в книге [Ifeachor, 2001] (см. предисловие). Для реализации на базе процессоров DSP56000 и DSP56300 коэффициенты квантовались до 24 бит.

Расширение методов реализации, описанных выше, на каскадную и параллельную структуры БИХ-фильтров высоких порядков выполняется относительно просто. Впрочем, если реализовать расчет звена второго порядка как подпрограмму, можно получить более компактный код.

Таблица 12.5. Реализация БИХ-фильтра четвертого порядка из примера 12.5: коэффициенты фильтра до и после квантования до 16 бит

	<i>Неквантованные коэффициенты</i>	<i>Квантованные коэффициенты</i>
SF ₁	0,18100	5931
C ₀	0,24992379	8190
a ₀₁	−0,1329225	−24063
a ₁₁	−0,1805232	−32670
b ₁₁	0,028994	16251
b ₂₁	−0,0445416	−24965
a ₀₂	−0,058534	−4756
a ₁₂	0,5084205	20653
b ₁₂	0,0484899	27178
b ₂₂	−0,017951	−10061
SF ₂	0,40332	13216

12.5.3. Расчет БПФ

Дискретное преобразование Фурье (ДПФ) конечной последовательности данных $x(n)$ определяется следующим образом:

$$X(k) = \sum_{n=0}^{N-1} x(n)W_N^{nk},$$

где W_N , часто именуемое *настроечным параметром*, — это набор комплексных коэффициентов.

Прямое вычисление коэффициентов ДПФ $X(k)$ является трудоемким процессом при больших N . Значительно сократить время расчета $X(k)$ можно с помощью алгоритмов быстрого преобразования Фурье (БПФ). Как обсуждалось в главе 3, основными параметрами алгоритмов БПФ являются коэффициент “бабочки” и настроечный коэффициент.

12.5.3.1. Реализация “бабочки”

На рис. 12.31, a и b изображено два типа “бабочек”, используемых в двоичном БПФ. Применение быстрого преобразования Фурье по обеим схемам дает одинаковые результаты. Для децимации во времени (рис. 12.31, a) на вход “бабочки” подается два значения A и B , по которым вычисляется пара выходов:

$$A' = A + BW_N^k, \quad (12.11, a)$$

$$B' = A - BW_N^k. \quad (12.11, b)$$

Вообще, входные и выходные выборки данных, а также настроечные параметры являются комплексными и в общем виде их можно выразить следующим образом:

$$A = A_{re} + iA_{im}, \quad (12.12, a)$$

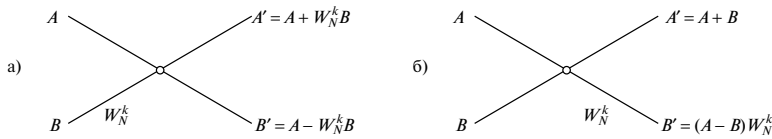


Рис. 12.31. Два типа “бабочек”, используемых в алгоритмах двоичного БПФ: а) “бабочка” для двоичного БПФ с децимацией во времени; б) “бабочка” для двоичного БПФ с децимацией по частоте

$$B = B_{re} + iB_{im}, \quad (12.12, б)$$

$$W_N^k = e^{-2\pi i k/N} = \cos(2\pi k/N) - i \sin(2\pi k/N), \quad (12.12, в)$$

где индекс *re* обозначает действительную часть величины, а *im* — мнимую. Операция “бабочка” согласно уравнению (12.11) включает действия с комплексной арифметикой, но на практике она часто выполняется с использованием действительной арифметики. Чтобы выразить эту операцию в форме, подходящей для действительной арифметики, отметим, что произведение B и W в уравнении (12.11) имеет следующий вид:

$$BW_N^k = B_{re} \cos(X) + B_{im} \sin(X) + i[B_{im} \cos(X) - B_{re} \sin(X)], \quad (12.13)$$

где $X = 2\pi k/N$. Подставляя уравнения (12.12) и (12.13) в уравнения (12.11, а и б), получаем:

$$A' = A_{re} + [B_{re} \cos(X) + B_{im} \sin(X)] + i\{A_{im} + [B_{im} \cos(X) - B_{re} \sin(X)]\} \quad (12.14, а)$$

$$B' = A_{re} - [B_{re} \cos(X) + B_{im} \sin(X)] + i\{A_{im} - [B_{im} \cos(X) - B_{re} \sin(X)]\}. \quad (12.14, б)$$

Теперь выходы “бабочки” A' и B' представлены в нужной форме. Следовательно, для данной пары комплексных точек A и B в ПДСК по формулам (12.14, а и б) можно вычислить выход “бабочки”, используя только действительную арифметику.

Вычисление синуса и косинуса в формуле (12.14) является трудоемкой операцией. В БПФ реального времени эффективнее заранее вычислить действительную и мнимую части настроечного параметра (уравнение (12.12, в)) и сохранить данные значения в таблице соответствий. Предварительный расчет значений настроечного параметра иллюстрируется в программе 12.7.

Программа 12.7. Псевдокод С для предварительного расчета значений настроечного коэффициента

```

pi=6.28315307179586/N;
for(k=0; k<N/2; ++k){
    X=k*pi;
    w.real[k]=cos[X];
    w.imag[k]=sin[X];
}

```

В программе 12.8 приводится псевдокод С для двоичной “бабочки”, значения настроечного параметра которой вычислены заранее и занесены в таблицу соответствий.

Программа 12.8. Псевдокод С для расчета “бабочки”

```
t.real=br*w.real[k]+bi*w.imag;
t.imag=bi*w.real[k]-br.w.imag[k];
b.real[j]=a.real-t.real;
b.imag[j]=a.imag-t.imag;
a.real[j]=a.real+t.real;
a.imag[j]=a.imag+t.imag;
```

Псевдокод С для реализации описанного алгоритма на базе процессора TMS320C25 приведен в программе 12.9. Значения настроечного множителя рассчитаны предварительно и сохранены в формате Q15. Предполагается, что входные данные комплексные, их действительные и мнимые части хранятся в последовательных ячейки ОЗУ данных. Для комплексных выходных данных A' и B' могут иметь максимальные значения 2,41442; для действительных входов максимальное значение равно 2. В арифметике с фиксированной запятой это может стать причиной переполнения. Чтобы избежать этого, данные на входе “бабочки” следует масштабировать. В реализации на базе TMS320C50 масштабирование является динамическим, здесь используются преимущества того, что при умножении двух чисел с фиксированной запятой получается один лишний бит знака. Этот дополнительный бит обычно удаляется путем левого сдвига, но результат при этом эффективно масштабируется в два раза.

Программа 12.9. Код для реализации “бабочки” на базе процессора TMS320C50

```
*
*   расчет членов, общих для двух "бабочек", A' и B'
*
*
*       LT      BR      ;расчет 1/2*[b.real*cos(X)+b.imag*sin(X)]
*       MPY      WREAL   ;1/2*b.real*cos(X)
*       LTP      BI
*       MPY      WIMAG   ;1/2*b.imag*sin(X)
*       APAC     ;1/2[b.real*cos(X)+b.imag*sin(X)]
*       MPY      WREAL   ;1/2*b.imag*cos(X)
*       LT      BR
*       SACH     BR      ;1/2[b.real*cos(X)+b.imag*sin(X)]
*       PAC      ;расчет [q.imag*cos(X)-q.real*sin(X)]
*       MPY      WIMAG
*       SPAC
*       SACH     BI
*
*
*   расчет и запись выходов "бабочки"
*
*       LACC     AR, 14   ;расчет и запись действительных частей выхода
*       ADD      BR, 15
*       SACH     AR, 1    ;запись a.real
*       SUB      BR
```

SACH	BR, 1	; запись b.real
LAC	AI, 14	; расчет и запись мнимых частей выхода
ADD	BI, 15	
SACH	AI, 1	; запись a.imag
SUB	BI	
SACH	BI, 1	; запись b.imag

12.5.3.2. Вычисление с замещением и постоянная геометрия

На рис. 12.32 показана схема прохождения сигнала по восьмиточечному БПФ. Из рисунка очевидно, что для получения коэффициентов ДПФ $X(k)$, показанных справа, при данном входе необходимо рассчитать несколько “бабочек”. Для упорядоченного проведения ряда вычислений по схеме “бабочка” обычно используется эффективный алгоритм двоичного БПФ. На приведенной схеме сигнал проходит слева направо. Следовательно, после вычисления выходов “бабочки” A' и B' входы A и B уже не нужны, поэтому выходы можно записывать поверх них. Данная концепция получила название *вычисление с замещением*.

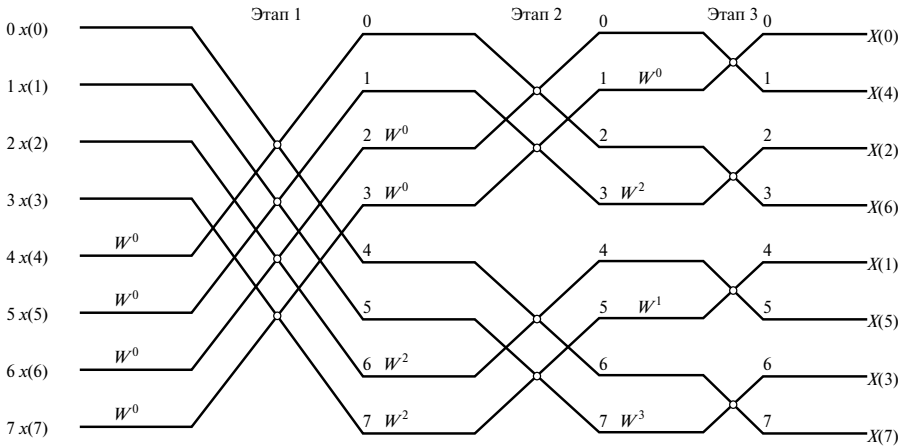


Рис. 12.32. Диаграмма прохождения сигнала по БПФ с децимацией во времени с замещением, где вход идет в естественном порядке, а выход записан в порядке, определяемом обращенными битами

Алгоритмы с замещением позволяют эффективно использовать доступную память, поскольку преобразованные данные записываются на место входных. В прошлом, когда память была очень дорогой, это соображение было существенным. В то же время, при вычислениях с замещением система индексации, позволяющая определить, откуда из памяти извлекать входные данные каждой “бабочки”, достаточно сложна. Например, на рис. 12.32 верхняя “бабочка” на этапе 1 получает вход с адресов 0 и 4 и записывает выход по тем же адресам. С другой стороны, верхняя “бабочка” на этапе 2 принимает вход с адресов 0 и 2. Вообще, при БПФ с замещением входные (выходные) адреса меняются на каждом этапе. Более того, для высокоскоростных БПФ использование одной памяти для входа и выхода замедляет вычисления из-за большого времени доступа

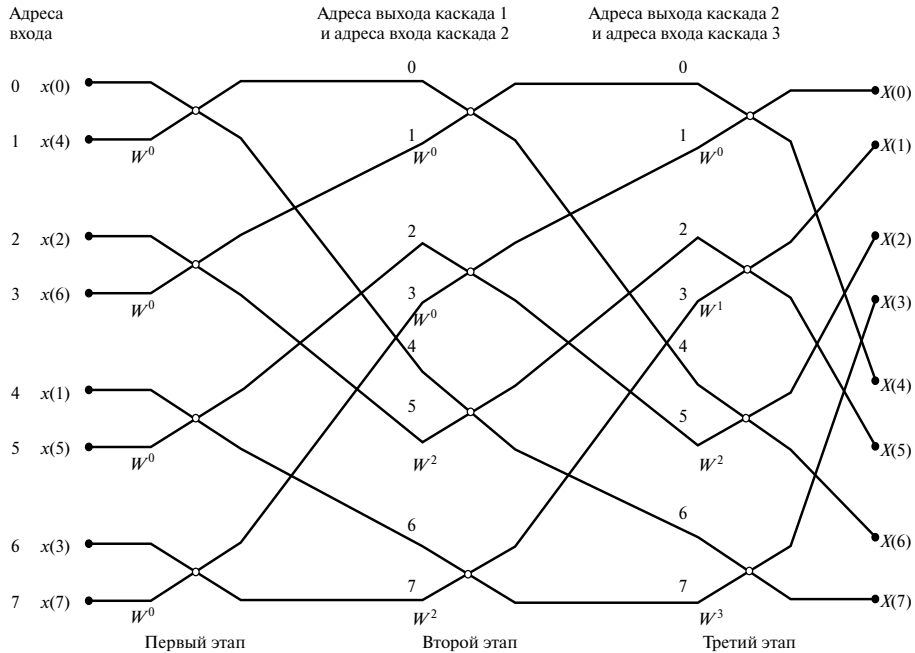


Рис. 12.33. Двоичное БПФ с постоянной геометрией, где вычисления выполняются без замещения ячеек памяти. Обратите внимание на то, что для каждой “бабочки” расстояние между входами и выходами одинаковое

к памяти (исключая, например, случаи использования двухпортовых ОЗУ). Поскольку сейчас память и умножители дешевы, наблюдается тенденция к повышению скорости путем оптимизации всего процессора БПФ.

В альтернативной реализации БПФ, известной как вычисления без замещения, или вычисления с постоянной геометрией, входные данные “бабочки” считываются по паре адресов, а выход записывается в пару ячеек с другими адресами, как показано на рис. 12.33. В отличие от БПФ с замещением, где адреса входа-выхода каждой “бабочки” отличаются для разных этапов, в описываемом случае адресация каждой “бабочки” фиксирована и значительно проще. Для N -точечного БПФ входы n -й “бабочки” на каждом этапе находятся в ячейках с адресами $2n$ и $2n+1$, $n = 0, 1, \dots, N/2-1$. Выходы n -й “бабочки” записываются в ячейки с адресами n и $N/2 + n$. Например, на втором этапе (рис. 12.33) верхняя “бабочка” берет входные значения в ячейках с адресами 0 и 1 и записывает выход в ячейки с адресами 0 и 4. Очевидно, что для работы БПФ без замещения требуются две отдельные области памяти или массива; в одном хранятся входы, а в другом — выходы. После каждого этапа роли областей памяти меняются.

12.5.3.3. Скремблирование данных и обращение битов

Если при БПФ с децимацией во времени последовательность входных данных подается на процессор БПФ в естественном порядке, выход БПФ кажется скремблированным (см. рис. 12.32). Чтобы гарантировать поступление выхода в правильном порядке

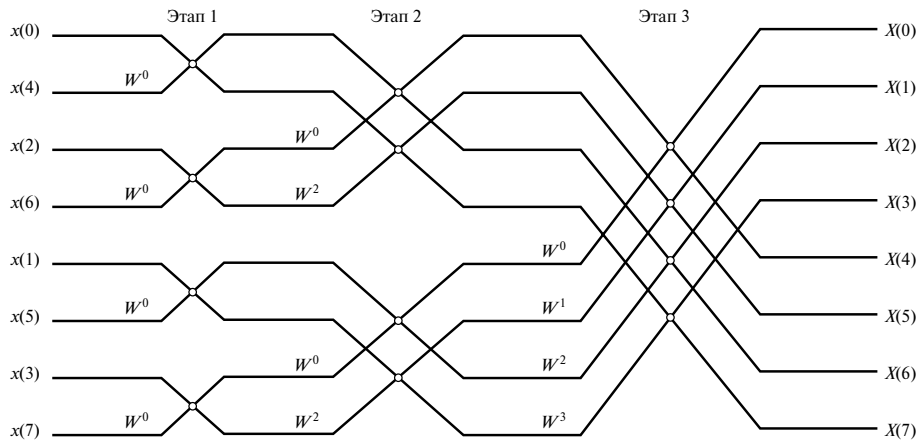


Рис. 12.34. Диаграмма прохождения сигнала для БПФ с децимацией во времени и замещением ячеек памяти; вход представлен в порядке, определяемом обращенными битами, а выход идет в естественном порядке

(т.е. $X(0), X(1), \dots, X(N-1)$), требуется либо скремблировать последовательность входных данных перед БПФ (см. рис. 12.33 и 12.34), либо восстановить естественный порядок выхода после применения БПФ (см. рис. 12.32).

Скремблирование входных данных для двоичного БПФ выполняется путем записи входной последовательности в порядке, определяемом обращенными битами. Предположим, что входные данные записаны в естественном порядке (т.е. как $x(0), x(1), x(2), \dots, x(N-1)$), для записи в порядке, определяемом обращенными битами, индексы входных данных представляются в двоичном виде, как показано во втором столбце табл. 12.6 для восьмиточечного БПФ, а затем биты переставляются относительно центра (четвертый столбец табл. 12.6). Например, в таблице индекс выборки $x(3)$ представляется в двоичной форме как 011. Меняя местами первый и третий биты, получаем 110 (средний бит остается без изменений). Код 110 представляет десятичное число 6. Для создания эффекта скремблирования необходимо переставить выборки $x(3)$ и $x(6)$. Применяя тот же принцип к остальным входам, получаем последовательность, порядок элементов которых определяется обращенными битами (представлена в третьем столбце табл. 12.6). Обратите внимание на то, что после скремблирования положение первой и последней выборки данных не меняется. Это объясняется тем, что обращение битов применительно к индексам 000 и 111 не дает эффекта. Вообще, процесс скремблирования в двоичном БПФ не затрагивает первую и последнюю точки данных. Внимательный читатель может отметить и другие выборки, на которые описанный процесс не влияет.

Когда входные данные содержатся в памяти или в массиве, скремблирование входных данных включает определение пар ячеек с входными данными и перестановкой данных, расположенных в этих ячейках. Для определения индексов ячеек памяти, которые будут обмениваться содержимым, наиболее широко используется алгоритм Рейдера (Rader) (см. [9]). В программе 12.10 представлен псевдокод С с реализацией алгоритма обращения битов.

Таблица 12.6. Демонстрация концепции обращения битов на данных восьмиточечного БПФ

<i>Входная последовательность, естественный порядок</i>	<i>Двоичный код последовательности</i>	<i>Входная последовательность, биты обращены</i>	<i>Двоичный код последовательности (с обращением битов)</i>
$x(0)$	000	$x(0)$	000
$x(1)$	001	$x(4)$	100
$x(2)$	010	$x(2)$	010
$x(3)$	011	$x(6)$	110
$x(4)$	100	$x(1)$	001
$x(5)$	101	$x(5)$	101
$x(6)$	110	$x(3)$	011
$x(7)$	111	$x(7)$	111

Программа 12.10. Обращение битов с замещением ячеек памяти в двоичном БПФ

```
/* обращение битов с замещением */

j=1; for(i=1; i,N; ++i){
    if(i,j){
        tr=x.real[j]; /* поменять местами x[j] и x[i] */
        ti=x.imag[j];
        x.real[j]=x.real[i];
        x.imag[j]=x.imag[i];
        x.real[i]=tr;
        x.imag[i]=ti;
        k=N/2;
        while(k,j){
            j=j-k;
            k=k/2;
        }
    }
    else {
        k=N/2;
        while(k,j){
            j=j-k;
            k=k/2;
        }
    }
    j=j+k;
}
```

Существующие чипы ЦОС содержат команды для выполнения обращения битов при извлечении выходных выборок данных из памяти (подготовка к БПФ) или при записи выходных данных после БПФ в память.

12.5.4. Обработка при нескольких скоростях

Как обсуждалось в главе 9, обработка при нескольких скоростях включает выполнение операций ЦОС с более чем одной частотой дискретизации. Двумя фундаментальными операциями обработки при нескольких скоростях являются децимация (снижение

FF00	$h_1(11)$	$x_1(n)$	300h
	$h_1(10)$	$x_1(n-1)$	
	$\vdots$	$\vdots$	
	$h_1(1)$	$x_1(n-10)$	
	$h_1(0)$	$x_1(n-11)$	
FF20	$h_2(12)$	$x_2(n)$	320h
	$h_2(11)$	$x_2(n-1)$	
	$\vdots$	$\vdots$	
	$h_2(1)$	$x_2(n-11)$	
	$h_2(0)$	$x_2(n-12)$	
FF80	$h_3(47)$	$x_3(n)$	380h
	$h_3(46)$	$x_3(n-1)$	
	$\vdots$	$\vdots$	
	$h_3(1)$	$x_3(n-46)$	
	$h_3(0)$	$x_3(n-47)$	
	B0	B1	

Рис. 12.35. Параметры трехкаскадного дециматора

частоты дискретизации) и интерполяция (повышение частоты дискретизации). Проиллюстрируем реализацию дециматора реального времени на примере.

Пример 12.6

Частоту дискретизации сигнала требуется в ходе трехэтапного процесса децимации уменьшить с 30 до 1 кГц. Наибольшая частота, представляющая практический интерес после децимации, равна 400 Гц, неравномерность в полосе пропускания равна 0,08 дБ, затухание в полосе подавления равно 50 дБ. Дециматор нужно реализовать на базе процессора TMS320C50.

Решение

Используя программу разработки систем обработки с несколькими скоростями (см. [Ifeachor, 2001] и предисловие), получим параметры трехкаскадного дециматора (рис. 12.35). Коэффициенты трех фильтров, полученные с помощью программы разработки оптимального КИХ-фильтра, представлены в табл. 12.7. Согласно спецификациям длины фильтров взяты несколько больше оценок, предсказанных программой разработки дециматоров (12, 13 и 48 вместо 13, 12 и 46).

На рис. 9.15 представлена функциональная схема общего трехкаскадного дециматора. Схемы хранения коэффициентов и данных для дециматора на основе TMS320C50 показаны на рис. 12.36. Коэффициенты фильтра квантуются до 16 бит путем умножения каждого коэффициента на 2^{15} с последующим округлением результата до ближайшего целого. Приведенная программа децимации на языке ассемблера TMS320C50 является универсальной и ее можно модифицировать для выполнения одно-, двух- или трехкаскадной децимации путем замены коэффициентов, задающих длины фильтров, число каскадов децимации и коэффициенты децимации.

Таблица 12.7. Коэффициенты фильтров трехкаскадного дециматора

Длина фильтра = 12	
***** Импульсная характеристика *****	
$H(1) = 0,73075550E - 02 = H(12)$	239
$H(2) = 0,27123260E - 01 = H(11)$	889
$H(3) = 0,59286430E - 01 = H(10)$	1943
$H(4) = 0,10198970E + 00 = H(9)$	3342
$H(5) = 0,14187870E + 00 = H(8)$	4649
$H(6) = 0,16675770E + 00 = H(7)$	5464
***** Импульсная характеристика *****	
$H(1) = -0,86768190E - 02 = H(13)$	-284
$H(2) = -0,25476870E - 01 = H(12)$	-835
$H(3) = -0,25468170E - 01 = H(11)$	-834
$H(4) = 0,24184320E - 01 = H(10)$	792
$H(5) = 0,13238570E + 00 = H(9)$	4338
$H(6) = 0,24907950E + 00 = H(8)$	8162
$H(7) = 0,30075170E + 00 = H(7)$	9855
Длина фильтра = 48	
***** Импульсная характеристика *****	
$H(1) = 0,17780220E - 02 = H(48)$	585
$H(2) = -0,17396640E - 02 = H(47)$	-57
$H(3) = -0,49461790E - 02 = H(46)$	-162
$H(4) = -0,25451430E - 02 = H(45)$	-83
$H(5) = 0,40843330E - 02 = H(44)$	134
$H(6) = 0,42773070E - 02 = H(43)$	140
$H(7) = -0,45042640E - 02 = H(42)$	-148
$H(8) = -0,80385180E - 02 = H(41)$	-263
$H(9) = 0,29002500E - 02 = H(40)$	95
$H(10) = 0,12193670E - 01 = H(39)$	400
$H(11) = 0,92281120E - 03 = H(38)$	30
$H(12) = -0,16199860E - 01 = H(37)$	-531
$H(13) = -0,76966970E - 02 = H(36)$	-252
$H(14) = 0,18898710E - 01 = H(35)$	619
$H(15) = 0,17966280E - 01 = H(34)$	589
$H(16) = -0,18756490E - 01 = H(33)$	-615
$H(17) = -0,32451860E - 01 = H(32)$	-1063
$H(18) = 0,13458800E - 01 = H(31)$	441
$H(19) = 0,52945520E - 01 = H(30)$	1735
$H(20) = 0,17620600E - 02 = H(29)$	58
$H(21) = -0,86433440E - 01 = H(28)$	-2832
$H(22) = -0,44585360E - 01 = H(27)$	-1461
$H(23) = 0,18176500E + 00 = H(26)$	5956
$H(24) = 0,41039480E + 00 = H(25)$	13448

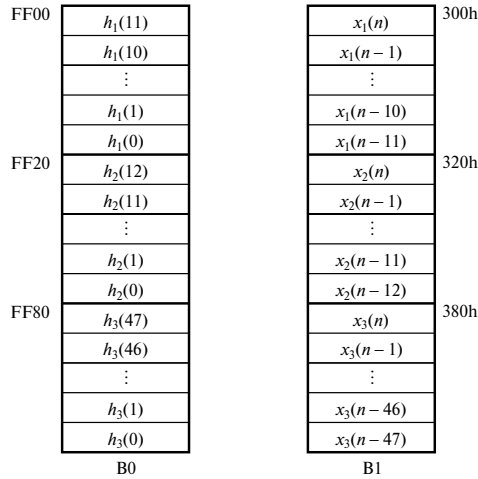


Рис. 12.36. Схемы хранения коэффициентов и данных трехкаскадного дециматора

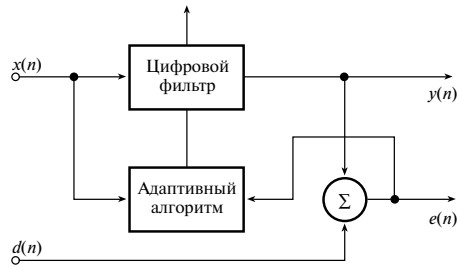


Рис. 12.37. Общая структура адаптивного фильтра: пара входов и пара выходов

12.5.5. Адаптивная фильтрация

Общая структура адаптивного фильтра изображена на рис. 12.37. Как обсуждалось в главе 10, адаптивная фильтрация включает два процесса.

1. *Цифровая фильтрация.* С помощью коэффициентов фильтра, изображенного на рис. 12.37, из входного сигнала $x(n)$ извлекается нужная информация, в результате чего получается $y(n)$.

Считая, что используется трансверсальная структура, фильтр можно записать следующим образом:

$$y(n) = \sum_{k=0}^{N-1} w_n(k)x(n-k), \quad (12.15)$$

где $w_k(k)$, $k = 0, 1, \dots, N-1$ — коэффициенты цифрового фильтра (часто именуемые *весовыми коэффициентами*), а $x(n-k)$, $k = 0, 1, \dots, N-1$ — последовательность входных данных.

Реализация цифрового фильтра в форме, представленной в уравнении (12.15), весьма схожа с рассмотренной ранее реализацией стандартного КИХ-фильтра. Так что реализация фильтра на языке С будет иметь знакомый вид:

```
y[n]=0;
for(k=0; k<N; k++){
y[n]=y[n]+wn[k]*xn[k];
}
```

2. *Адаптивный процесс.* Данный процесс включает обновление, т.е. настройку коэффициентов фильтра до достижения ими оптимальных значений. При использовании стандартного алгоритма наименьших квадратов коэффициенты обновляются следующим образом:

$$w_{n+1}(k) = w_n(k) + 2\mu e(n)x(n-k), \quad k = 0, 1, 2, \dots, N-1, \quad (12.16)$$

где $w_n(k)$ — k -й коэффициент цифрового фильтра в n -й момент выборки, μ — параметр устойчивости, $x(n-k)$ — k -я входная выборка в k -й линии задержки. Реализация стандартного обновления по схеме наименьших квадратов на языке С приведена в программе 12.11. Член $2\mu e[n]$ является скаляром, будучи одинаковым для всех коэффициентов, он вычисляется один раз и выносится за пределы цикла. Реализация адаптивного процесса на базе TMS320C50 представлена в программе 12.12.

Программа 12.11. Псевдокод С для обновления коэффициентов адаптивного фильтра по схеме наименьших квадратов

```
uen=2*u*e[n]
for(k=0; k<N; k++){
    wn[k]=wn[k]+uen*xn[k];
}
```

Программа 12.12. Псевдокод TMS320C50 для обновления коэффициентов адаптивного фильтра по схеме наименьших квадратов

	LT	ERR	
	MPY	U	; вычисляет $m * e(n)$
	PAC		
	ADD	ONE, 15	
	SACH	ERRF	
	LACC	#N-1	; задает длину фильтра
	SAMM	BRCR	
	LAR	AR2, #WNM1	; указывает на последний коэффициент, $w_k(N-1)$
	LAR	AR3, #XNM1	; указывает на $x(n-(N-1))$
	LT	ERRF	
	MPY	-, AR2	; вычисляет $\mu e(n) + x(n-k)$
	RPTB	LMS-1	; обновляет коэффициенты, $w_{k+1}(n)$
	ZALR	*, AR3	
	MPYA	*-, AR2	
	SACH	*+	
LMS	ZALR	*, AR3	
	APAC		
	SACH	*+	

12.6. Специализированная аппаратура ЦОС

Почему специализированная?

Операции цифровой обработки сигналов являются вычислительно интенсивными. В приложениях с широкополосными сигналами, где скорость поступления (выдачи) данных велика, большинство универсальных процессоров ЦОС не могут достаточно быстро выполнить нужные вычисления. Разумеется, по этой причине универсальные процессоры ЦОС часто используются в приложениях, где задействованы аудиочастоты. Кроме того, для данных приложений большинство универсальных процессоров ЦОС содержат много встроенных ресурсов, которые либо излишни, либо недостаточно эффективно используются, например, это относится к режимам адресации, наборам команд и периферийным устройствам ввода-вывода. В специализированных процессорах ЦОС аппаратное обеспечение оптимизировано для выполнения определенного алгоритма или некоторых функций специфического приложения. Это позволяет эффективнее использовать встроенные ресурсы и повышать скорость работы.

Специализированное аппаратное обеспечение можно реализовать как однокиповый продукт или как блок отдельных компонентов. Использование блочной конструкции гибче и позволяет повысить скорость, но внедрение подобных аппаратных средств сложнее и дороже. Однокиповые процессоры ЦОС (если они разработаны для выполнения данной задачи) имеют меньшее число интегральных схем, не требуют знания малопонятного языка ассемблера и не имеют проблем программной отладки.

Основные требования к специализированным процессорам ЦОС

Наиболее распространенной арифметической операцией в таких алгоритмах ЦОС, как цифровая фильтрация, корреляция и преобразования, является сумма произведений:

$$y = \sum_{k=0}^{N-1} a_k x_k, \quad (12.17)$$

где a_k — набор коэффициентов или переменных, x — последовательность данных.

Характерное уравнение (12.17) можно для более эффективного вычисления суммы произведений записать в рекурсивной форме:

$$y_k = a_k x_k + y_{k-1}, \quad k = 0, 1, \dots, N-1, \quad (12.18)$$

где

$$\begin{aligned} y_{-1} &= 0, \\ y &= y_{N-1}. \end{aligned}$$

В специализированных процессорах ЦОС уравнение (12.18) вычисляется с помощью умножителя-накопителя с очень большой скоростью, например, 40 нс на одну операцию умножения-накопления.

Подобно универсальным процессорам ЦОС архитектура специализированных процессоров включает память (ОЗУ и/или ПЗУ) для хранения данных, память (ОЗУ и/или ПЗУ) для хранения данных и переменных (таких как коэффициенты фильтра или БПФ),

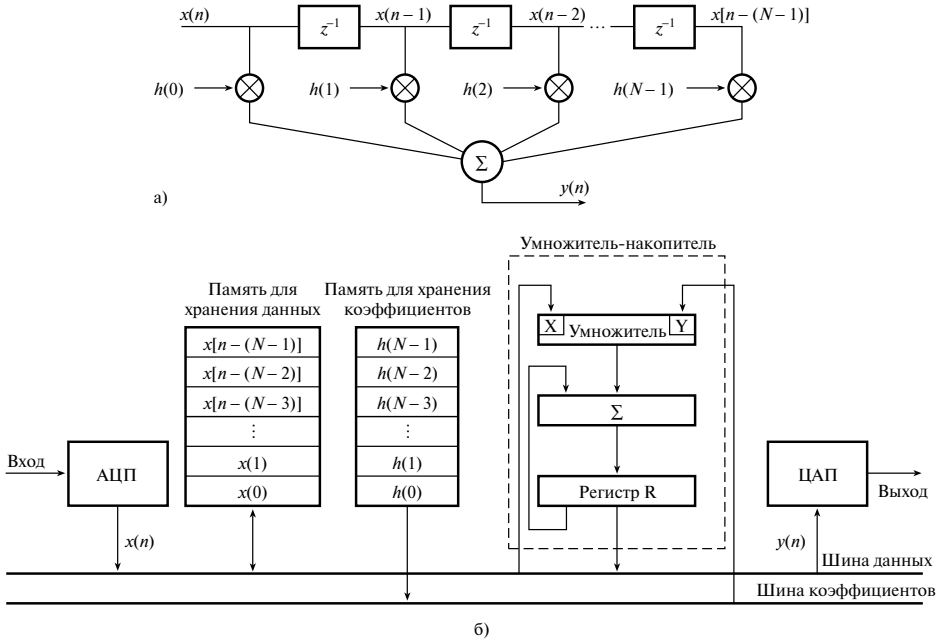


Рис. 12.38. Архитектура аппаратного цифрового КИХ-фильтра

быстрый аппаратный умножитель-накопитель и временные регистры для хранения данных или промежуточных результатов. Для достижения максимальной скорости интенсивно используются параллелизм, уплотнение и конвейерная обработка.

В следующих нескольких разделах рассмотрены некоторые базовые вопросы, касающиеся разработки специализированного аппаратного обеспечения для цифровой обработки сигналов.

12.6.1. Аппаратные цифровые фильтры

12.6.1.1. Цифровые КИХ-фильтры

КИХ-фильтр, реализованный в прямой форме, характеризуется следующим уравнением:

$$y(n) = \sum_{k=0}^{N-1} h(k)x(n-k).$$

На рис. 12.38 показана стандартная архитектура цифрового КИХ-фильтра с использованием блоков стандартных компонентов. Основные компоненты здесь — это память для хранения данных и память для хранения коэффициентов, блоки аналогового ввода-вывода (АЦП и ЦАП), умножитель-накопитель и контроллер (не показан). Для реализации всех компонентов КИХ-фильтра можно использовать быстрые готовые продукты.

В каждый момент взятия выборки новая выборка данных $x(n)$ считывается с АЦП и записывается в память для хранения данных. Каждая входная выборка данных и соответствующий коэффициент извлекаются из памяти одновременно и подаются на умножитель. Затем получающиеся произведения накапливаются и на их основе получается выходная выборка. Расчет каждой выходной выборки $y(n)$ потребует N операций извлечения данных (коэффициентов) из памяти и N операций умножения-накопления.

Операция КИХ-фильтрации является регулярной с хорошей структурой и ее легко реализовать в одной интегральной схеме. В настоящее время существуют такие специализированные однокристалльные КИХ-фильтры, как перепрограммируемый КИХ-фильтр PDSP16256 (Mitel).

12.6.1.2. Цифровой БИХ-фильтр

На рис. 12.39 показана архитектура канонического БИХ-фильтра второго порядка. В данном случае память для хранения данных содержит данные внутреннего узла $w(n)$. Стандартное звено второго порядка на рис. 12.39, *a* характеризуется следующими уравнениями:

$$\begin{aligned}w(n) &= s_1 x(n) - a_1 w(n-1) - a_2 w(n-2), \\y(n) &= b_0 w(n) + b_1 w(n-1) + b_2 w(n-2),\end{aligned}$$

где $x(n)$ представляет входные данные, а $w(n)$ — внутренний узел, $y(n)$ — выходная выборка фильтра, s_1 — масштабный коэффициент.

12.6.2. Аппаратные процессоры БПФ

ДПФ принимает набор из N выборок во временной области и преобразовывает их в набор из N выборок в частотной области $X(k)$. Для эффективного расчета коэффициентов ДПФ $X(k)$ используется БПФ. Основной операцией в БПФ является “бабочка”, которая характеризуется следующими уравнениями:

$$\begin{aligned}A' &= A + W_N^k B, \\B' &= A - W_N^k B,\end{aligned}$$

где A и B — пара комплексных выборок на входе “бабочки”, A' и B' — выходы “бабочки”, W_N — комплексный настроечный параметр.

Каждая “бабочка” требует комплексного умножения, т.е. $W_N^k B$, комплексного сложения и комплексного вычитания. На рис. 12.40 показана прямая аппаратная реализация процессора для вычисления “бабочки” с использованием отдельных блоков комплексных арифметических устройств: комплексного умножителя и пары комплексных накопителей. Комплексный умножитель вычисляет общие члены $W_N^k B$. В двух комплексных накопителях вычисляются два выхода “бабочки”, A' и B' .

Используя комплексный умножитель PDSP16112A (Mitel) и пары комплексных накопителей (PDSP16318A), можно реализовать одноканальный процессор для расчета “бабочки” со временем такта 50 нс. При использовании стандартных действительных

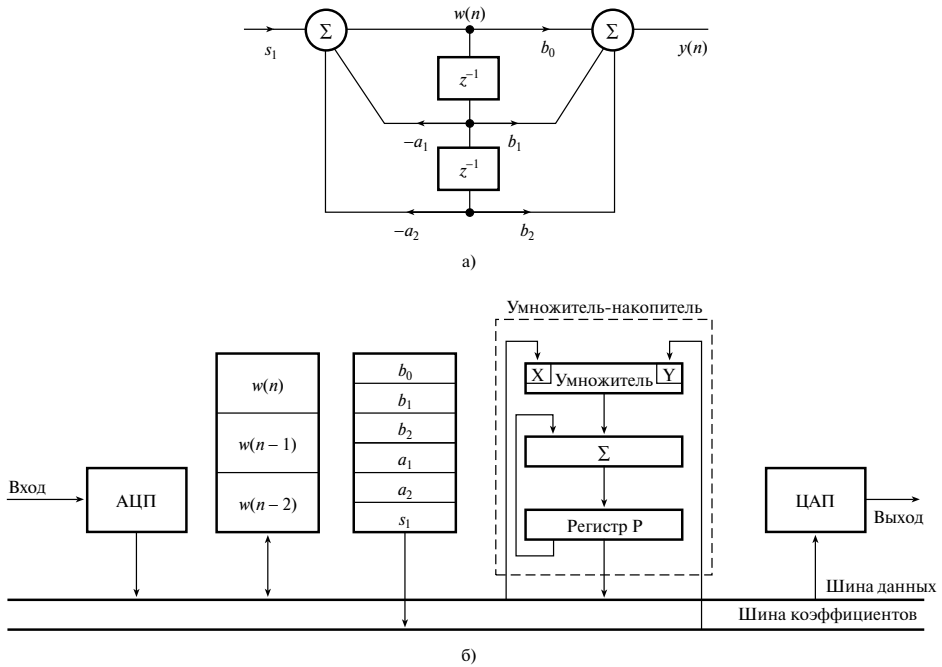


Рис. 12.39. БИХ-фильтр: а) структура; б) гарвардская архитектура биквадратного звена

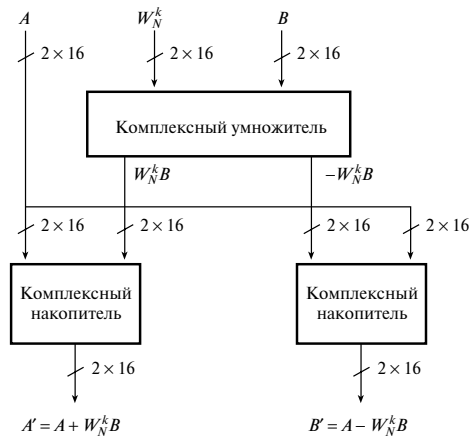


Рис. 12.40. Схема аппаратного процессора для вычисления "бабочки" с использованием отдельных блоков комплексной арифметики

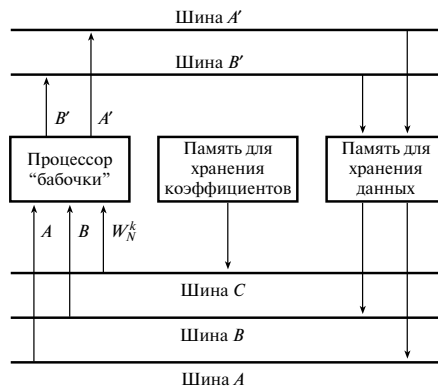


Рис. 12.41. Упрощенная архитектура аппаратного процессора БПФ. Контроллер и генератор адресов не показаны

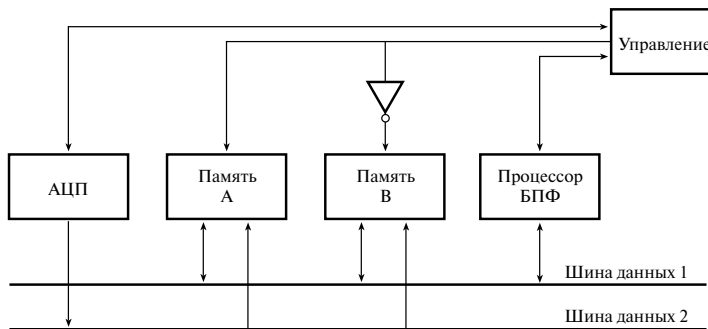


Рис. 12.42. Двойная буферизация в БПФ реального времени

арифметических устройств эквивалентный процессор будет состоять из четырех умножителей и шести сумматоров. Использование комплексных арифметических устройств, очевидно, позволяет обойтись меньшим числом интегральных схем и, возможно, повысить эффективность системы. На рис. 12.41 показан аппаратный процессор БПФ, построенный на базе описанного выше процессора для расчета “бабочки”. Кроме того, существуют и самостоятельные высокоскоростные процессоры БПФ, такие как PDSP16510 (Mitel).

На рис. 12.42 изображена конфигурация БПФ реального времени с двойным буфером. При расчете N -точечного БПФ попеременно используются оба буфера. Пока БПФ применяется к N -точечным данным из буфера А, буфер В заполняется новыми данными. Подобная двойная буферизация позволяет непрерывно выполнять БПФ в реальном времени без потери данных. Максимальное время завершения N -точечного БПФ равно промежутку $T_f = NT$ (с).

12.7. Резюме

Алгоритмы ЦОС подразумевают трудоемкие арифметические операции, в частности, умножение и сложение с интенсивным потоком данных через центральный процессор. Эффективное выполнение подобных алгоритмов в реальном времени требует аппаратной архитектуры и набора команд, радикально отличающихся от архитектуры и команд стандартных микропроцессоров. В цифровых процессорах сигналов для этого используются гарвардская архитектура, конвейерная обработка и специализированное аппаратное обеспечение, например, быстрые аппаратные умножители-накопители и схемы сдвига, скоростная внешняя память, также в процессоры внедряется множество команд, ориентированных на ЦОС.

Чтобы удовлетворять требованиям таких многоканальных, вычислительно-интенсивных приложений, как современные модемы удаленного доступа к серверу, системы мобильной связи третьего поколения, обработка мультимедийной информации, вводятся новые архитектуры. В частности, это архитектура *VLIW* (very long instruction word — с командными словами сверхбольшой длины) и *статическая суперскалярная архитектура*, которые используются в последнем поколении процессоров ЦОС. В обеих архитектурах применяются множественные тракты обработки данных и арифметические устройства, также для повышения эффективности используется параллелизм на уровне команд.

Существует два типа цифровых процессоров сигналов: универсальные процессоры (которые сродни стандартным микропроцессорам, за исключением архитектур и наборов команд, приспособленных для выполнения операций ЦОС) и специализированные процессоры. Последние используются для выполнения специфических алгоритмов ЦОС, например, цифровой КИХ-фильтрации (алгоритмические процессоры ЦОС), или для эффективного выполнения некоторых операций конкретных приложений (прикладные процессоры, или процессоры специального назначения). По сравнению с универсальными процессорами специализированные более быстрые, но менее гибкие.

В данной главе подробно рассмотрены основные идеи, реализованные в аппаратуре ЦОС, и влияние алгоритмов ЦОС на архитектуру процессоров ЦОС. Для иллюстрации существенных моментов разобрана реализация нескольких ключевых алгоритмов ЦОС с использованием универсальных цифровых и специализированных процессоров сигналов.

Задачи

12.1. Напишите краткое резюме по следующим концепциям, используя для иллюстрации ответа подходящие диаграммы:

- гарвардская архитектура;
- конвейерная обработка;
- умножитель-накопитель;

- специальные команды;
- память для хранения данных и программы.

Объясните, чем гарвардская архитектура, используемая в семействе TMS320, отличается от строгой гарвардской архитектуры. Сравните ее с архитектурой стандартного фоннеймановского процессора.

- 12.2. 1. В цифровом процессоре сигналов требуется умножитель-накопитель с трех-каскадным конвейером. Изобразите блок-схему умножителя-накопителя подходящей конфигурации. С помощью временной диаграммы объясните работу данного умножителя-накопителя.
2. Пусть время обращения к памяти равно 150 нс, время умножения — 100 нс, время сложения — 100 нс, служебные издержки на каждом каскаде конвейера — 5 нс. Определите пропускную способность умножителя-накопителя. Ответ прокомментируйте.

3. Требуется система ЦОС, в реальном времени выполняющая следующий алгоритм:

$$y(n) = a_0x(n) + a_1x(n-1) + a_2x(n-2) + \dots + a_{N-1}x[n - (N-1)].$$

Сколько времени затратит умножитель-накопитель на получение каждой выходной выборки?

- 12.3. В работе [3] высокоскоростные компьютеры разбиты на следующие четыре категории:
- а) с одним потоком команд и одним потоком данных (single instruction stream, single data stream — SISD);
 - б) с одним потоком команд и многими потоками данных (single instruction stream, multiple data stream — SIMD);
 - в) со многими потоками команд и одним потоком данных (multiple instruction stream, single data stream — MISD);
 - г) со многими потоками команд и многими потоками данных (multiple instruction stream, multiple data stream — MIMD),

где поток команд — это последовательность команд программы, выполняемой компьютером. а поток данных — это последовательность данных, необходимая компьютеру для выполнения команд.

Классифицируйте перечисленные ниже процессоры. Выбор аргументируйте.

- Motorola 68000;
- Motorola DSP56000;
- Analog Devices ADSP2100;
- Texas Instruments TMS320C50;
- Texas Instruments TMS320C30;
- Texas Instruments TMS320C40;
- Texas Instruments TMS320C62x;
- Analog Devices TS001.

- 12.4. 1. Объясните, почему такие традиционные меры, как тактовая частота процессора, MIPS и MFLOPS, могут не подходить для сравнения вычислительной эффективности процессоров ЦОС. Предложите альтернативный метод сравнения вычислительной эффективности.
2. Сформулируйте и обсудите ключевые факторы, отличные от скорости работы, которые следует учитывать при выборе процессора ЦОС для следующих приложений:
- а) высококачественное воспроизведение цифрового аудио;
 - б) передача речи посредством IP-телефонии;
 - в) обработка физиологических сигналов в сфере диагностики в биомедицине.
- 12.5. 1. Сравните вычислительную эффективность процессоров с фиксированной запытой TMS320C50, DSP56000 и ADSP2100 на основе выполнения внутреннего цикла N -точечного КИХ-фильтра. Укажите сделанные предположения.
2. Повторите п. 1 для БИХ-фильтра n -го порядка с M каноническими звеньями второго порядка, соединенными каскадно. Число N считайте четным.
3. Повторите п. 1 для адаптивного фильтра, работающего по схеме наименьших квадратов, который определен в уравнении (12.16).
- 12.6. Напишите краткое резюме по следующим концепциям, используя для иллюстрации ответа подходящие диаграммы:
- а) круговая адресация;
 - б) архитектура с одним потоком команд и многими потоками данных (single input, multiple data — SIMD);
 - в) суперскалярная архитектура;
 - г) архитектура с командными словами сверхбольшой длины (Very Long Instruction Word — VLIW);
 - д) организация циклов с нулевыми служебными издержками.
- В каждом случае четко укажите достоинства и недостатки метода применительно к цифровой обработке сигналов.
- 12.7. Используя представленную на рис. 12.43 диаграмму прохождения сигнала для 16-точечного БПФ с децимацией по времени, постройте эквивалентную диаграмму для БПФ с постоянной геометрией. Прокомментируйте относительные преимущества двух диаграмм.
- 12.8. На рис. 12.32 изображена схема прохождения сигнала по 8-точечному БПФ с децимацией во времени и скремблированным выходом. Покажите, что выход можно получить в естественном порядке, записав входные данные в порядке, определяемом обращенными битами, показав таким образом, что в БПФ с децимацией во времени окончательный выход будет правильно упорядочен, если последовательность входных данных перед БПФ скремблирована, либо в выходной последовательности после БПФ восстановлен естественный порядок данных.

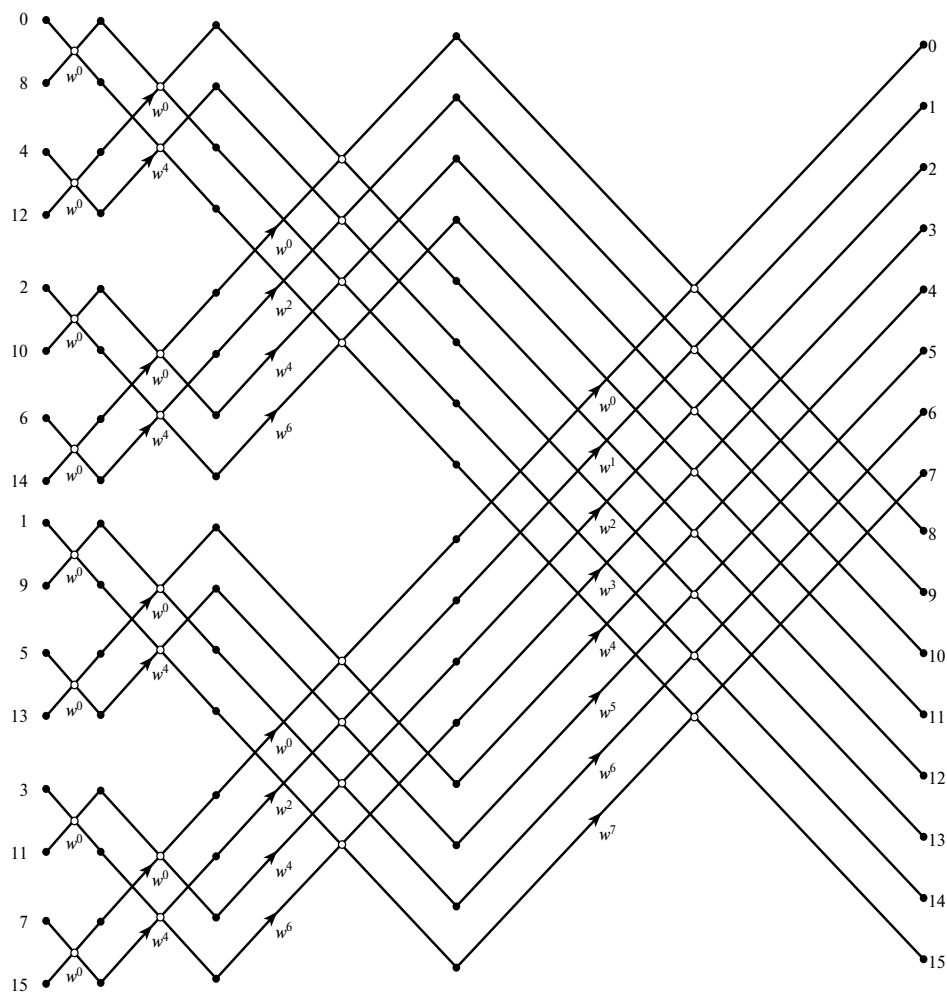


Рис. 12.43. Схема прохождения сигнала 16-точечного двоичного БПФ с децимацией по времени

- 12.9. В табл. 12.8 представлена последовательность входных данных и двоичное представление ее индексов. Определите входную последовательность, записанную в порядке, определяемом обращенными битами, и заполните таблицу.

Таблица 12.8. Последовательность входных данных задачи 12.9

<i>Входная последовательность, естественный порядок</i>	<i>Двоичный код последовательности</i>	<i>Входная последовательность, биты обращены</i>	<i>Двоичный код последовательности (с обращением битов)</i>
$x(0)$	0000		
$x(1)$	0001		
$x(3)$	0011		
$x(5)$	0101		
$x(6)$	0110		
$x(7)$	0111		
$x(8)$	1000		
$x(9)$	1001		
$x(10)$	1010		
$x(11)$	1011		
$x(12)$	1100		
$x(13)$	1101		
$x(14)$	1110		
$x(15)$	1111		

- 12.10. Разработайте эффективное специализированное аппаратное обеспечение, используя отдельные блоки арифметических элементов, для цифрового БИХ-фильтра N -го порядка реального времени, реализованного как каскад звеньев второго порядка. Разрешение АЦП/ЦАП равно 12 бит, длина коэффициентов — 16 бит. Требуемая частота дискретизации — 100 кГц. Укажите сделанные предположения.
- 12.11. 1. Напишите краткое резюме по следующим концепциям:
- а) динамический диапазон;
 - б) арифметика с фиксированной и плавающей запятой.
- Для иллюстрации ответа укажите определенные приложения в аудиотехнологиях, связи и биомедицине.
- 2, а. В системе на основе ЦОС используется цифровой процессор сигналов с фиксированной запятой и длиной слова 16 бит. Оцените динамический диапазон системы.
- 2, б. Повторите п. 2, а, если длина слова равна 24 бит.
3. Повторите п. 2, если для предотвращения переполнения умножитель-накопитель предоставляет один бит защиты.
4. Повторите п. 3, если используется восемь бит защиты.
- Оцените и сравните степени расширения динамического диапазона, которое происходит в рассмотренных случаях.

Литература

1. Berkeley Design Technology (1999) *Buyer's Guide to DSP Processors*. Fremont CA: Berkeley Design Technology Inc. Подробности см. на www.BDTI.com.
2. Blalock G. (1997) General-purpose μ Ps for DSP applications: consider the trade-offs. *EDN*, 23 October, pp. 165–172.
3. Flynn M. J. (1966) Very high-speed computing systems. *Proc. IEEE*, **54**(12), 1901–1909.
4. Hacker S. (1999) Static superscalar design: a new architecture for the TigerSHARC DSP processor. Analog Devices Whitepaper. www.analog.com/publications/whitepapers/products/sharc.html
5. Hayes J. P. (1998) *Computer Architecture and Organization*, 3rd edn. Boston MA: McGraw-Hill.
6. Hennessy J. L. and Patterson D.A. (1990) *Computer Architecture: A Quantitative Approach*. San Mateo CA: Morgan Kaufmann.
7. Levy M. (1998) DSP architecture directory. *EDN*, 23 April, pp. 40–110.
8. Levy M. (1999) DSP architecture directory. *EDN*, 15 April, pp. 67–102.
9. Rabiner L. R. and Gold B. (1975) *Theory and Applications of Digital Signal Processing*. Englewood Cliffs NJ: Prentice-Hall.
10. Texas Instruments (1988) *TMS320C2x Software Development System User's Guide*. Dallas TX: Texas Instruments.
11. Texas Instruments (1999) TMS320C6000 Technical Brief, Literature No. SPRU197D, Texas Instruments, Austin TX. Документ также можно найти на www.ti.com.

Дополнительная литература

- Casey P. E. and Simmers L. (1986) Digital signal processing IC helps to shed new light on image processing applications. *Electronic Design*, 20 March, 135.
- Chassaing R. and Horning D. W. (1990) *Digital Signal Processing with the TMS320C25*. New York: Wiley.
- Cragon H. (1980) The elements of single-chip microcomputer architecture. *Computing Mag.*, **13**(10), 27–41.
- Croisier A., Estaban D. J., Levilion M. E. and Rizo W. (1973) *Digital Filter for PCM Encoded Signal*. US Patent 3777130, 3 December, 1973.
- Dahnoun N. (2000) *Digital Signal Processing Implementation Using the TMS320C6000 DSP Platform*. Englewood Cliffs NJ: Prentice-Hall.
- De Roberts R. B. and Rifaat R. (1999) DSPs enhance flexible third-generation base-station design. *Wireless System Design*, November, www.wsdmag.com.
- DSP-architecture directory, 1997, *EDN*, 8 May, 43–107.
- DSP56300 Family Manual, Motorola, www.motorola.com/SPS/DSP/documentation
- DSP56000 Digital Signal Processor Family Manual, Motorola, Austin TX, 1995.
- Fine B. and McGuire G. Considerations for selecting a DSP processor P ADSP-2101 vs TMS320C50, AN-233 Analog Devices Application Note, Digital Signal Processing Products 9-77-9-92.
- Gallant J. (1990) Plug-in DSP boards. *EDN*, 35, 142–160.
- Ganesan S. (1991) A dual-DSP microprocessor system for real-time digital correlation. *Microprocessor and Microsystems*, **15**(7), 379–384.
- Gore A. E. (1986) *Cascadable digital signal processor*. New Electronics, 14 October, 39–40.
- Jouppi P. and Wall D. W. (1989) Available instruction-level parallelism for superscalar and superpipelined machines. In *Proc. Third Conf. on Architectural Support for Programming Languages and Operating Systems*. IEEE/ACM (April), Boston MA, pp. 272–282.
- Kloker K. L. (1986) The Motorola DSP56000 Digital Signal Processor, *IEEE Micro*, December, pp. 29–48.
- Kogge P. M. (1981) *The Architecture of Pipelined Computers*. New York: McGraw-Hill.

- Leary K. and Morgan D. (1986) Fast and accurate analysis with LPC gives a DSP chip speech-processing power. *Electronic Design*, 17 April, 153.
- Levy M. (1996) DSP-chip directory. *EDN*, March, 42–103.
- Levy M. (1997) C compilers for DSPs flex their muscles. *EDN*, 7 June, 93–107.
- Lin K. S. (ed.) (1988) *Digital Signal Processing Applications with the TMS320 Family*, Vol. 1. Englewood Cliffs NJ: Prentice-Hall.
- Lin K. S., Frantz G.A. and Simar R. (1987) The TMS320 family of digital signal processors. *Proc. IEEE*, **75**(9), 1143–1159.
- McKee D. (1990) TMS32010 routine finds phase. *EDN*, **35**, 148.
- Mennen P. (1991) DSP chips can produce random numbers using proven algorithm. *EDN*, 36, 141–146.
- Messer D. D. (1991) Convolutional encoding and viterbi decoding using the DSP56001. *Microprocessors and Microsystems*, **15**(1), 54–62.
- Motorola (1986) *DSP56000 Digital Signal Processor User's Manual*. Motorola.
- Nath N. S. M. (1999) C compilers and development tools simplify DSP assembly-language programming. *EDN*, 21 January, 103–110.
- Papamichalis P. (1990) *Digital Signal Processing Applications with the TMS320 Family. Theory, Algorithms, and Implementations*, Vol. 3. Dallas TX: Texas Instruments.
- Papamichalis P. and Simar R. (1988) The TMS320C30 floating-point digital signal processor. *IEEE Micro Mag.*, **8**(6), 10–28.
- Roesgen J. (1986) Fast modem designs benefit from DSP chip's versatility. *Electronic Design*, 12 June.
- Roesgen J. and Tung S. (1986) Moving memory off chip, DSP μ P squeezes in more computational power. *Electronic Design*, 20 February, 131.
- Rosen S. (1969) Electronic computers: a historical survey. *Computer Survey*, **1**(1), 7–36.
- Schmalzel J., Hein D. and Ahmed N. (1980) Some pedagogical considerations of digital filter hardware implementation. *IEEE Circuits and Systems Mag.*, **2**(1), 4–13.
- So J. (1983) TMS 320 P step forward in digital signal processing. *Microprocessors and Microsystems*, **7**(10), 451–460.
- Stokes J. and Sohie G. R. L. (1991) Implementation of PID controllers on the Motorola DSP56000/DSP56001. Part 1. *Microprocessors and Microsystems*, **15**(6), 321–331.
- Stokes J. and Sohie G. R. L. (1991) Implementation of PID controllers on the Motorola DSP56000/DSP56001. Part 2. *Microprocessors and Microsystems*, **15**(7), 385–392.
- Texas Instruments (1989) *Second-Generation TMS320 User's Guide*. Dallas, TX: Texas Instruments.
- TMS320C5x User's Guide, Texas Instruments, 1995.
- TMS320C54x DSP Reference Set, Volume 4: Applications Guide. Texas Instruments, October 1996. www.ti.com
- Tomarakos J. and Ledger D. (1998) Using the Low-cost, High Performance ADSP-21065L Digital Signal Processor for Digital Audio Applications. Analog Devices DSP Application. Подробности можно найти на www.analog.com.
- Zolzer U. (1997) *Digital Audio Signal Processing*. Wiley.

Другие полезные Web-адреса

Специализированное аппаратное обеспечение, например, многоканальный эхоподавитель MT9300, программируемые КИХ-фильтры PDSP16256. Mitel Semiconductor, www.mitelsemi.com.

Аудиокодеки и процессоры, например, 24-битовый кодек 96 кГц объемного звучания CS4228, цифровые конвертеры частоты дискретизации аудиосигнала. Cirrus, www.cirrus.com.

Приложение

12.A. Программы на языке ассемблера TMS320 для обработки сигналов в реальном времени и программа на языке C для двоичного БПФ с постоянной геометрией

На компакт-диске к книге [Ifeachor, 2001] имеются следующие программы для процессоров TMS320C10/C25 (подробности см. в предисловии):

- 1) реализация цифрового узкополосного режекторного КИХ-фильтра на основе TMS320C10;
- 2) реализация цифрового полосового КИХ-фильтра на основе TMS320C10;
- 3) реализация цифрового узкополосного режекторного КИХ-фильтра на основе TMS320C25;
- 4) реализация с помощью каскадного соединения звеньев второго порядка цифрового БИХ-фильтра четвертого порядка на основе TMS320C10;
- 5) реализация с помощью каскадного соединения звеньев второго порядка цифрового БИХ-фильтра четвертого порядка на основе TMS320C25;
- 6) реализация с помощью параллельного соединения звеньев второго порядка цифрового БИХ-фильтра четвертого порядка TMS320C25;
- 7) программа на языке C для двоичного БПФ с постоянной геометрией;
- 8) двоичный алгоритм БПФ на основе TMS320C25;
- 9) адаптивный фильтр на основе TMS320C25.

Из-за недостатка места в данную книгу включены только программы 2 и 5.

Программа 12A.1. Реализация цифрового полосового КИХ-фильтра на основе TMS320C10

METAi Assembler 4.00 ©1988 Crash Barrier Thu Nov 19 00:37:40 1992

Page 1 Assembler

```

                                targbpf.asm
00000000      1      c:\metai\32010.tab/
                                2
00000000      3      .ctrI      27, 15
00000000      4      SEGMENT    word at 0000 'ram'
00000000      5      ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
00000000      6      ;
00000000      7      ; Полосовой КИХ-фильтр
00000000      8      ;
00000000      9      ; Спецификация фильтра:

```

```

00000000      10 ; ;
00000000      11 ; Тип фильтра полосовой фильтр ;
00000000      12 ; частота дискретизации 15 кГц ;
00000000      13 ; полоса пропускания 900--1100 Гц ;
00000000      14 ; ширина полосы перехода 450 Гц ;
00000000      15 ; неравномерность в полосе пропускания <0.87 дБ ;
00000000      16 ; затухание в полосе подавления >30 дБ ;
00000000      17 ; длина фильтра 41 ;
00000000      18 ; ;
00000000      19 ; Аппаратура: целевая плата TMS320C10 ;
00000000      20 ; с 8-битовыми АЦП/ЦАП ;
00000000      21 ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
00000000      22 ;
00000000      23
00000000 F900002B      24 B START
00000000      25
00000028      26 NM1 EQU 40 ;N-1
00000000      27 XN EQU 0 ;текущая входная выборка
00000028      28 XNM1 EQU NM1
00000029      29 H0 EQU NM1+1
00000051      30 HNM1 EQU H0+NM1
0000007B      31 YN EQU 123
0000007C      32 ONE EQU 124
00000000      33 PA0 EQU 0 ;адрес ввода-вывода для прямого
00000001      34 PA1 EQU 1 ;доступа к памяти целевой платы
00000002      35 PA2 EQU 2
00000003      36 PA3 EQU 3
00000002      37 COEFF EQU 2 ;начальный адрес коэффициентов.
00000000      38 R0 EQU 0
00000001      39 R1 EQU 1
00000002      40 ;
00000002      41 ;таблица коэффициентов, которые изначально
00000002      42 ;хранятся в программной памяти.
00000000      43
00000000      44
00000002 FE09FFFEFF5B019F 45
00000002 02B3038E03D9 45 DC.W -503,-2,-165,415,691,910,985
00000009 035001 D9FF97FCE8 46
00000009 FA57F881 46 DC.W 848,473,-105,-792,-1449,-1919
0000000F F7EAF8DDFB52FEE8 47
0000000F 02F406A8 47 DC.W -2070,-1827,-1198,-280,756,1704
00000015 093F0A2E093F06A8 48
00000015 02F4FEE8 48 DC.W 2367,2606,2367,1704,756,-280
0000001B FB52F8DDF7EAF881 49
0000001B FA57 49 DC.W -1198,-1827,-2070,-1919,-1449
00000020 FCE8FF9701D90350 50
00000020 03D9038E02B3 50 DC.W -792,-105,473,848,985,910,691
00000027 019F00A5FFFEFE09 51 DC.W 415,165,-2,-503

```


METAi Assembler 4.00 © 1988 Crash Barrier Thu Nov 19

00:37:40 1992 Page 2 Assembler

targbpf.asm

```

52
53
0000002B 54 ;===Начало основной программы ===
0000002B 55 ;
0000002B 56 ; инициализация
0000002B 57 ;
0000002B 7E01 58 START LACK 1
0000002C 507C 59 SACL ONE
0000002D 6E00 60 LDPK 0 ;указатель на нулевую страницу
0000002E 61 ;памяти данных
0000002E 62 ;передача коэффицентов из программной памяти
;в ПСПЗУ в память для хранения данных
0000002E 63 ;
0000002E 7E02 64 LACK COEFF ;загрузка адресов коэффицентов
;в накопитель
0000002F 7028 65 LARK AR0,NM1 ;числа коэффицентов
;во вспомогательный регистр 0
00000030 7129 66 LARK AR1,H0 ;и адреса в памяти для хранения данных
;в регистр AR1
00000031 6881 67 LOAD LARP R1 ;выбрать AR1 и начать передавать
;коэффиценты
00000032 67A0 68 TBLR *+,R0 ;в память для хранения данных, затем
;увеличить содержимое регистра
;AR1 на 1
00000033 007C 69 ADD ONE ;увеличить значение в накопителе
00000034 F4000031 70 BANZ LOAD ;уменьшить значение в AR0 на 1
00000036 71 ;и проверить, не равно ли оно 0
00000036 72 ;дождаться новой входной выборки
00000036 73
00000036 6880 74 LARP R0
00000037 F600003B 75 WAIT BIOZ NXTPT ;проверить, готова ли выборка,
00000039 F9000037 76 B WAIT ;если нет - подождать
77
0000003B 4000 78 NXTPT IN XN,PA0 ;если готова - считать выборку
0000003C 79 ;
0000003C 80 ;вычислить выход фильтра в регистре YN
0000003C 81 ;и передать его ЦАП
0000003C 7028 82 skip LARK AR0,XNM1
0000003D 7151 83 LARK AR1,NNM1
0000003E 7F89 84 ZAC
85
0000003F 6A91 86 LT *-,R1 ;загрузить XN(N-1) выборку
00000040 6D90 87 MPY *-,R0 ;вычислить H(N-1)*XN(N-1)
00000041 6B81 88 LOOP LTD *,R1 ;вычислить SIG[H(K)*X(N-K)]
00000042 6D90 89 MPY *-,R0
00000043 F4000041 90 BANZ LOOP
00000045 7F8F 91 APAC ;прибавить H(N-1)*X(N-1)
00000046 597B 92 SACH YN,1 ;подать выборку на выход

```

```

00000047 487B 93          OUT    YN,PA0    ;
          94
00000048 F6000048 95  onhi  BIOZ    onhi  ;оставаться на этом этапе,
          ;пока строка BIO не станет достаточно большой
          96
0000004A F9000037 97  B      WAIT
          98
0000004C          99  end
No errors on assembly of 'targbpf.asm'

```

Программа 12A.2. Реализация на основе TMS320C25 цифрового БИХ-фильтра четвертого порядка с помощью каскадного соединения звеньев второго порядка

```

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
; Эллиптический фильтр четвертого порядка, представленный как      ;
; каскад двух биквадратных канонических звеньев                    ;
; Manny Ifeachor, Jan., 1992                                         ;
;                                                                      ;
; Спецификации фильтра:                                             ;
;                                                                      ;
; Тип фильтра:    фильтр нижних частот                             ;
; Частота дискретизации:      15 кГц                                 ;
; Полоса пропускания:        0-3 кГц                                ;
; Ширина полосы перехода:    450 кГц                                ;
; Неравномерность в полосе пропускания: 0,5 дБ                      ;
; Затухание в полосе подавления: 45 дБ                             ;
;                                                                      ;
; Аппаратура: TMS320C25 с                                           ;
; 1-битовым АЦП/ЦАП (фильтр В)                                     ;
;-----;
;
XN      .set      0
YN1     .set      1
W1N     .set      2
W1NM1   .set      3
W1NM2   .set      4
YN2     .set      5
W2N     .set      6
W2NM1   .set      7
W2NM2   .set      8
SF1     .set      9
A01     .set     10
A11     .set     11
A21     .set     12
B11     .set     13
B21     .set     14
A02     .set     15
A12     .set     16
A22     .set     17
B12     .set     18
B22     .set     19

```

```

SF2      .set      20
ONE      .set      21
RATED    .set      22
MODED    .set      23
WONE     .set      24
TEMP     .set      25
PBM1     .set      0300 h
;
      .sect      TIRUPTSU
START B   INIT
;
      .text
COEFFS   .word      13217      ; SF1 = 0.4033627 в формате Q15
      ; (SCALE FACTOR)
;
      .word      4296      ; a01=0.1311136
      .word      7087      ; a11=0.2162924
      .word      4296      ; a21=0.1311136
      .word      27175     ; -b11=0.829328
      .word      -10061    ; -b21=-0.307046
;
      .word      32767     ; a02=0.9999695
      .word      22142     ; a12=0.675718
      .word      32767     ; a22=0.9999695
      .word      16251     ; -b12=0.495935
      .word      -24965    ; -b22=-0.761864
;
      .word      29769     ; SF2=0.90847
;
MODEP    .word      0Ah
RATEP    .word      0299h
;
*
**      Инициализация AIB **
*
INIT     LDPK      6
      SSXM
      LACK      MODEP
      TBLR      MODED
      OUT       MODED, PA0
      LACK      RATEP      ; устанавливается частота дискретизации
      TBLR      RATED
      OUT       RATED, PA1
      OUT       RATED, PA3
*
**      Коэффициенты передаются из памяти программы в память данных
*
      LARP      AR0
      LRLK      AR0, PMB1+SF1
      RPTK      11
      BLKP      COEFFS, *+

```

```

*
**      Инициализируется прямой доступ к памяти для данных внутреннего узла
*
INITWN      ZAC
      SACL      W1N
      SACL      W1NM1
      SACL      W1NM2
      SACL      W2N
      SACL      W2NM1
      SACL      W2NM2
*
**      Ожидать новой выборки данных
*
RDATA      BIOZ      NXTPT      ;Извлечь новую выборку данных
      B      RDATA
NXTPT      IN      XN, PA2
      LT      XN
*
**      Начало блока фильтра 1
*
BLOCK1      MPY      SF1      ;Масштабируется входная выборка данных: SF*X(N)
      PAC
      LT      W1NM1      ;В регистр Т загружается W(N-1)
      MPY      B11      ;B11W1(N-1)
      LTA      W1NM2      ;SFX(N)+B11W1(N-1)
      MPY      B21      ;B21W1(N-2)
      APAC      ;SFX(N)+B11W1(N-1)+B21W1(N-2)
      SACH      W1N
      ZAC
      MPY      A21      ;A21W1(N-2)
      LTD      W1NM1      ;SUM = A21W(N-2); W1(N-2)=W1(N-1)
      MPY      A11      ;A11W1(N-1)
      LTD      W1N      ;SUM=A21W1(N-2) + A11W1(N-1); W1(N-1)=W1(N)
      MPY      A01      ;A01W1(N)
*
**      Начало блока фильтра 2
*
BLOCK2      LTA      W2NM1      ;Y1(N)=A21W(N-2)+A11W1(N-1)+A01W1(N)
      MPY      B12      ;B12W2(N-1)
      LTA      W2NM2      ;Y1(N)+B12W2(N-1)
      MPY      B22      ;B22*W2(N-2)
      APAC      ;Y1(N)+B12*W2(N-1)+B22*W2(N-2)
;      ;SUM=Y1(N)+B12*W2(N-1)+2*B22*W2(N-2)
      SACH      W2N      ;STORE HIGH 16 BIT WORD OF SUM IN W2(N)
      MPY      A22      ;A22*W2(N-2)
      ZAC      ;SUM=SUM+A22*W2(N-2)
      LTD      W2NM1      ;W2(N-2)=W2(N-1); SUM = A22*W2(N-2)
      MPY      A12      ;A12*W2(N-1)
      APAC      ;SUM=A22*W2(N-2)+A12*W2(N-1)
      LTD      W2N      ;W2(N-1)=W2(N)
      MPY      A02      ;A02*W2(N)

```

```

APAC
SACH      YN2      ;
*
**      Выходная выборка масштабируется и передается ЦАП
*
LT      YN2      ;
MPY      SF2      ;SF2*YN2
PAC      ;
APAC
APAC
APAC
APAC
APAC
APAC
APAC
APAC
APAC
APAC
APAC
APAC
APAC
APAC
APAC
APAC
APAC
APAC
APAC
SACH      YN2      ;21*SF2*YN2
;
OUT      YN2,PA2
B      RDATA
;
END
;@\\000300AB1800001100AB18;
/* Командный файл связей*/
MEMORY
{
    /* Память программы*/
    PAGE 0:      VECTORS:origin=0h, length=01Fh
                CODE:origin=20h, length=0F90h
    /* Память для хранения данных*/
    PAGE 1:      RAMB2:origin=60h, length=020h
                RAMB0:origin=200h, length=0FFh
                RAMB1:origin=300h, length=0FFh
}
SECTIONS
{
    IRUPTS      :{} . VECTORS      PAGE 0
    .text       :{} . CODE      PAGE 0
    .data       :{} . RAMB2      PAGE 1
    .bss        :{} . RAMB0      PAGE 1
}

```

Анализ эффектов конечной разрядности в системах ЦОС с фиксированной запятой

13.1. Вступление	867
13.2. Арифметика ЦОС	868
13.3. Шум квантования АЦП и качество сигнала	877
13.4. Эффекты конечной разрядности в цифровых БИХ-фильтрах	880
13.5. Эффекты конечной разрядности слова в алгоритмах БПФ	923
13.6. Резюме	927
Задачи	928
Литература	930
Дополнительная литература	931
Приложения	932

Цель данной главы — изучить ошибки, которые возникают в практических системах ЦОС в результате квантования и использования для операций ЦОС арифметических устройств со словами конечной длины. Обсуждается влияние данных ошибок на качество сигнала и методы борьбы с этими ошибками. Понимание вопросов, освещенных в данной главе, должно помочь при разработке систем ЦОС с предсказуемой производительностью.

Поскольку системы ЦОС с фиксированной запятой более распространены, в данной главе, в основном, рассматривается разработка именно таких систем.

13.1. Вступление

В большинстве случаев системы ЦОС разрабатываются для реализации функции ЦОС, например фильтрации или БПФ, на цифровом процессоре. На практике для представления точных переменных и выполнения арифметических операций используется конечное число битов. Типичными длинами слов в современных процессорах ЦОС является 16 бит (например, TMS320CC54), 24 бит (например, DSP56300) и 32 бит (например, ADSP-21065). Поскольку слова конечной длины вызывают ошибки, влияющие на производительность систем ЦОС, перед реализацией функции ЦОС разработчик

должен определить степень такого ухудшения и при необходимости найти эффективное “лекарство”.

Ниже приводятся основные ошибки в ЦОС.

1. Ошибка квантования АЦП, вызванная представлением входных данных ограниченным числом битов.
2. Ошибка квантования коэффициентов, которая возникает при представлении коэффициентов или параметров ЦОС конечным числом битов. Коэффициенты a_k и b_k , которые используются на этапе 2 разработки фильтра, например, обычно считаются с очень большой точностью, но в процессоре ЦОС они должны квантоваться (обычно до длины слова процессора).
3. Ошибки переполнения, к которым приводит сложение двух больших чисел одного знака, если в результате получается число, превышающее разрешенную длину слова.
4. Ошибки округления, которые возникают, когда результат умножения округляется (или усекается) до ближайшего дискретного значения или приемлемой длины слова.

Влияние ошибок, возникающих при обработке сигнала, зависит от многих факторов, в том числе типа используемой арифметики, качества входного сигнала и типа реализованной функции или алгоритма ЦОС. Ниже для изучения влияния конечной разрядности на производительность системы ЦОС используются цифровые БИХ-фильтры, поскольку в них проявляются большинство проблем, которые могут встретиться на практике.

13.2. Арифметика ЦОС

В предыдущих главах было показано, что основными операциями в ЦОС являются умножение, сложение и задержка (сдвиг). Например, при цифровой КИХ-фильтрации коэффициенты $h(k)$, ($k = 0, 1, \dots, N - 1$) и входные выборки $x(n)$ ($n = 0, 1, \dots$) умножаются, а полученные произведения суммируются:

$$y(n) = \sum_{k=0}^{N-1} h(k)x(n-k) = h(0)x(n) + h(1)x(n-1) + \dots + h(N-1)x[n-(N-1)]. \quad (13.1)$$

На практике арифметические операции, входящие в ЦОС (подобные операции, приведенной выше), часто выполняются с использованием арифметики с плавающей или фиксированной запятой (см. [16]). Иногда используются арифметики и других типов, например, блочная арифметика с плавающей запятой, которая в чем-то объединяет преимущества двух арифметик, названных выше. В ЦОС более распространена арифметика с фиксированной запятой, так как ей сопутствует быстрая и недорогая реализация, хотя диапазон представляемых чисел ограничен, а итоговая система подвержена проблемам переполнения, которое может произойти, когда результат сложения не входит в диапазон приемлемых чисел (например, масштабные предельные циклы в БИХ-фильтрах

и перегрузка в БПФ высоких порядков). Чтобы предотвратить выход результата арифметических операций из диапазона приемлемых значений, операнды масштабируются. Подобное масштабирование снижает производительность систем ЦОС, т.е. уменьшает достижимое отношение сигнал-шум.

Арифметика с плавающей запятой предпочтительнее, если амплитуды переменных или коэффициентов системы могут значительно изменяться [6]. Она допускает использование значительно более широкого динамического диапазона и практически снимает проблему переполнения. Более того, обработка с плавающей запятой упрощает программирование. Алгоритмы ЦОС, разработанные на больших машинах, например, на персональных компьютерах или мэйнфреймах, на языке высокого уровня, могут реализовываться прямо на аппаратуре ЦОС с незначительными изменениями основных алгоритмов. В то же время арифметика с плавающей запятой — это более дорогая и часто медленная реализация, хотя в последние годы широкое распространение начинают получать высокоскоростные цифровые процессоры сигналов со встроенным процессором с плавающей запятой (такие как Texas Instruments TMS320C30). Кроме того, в литературе (например, [18]) описаны эффективные программные процедуры, рассчитанные на действия с плавающей запятой. Отметим также, что за последнее время разница в цене и скорости между процессорами с плавающей и фиксированной запятой значительно сократилась.

Методы ЦОС все активнее используются в приложениях, где требуются и широкий динамический диапазон, и высокая точность. Цифровые процессоры сигналов с фиксированной запятой, с длинными словами (24 бит и больше) могут удовлетворять таким требованиям, но гораздо естественнее и проще в таких случаях использовать обработку с плавающей запятой.

Областей применения, в которых требуются широкий динамический диапазон и высокая точность, и где использование арифметики с плавающей запятой необходимо или желательно, множество. В качестве примера можно привести параметрическое выравнивание в реальном времени цифровых аудиосигналов с использованием цифровых фильтров, где значения коэффициентов фильтра меняются при прохождении через полосу аудиочастот и подгонке параметров эквалайзера. В определенных задачах обработки сигналов (например, анализ спектра в радаре или сонаре, сейсмология или биомедицина) часто требуется разрешать очень низкоуровневые компоненты сигнала с широким динамическим диапазоном. В подобных случаях требуется широкий динамический диапазон и высокая точность. В число аналогичных областей применения входят обработка графическими станциями с высоким разрешением и универсальные инженерные вычисления. Необходимый максимальный динамический диапазон и требования к точности в нескольких приложениях сведены в табл. 13.1 [20].

Ключевой вывод из сказанного можно сформулировать следующим образом: тип используемой арифметики — это основной фактор, влияющий на производительность систем ЦОС. Основные концепции арифметик двух типов (т.е. с плавающей и фиксированной запятой) рассмотрены ниже.

Таблица 13.1. Динамический диапазон и требования к точности

	Динамический диапазон (бит)	Точность (бит)
Подавление шума	32	20
Обработка сигналов радаров	32	20
Обработка изображений со студийным качеством	20	20
Обработка изображений	30	20
Медицинский спектральный анализ	20	20
Обработка сейсмических данных	70	20

13.2.1. Арифметика с фиксированной запятой

13.2.1.1. Представление в форме с фиксированной запятой

В ЦОС переменные часто представляются как дробные числа в форме дополнения до двух с фиксированной запятой; см., например, табл. 13.2. В таком представлении двоичная запятая находится справа от самого старшего бита, который одновременно является и битом знака. Каждое число принадлежит диапазону от -1 до $1 - 2^{-(B-1)}$, где B — число битов, используемых для представления числа. В ЦОС обычным является представление в так называемом формате Q15, где используется 16 бит (1 бит знака и 15 бит дробной части):

0110 0000 0000 0000
| двоичная запятая

Положительные числа в форме дополнения до двух не отличаются от обычного двоичного представления (см. табл. 13.2). Отрицательное число получается из положительного путем дополнения до двух всех его битов с последующим прибавлением 1 младшего разряда. Например, представление числа $-3/8$ в форме дополнения до двух получается из $3/8$ (т.е. 0011) так: $1100 + 0001 = 1101$.

Когда сигнал на вход системы ЦОС поступает с АЦП (аналога-цифрового преобразователя), данные, поступающие на цифровой процессор, могут представляться в двоичном формате со сдвигом. Подобным образом выход системы ЦОС может понадобиться преобразовать в такой формат, если он должен подаваться на ЦАП (цифроаналоговый преобразователь). Чтобы преобразовать двоичный формат со сдвигом в формат дополнения до двух, к числу в двоичном формате со сдвигом следует добавить старший бит. Например, в табл. 13.2 двоичный код со сдвигом 1111 (т.е. $7/8$) легко превращается в код дополнения до двух (0111) путем введения дополнительного старшего бита. На практике шины процессоров ЦОС часто шире, чем разрешение АЦП. В таком случае после преобразования в формат дополнения до двух оставшиеся слева позиции заполняет бит знака. Например, код 111 1101 в формате дополнения до двух выглядит как 1111 1111 1111 1101.

Если в представлении с дополнением до двух с фиксированной запятой каждое число представить B бит, то максимум можно представить 2^B различных чисел, причем соседние числа будут разнесены приблизительно на 2^{-B} . В связи с этим полезно определить точность, с которой можно представить любое число, по сравнению с десятичным представлением.

Таблица 13.2. Сравнение систем дополнения до двух и двоичной со сдвигом при длине слова 4 бит

<i>Число</i>	<i>Десятичная дробь</i>	<i>Дополнение до двух</i>	<i>Двоичная со сдвигом</i>
7	7/8	0111	1111
6	6/8	0110	1110
5	5/8	0101	1101
4	4/8	0100	1100
3	3/8	0011	1011
2	2/8	0010	1010
1	1/8	0001	1001
0	0	0000	1000
-1	-1/8	1111	0111
-2	-2/8	1110	0110
-3	-3/8	1101	0101
-4	-4/8	1100	0100
5	-5/8	1011	0011
-6	-6/8	1010	0010
-7	-7/8	1001	0001
-8	-1	1000	0000

Таблица 13.3. Связь между числом битов и точностью в десятичных знаках

<i>Число битов</i>	<i>Точность (число десятичных цифр)</i>
7	2,1
8	2,4
10	3
12	3,6
14	4,2
15	4,5
16	4,8
18	5,4
20	6,1
23	7,0
24	7,3
64	19,4

Для данной десятичной дроби X , состоящей из d цифр, точность равна $\pm 0,5 \times 10^{-d}$. Если представить то же число с помощью B бит, точность будет равна $\pm 0,5 \times 2^{-B}$. Чтобы сохранить ту же точность представления, необходимо

$$0,5 \times 10^{-d} = 0,5 \times 2^{-B}, \text{ т.е. } B = d \log_2 10 \simeq 3,3d \text{ бит.} \quad (13.2)$$

Например, предположим, что десятичное число 0,23456 нужно представить в двоичной форме. Тогда для представления с прежней точностью потребуется $3,3 \times 5 = 17$ бит. В табл. 13.3 сведены основные соотношения между числом битов двоичной системы и их точностью в десятичных знаках.

Пример 13.1

Представьте десятичное число 0,95624 следующим образом.

1. Как число в формате Q3.
2. Как число в формате Q4.

Сравните ошибки в двух случаях.

3. Оцените число битов, требуемых для представления указанного десятичного числа с одинаковой точностью.

Решение

1. Число в формате Q3 — это число в форме дополнения до двух с одним бит знака и тремя бит дробной части. Чтобы преобразовать десятичное число в формат Q3, его нужно просто умножить на 2^3 , а полученное произведение округлить до ближайшего приемлемого числа: $0,95624 \times 2^3 = 7,64992$, что округляется до $7 = 0111$ (наибольшее приемлемое число).
2. В данном случае число представляется с использованием одного бит знака и четырех бит дробной части: $0,95624 \times 2^4 = 15,29984$, что округляется до $15 = 01111$.

Ошибки в представлении чисел в п. 1 и 2 равны соответственно $0,64992/8 = 0,08124$ и $0,29984/16 = 0,01874$. Ошибка в представлении числа часто называется *ошибкой квантования коэффициентов*.

3. Из уравнения (13.2) находим, что требуется $3,3 \times 5 = 16,5$ бит $\simeq 17$ бит.
-

13.2.1.2. Умножение с фиксированной запятой

При умножении с фиксированной запятой используется тот факт, что произведение двух дробных чисел также является дробным числом, а произведение двух целых чисел — это целое число. Проиллюстрируем сказанное на примере.

Пример 13.2

Возведите $0,5625$ в квадрат, используя арифметику с дополнением до двух с фиксированной запятой. Предполагается формат Q4.

Решение

$$\begin{array}{r}
 0 \quad 1001 = 0,5625 \\
 0 \quad 1001 = 0,5625 \\
 \hline
 00000 \\
 0100 \quad 1 \\
 000 \quad 00 \\
 00 \quad 000 \\
 0 \quad 1001 \\
 \hline
 000101 \quad 0001 \\
 \begin{array}{cc}
 \uparrow & \nwarrow \\
 \text{двоичная запятая} & \text{при квантовании произведения бит теряется}
 \end{array}
 \end{array}$$

После сдвига влево (для удаления дополнительного бита) и округления получаем окончательный ответ: $0 \ 0101 = 0,25 + 2^{-4} = 0,3125$ (точный ответ — $0,31640625$).

Из примера 13.2 видно, что после умножения получается дополнительный бит знака, и что произведение двух 5-битовых чисел имеет длину 10 бит, так что результат следует округлить или усечь до 5 бит перед записью в память. Вообще, произведение двух B -битовых чисел имеет длину $2B$ бит. Например, 10-битовый результат смещается на один элемент влево для удаления добавочного бита знака, а затем округляется до 0,0101. Округление (или усечение) результата, как иллюстрируется выше, является одним из главных преимуществ дробной арифметики с дополнением до двух, поскольку оно означает, что при умножении никогда не произойдет переполнения. Впрочем, подобное округление (или усечение) вносит в сигнал ошибки, которые могут привести к неустойчивости или некоторым нежелательным побочным эффектам в системах ЦОС с обратной связью.

13.2.1.3. Сложение с фиксированной запятой

Сложение двух дробей с фиксированной запятой значительно сложнее умножения. Это объясняется тем, что операнды должны представляться в одном Q-формате, кроме того, следует учитывать возможность переполнения.

Пример 13.3

Найдите сумму следующих чисел, представленных в формате дополнения до двух: 0001 1001 и 0110 1101 011 1101.

Решение

Вначале операнды выражаются в одном Q-формате, а затем складываются:

```

0110 1101 0111 1101
0001 1001 0000 0000
1000 0110 0111 1101
| переполнение

```

Один из способов скорректировать переполнение — это сместить результат на одну позицию вправо, а затем установить метку экспоненты. Таким образом, ответ получается следующим:

```

1 0100 0011 0011 1110
↑
метка экспоненты

```

Другая альтернатива — это представить результат с использованием двойной точности или предоставить достаточное свободное пространство, чтобы разрешить расширение вследствие переполнения.

Пример 13.4

Пусть размер регистра составляет 4 бит (1 бит знака и 3 бит данных). Найдите сумму следующих чисел:

- 1) $-0,25$ и $0,75$;
- 2) $0,5$; $0,75$; $-0,5$.

Решение

Из табл. 13.2 находим: $0,25_{10} = 0,010_2$; $0,5_{10} = 0,100_2$ и $0,75_{10} = 0,110_2$.

1.

$$\begin{array}{r} 0,75 \quad 0,110 \\ -0,25 \quad 0,110 \\ \hline 0,50 \quad 1\,0,100 \end{array}$$

Ответ: $0,100$.

2.

$$\begin{array}{r} 0,5 \quad 0,100 \\ +0,75 \quad 0,110 \\ \hline 1,25 \quad 1,010 \leftarrow \text{частичная сумма} \\ -0,50 \quad 1,100 \\ \hline 0,75 \quad 1\,0,110 \leftarrow \text{окончательная сумма} \end{array}$$

Ответ: $0,110$.

13.2.2. Арифметика с плавающей запятой**13.2.2.1. Представление в формате с плавающей запятой**

Двоичное число X в формате с плавающей запятой представляется как произведение двух чисел со знаком — мантиссы M и экспоненты E :

$$X = M \cdot 2^E, \quad (13.3)$$

где 2 — основание двоичной системы.

Экспонента определяет диапазон чисел, которые можно представить, мантисса — это точность чисел. Например, если экспонента и мантисса представлены 8 и 16 бит соответственно, диапазон чисел с плавающей запятой, которые можно представить, в данном случае — от $0,5 \times 2^{-128}$ до $1 - (2^{-15}) \times 2^{128}$. Из 16 бит, используемых для представления мантиссы, один — это бит знака, а самый младший разряд имеет сомнительную точность из-за эффектов округления. Следовательно, точность чисел с плавающей запятой составляет 1 в степени 2^{14} ($0,61 \times 10^4$), т.е. порядка 4 десятичных цифр.

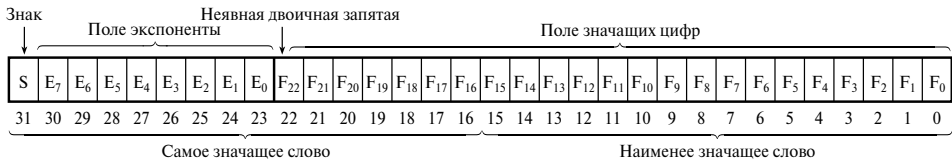


Рис. 13.1. Представление с плавающей запятой (формат IEEE обычной точности)

13.2.2.2. Формат с плавающей запятой IEEE

Одной из наиболее широко используемых двоичных систем с плавающей запятой является стандарт IEEE 754 [9, 15]. Формат обычной точности представлен на рис. 13.1. В данном случае числа с плавающей запятой с экспонентой в диапазоне $0 < E < 255$ называются *нормированными*.

Десятичный эквивалент X нормированного числа с плавающей запятой в формате IEEE равен

$$X = (-1)^s (1 \cdot F) 2^{E-127},$$

где

- F — мантисса дробной части в формате дополнения до двух, представленная битами от 0 до 22.
- E — экспонента в форме представления “с избытком 127”.
- $s = 0$ для положительных чисел, $s = 1$ для отрицательных чисел.

Две важные особенности формата IEEE с плавающей запятой — перед мантиссой предполагается наличие 1 и используется смещенный порядок (biased exponent).

Пример 13.5

1. Для числа с плавающей запятой

$$0\ 1000\ 0011.1100 \dots 00000$$

экспонента равна $1000\ 0011 = 131$, мантисса $-0,1100 \dots = 0,75$ и $s = 0$. Следовательно, $X = (1,75)2^{131-127} = 1 \times 1,75 \times 2^4$.

2. Для числа с плавающей запятой

$$1\ 0000\ 1111.0110 \dots 00000$$

экспонента равна $0000\ 1111 = 15$, мантисса $-0,0110 \dots = 0,375$ и $s = 1$. Следовательно, $X = (1,375)2^{15-127} = -1 \times 1,375 \times 2^{-112}$.

13.2.2.3. Сложение и умножение с плавающей запятой

Если X_1 и X_2 — два числа с плавающей запятой, сумму которых нужно найти, причем $X_1 = M_1 \times 2^{E_1}$ и $X_2 = M_2 \times 2^{E_2}$, их сумма X равна

$$X = M \times 2^E,$$

где

$$M = M_1 + M_2 \times 2^{E_1 - E_2}; E = E_1, \text{ предполагая, что } X_1 > X_2. \quad (13.4)$$

Перед сложением двух чисел с плавающей запятой необходимо уравнивать их экспоненты. Этот процесс называется *выравниваем* и включает сдвиг вправо мантиссы меньшего операнда и увеличение его экспоненты, пока она не сравняется с экспонентой большего операнда.

Если X_1 и X_2 — два надлежащим образом нормированных числа с плавающей запятой, произведение которых нужно найти, причем

$$X_1 = M_1 \times 2^{E_1}, \quad X_2 = N_2 \times 2^{E_2}, \quad (13.5)$$

тогда их произведение X записывается следующим образом:

$$X = M \times 2^E,$$

где

$$M = M_1 \times M_2, E = E_1 + E_2.$$

Таким образом, мантиссы умножаются и их экспоненты складываются. Поскольку M_1 и M_2 — нормированы, их произведение M будет принадлежать диапазону $0,25 < M < 1$. Следовательно, произведение M не вызывает переполнения, но может быть неверно нормировано (“антипереполнение” мантиссы).

Пример 13.6

1. Найдите сумму двух чисел A и B , где $A = 9,985 \times 10^4$ и $B = 5,6756 \times 10^2$.
2. Найдите произведение двух чисел A и B , где $A = 2,75 \times 10^{-16}$ и $B = 4,5 \times 10^{10}$.

Решение

1. Прежде всего экспоненты двух чисел сравниваются, и если они не равны, мантисса операнда с меньшей экспонентой смещается так, чтобы две экспоненты стали равными:

$$5,6756 \times 1062 = 0,056756 \times 10^4.$$

Затем мантиссы складываются:

$$M = 9,985 + 0,056756 = 10,041756; \quad E = 10^4$$

и их сумма равна $10,041756 \times 10^4$. Затем сумма нормируется — смещается десятичная точка мантиссы и (при необходимости) выравнивается экспонента — и получается окончательный ответ:

$$1,0041756 \times 10^5.$$

2. Мантиссы перемножаются и экспоненты складываются:

$$M = 2,75 \times 4,5 = 12,375; E = -16 + 10 = -6,$$

в результате получается произведение $A \times B = 12,375 \times 10^{-6}$. Далее произведение нормируется, причем предполагается, что должным образом нормированное число с плавающей запятой меньше 10, — для этого выравнивается экспонента и меняется положение десятичной точки в мантиссе. В результате получаем

$$A \times B = 1,2375 \times 10^{-5}.$$

В арифметике с плавающей запятой ошибки округления возникают и при сложении, и при умножении, тогда как в арифметике с фиксированной запятой ошибки округления возможны только при умножении. Впрочем, в отличие от переполнения в арифметике с фиксированной запятой, переполнение при сложении с плавающей запятой крайне маловероятно из-за очень большого доступного динамического диапазона: чем больше битов в экспоненте, тем шире динамический диапазон. В настоящее время значительное распространение получили программы и процессоры ЦОС с плавающей запятой (см. главу 12).

13.3. Шум квантования АЦП и качество сигнала

АЦП квантует аналоговый входной сигнал в конечное число битов, обычно 8, 12 или 16, что приводит к появлению шума квантования.

Как обсуждалось в главе 2, мощность шума квантования (или дисперсия, $\sigma_{\text{АЦП}}^2$) записывается следующим образом:

$$\sigma_A^2 = \frac{q^2}{12} = \frac{2^{-2B}}{3}, \quad (13.6)$$

где q — размер шага квантования, а B — число битов АЦП. Очевидно, что уровень шума можно легко уменьшить, увеличив число битов АЦП. Кроме того, можно также снизить этот шум, используя методы обработки при нескольких скоростях (см. главу 9). Вообще, для значений B , превышающих 12 бит, шум из-за ошибки квантования незначителен, исключая такие приложения, как профессиональное аудио, где для получения приемлемого качества требуется не менее 16 бит.

Шум, вызванный квантованием АЦП, существует в системе ЦОС как неизбежная ошибка. Мощность шума, введенного АЦП, на выходе системы ЦОС записывается следующим образом:

$$\sigma_{oA}^2 = \sigma_A^2 \left[\frac{1}{2\pi i} \oint H(z) H(z^{-1} \frac{dz}{z}) \right] = \quad (13.7, a)$$

$$= \sigma_A^2 \sum_{k=0}^{\infty} h^2(k), \quad (13.7, б)$$

где

σ_{oA}^2 — шум квантования АЦП на выходе системы;

$\oint_c$ — контурный интеграл;

$h(k)$ — импульсная характеристика системы.

Выражение в квадратных скобках можно рассматривать как “коэффициент усиления системы по мощности”, который определяет усиление (или изменение) шума АЦП в зависимости от характеристик системы ЦОС.

Пример 13.7

На входе системы ЦОС, которая характеризуется указанной ниже передаточной функцией, расположен 8-битовый АЦП.

$$H(z) = \frac{1}{z + 0,5}.$$

Оцените стационарную мощность шума квантования на выходе системы.

Решение

Мощность шума, введенного АЦП, на входе системы составляет (уравнение (13.6)):

$$\sigma_A^2 = \frac{2^{-16}}{3}.$$

Выходной шум, порожденный АЦП, выражается следующим образом:

$$\sigma_{oA}^2 = \sigma_A^2 \left[\frac{1}{2\pi i} \oint_c H(z) H(z^{-1} \frac{dz}{z}) \right] = \sigma_A^2 \sum_{k=0}^{\infty} h^2(k), \quad (13.8)$$

где $\oint_c$ обозначает контурный интеграл, а $h(k)$ — импульсная характеристика системы (см. главу 4). Выражение в квадратных скобках можно рассматривать как коэффициент усиления системы, который отражает изменение мощности шума АЦП.

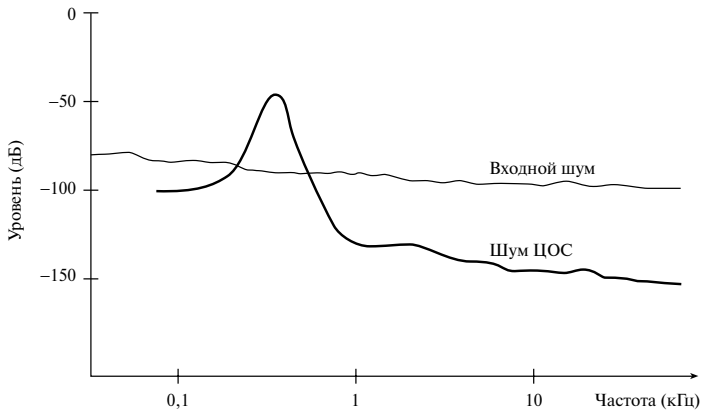


Рис. 13.2. Иллюстрация влияния шума округления в обработке сигналов на уровень собственных шумов системы

Для выбранной простой передаточной функции выражение в квадратных скобках можно легко вычислить, используя метод вычетов, рассмотренный в главе 4. Используя результаты, полученные в главе 4, и единичную окружность в качестве контура интегрирования, получаем, что выражение в квадратных скобках равно $4/3$. Следовательно, мощность шума квантования равна

$$\sigma_{oA}^2 = \frac{2^{-16}}{3} \times \frac{4}{3}.$$

Шум АЦП плюс шум, присущий сигналу, определяет минимальный уровень шума на входе. Для поддержания качества сигнала уровень искажения, введенный последующей цифровой обработкой, должен быть ниже этого минимального уровня шумов. На рис. 13.2 приведен пример системы ЦОС, в которой шум, введенный цифровой обработкой сигнала (в данном случае — КИХ-фильтрацией) превышает уровень входного шума [21]. В подобных случаях требуется принимать меры для подавления шума до уровня собственных шумов (например, с помощью системы обратной связи по ошибке округления или процессора с достаточно большой длиной слова).

На практике число битов, используемых для представления входных данных, определяет динамический диапазон и качество сигнала. Например, в потребительских системах проигрывания аудиокомпакт-дисков аудиовход представляется как 16-битовые выборки, что соответствует теоретическому уровню сигнал-шум в 96 дБ. Для поддержания требуемого качества операции ЦОС должны выполняться с длиной слова не менее 24 бит. При длине слова процессора 16 бит динамический диапазон равен 96 дБ, но с учетом ошибок вычисления эффективный динамический диапазон будет меньше. Следовательно, при анализе эффектов конечной разрядности важно также рассматривать качество входного сигнала.

13.4. Эффекты конечной разрядности в цифровых БИХ-фильтрах

Коэффициенты a_k и b_k , полученные на этапе 2 разработки БИХ-фильтра, имеют бесконечную или очень высокую точность, обычно это шесть десятичных разрядов. При реализации цифрового БИХ-фильтра на небольшой системе, такой как 8-битовый микрокомпьютер, возникают ошибки в представлении коэффициентов фильтра и выполнении арифметических операций, определяемых разностным уравнением. Данные ошибки снижают производительность фильтра и в крайних случаях приводят к неустойчивости фильтра.

Перед реализацией БИХ-фильтра важно убедиться, что снижение производительности, вызванное эффектами конечной разрядности, минимально, и, если это не так, принять необходимые меры. Вообще, влияние указанных ошибок можно снизить до приемлемого уровня, используя больше битов, но это может привести к повышению стоимости системы.

Перечислим основные ошибки цифровых БИХ-фильтров.

- Шум квантования АЦП, проистекающий от представления выборок входных данных $x(n)$ малым числом битов.
- Ошибки квантования коэффициентов, вызванные представлением коэффициентов БИХ-фильтров конечным числом битов.
- Ошибки переполнения, которые происходят при сложении или промежуточном суммировании частичных результатов в регистре ограниченной длины.
- Ошибки округления результата, когда выход $y(n)$ и результаты внутренних арифметических операций округляются (или усекаются) до разрешенной длины слова.

Степень ухудшения фильтра зависит от, во-первых, длины слова и типа арифметики, используемой для выполнения операции фильтрации, во-вторых, метода, используемого для квантования коэффициентов и переменных фильтра, и, в-третьих, структуры фильтра. Зная данные факторы, разработчик может оценить влияние конечной разрядности на производительность фильтра и при необходимости принять меры. В зависимости от реализации фильтра, некоторые эффекты могут быть незначительными. Например, если фильтр реализуется как программа на языке высокого уровня преимущественно на больших компьютерах, ошибки квантования коэффициентов и округления несущественны. При обработке в реальном времени входные и выходные сигналы, коэффициенты фильтра и результаты арифметических операций представляются с использованием слов конечной длины (обычно 8, 12 и 16 бит). В таких случаях практически всегда необходимо проанализировать влияние квантования на производительность фильтра.

Влияние конечной разрядности на производительность более сложно проанализировать для БИХ-фильтров, чем для КИХ-фильтров, вследствие наличия обратной связи. В то же время, компьютерная поддержка позволяет для определенных фильтров найти выгодные практические решения. Влияние каждого из четырех перечисленных источников ошибок рассматривается ниже в этой главе.

13.4.1. Влияние структуры фильтра на эффекты конечной разрядности

Как уже говорилось, БИХ-фильтры можно представить с помощью разнообразных структур, которые теоретически эквивалентны. В то же время, при реализации на процессорах ЦОС с фиксированной или плавающей запятой поведение фильтров может существенно различаться.

На практике БИХ-фильтры часто реализуются с использованием структур второго порядка прямой формы I или прямой формы II (канонической) (рис. 13.1 и 13.2). Прямая форма I характеризуется следующим образом:

$$y(n) = \sum_{i=0}^2 b_i x(n-i) - \sum_{i=1}^2 a_i y(n-i) \text{ — разностное уравнение}$$

$$H(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} \text{ — передаточная функция}$$

- 5 коэффициентов фильтра;
- 4 элемента задержки;
- 1 сумматор (4 сложения)
- 1 точка квантования для сумм произведений;
- 1 умножитель (5 умножений);
- 9 ячеек памяти для хранения данных и коэффициентов.

Каноническое звено характеризуется следующим образом:

$$\left. \begin{aligned} y(n) &= \sum_{i=0}^2 b_i w(n-i) \\ w(n) &= x(n) - \sum_{i=1}^2 a_i w(n-i) \end{aligned} \right\} \text{ — двухэтапное разностное уравнение}$$

$$H(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} \text{ — передаточная функция}$$

- 5 коэффициентов фильтра;
- 2 элемента задержки;
- 2 сумматора (4 сложения)
- 2 точки квантования для сумм произведений;
- 1 умножитель (5 умножений);
- 7 ячеек памяти для хранения данных и коэффициентов.

Отметим, что хотя две передаточные функции теоретически одинаковы, между ними существуют важные отличия. Например, в структуре, реализованной в прямой форме I (рис. 13.3, а), члены прямой связи (относятся к нулям) предшествуют членам обратной связи (относятся к полюсам). Для канонического звена (рис. 13.3, б) все наоборот. В результате, на практике полюса канонического звена имеют тенденцию к дальнейшему усилению шума, генерируемого при вычислениях.

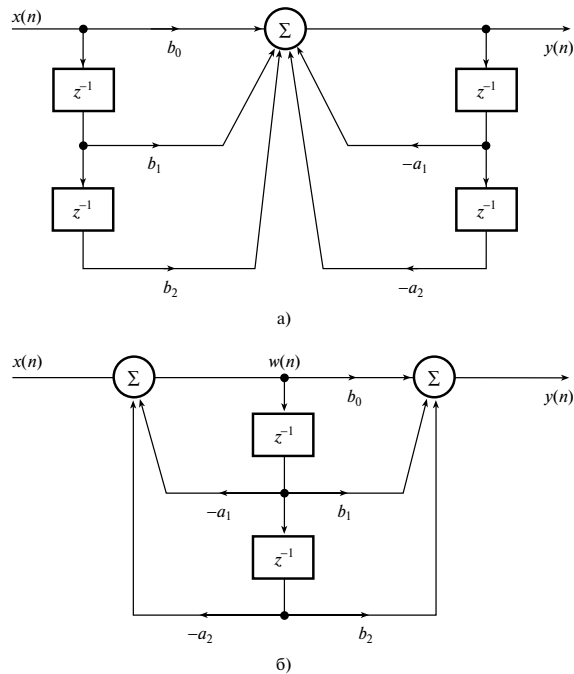
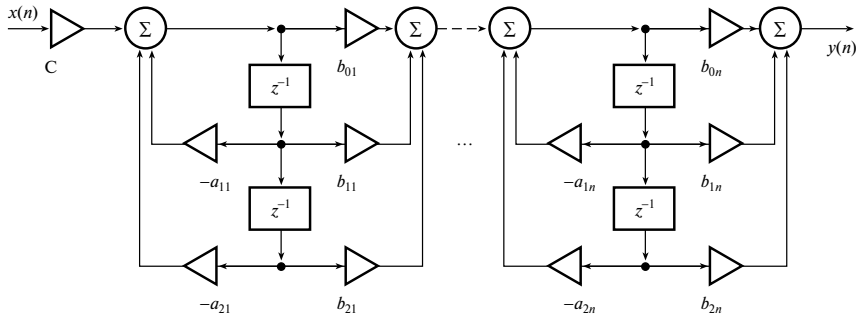


Рис. 13.3. Стандартные блоки БИХ-фильтров: а) звено второго порядка, реализованное в прямой форме I; б) каноническое звено второго порядка

Различия между двумя структурами более очевидны, если непосредственно сравнивать рис. 13.3, а и б. Реализация в прямой форме I (рис. 13.3, а) имеет только 1 сумматор и 1 точку квантования для сумм произведений, тогда как каноническое звено (рис. 13.3, б) имеет 2 сумматора и 2 точки квантования. Выход левого сумматора на рис. 13.3, б приходится на внутренний узел $w(n)$. Следовательно, каноническое звено подвержено внутреннему самоподдерживающемуся переполнению. Реализация в прямой форме I не содержит внутреннего узла, и переполнение на ее выходе при сложении в арифметике дополнения до двух является либо самоисправляющимся, либо его легко скорректировать извне. Более того, вход такого звена $x(n)$ масштабируется на коэффициент b_0 , в отличие от канонического звена, где вход никак не ограничивается. Следовательно, видно, что структура, использованная для реализации функции ЦОС (в данном случае — КИХ-фильтрации), существенно влияет на производительность итоговой системы ЦОС.

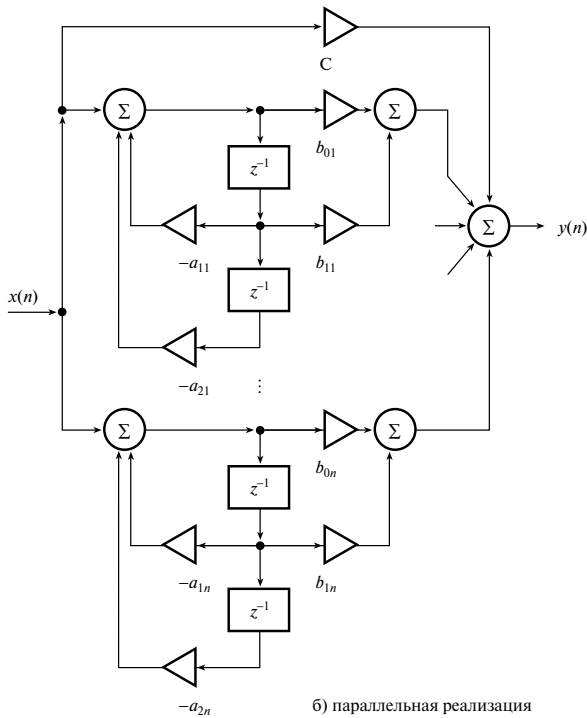
На практике БИХ-фильтры больших порядков реализуются в форме каскадного или параллельного соединения стандартных блоков второго порядка (рис. 13.4). При каскадном соединении возникает три вопроса:

- как выбрать пары “множитель числителя-множитель знаменателя” и определить коэффициенты блоков второго порядка;



а) каскадная реализация

$$H(z) = C \times \prod_{k=1}^n \frac{b_{0k} + b_{1k}z^{-1} + b_{2k}z^{-2}}{1 + a_{1k}z^{-1} + a_{2k}z^{-2}}$$



б) параллельная реализация

$$H(z) = C + \sum_{k=1}^n \frac{b_{0k} + b_{1k}z^{-1}}{1 + a_{1k}z^{-1} + a_{2k}z^{-2}}$$

Рис. 13.4. Структуры реализации БИХ-фильтров больших порядков

- в какой последовательности соединить блоки второго порядка;
- как масштабировать уровни сигналов в различных точках суммарного фильтра, чтобы они согласовывались с разрешенной длиной слова.

Как обсуждалось в главе 8, разбиение на пары и упорядочение звеньев фильтра тесно связано с эффектами конечной разрядности. В зависимости от порядка фильтра допускается существование множества эквивалентных структур, образованных перепорядочением звеньев второго порядка, но с точки зрения ошибок конечной разрядности данные структуры не равнозначны. Чтобы определить подходящую конфигурацию фильтра, можно, например, использовать программу анализа эффектов конечной разрядности из книги [10].

При параллельной реализации порядок соединения звеньев значения не имеет.

13.4.2. Ошибки квантования коэффициентов

БИХ-фильтр характеризуется следующим уравнением:

$$H(z) = \frac{\sum_{k=0}^N b_k z^{-k}}{1 + \sum_{k=1}^M a_k z^{-k}},$$

$$[H(z)]_q = \frac{\sum_{k=0}^N [b_k]_q z^{-k}}{1 + \sum_{k=1}^M [a_k]_q z^{-k}},$$

где

$$[a_k]_q = a_k + \Delta a_k; \quad [b_k]_q = b_k + \Delta b_k;$$

$\Delta a_k, \Delta b_k$ — изменение коэффициентов a_k и b_k соответственно,

q обозначает квантованную величину.

Основное влияние квантования коэффициентов фильтра с использованием конечного числа битов проявляется в изменении положения полюсов и нулей $H(z)$ на комплексной плоскости. В узкополосных фильтрах, например, полюса расположены близко к единичной окружности, так что любое значительное изменение их положения может сделать фильтр неустойчивым. Чем меньше число битов, используемых для представления коэффициентов, тем большим будет изменение положения полюсов и нулей.

Помимо потенциальной неустойчивости смещение положений полюсов и нулей также приводит к изменению частотной характеристики. Квантованный фильтр всегда следует проанализировать, чтобы убедиться, что используемой длины слова достаточно для обеспечения устойчивости и удовлетворительной частотной характеристики. Следовательно, на этом этапе разработчик должен определить число битов, требуемых для представления коэффициентов фильтра.

Пример 13.8

Полосовой цифровой фильтр планируется использовать для цифрового восстановления тактовой синхронизации в модеме 4,8 Кбит/с при частоте дискретизации 153,6 кГц. Фильтр характеризуется следующей передаточной функцией:

$$H(z) = \frac{1}{1 + a_1 z^{-1} + a_2 z^{-2}},$$

где

$$a_1 = -1,957558 \text{ и } a_2 = 0,995913.$$

Оцените влияние квантования коэффициентов до 8 бит на положение полюсов, а следовательно, центральную частоту.

Решение

Вначале найдем положения всех полюсов неквантованного фильтра. Радиус r и угол θ полюса (см. уравнение (13.10)) записываются как

$$r = \sqrt{0,995913} = 0,99795, \quad \theta = \arccos\left(\frac{1,957558}{2r}\right) = 11,25^\circ.$$

Найденные величины соответствуют центральной частоте

$$\left(\frac{11,25}{360}\right) \times 153,6 \times 10^3 = 4,7999 \text{ кГц.}$$

Далее квантуем коэффициенты до 8 бит. Поскольку один коэффициент больше единицы, выделим 1 бит для хранения бита знака, 1 бит для целой и 6 бит — для дробной части коэффициента. Таким образом, после квантования коэффициенты приобретают такие значения:

$$a_1 = -1,957558 \times 2^6 = -125 \equiv 10000100,$$

$$a_2 = 0,995913 \times 2^6 = 63 \equiv 00111111.$$

В дробной записи значения квантованных коэффициентов имеют такой вид:

$$a_1 = -\frac{125}{64} = -1,953125; \quad a_2 = \frac{63}{64} = 0,984375.$$

Новое положение полюса: $r = 0,992156$, $\theta = 10,171853^\circ$; центральная частота смещается в положение

$$f_0 = \left(\frac{10,171853}{360}\right) \times 153,6 \times 10^3 = 4,3399 \text{ кГц.}$$

13.4.3. Требования к длинам коэффициентов с точки зрения устойчивости и желаемой частотной характеристики

Далее обсуждение устойчивости будет касаться только звеньев второго порядка, являющихся основой любого фильтра. Рассмотрим звено второго порядка, которое характеризуется следующими уравнениями:

$$H(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}},$$

$$y(n) = \sum_{k=0}^2 b_k x(n-k) - \sum_{k=1}^2 a_k y(n-k).$$

Полюса (или корни знаменателя) расположены в точках

$$p_1 = \frac{1}{2}[-a_1 + (a_1^2 - 4a_2)^{1/2}], \quad (13.9, a)$$

$$p_2 = \frac{1}{2}[-a_1 - (a_1^2 - 4a_2)^{1/2}]. \quad (13.9, б)$$

Каждое звено второго порядка может иметь полюса трех типов: комплексно-сопряженные, неравные действительные и равные действительные. Наиболее распространенный случай — это комплексно-сопряженные полюса, которые появляются при $a_1^2 < 4a_2$. В данном случае полюса расположены на окружности радиуса r с центром в начале координат с угловыми координатами

$$p_1 = r \angle \theta, p_2 = r \angle -\theta, \quad (13.10)$$

где

$$r = a_2^{1/2}, \theta = \arccos\left(-\frac{a_1}{2r}\right).$$

Небольшие изменения коэффициентов a_1 и a_2 после квантования коэффициентов приведут к изменениям r и θ . Чтобы обеспечить устойчивость, коэффициенты фильтра должны лежать внутри треугольника устойчивости (рис. 13.5), который ограничен следующими линиями:

$$0 \leq |a_2| < 1, \quad (13.11, a)$$

$$|a_1| \leq 1 + a_2. \quad (13.11, б)$$

Выполнение первого условия гарантирует, что полюса будут лежать внутри единичной окружности, поскольку радиус полюса определяется уравнением (13.10). Из уравнений (13.10) и (13.11) находим, что для оценки числа битов, требуемых для поддержания устойчивости, можно получить несколько простых формул, которые, впрочем, применимы к ограниченному набору классов. Альтернативным способом оценки подходящих

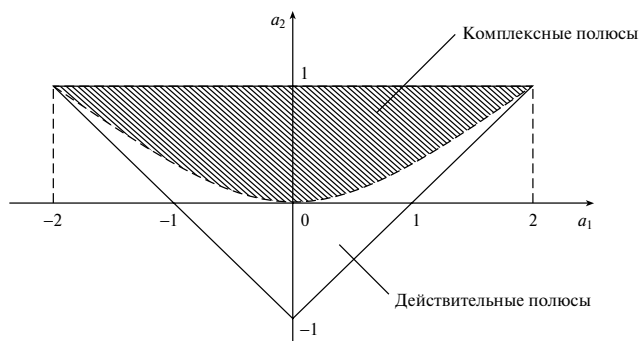


Рис. 13.5. Треугольник устойчивости, показывающий значения коэффициентов фильтра a_1 и a_2 , при которых фильтр устойчив

длин коэффициентов является анализ отдельных блоков второго порядка при различных значениях длин коэффициентов (см. пример 13.9).

Числа битов, необходимых для обеспечения устойчивости, может быть недостаточно для получения приемлемой характеристики. Вообще, при представлении коэффициентов слишком малым числом битов меняется частотная характеристика в полосе пропускания и полосе подавления (см. рис. 13.6). Изменения в полосе пропускания вызваны в основном изменением положений полюсов, а изменения в полосе подавления — изменением положений нулей.

Для определения длины коэффициентов, обеспечивающей удовлетворительную частотную характеристику, обычно находится минимальная длина коэффициентов, при которой удовлетворяются требования в полосе подавления и пропускания. Хотя такой подход может быть вычислительно трудоемким, доступность персональных компьютеров и программ анализа эффектов конечной разрядности позволяет относительно просто определить длину слова для данного фильтра (пример подобной программы приводится в книге [10]). Альтернативный подход — использовать статистический метод, чтобы получить разумно точные оценки [2].

Для представления коэффициентов можно использовать меньшее число битов, если для квантования коэффициентов применить метод оптимизации или увеличить порядок фильтра, не меняя остальные параметры. Таким образом можно использовать компромиссы между длиной коэффициентов и порядком фильтра [16]. Например, перед разработчиком может стоять задача использования существующего процессора или системы с предопределенной длиной слова. Если разработчик видит, что требуемая длина слова больше длины слова процессора, может быть удобнее повысить порядок фильтра для уменьшения требований к длине слова до уровня, который предписывает процессор. Впрочем, использование фильтров больших порядков потребует больше вычислений, что может наложить определенные ограничения на скорость работы и сделать систему более уязвимой к шуму округления. Таким образом, все компромиссы следует рассматривать очень тщательно.

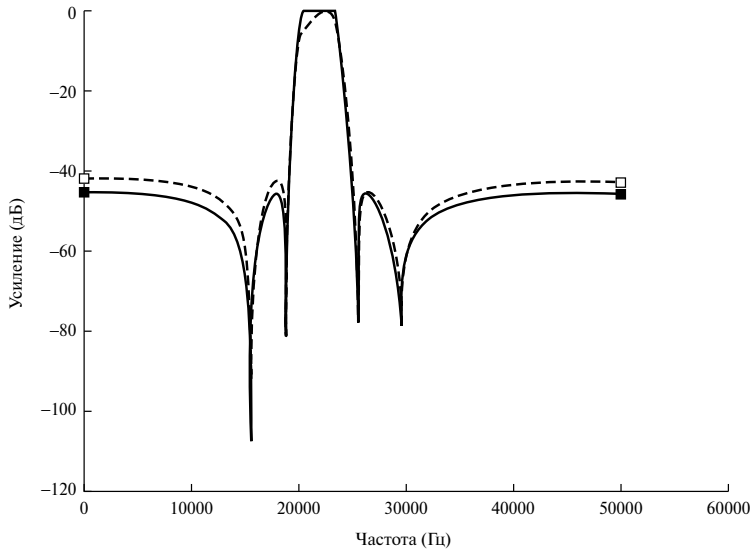


Рис. 13.6. Практическое влияние квантования коэффициентов на частотную характеристику: ■ — неквантованные коэффициенты; □ — коэффициенты квантованы до 5 бит

Пример 13.9

Требуется цифровой фильтр, удовлетворяющий следующим спецификациям частотной характеристики:

полоса пропускания	20,5–23,5 кГц,
полоса подавления	0–19, 25–50 кГц,
неравномерность в полосе пропускания	$\leq 0,25$ дБ,
затухание в полосе подавления	> 45 дБ,
частота дискретизации	100 кГц.

1. Найдите подходящую передаточную функцию фильтра.
2. Определите подходящую длину коэффициентов,
 - а) обеспечивающую устойчивость;
 - б) удовлетворяющую спецификациям частотной характеристики.
3. Изобразите частотную характеристику неквантованного и характеристику квантованных фильтров, полученных в п. 2.

Решение

1. Используя программу разработки (книга [10], подробности см. в предисловии), находим, что требованиям удовлетворяет эллиптический фильтр со следующей передаточной функцией:

$$H(z) = H_1(z)H_2(z)H_3(z)H_4(z),$$

где

$$H_1(z) = \frac{1 + 0,0339z^{-1} + z^{-2}}{1 - 0,1743z^{-1} + 0,9662z^{-2}},$$

$$H_2(z) = \frac{1 - 0,7563z^{-1} + z^{-2}}{1 - 0,5588z^{-1} + 0,9675z^{-2}},$$

$$H_3(z) = \frac{1 + 0,5331z^{-1} + z^{-2}}{1 - 0,2711z^{-1} + 0,9028z^{-2}},$$

$$H_4(z) = \frac{1 - 1,1489z^{-1} + z^{-2}}{1 - 0,4441z^{-1} + 0,9045z^{-2}}.$$

- 2, а. Коэффициенты знаменателя каждого звена второго порядка были проквантованы (округлены) до B бит ($B = 2, 3, \dots, 29$), включая бит знака. Затем для каждого значения B были вычислены квантованные коэффициенты и положения полюсов в полярных координатах. Рассмотрим для иллюстрации первое звено второго порядка с передаточной функцией $H_1(z)$. Для $B = 8$ бит коэффициенты знаменателя квантуются следующим образом:

$$a_1 = -(0,1743 \times 2^7 + 0,5) = -22,8104 = -22,$$

$$a_2 = 0,9662 \times 2^7 + 0,5 = 124,1736 = 124.$$

В дробной записи коэффициенты имеют следующий вид:

$$a_1 = -22/128 = -0,171875,$$

$$a_2 = 124/128 = 0,96875.$$

Из уравнения (13.10) находим, что радиус и угол полюса звена при $B = 8$ бит равны

$$r = \sqrt{0,96875} = 0,9843,$$

$$\theta = \arccos\left(-\frac{b_1}{2r}\right) = \arccos(0,087308) = 84,99^\circ.$$

Все квантованные коэффициенты и полярные координаты вычислены с использованием программы анализа. Если при некоторой длине слова радиальное расстояние от полюса до начала координат равно или превышает 1, возникает потенциальная неустойчивость. При анализе было найдено, что для поддержания устойчивости требуется не меньше $B = 5$ бит. Вообще, если полюс неквантованного звена второго порядка лежит на окружности с радиусом $r < 0,9$, неустойчивость маловероятна, если использовать длину слова 8 бит или больше.

- 2, б. Коэффициенты звеньев второго порядка квантовались до различных длин слов, как описано выше. Для каждой длины слова квантованные коэффициенты собирались в общую квантованную передаточную функцию фильтра, реализованного в прямой форме. Примеры коэффициентов при длинах слов 5 и 16 бит приведены в табл. 13.4. Были также рассчитаны неравномерность в полосе пропускания и

затухание в полосе подавления квантованного фильтра для различных размеров коэффициентов. Было найдено, что для удовлетворения спецификаций частотной характеристики в полосе подавления и полосе затухания требуется 16 или более бит. Отметим, что это больше, чем нужно для обеспечения устойчивости.

3. Частотные характеристики, масштабированные так, чтобы максимум располагался на 0 дБ, с квантованными и неквантованными коэффициентами ($B = 5$ бит) изображены на рис. 13.6. Визуально характеристика фильтра, квантованного до 16 бит, не отличается от характеристики фильтра с неквантованными коэффициентами, поэтому на рисунке она не показана.

Таблица 13.4. Коэффициенты фильтра (пример 13.9), демонстрирующие эффекты конечной разрядности

k	$B(k)$			$A(k)$		
	Точно	5 бит	16 бит	Точно	5 бит	16 бит
0	1,000 000	1,000 000	1,000 000	1,000 000	1,000 000	1,000 000
1	-1,338 200	-1,250 000	-1,338 165	-1,448 300	-1,437 500	-1,448 273
2	3,806 737	3,707 031	3,806 700	4,483 108	4,355 469 0	4,483 071
3	-3,556 357	-3,288 574	-3,556 255	-4,220 527	-4,060 791	-4,220 431
4	5,629 177	5,443 726	5,629 105	6,647 162	6,261 536	6,647 087
5	-3,556 357	-3,288 574	-3,556 255	-3,945 450	-3,677 216	-3,945 354
6	3,806 737	3,707 031	3,806 700	3,918 398 1	3,573 486	3,918 352
7	-1,338 200	-1,250 000	-1,338 165	-1,182 602 0	-1,067 047	-1,182 575
8	1,000 000	1,000 000	1,000 000	0,763 340 2	0,672 912	0,763 338

13.4.4. Ошибки переполнения при сложении

В арифметике с дополнением до двух при сложении двух больших чисел одного знака может произойти переполнение (т.е. получится результат, который превышает приемлемую длину слова), а это приведет к изменению знака выходной выборки. Таким образом, очень большое положительное число становится очень большим отрицательным числом, и наоборот (рис. 13.7). Рассмотрим каноническое звено, изображенное на рис. 13.8. Вследствие рекурсивной природы БИХ-фильтра переполнение в узле $w(n)$ переходит по обратной связи и используется для вычисления следующего выхода, где оно может вызвать дальнейшее переполнение и привести к нежелательным самоподдерживающимся осцилляциям. Подобные *масштабные граничные циклы переполнения* трудно остановить после их возникновения, чтобы их прекратить, следует повторно инициализировать фильтр.

Масштабное переполнение происходит на выходах сумматоров, и его можно предотвратить путем масштабирования входов сумматоров так, чтобы выходы были малыми, хотя такое решение возможно за счет снижения отношения сигнал-шум. Таким образом, важно выбрать масштабные множители так, чтобы предотвратить переполнение и при этом сохранить наибольшее возможное отношение сигнал-шум.

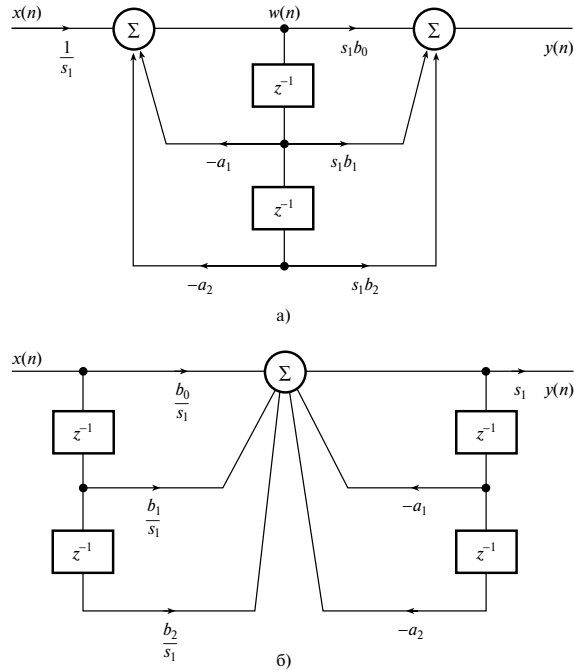


Рис. 13.9. Принципы масштабирования в звеньях второго порядка:
а) каноническое звено; б) реализация в прямой форме

13.4.5. Принципы масштабирования

13.4.5.1. Каноническое звено

Рассмотрим каноническое звено второго порядка (рис. 13.9, а). Масштабный множитель s_1 на входе фильтра выбирается так, чтобы предотвратить переполнение или снизить его вероятность на выходе левого сумматора. Чтобы поддержать общее усиление фильтра на предыдущем уровне, коэффициенты числителя умножаются на s_1 .

Существует три распространенных метода определения подходящих масштабных множителей фильтра. В первом методе, который часто называют *нормой* L_1 , масштабный множитель выбирается следующим образом:

$$s_1 = \sum_{k=0}^{\infty} |f(k)|, \quad (13.12)$$

где $f(k)$ — импульсная характеристика, связывающая вход системы с выходом первого сумматора, т.е. узлом $w(n)$. Выбранный таким способом масштабный множитель s_1 гарантирует, что общее усиление фильтра от входа до узла $w(n)$ равно единице, поэтому переполнение в $w(n)$ невозможно. Для получения импульсной характеристики $f(k)$ можно вначале определить соответствующую передаточную функцию $F(z)$, а затем применить к ней обратное z -преобразование.

Во втором методе, часто называемом *нормой* L_2 , масштабный множитель s_1 определяется следующим образом:

$$s_1 = \left[\sum_{k=0}^{\infty} f^2(k) \right]^{1/2}. \quad (13.13)$$

Нормировочный множитель L_2 можно получить иначе, используя контурный интеграл

$$\sum_{k=0}^{\infty} f^2(k) = \frac{1}{2\pi i} \oint F(z) F(z^{-1}) \frac{dz}{z}, \quad (13.14)$$

где $F(z)$ — z -образ (результат применения z -преобразования) $f(k)$, а $\oint$ обозначает контурный интеграл по единичной окружности $|z| = 1$. Вычисляя норму L_2 , получаем:

$$\begin{aligned} s_1^2 = \sum_{k=0}^{\infty} f^2(k) &= \frac{1}{2\pi i} \oint \frac{1}{1 + a_1 z^{-1} + a_2 z^{-2}} \frac{1}{1 + a_1 z + a_2 z^2} \frac{dz}{z} = \\ &= \frac{1}{1 - a_2^2 - a_1^2(1 - a_2)/(1 + a_2)}. \end{aligned} \quad (13.15)$$

Формула (13.15) позволяет обойтись без вычисления бесконечной суммы (13.13). На практике, впрочем, $f(k)$ имеет лишь конечное число значимых членов, и эту функцию легко вычислить с помощью подходящей программы анализа эффектов конечной разрядности.

В методе 3 используется так называемая *норма* L_{∞} и масштабный множитель определяется следующим образом:

$$s_1 = \max |F(w)|, \quad (13.16)$$

где $F(w)$ — максимальная амплитуда частотной характеристики между входом и выходом $w(n)$.

В методе 1 изначально предполагается, что вход ограничен, т.е. $|x(n)| < 1$. Схема масштабирования такова, что вне зависимости от типа входа переполнения не будет. Это чересчур радикальная схема масштабирования, поскольку она учитывает маловероятные ситуации. Норма L_2 соответствует наложению условий на энергию входного сигнала и передаточной функции. Основным привлекательным отличием нормы L_2 является то, что анализ эффектов конечной разрядности требует оценки норм L_2 (сравните, например, уравнение (13.8) и (13.14)). Кроме того, эта норма позволяет вывести ряд аналитических выражений для различных фильтрующих структур. Метод 3 гарантирует, что в фильтре не случится переполнения при подаче на его вход синусоидального сигнала, также этот фильтр предлагает наилучший компромисс. Обычно, кстати, именно такую схему масштабирования и выбирают, поскольку она позволяет проверить экспериментально эффективность масштабирования с помощью синусоидальных сигналов.

Для компактного представления i -го масштабного множителя используется запись

$$s_i = \|F\|_p,$$

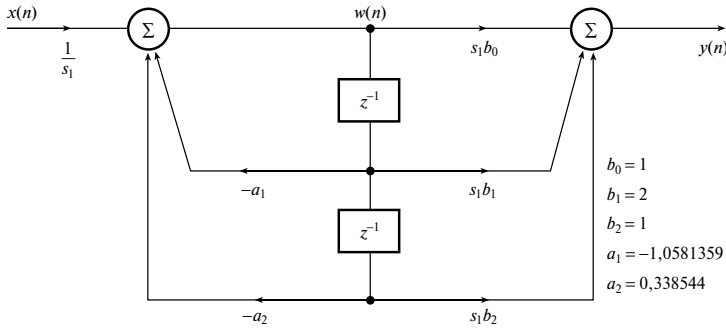


Рис. 13.10. Блок-схема фильтра (пример 13.9)

где символ $\|\cdot\|$ обозначает норму, а $p = 1, 2, \infty$ — тип нормы. Масштабные множители, полученные с использованием описанных трех методов, подчиняются следующему соотношению:

$$L_2 < L_\infty < L_1.$$

13.4.5.2. Прямая структура

Рассмотрим прямую структуру, изображенную на рис. 13.9, б. Поскольку фильтр имеет один накопитель, внутреннее переполнение проблемы не представляет, так что масштабирование входа не является обязательным. Этот момент — одно из привлекательных отличий прямой структуры. Промежуточное переполнение возможно на выходе сумматора при вычислении $y(n)$. Впрочем, это несущественно, если гарантировать, что в конечном выходе переполнения не будет. Если масштабирование необходимо, масштабные множители можно упорядочить, как показано на рис. 13.9.

Пример 13.10

Определите подходящий масштабный множитель, позволяющий предотвратить или снизить возможность переполнения в БИХ-фильтре нижних частот, который характеризуется следующей передаточной функцией:

$$H(z) = \frac{1 + 2z^{-1} + z^{-2}}{1 - 1,0581359z^{-1} + 0,338544z^{-2}}.$$

Решение

Блок-схема фильтра, построенного на канонических звеньях второго порядка, показана на рис. 13.10. Используя программу анализа эффектов конечной разрядности [10] для расчета уравнений (13.12), (13.13) и (13.16), получим масштабные множители для трех методов:

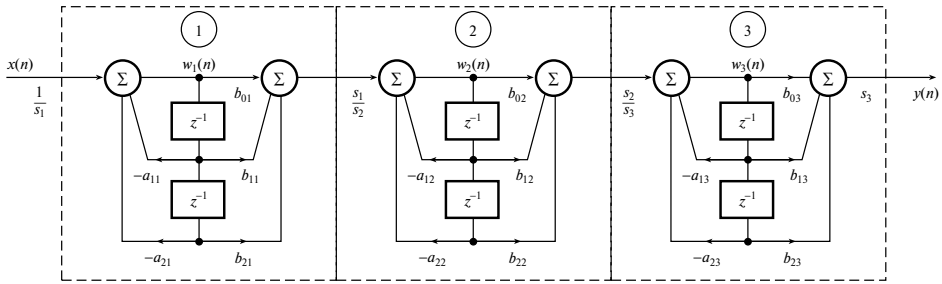


Рис. 13.11. Масштабирование в каскадной реализации БИХ-фильтра шестого порядка

	L_1	L_2	L_∞
s_1	3,7112	1,7352	3,5663

Просто для иллюстрации вычислим также норму L_2 , используя уравнение (13.15):

$$s_1^2 = \frac{1}{1 - (0,3385)^2 - (1,058)^2[(-0,3385)/(1 + 0,3385)]} =$$

$$= 1/0,3322 = 3,01,$$

$$s_1 = 1,7350.$$

13.4.6. Масштабирование при каскадной реализации

На практике фильтры реализуются как каскадные или параллельные комбинации звеньев второго порядка. На рис. 13.11 показана схема масштабирования каскадной реализации шестого порядка. Как и ранее, масштабные коэффициенты s_i , $i = 1, 2, 3$ выбираются так, чтобы предотвратить или минимизировать переполнение в звеньях фильтра в узлах, отмеченных $w_i(n)$. Схема масштабирования для каждого звена второго порядка, по сути, не отличается от схемы, используемой в базовых фильтрах, рассмотренных выше. Масштабные коэффициенты находятся следующим образом:

$$s_i = \|F_i(z)\|_p,$$

где p обозначает тип нормы: $p = 1, 2, \infty$. $F_i(z)$ — передаточная функция, связывающая вход фильтра с узлом $w_i(n)$, которая записывается так:

$$F_i(z) = \frac{\prod_{k=1}^{i-1} H_k(z)}{1 + a_{1i}z^{-1} + a_{2i}z^{-2}}, \quad i = 1, 2, 3.$$

При каскадной реализации входные масштабные коэффициенты s_1/s_2 обычно вносятся в числитель первого каскада, s_2/s_3 — в числитель второго, и т.д. Следовательно, масштабные коэффициенты на рис. 13.11 можно переупорядочить так, как показано

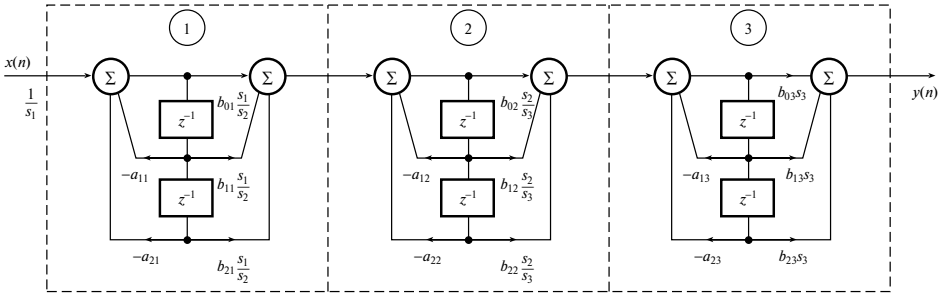


Рис. 13.12. Масштабирование в каскадной реализации БИХ-фильтра шестого порядка (масштабные коэффициенты внесены в числитель, т.е. представлены коэффициенты прямой связи)

на рис. 13.12. Следует также отметить, что передаточная функция фильтра после масштабирования, как они определены выше, идентичны соответствующим параметрам немасштабированного фильтра (по крайней мере, теоретически).

Пример 13.11

Используя описанные выше три метода, сравните масштабные коэффициенты фильтра с приведенной ниже передаточной функцией, предполагая каскадную реализацию со звеньями второго порядка:

$$H(z) = H_1(z)H_2(z)H_3(z),$$

$$H_1(z) = \frac{1 + 0,2189z^{-1} + z^{-2}}{1 - 0,0127z^{-1} + 0,9443z^{-2}},$$

$$H_2(z) = \frac{1 - 0,5291z^{-1} + z^{-2}}{1 - 0,1731z^{-1} + 0,7252z^{-2}},$$

$$H_3(z) = \frac{1 + 1,5947z^{-1} + z^{-2}}{1 - 0,6152z^{-1} + 0,2581z^{-2}}.$$

Решение

Используя программу анализа эффектов конечной разрядности и три метода определения нормы, получим масштабные множители s_1 , s_2 и s_3 .

	L_1	L_2	L_3
s_1	20,9608	3,0388	13,4098
s_2	19,0361	2,5358	10,1366
s_3	14,4467	2,9146	6,4087

Как отмечалось выше, норма L_1 всегда максимальна, а L_2 — минимальна.

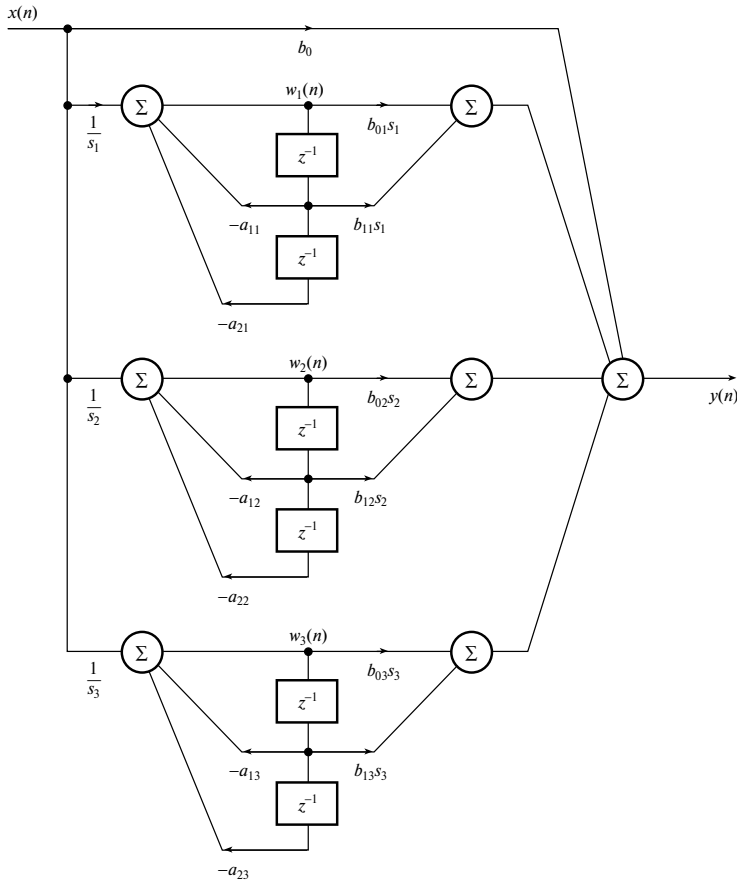


Рис. 13.13. Масштабирование в параллельной реализации БИХ-фильтра шестого порядка

13.4.7. Масштабирование в параллельной реализации

На рис. 13.13 изображена схема масштабирования для параллельной реализации БИХ-фильтра шестого порядка. Видно, что в данном случае звенья второго порядка масштабируются по отдельности, как описано выше. Масштабный множитель s_i на входе каждого фильтрующего звена гарантирует, что переполнения в соответствующем узле $w_i(n)$ не будет. Чтобы сохранить усиление звена таким же, каким оно было до масштабирования, коэффициенты прямой связи b_{ki} умножаются на s_i , как показано на диаграмме. Масштабные коэффициенты выражаются следующим образом:

$$s_i = \|F_i(z)\|_p, \quad (13.17)$$

где $F_i(z)$ — передаточная функция от входа $x(n)$ до узла $w_i(n)$, записываемая следующим образом:

$$F_i(z) = \frac{1}{1 + a_{1i} + a_{2i}z^{-2}}, \quad i = 1, 2, 3$$

Пример 13.12

Сравните масштабные множители фильтра, имеющего приведенную ниже передаточную функцию, используя три метода масштабирования.

$$H(z) = \frac{1,2916 - 0,0807z^{-1}}{1 - 0,131z^{-1} + 0,3355z^{-2}} + \frac{7,5268}{1 - 0,049z^{-1}} - 8,6788.$$

Решение

С помощью программы анализа эффектов конечной разрядности были получены следующие масштабные коэффициенты:

	L_1	L_2	L_∞
s_1	1,7345	1,0667	1,5126
s_2	1,0515	1,0012	1,0515

13.4.8. Детектирование и предотвращение переполнения на выходе

При использовании норм L_2 и L_∞ переполнение возможно, хотя и изредка. В таких случаях обычно на выходе фильтра используется арифметика насыщения. По сути, при переполнении на выходе значение устанавливается равным наибольшему положительному или отрицательному, в зависимости от знака истинной выборки данных. Данный подход оказался эффективным при обработке переполнений конечного выхода. Если выход не насыщается, то такой подход использовать нельзя, так как он может привести к нежелательным результатам, например, раздражающим звукам в цифровом аудио. Если применяется каноническое звено, к сказанному добавить нечего. Если же используется прямая реализация, и конечный выход не модифицируется, переполнение по обратной связи пойдет на умножители и затронет последующие выходные выборки.

На рис. 13.14, *a* и *b* показаны характеристики детектирования переполнения в арифметике с дополнением до двух и арифметике насыщения соответственно. На данных рисунках через y обозначен правильный выход, а через $\hat{y}$ — выход при переполнении. В современных процессорах ЦОС наблюдается тенденция к использованию в накопителе дополнительных защитных битов, позволяющих предотвратить или снизить вероятность ошибок переполнения. Например, в процессоре DSP56300 56-битовый накопитель включает 8 защитных бит. Следовательно, в 256 случаях переполнения возникнет только одна ошибка переполнения.

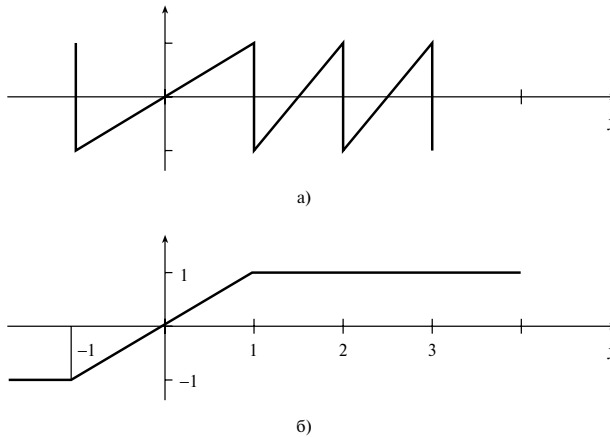


Рис. 13.14. Переполнение: *a* — характеристика переполнения; *б* — арифметика насыщения

13.4.9. Ошибки округления произведения в цифровых БИХ-фильтрах

Анализ ошибок округления произведения является обширной темой. Здесь мы будем кратки и просто расскажем о природе ошибок, их влиянии и способах снижения этого влияния.

Основные операции в БИХ-фильтрации определяются знакомым разностным уравнением второго порядка:

$$y(n) = \sum_{k=0}^2 b_k x(n-k) - \sum_{k=1}^2 a_k y(n-k), \quad (13.18)$$

где $x(n-k)$ и $y(n-k)$ — входная и выходная выборки данных, b_k и a_k — коэффициенты фильтра. На практике эти переменные часто представляются как числа с фиксированной запятой. Обычно каждое произведение $b_k x(n-k)$ или $a_k y(n-k)$ требует большего числа битов, чем каждый операнд по отдельности. Например, произведение B -битового элемента данных и B -битового коэффициента имеет размер $2B$ бит. Для рекурсивных систем, если результат не сократить, последующие вычисления приведут к неограниченному росту числа битов.

Усечение или округление используется для квантования произведения до приемлемой длины слова. Квантование произведения приводит к *ошибкам округления* в выходных данных, а следовательно, к снижению отношения сигнал-шум. Данные ошибки могут также привести к мелкомасштабным осцилляциям выхода цифрового фильтра даже при отсутствии входа.

На рис. 13.15, *a* представлена блок-схема процесса квантования произведения, а на рис. 13.15, *б* — линейная модель эффекта квантования произведения. Модель включает идеальный умножитель с бесконечной точностью, последовательно соединенный с сумматором, на который подается выборка шума $e(n)$, представляющая ошибку в

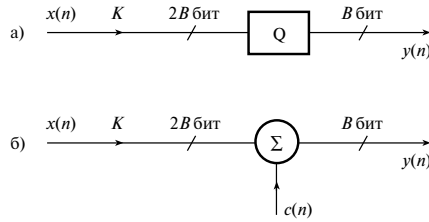


Рис. 13.15. Представление ошибки квантования произведения: *а* — блок-схема процесса квантования; *б* — линейная модель процесса квантования

квантованном произведении, причем для простоты предполагается, что $x(n)$, $y(n)$ и K представлены B бит каждый. Таким образом,

$$y(n) = Kx(n) + e(n). \quad (13.19)$$

Мощность шума, порождаемого квантованием произведения, выражается следующим образом:

$$\sigma_r^2 = \frac{q^2}{12},$$

где r обозначает ошибку округления, а q — шаг квантования, определяемый длиной слова, до которой квантуется произведение. Считается, что шум округления представляется случайной переменной с нулевым средним и постоянной дисперсией. Хотя такое предположение не всегда справедливо (например, при наличии слабых узкополосных сигналов), оно полезно при оценке производительности фильтра.

Шум округления может передаваться на последующие звенья системы ЦОС, где он усиливается, подавляется или модифицируется. Суммарный выходной шум, порожденный ошибками округления, зависит от структуры реализации. Если фильтр реализован в виде каскадной структуры, шум, порожденный одним звеном, подается на последующие звенья. Следовательно, звенья должны упорядочиваться так, чтобы минимизировать общий шум, вызванный ошибками квантования.

13.4.10. Влияние ошибок квантования на производительность фильтра

Влияние шума квантования на производительность фильтра зависит от типа используемой фильтрующей структуры и точки, в которой квантуются результаты. На рис. 13.16, *а* показана модель шума квантования для описанного ранее стандартного компоновочного блока, реализованного в прямой форме. На рисунке предполагается, что входные данные $x(n)$, выход $y(n)$ и коэффициенты фильтра представляются как B -битовые числа (включая бит знака). После умножения произведения квантуются до B бит путем округления (или усечения).

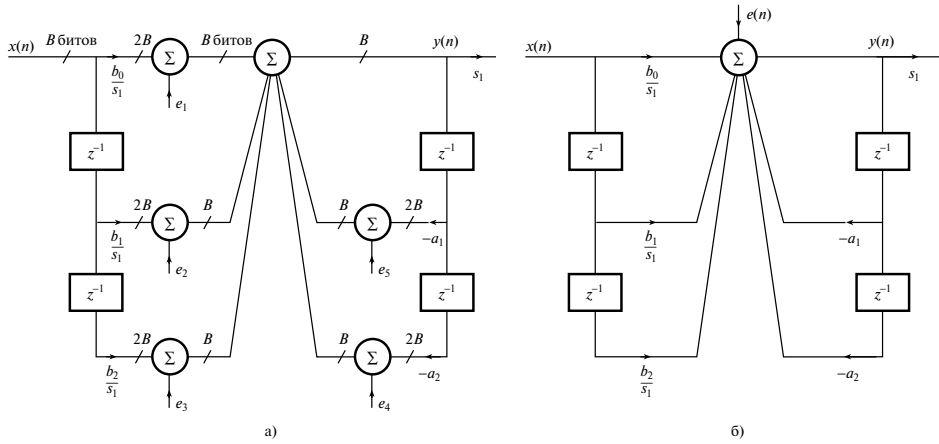


Рис. 13.16. Модель шума квантования произведения для фильтрующего звена, реализованного в прямой форме. Все источники шума, представленные на панели *а*, на панели *б* объединены, поскольку выход с них поступает в одну точку

$$\begin{aligned}\sigma_{or}^2 &= \frac{5q^2}{12} \left[\frac{1}{2\pi i} \oint_c F(z) F(z^{-1}) \frac{dz}{z} \right] s_1^2 = \\ &= \frac{5q^2}{12} \left[\sum_{k=0}^{\infty} f^2(k) \right] s_1^2 =\end{aligned}\quad (13.20)$$

$$= \frac{5q^2}{12} \|F(z)\|_2^2 s_1^2, \quad (13.21)$$

где

$$\begin{aligned}F(z) &= \frac{1}{1 + a_1 z^{-1} + a - 2z^{-2}} \\ f(k) &= Z^{-1}[F(z)] -\end{aligned}$$

результат применения обратного z -преобразования к $F(z)$. Кроме того, $f(k)$ — это импульсная характеристика каждого источника шума, дающего вклад в выход фильтра, $\|\cdot\|_2^2$ — квадрат нормы L_2 , а $q^2/12$ — мощность шума округления внутреннего произведе-

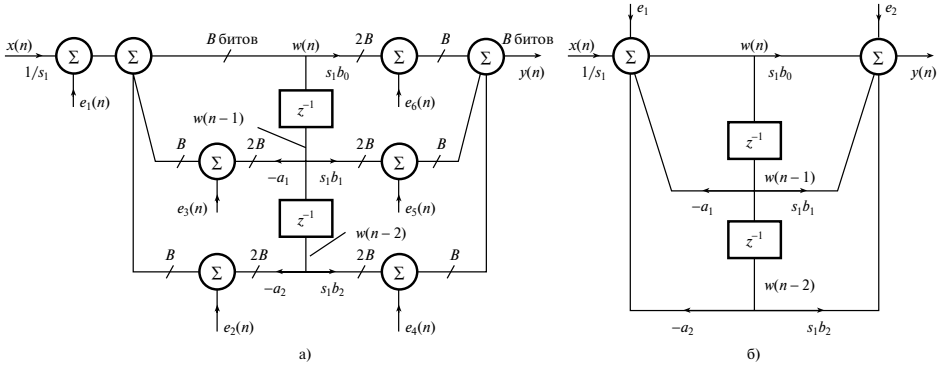


Рис. 13.17. Модель шума квантования произведения для канонического фильтрующего звена. Все источники шума, выход которых на панели *a* поступает в одну точку, на панели *б* объединены

ния. Общая мощность шума на выходе фильтра — это сумма мощности шума округления произведений и мощности шума квантования АЦП (формулы (13.8) и (13.20)):

$$\begin{aligned}
 \sigma_o^2 &= \sigma_{0A}^2 + \sigma_{or}^2 = \\
 &= \frac{q^2}{12} \left[\sum_{k=0}^{\infty} h^2(k) + 5s_1^2 \sum_{k=0}^{\infty} f^2(k) \right] = \\
 &= \frac{q^2}{12} [\|H(z)\|_2^2 + 5s_a^2 \|F(z)\|_2^2].
 \end{aligned} \tag{13.22}$$

Для канонического звена (рис. 13.17, *a*) модель шума также включает масштабный множитель, поскольку само каноническое звено вносит шум округления. Выход источников шума $e_1(n)$, $e_2(n)$ и $e_3(n)$ подается на левый сумматор, тогда как выход источников e_4 , e_5 и e_6 подается непосредственно на выход фильтра. Объединяя источники шума, выход которых подается на одну точку, получаем модель шума, изображенную на рис. 13.17, *б*. Предполагая, что источники шума не коррелируют, общий вклад шума в полезный сигнал — это просто сумма отдельных шумовых составляющих:

$$\sigma_{or}^2 = \frac{3q^2}{12} \sum_{k=0}^{\infty} f^2(k) + \frac{3q^2}{12} = \frac{3q^2}{12} [\|F(z)\|_2^2 + 1], \tag{13.23}$$

где $f(k)$ — импульсная характеристика, связывающая выход источника шума e_1 с выходом фильтра, а $F(z)$ — соответствующая передаточная функция:

$$F(z) = s_1 \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} = s_1 H(z). \tag{13.24}$$

Общий шум (шум АЦП + шум округления) на выходе фильтра записывается следующим образом:

$$\begin{aligned}\sigma_0^2 &= \sigma_{oA}^2 + \sigma_{or}^2 = \\ &= \frac{q^2}{12} \left\{ 3 \left[1 + s_1^2 \sum_{k=0}^{\infty} h^2(k) \right] + \sum_{k=0}^{\infty} h^2(k) \right\} = \\ &= \frac{q^2}{12} \{ 3[1 + s_1^2 \|H(z)\|_2^2] + \|H(z)\|_2^2 \}.\end{aligned}\quad (13.25)$$

Формулу (13.25) стоит сравнить с формулой (13.22). Отметим также, что использование масштабного множителя увеличивает выходной шум.

Пример 13.13

Фильтр характеризуется следующей передаточной функцией:

$$H(z) = \frac{0,1436 + 0,2872z^{-1} + 0,1436z^{-2}}{1 - 1,8353z^{-1} + 0,9747z^{-2}}.$$

Фильтр нужно реализовать в 8-битовой системе со входом $x(n)$, выходом $y(n)$ и коэффициентами фильтра, представленными 8-битовыми дробными величинами с фиксированной запятой в арифметике с дополнением до двух.

Предполагается, что для реализации фильтра используется каноническое звено второго порядка, и что каждое произведение (представленное 16 бит) квантуется до 8 бит непосредственно после умножения.

- 1, а. Изобразите диаграмму реализации с указанными источниками ошибок округления и определите подходящий масштабный множитель системы.
- 1, б. Оцените общую входную мощность шума в холостом состоянии и ухудшение выходного отношения сигнал-шум (в децибелах), вызванное ошибками округления.
2. Повторите п. 1 при условии, что для реализации фильтра используется прямая форма.

Решение

- 1, а. Диаграмма реализации системы с указанными источниками шума приведена на рис. 13,17, б. Источник шума e_1 представляет сумму ошибок, вызванных округлением (или усечением) произведений $a_1w(n-1)$, $a_2w(n-2)$ и $x(n)/s_1$ с 16 до 8 бит. Источник шума e_2 представляет сумму трех 16-битовых входов правого сумматора, квантованную до 8 бит.

С помощью программы анализа эффектов конечной длины слова были получены следующие масштабные множители: $s_1 = 133,89966$ (норма L_1), $s_1 = 12,1395$ (норма L_2) и $s_1 = 102,088$ (норма L_∞).

- 1, б. Для определения выходной мощности шума, порожденной ошибками округления, использовалась программа анализа эффектов конечной разрядности (за основу взята формула (13.23)):

$$\sigma_{or}^2 = 1668,03q^2,$$

где предполагается использование нормы L_2 .

Из уравнения (13.8) выходной шум, порожденный АЦП, равен $3,7724q^2$. Следовательно, общая мощность выходного шума равна:

$$\sigma_o^2 = (1668,03 + 3,7724)q^2 = 1671,8024q^2.$$

Выходная мощность сигнала (на входе предполагается случайный сигнал) равна

$$\sigma_x^2 = \frac{1}{3} \|H(z)\|_2^2 = 15,0896.$$

Отношение сигнал-шум (без ошибок округления) равно:

$$\frac{15,0896}{3,7724q^2} = \frac{4}{q^2}.$$

Отношение сигнал-шум (с ошибкой округления) равно:

$$\frac{15,0896}{1671,8024q^2} = \frac{9,0260 \times 10^{-3}}{q^2}.$$

Ухудшение отношения сигнал-шум вследствие ошибки округления равно:

$$10 \lg \left(\frac{4/q^2}{9,0260 \times 10^{-3}/q^2} \right) = 26,47 \text{ дБ}.$$

2. При реализации в прямой форме общая выходная мощность шума, порожденного ошибкой округления, равна $9048,82q^2$. Ухудшение отношения сигнал-шум вследствие ошибки округления составляет 33,8 дБ при масштабировании с нормой L_2 . Без масштабирования ухудшение отношения сигнал-шум составляет порядка 1,11 дБ. Такое небольшое снижение отношения сигнал-шум при отсутствии масштабирования объясняет, почему в некоторых приложениях прямая реализация выгоднее — она позволяет избежать масштабирования [5].

13.4.11. Шум округления в каскадных и параллельных реализациях

13.4.11.1. Каскадная структура

На рис. 13.18 показана каскадная реализация БИХ-фильтра шестого порядка с использованием канонических звеньев второго порядка, где для простоты источники шума, выход с которых поступает в одну точку, объединены (см. выше) и перенумерованы. Так, e_1 — это сумма трех источников шума, поступающего с трех умножителей на крайний слева сумматор. Шум составного источника e_1 проходит через три звена фильтра $H_1(z)$, $H_2(z)$ и $H_3(z)$. Шум составного источника e_2 вносит вклад в две передаточные функции $H_2(z)$ и $H_3(z)$, и т.д.

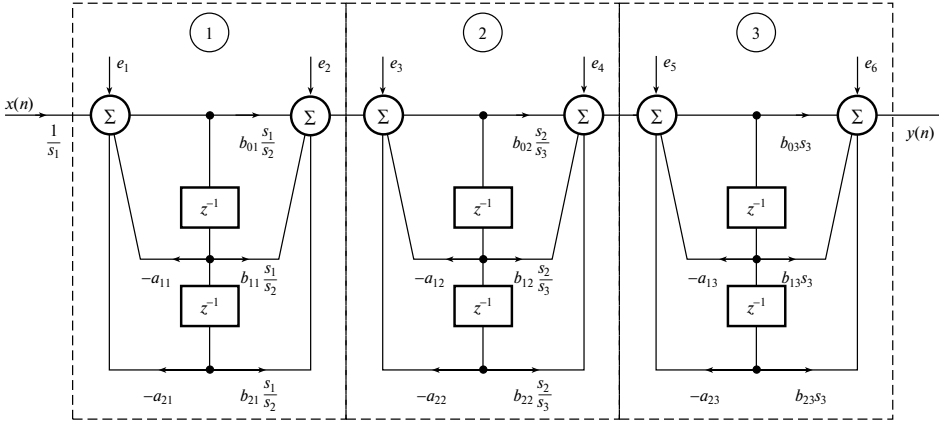


Рис. 13.18. Шумовая модель каскадной реализации БИХ-фильтра шестого порядка

Общий выходной шум, возникший вследствие ошибок округления, представляет собой сумму шести источников шума:

$$\begin{aligned}
 \sigma_{or}^2 &= \frac{3q^2}{12} \sum_{k=0}^{\infty} f_1^2(k) + \frac{3q^2}{12} \sum_{k=0}^{\infty} f_2^2(k) + \frac{2q^2}{12} \sum_{k=0}^{\infty} f_3^2(k) + \frac{3q^2}{12} \sum_{k=0}^{\infty} f_4^2(k) + \\
 &+ \frac{2q^2}{12} \sum_{k=0}^{\infty} f_5^2(k) + \frac{3q^2}{12} = \\
 &= \frac{q^2}{12} \left[3 \sum_{k=0}^{\infty} f_1^2(k) + 5 \sum_{k=0}^{\infty} f_3^2(k) + 3 \sum_{k=0}^{\infty} f_5^2(k) + 3 \right] = \\
 &= \frac{q^2}{12} [3\|F_1(z)\|_2^2 + 5\|F_3(z)\|_2^2 + 5\|F_1(z)\|_2^2 + 3],
 \end{aligned} \tag{13.26}$$

где $f_i(k)$ — импульсная характеристика, связывающая источник шума e_i и выход. Шумовые компоненты, порожденные источниками e_2 и e_3 (рис. 13.18), проходят через одни звенья фильтра, т.е. множатся на передаточные функции $H_2(z)$ и $H_3(z)$, поэтому их выходы объединены. То же относится и к шумовым компонентам e_4 и e_5 .

13.4.11.2. Параллельная структура

На рис. 13.19 представлена модель шума округления для параллельной реализации фильтра шестого порядка. Как и ранее, источники шума, возникающего при квантовании отдельных произведений, объединены. Выходы источников шума e_1 , e_2 и e_3 проходят через фильтрующее звено и подаются на выход, тогда как выход остальных источников шума поступает непосредственно на выход фильтра. Вклад источников e_1 , e_2 и e_3 в выходной шум записывается следующим образом:

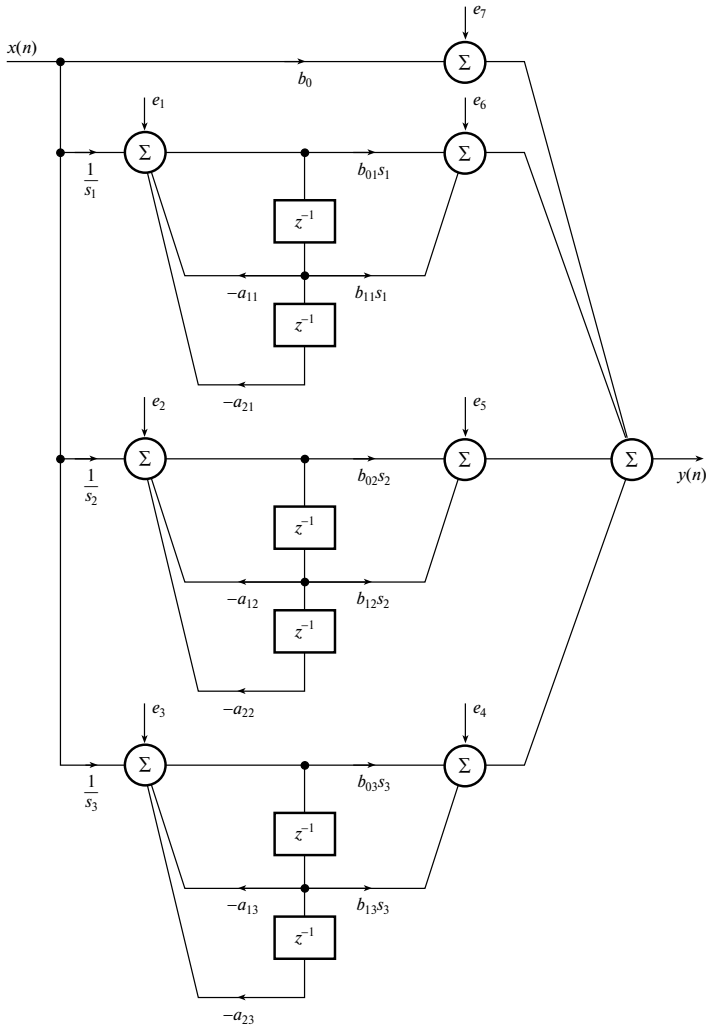


Рис. 13.19. Шумовая модель каскадной реализации БИХ-фильтра шестого порядка

$$\begin{aligned}
\sigma_{r,i}^2 &= \frac{3q^2}{12} \sum_{k=0}^{\infty} f_i^2(k) = \frac{3q^2}{12} \|F_i(z)\|_2^2, \quad i = 1, 2, 3 \\
&= \frac{3q^2}{12} s_i^2 \sum_{k=0}^{\infty} h_i^2(k) = \frac{3q^2}{12} s_i^2 \|H_i(z)\|_2^2,
\end{aligned} \tag{13.27}$$

где $H_i(z)$ и $h_i(k)$ — передаточная функция i -го звена и соответствующая импульсная характеристика. $F_i(z)$ и $f_i(k)$ — передаточная функция i -го источника шума и соответствующая импульсная характеристика.

Выход источников шума e_i , $i = 4, 5, 6, 7$ передается непосредственно на выход. Следовательно, общая шумовая мощность записывается так:

$$\begin{aligned}
\sigma_{or}^2 &= \frac{q^2}{12} \left\{ 7 + 3 \sum_{i=1}^3 \left[s_i^2 \sum_{k=0}^{\infty} h_i^2(k) \right] \right\} = \\
&= \frac{q^2}{12} \left\{ 7 + 3 \sum_{i=1}^3 [s_i^2 \|H_i(z)\|_2^2] \right\}.
\end{aligned} \tag{13.28}$$

Вообще, при параллельной реализации с L звеньями выходная мощность шума, порожденного ошибкой округления, записывается следующим образом:

$$\sigma_{or}^2 = \frac{q^2}{12} \left[2L + 1 + 3 \sum_{i=1}^3 s_i^2 \|H_i(z)\|_2^2 \right]. \tag{13.29}$$

Используя простую компьютерную программу (например, описанную ранее программу анализа эффектов конечной разрядности), можно легко вычислить мощность шума округления на основе приведенных выше соотношений.

Пример 13.14

Используя приведенную передаточную функцию БИХ-фильтра четвертого порядка [12], определите снижение отношения сигнал-шум (в децибелах), если фильтр реализован:

- как каскад двух звеньев второго порядка, упорядоченных так, как следует из функции $H(z)$;
- как параллельная комбинация двух звеньев второго порядка.

$$H(z) = \frac{1 - 2z^{-1} + z^{-2}}{1 + 0,777z^{-1} + 0,3434z^{-2}} \frac{1 - 0,707z^{-1} + z^{-2}}{1 + 0,01877z^{-1} + 0,801z^{-2}} 0,093226.$$

Решение

Модели шума для каскадной и параллельной структур идентичны представленным на рис. 13.18 и 13.19 соответственно, если на рисунках отбросить третий каскад.

1. При каскадной реализации масштабные множители L_1 равны

$$\begin{aligned}s_1 &= 2,395746, \\ s_2 &= 15,703627.\end{aligned}$$

Мощность шума округления, мощность шума АЦП и мощность сигнала на выходе фильтра равны

$$\begin{aligned}\sigma_{or}^2 &= 345,0391q^2, \\ \sigma_{oA}^2 &= 0,03976q^2, \\ \sigma_y^2 &= 0,1591q^2.\end{aligned}$$

Отношение сигнал-шум без ошибки округления равно

$$\frac{0,1591}{0,0397q^2},$$

а отношение сигнал-шум с ошибкой округления —

$$\frac{0,1591}{345,0789}.$$

Снижение отношения сигнал-шум вследствие ошибки округления равно

$$10\lg(345,0789/0,03976) \approx 39,4 \text{ дБ}.$$

2. Используя для параллельной реализации разложение на простые дроби [12], получаем

$$H(z) = 0,093326 \left(1 + \frac{-5,162 + 0,7867z^{-1}}{1 + 0,777z^{-1} + 0,3434z^{-2}} + \frac{1,65736 + 0,2759z^{-1}}{1 + 0,01877z^{-1} + 0,801z^{-2}} \right).$$

Мощность шума округления на выходе фильтра выражается так:

$$\sigma_{or}^2 \frac{q^2}{12} \left[5 + 3 \sum_{i=1}^2 s_i^2 \|H_i(z)\|_2^2 \right].$$

Используя программу анализа эффектов конечной длины слова, получаем масштабные множители (норма L_2) $s_1 = 2,395746$, $s_2 = 5,450612$, $\|H_1(z)\|_2^2 = (7,378492)^2$, $\|H_2(z)\|_2^2 = (2,801937)^2$. Следовательно,

$$\begin{aligned}\sigma_{or}^2 &= \frac{q^2}{12} \{ 5 + 3[(2,395746)^2 (7,378492)^2 + (5,450612)^2 (2,801937)^2] \} = \\ &= 136,85q^2.\end{aligned}$$

Таким образом, если выходной сигнал и шум АЦП те же, что и в предыдущем случае, на выходе фильтра получаем такое отношение сигнал-шум с ошибкой округления:

$$0,1591/(136,85 + 0,03976)q^2 = 1,16228 \times 10^{-3}/q^2.$$

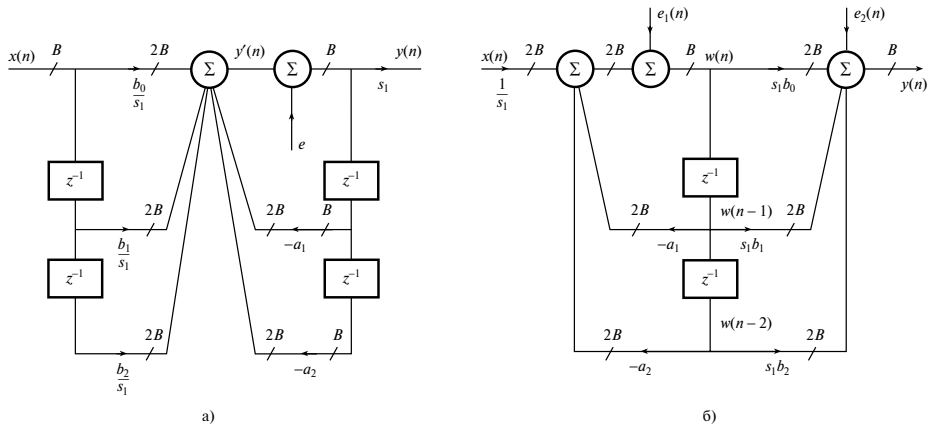


Рис. 13.20. Модели шума для звеньев БИХ-фильтров в современной системе ЦОС. Показаны длины слов в различных точках фильтра. Предполагается, что входные данные и коэффициенты имеют размер по B бит

Снижение отношения сигнал-шум, вызванное ошибкой округления, равно

$$10 \lg [(136,85 + 0,03976)/0,03976] = 35,37 \text{ дБ.}$$

13.4.12. Влияние шума округления произведения в современных системах ЦОС

Ранние работы по влиянию ошибок округления произведения на производительность фильтра основывались на единой, фиксированной внутренней длине слова и преимущественно каждое $2B$ -битовое произведение (строго говоря, $(2B - 1)$ -битовое) перед суммированием квантовалось до прежних B бит. В современных процессорах ЦОС подобных условий нет, поскольку процессоры поддерживают накопление в регистрах двойной длины. Все современные процессоры ЦОС имеют по крайней мере встроенный умножитель 16×16 бит и 32-битовый регистр произведений и позволяют накапливать произведения как 32-битовые числа — используется так называемая *16/32-битовая архитектура*.

На рис. 13.20, а показана модель шума для звена второго порядка, реализованного в прямой форме, где квантование выполняется после сложения произведений. На данном рисунке $2B$ -битовая сумма произведений $y'(n)$ квантуется до B бит. Чтобы отличить данную схему от метода с отдельным квантованием каждого произведения, будем называть ее *квантованием после накопления* (post-accumulation quantization). Очевидно, что в данном случае после квантования остается только один источник шума, и выходная мощность шума записывается как

$$\sigma_{or}^2 = \frac{q^2}{12} \|F(z)\|_2^2, \quad (13.30)$$

где

$$F(z) = \frac{1}{1 + a_1 z^{-1} + a_2 z^{-2}}.$$

При использовании канонических звеньев второго порядка (рис. 13.20, б) выходной шум, вызванный источниками шума, равен

$$\sigma_{or}^2 = \frac{q^2}{12} [s_1^2 \|H(z)\|_2^2 + 1]. \quad (13.31)$$

Очевидно, что округление после накопления произведений (формулы (13.30) и (13.31)) приводит к значительному снижению шума округления по сравнению с округлением после каждого умножения.

13.4.13. Схемы снижения шума округления

На практике округление или усечение в некоторой точке внутри фильтра необходимо, чтобы приспособить его к умножителям, ячейкам памяти и интерфейсам по фактору длины слова. Ошибки округления произведений, вызванные округлением или усечением произведений, могут привести к заметному искажению выхода фильтра с малыми сигналами, и при создании систем высокой точности их следует минимизировать. Придумано несколько схем снижения или устранения эффектов ошибок округления в БИХ-фильтрах. Эти схемы так эффективно формируют спектр шума, что шум в определенных полосах фильтра снижается или устраняется. В сумме подобные схемы называют *формированием спектра ошибки* (Error Spectral Shaping — ESS).

Введем основные принципы снижения шума округления с использованием фильтрующего звена второго порядка I (рис. 13.21, а). На данном рисунке параметры фильтра (коэффициенты и данные) представлены как B -битовые числа с фиксированной запятой, а накопитель имеет размер $2B$ бит (т.е. используется реализация “ $B/2B$ бит”). В современных процессорах ЦОС B обычно равно 16 или 24 бит. При квантовании выхода накопителя до B бит создается шум округления. Можно показать, что преобразование квантованного выхода записывается как

$$Y(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} X(z) - \frac{1}{1 + a_1 z^{-1} + a_2 z^{-2}} E(z). \quad (13.32)$$

Видно, что спектр квантованного выхода равен спектру идеального выхода плюс масштабированный спектр ошибки. Спектр ошибки усиливается полюсами фильтра (корни полинома $1 + a_1 z^{-1} + a_2 z^{-2}$) вне зависимости от характеристик фильтра. В зависимости от типа фильтра шум может усиливаться в нижней средней или верхней части частотного диапазона.

Схема обратной связи по ошибке первого порядка, используемая для снижения влияния шума округления, изображена на рис. 13.21, б. Можно показать (см. пример 13.16), что в этом случае выходное преобразование звена фильтра с обратной связью по ошибке записывается так:

$$Y(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} X(z) - \frac{1 - k z^{-1}}{1 + a_1 z^{-1} + a_2 z^{-2}} E(z). \quad (13.33)$$

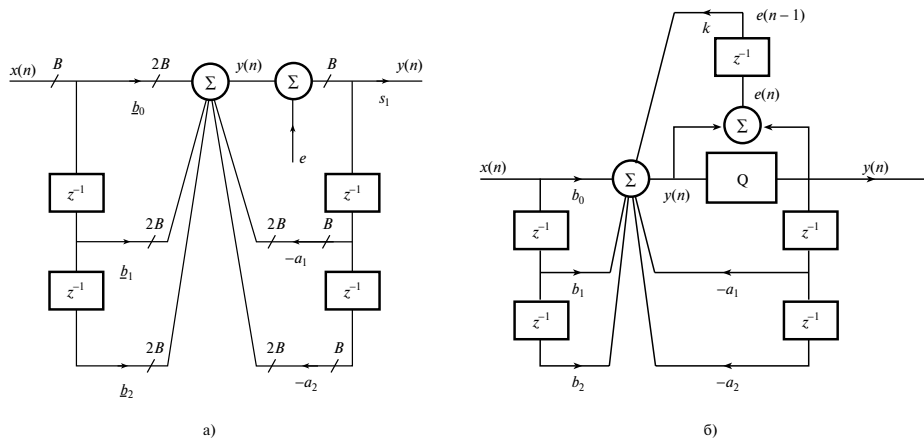
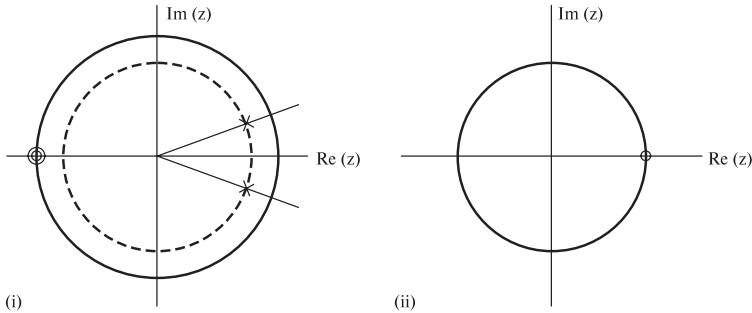


Рис. 13.21. Шум округления в звене второго порядка прямой формы I: *a* — длины слов в различных точках фильтра; *б* — схема снижения шума первого порядка

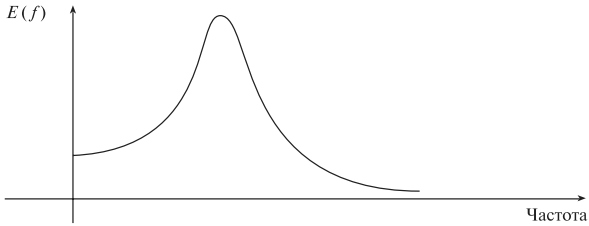
Коэффициент обратной связи k эффективно вводит нуль на пути ошибки, компенсируя таким образом влияние полюсов фильтра. Влияние нуля с обратной связью по ошибке на спектр ошибки демонстрируется на рис. 13.22. Очевидно, что без обратной связи по ошибке спектр ошибки усиливается полюсами фильтра. При использовании обратной связи по ошибке на пути ошибки вводится нуль, что приводит к уменьшению спектра ошибки. Чтобы компенсировать влияние полюсов, нуль в цепи обратной связи обычно помещают как можно ближе к частоте полюса.

Чтобы максимально сократить спектр шума, можно использовать схему второго порядка, изображенную на рис. 13.23.

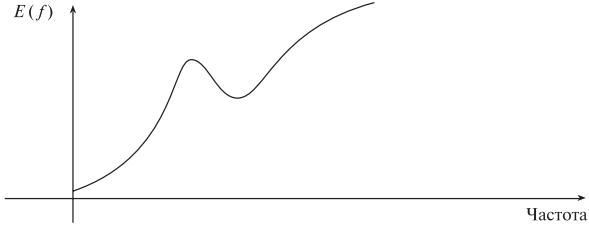
Обобщенные стратегии снижения шума для звена в прямой форме и канонического звена изображены на рис. 13.24, *a* и *б* соответственно. На обоих рисунках коэффициенты прямой и обратной связи a'_i и b'_i используются для изменения передаточной функции, искаженной ошибками округления, так что выходной шум округления минимизируется, не затрагивая полезный сигнал. На рис. 13.24, *a* квантование суммы произведений на выходе левого сумматора порождает ошибку $e_1(n)$, которая представляет собой нижнюю половину переменной двойной точности $y(n)$. Произведения $a'_1 e_1(n-1)$ и $a'_2 e_2(n-2)$ не имеют такого веса, как другие выходы сумматора, хотя все они имеют длину $2B + 1$ бит, так что их следует выровнять или проквантовать перед сложением с другими входами. Ошибка квантования в этом случае обозначается через $e_2(n)$. Подобным образом члены $b'_0 e(n)/s_1$, $b'_1 e(n-1)/s$ и $b'_2 e(n-2)/s_1$ нужно проквантовать перед суммированием с другими входами правого сумматора, что сводится к ошибке $e_3(n)$. Наконец, выход правого сумматора квантуется с $2B + 1$ бит до $B + 1$ бит, что дает ошибку $e_4(n)$. Подобные соображения можно привести и для канонического звена, изображенного на рис. 13.24, *б*.



а) диаграмма нулей и полюсов (i) фильтра, (ii) сети с обратной связью по ошибке при $k = 1$



б) спектр ошибки при $k = 0$



в) спектр ошибки при $k = 1$

Рис. 13.22. Иллюстрация влияния коэффициента обратной связи. Обратите внимание на то, что положение нуля в цепи обратной связи близко к положению полюсов фильтра

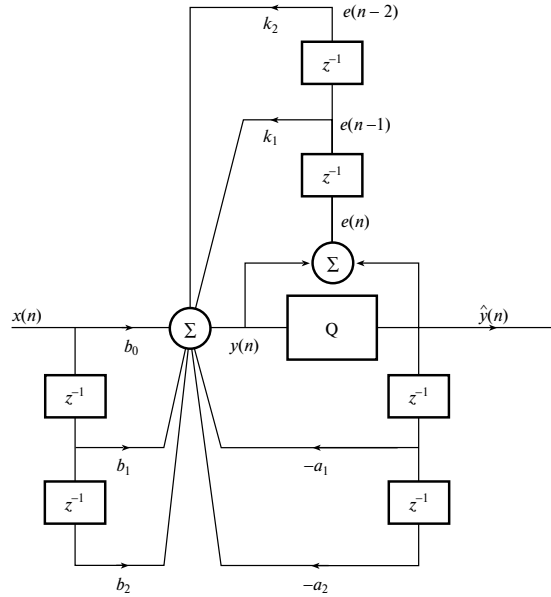


Рис. 13.23. Схема снижения шума второго порядка

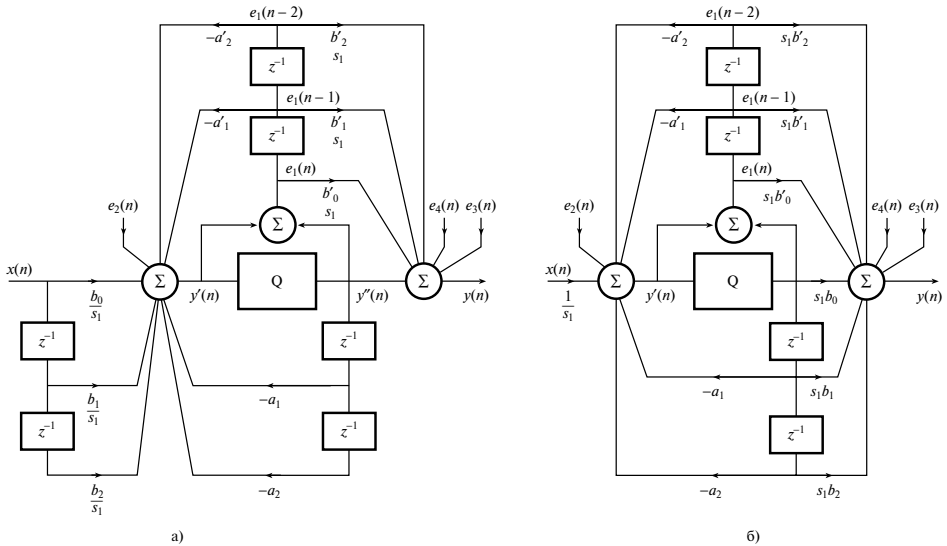


Рис. 13.24. Обобщенные схемы снижения шума: *a* — фильтр, реализованный в прямой форме; *б* — канонический фильтр

При реализации в прямой форме со схемой ESS (рис. 13.24, *а*) выходной шум выражается следующим образом:

$$\sigma_{or}^2 = \frac{q^2}{12} \left[\sum_{k=0}^{\infty} f_1^2(k) + \sum_{k=0}^{\infty} f_2^2(k) + 2s_1^2 \right], \quad (13.34)$$

где $f_1(k)$ и $f_2(k)$ — импульсная характеристика, связывающая источники шума 1 и 2 с выходом фильтра.

Для канонического звена (рис. 13.24, *б*) выходная мощность шума, порожденного округлением, записывается так:

$$\begin{aligned} \sigma_{or}^2 &= \frac{q^2}{12} \left[\sum_{k=0}^{\infty} f_1^2(k) + \sum_{k=0}^{\infty} f_2^2(k) + 2 \right] = \\ &= \frac{q^2}{12} [\|F_1(z)\|_2^2 + \|F - 2(z)\|_2^2], \end{aligned} \quad (13.35)$$

где

$$\begin{aligned} F_1(z) &= (b'_0 + b'_1 z^{-1} + b'_2 z^{-2})s_1 + \frac{(1 + a'_1 z^{-2})(b_0 + b_1 z^{-1} + b_2 z^{-2})s_1}{1 + a_1 z^{-1} + a_2 z^{-2}} \\ F_2(z) &= \frac{(b_0 + b_1 z^{-1} + b_2 z^{-2})s_1}{1 + a_1 z^{-1} + a_2 z^{-2}}. \end{aligned}$$

Выбор коэффициентов формирования спектра ошибки a'_i и b'_i определяет эффективность схемы по снижению шума. На практике обычно коэффициенты ESS выбираются целыми, чтобы избежать дальнейшего квантования, и используются схемы ESS первого и второго порядка.

В схемах снижения шума первого порядка коэффициенты прямой связи по ошибке устанавливаются равными нулю, а коэффициенты обратной связи по ошибке — равными целым числам. Схемы снижения шума первого порядка частично помогают снизить шум округления в узкополосных фильтрах нижних и верхних частот, поскольку позволяют помещать на пути ошибки округления единственный нуль либо в нижней, либо в верхней области частотного диапазона. Схема привлекательна тем, что она лишь незначительно увеличивает вычислительную сложность фильтра.

Оптимум ESS достигается при использовании схем второго порядка, в которых эффект шума округления на выходе фильтра полностью устраняется. Для звена, реализованного в прямой форме (рис. 13.24, *а*), оптимальная схема ESS записывается следующим образом:

$$b'_i = 0, i = 0, 1, 2; \quad a'_i = a_i, i = 1, 2. \quad (13.36)$$

В данном случае выходной шум округления снижается до уровня, не превышающего внутреннего шума округления. Выходная мощность шума округления снижается до величины

$$\sigma_{or}^2 = \frac{q^2}{12} \left[1 + \sum_{k=0}^{\infty} f^2(k) \right] = \quad (13.37)$$

$$= \frac{q^2}{12} [1 + \|F(z)\|_2^2],$$

где

$$F(z) = \frac{s_1}{1 + a_1 z^{-1} + a_2 z^{-2}}.$$

Для канонического звена следует использовать такие значения:

$$b'_i = -b_i, i = 0, 1, 2; \quad a'_i = a_i, i = 1, 2. \quad (13.38)$$

Выходная мощность шума записывается следующим образом:

$$\sigma_{or}^2 = \frac{q^4}{12} \sum_{k=0}^{\infty} f^2(k) = \frac{q^4}{12} \|F(z)\|_2^2,$$

где

$$F(z) = \frac{(b_0 + b_1 z^{-1} + b_2 z^{-2}) s_1}{1 + a_1 z^{-1} + a_2 z^{-2}}.$$

Оптимальное решение вычислительно дороже и, как отмечается в работе [13], включает представление с двойной точностью внутренних переменных фильтра. Помимо решения с целыми числами, рассмотренного выше, существует несколько других субоптимальных решений (см., например, [7]).

Пример 13.15

Сравните характеристики шума округления БИХ-фильтра второго порядка, который характеризуется передаточной функцией

$$H(z) = \frac{0,1436(1 + 2z^{-1} + z^{-2})}{1 - 1,8353z^{-1} + 0,9748z^{-2}},$$

если фильтр реализован а) как каноническое звено; б) в прямой форме для следующих случаев:

- 1) $a'_i = 0, i = 1, 2$ (без обратной связи по ошибке);
- 2) $a'_1 = -1, a'_2 = 0$;
- 3) $a'_1 = -2, a'_2 = 0$;
- 4) $a'_1 = -1, a'_2 = 1$;
- 5) $a'_1 = -2, a'_2 = 1$.

Предполагайте, что все коэффициенты прямой связи по ошибке во всех случаях равны нулю.

Таблица 13.5. Выход программы анализа эффектов конечной разрядности (пример 13.15)

№	<i>Мощность шума</i>	
	Каноническое звено	Реализация в прямой форме
1	$556,0108q^2$	$556,0108q^2$
2	$77,1261q^2$	$78,6247q^2$
3	$710,0842q^2$	$713,0933q^2$
4	$414,1014q^2$	$413,845q^2$
5	$12,2659q^2$	$14,9999q^2$

Решение

Структуры реализации фильтров со схемами ESS показаны на рис. 13.25. Выходная мощность шума округления для канонической структуры и структуры с реализацией в прямой форме равны соответственно:

$$\sigma_{or}^2 = \frac{q^2}{12} [||F_1(z)||_2^2 + 1]$$

и

$$\sigma_{or}^2 = \frac{q^2}{12} [||F_2(z)||_2^2 + 1],$$

где

$$F_1(z) = (1 + a'_1 z^{-1} + a'_2 z^{-2}) \frac{(b_0 + b_1 z^{-1} + b_2 z^{-2})s_1}{1 + a_1 z^{-1} + a_2 z^{-2}}, \quad s_1 = 12,1395 \text{ (масштабирование } L_2)$$

$$F_2(z) = \frac{(1 + a'_1 z^{-1} + a'_2 z^{-2})s_1}{1 + a_1 z^{-1} + a_2 z^{-2}}, \quad s_1 = 6,7282 \text{ (масштабирование } L_2).$$

С помощью программы анализа эффектов конечной разрядности была найдена выходная мощность шума в каждом случае, результаты сведены в табл. 13.5. Отметим, что в ситуации 3 ($a'_1 = -2, a'_2 = 0$) выход шума на самом деле увеличился, а не уменьшился, откуда следует, что коэффициенты нужно выбирать аккуратно. Отметим также, что эффективно снизить выходной шум позволяет схема первого порядка, предложенная в п. 2.

13.4.14. Определение практических значений коэффициентов обратной связи по ошибке

Из сказанного выше очевидно, что на спектр ошибки $E(z)$ влияют полюса фильтра. По сути, спектр ошибки усиливается полюсами фильтра. Если предположить, что ошибка имеет плоский спектр, то на выходе фильтра шум будет усилен около частоты полюса. Коэффициенты обратной связи по ошибке компенсируют усиление спектра ошибки, для чего на пути шума вводится один или несколько нулей. В цепи обратной связи по ошибке первого порядка вводится один ноль в числитель передаточной функции ошибки. В цепи второго порядка коэффициенты обратной связи вводят два нуля в передаточную функцию шума. В обоих случаях нули не влияют на выход фильтра.

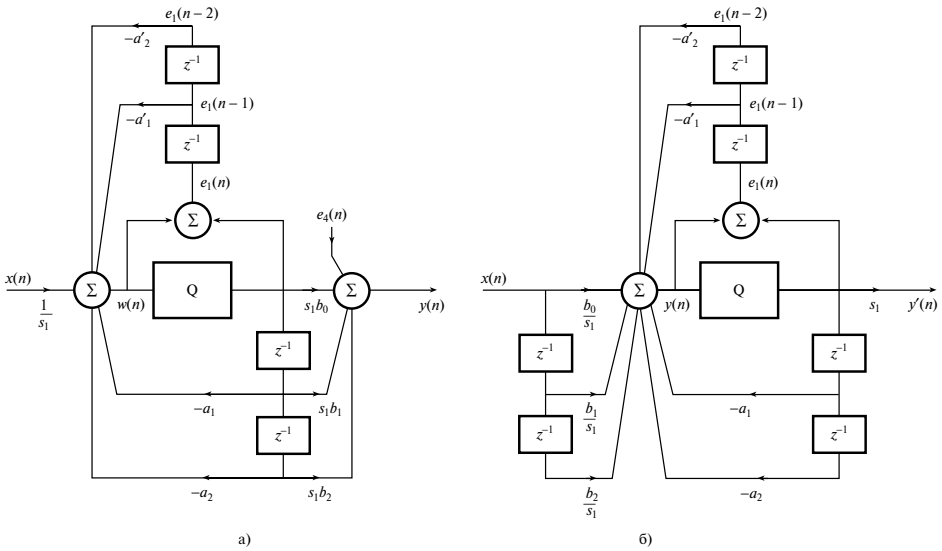


Рис. 13.25. Две различные структуры реализации БИХ-фильтра второго порядка

Простая, но эффективная стратегия заключается в размещении нулей обратной связи максимально близко к частотам полюсов, чтобы противодействовать влиянию полюсов (см. рис. 13.21). На практике в число факторов, влияющих на выбор коэффициентов обратной связи по ошибке, входят желание избежать дальнейших ошибок округления, использование двойной точности, желание упростить умножители и необходимость расположить нули цепи обратной связи по ошибке максимально близко к полюсам фильтра, чтобы компенсировать влияние шума округления. По этим причинам значения коэффициентов обратной связи по ошибке k_1 и k_2 часто выбираются из простых целых чисел $0, \pm 1, \pm 2$.

Чтобы коэффициенты обратной связи были простыми числами, нули цепи обратной связи должны располагаться в точках $0, \pm 60^\circ, \pm 90^\circ, \pm 120^\circ, \pm 180^\circ$, в зависимости от значений k_1 и k_2 . Возможные значения коэффициентов обратной связи по ошибке и положения соответствующих нулей сведены в табл. 13.6.

Таблица 13.6. Целые коэффициенты обратной связи по ошибке и соответствующие положения нулей

Номер варианта	Значения k_1	Значения k_2	Положение нулей
1	0	1	Пара нулей на 0° и 180° (т.е. на постоянной составляющей и $F_s/2$)
2	0	-1	Пара комплексно-сопряженных нулей на $\pm 90^\circ$ (т.е. на $\pm F_s/4$)
3	1	0	Один нуль на 0° (т.е. на постоянной составляющей)
4	1	-1	Пара комплексно-сопряженных нулей на $\pm 60^\circ$ (т.е. на $\pm F_s/6$)
5	2	-1	Двойной нуль на 0° (т.е. на постоянной составляющей)
6	-2	-1	Двойной нуль на 180° (т.е. на $F_s/2$)
7	-1	-1	Пара комплексно-сопряженных нулей на $\pm 120^\circ$ (т.е. на $\pm F_s/3$)
8	-1	0	Один нуль на 180° (т.е. на $F_s/2$)

Выбор коэффициентов обратной связи по ошибке зависит также от типа фильтра. Например, фильтр нижних частот будет иметь полюса на постоянной составляющей или возле нее. Таким образом, в таблице показано, что значения k_1 и k_2 можно выбирать лишь из позиций 1, 3, 4 или 5, поскольку только так можно получить нули, близкие к полюсам фильтра. С другой стороны, фильтр верхних частот имеет полюса вблизи верхней границы полосы частот доступного спектра (т.е. возле $F_s/2$). Следовательно, в этом случае значения k_1 и k_2 можно выбирать из вариантов 1, 6, 7 или 8.

Пример 13.16

1. Используя необходимые диаграммы, обсудите проблемы шума округления в цифровых БИХ-фильтрах с фиксированной запятой. В ответе нужно осветить следующие моменты:
 - а) возникновение шума округления в БИХ-фильтрах;
 - б) влияние шума округления на производительность БИХ-фильтров.
2. На рис. 13.26 показана структура звена фильтра второго порядка со схемой обратной связи по ошибке. Предположим, что данное звено нужно реализовать с использованием арифметики с фиксированной запятой с дополнением до двух, причем квантование выполняется после сложения произведений.
 - А. Выведите выражение для преобразования квантованного выхода $\hat{Y}(z)$, записанное через преобразование входа $X(z)$ и ошибку квантования $E(z)$, показав таким образом, что цепь обратной связи по ошибке не оказывает неблагоприятного воздействия на входной сигнал.
 - Б. Выведите выражение для функции обратной связи по ошибке.
 - В. Какие ключевые факторы на практике влияют на выбор значений коэффициентов обратной связи по ошибке?
3. Из анализа положений полюсов и нулей найдите подходящие пары целых значений коэффициентов обратной связи по ошибке, при которых минимизируется уровень шума на выходе указанных ниже фильтров:
 - а) $H(z) = \frac{1+2z^{-1}+z^{-2}}{1-1,75z^{-1}+0,81z^{-2}}$;
 - б) $H(z) = \frac{1-2z^{-1}+z^{-2}}{1+1,75z^{-1}+0,81z^{-2}}$;
 - в) $H(z) = \frac{1-z^{-2}}{1+0,81z^{-2}}$.

Допустим, что все фильтры реализованы с использованием структуры, изображенной на рис. 13.26, и что корни полинома второго порядка

$$1 + d_1 z^{-1} + d_2 z^{-2}$$

равны $(r, \angle\theta)$ и $(r, \angle-\theta)$, где

$$r = \sqrt{d_2} \text{ и } \theta = \arccos\left(\frac{-d_1}{2r}\right).$$

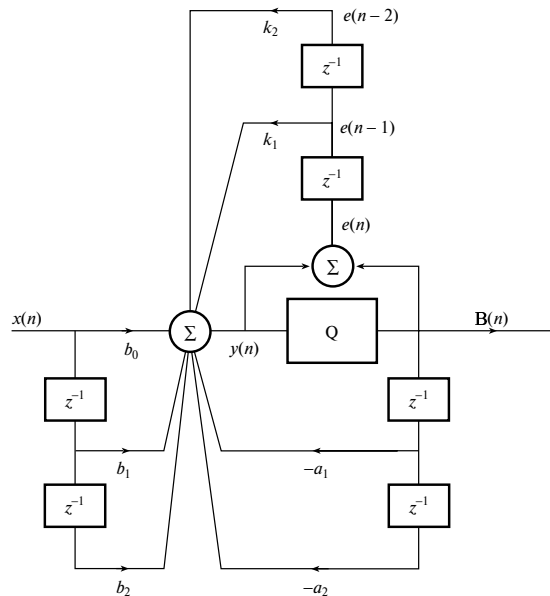


Рис. 13.26. Схема снижения шума второго порядка — выбор коэффициентов обратной связи по ошибке

Решение

1. Шум округления проистекает от квантования произведений (округления или усечения) и/или квантования суммы произведений, необходимого в рекурсивной реализации, чтобы удерживать переменные согласованными с разрешенной длиной слова. Например, умножение двух чисел, представленных B бит каждое, дает результат размером $2B$ бит. Если результат не проквантовать до (например) B бит, длина слова последующих результатов будет расти бесконечно. Ошибки квантования усиливаются полюсами фильтра и поступают на выход в виде шума. Данный шум увеличивает общий собственный шум системы. При этом искажаются сигналы низкого уровня, и в приложениях, требующих высокоточного воспроизведения, это может быть неприемлемо. Ошибки округления могут также привести к мелкомасштабным осцилляциям на выходе фильтра даже при отсутствии входа. Для иллюстрации ответа можно привести диаграммы топологий фильтра (например, каноническое звено второго порядка) и модели шума округления.
- 2.А. С помощью рис. 13.26 находим, что ошибка округления $e(n)$ и квантованный и неквантованный выходы фильтра $y'(n)$ и $y(n)$ передаются следующим образом:

$$e(n) = y(n) - \hat{y}(n)$$

$$y(n) = \sum_{i=0}^2 b_i x(n-i) - \sum_{i=1}^2 a_i \hat{y}(n-i) + \sum_{i=1}^2 k_i e(n-i) \quad (13.39)$$

Используя формулы (13.39, а и б), применяя z -преобразования и упрощая результат, получаем искомое уравнение:

$$\hat{Y}(z) = \left(\frac{\sum_{i=0}^2 b_i z^{-i}}{1 + \sum_{i=1}^2 a_i z^{-i}} \right) X(z) - \left(\frac{1 - \sum_{i=1}^2 k_i z^{-i}}{1 + \sum_{i=1}^2 a_i z^{-i}} \right) E(z).$$

- 2.Б. Передаточную функцию шума можно получить, положив вход равным нулю. Из уравнения (13.39, в) следует:

$$H_e(z) = \frac{1 - \sum_{i=1}^2 k_i z^{-i}}{1 + \sum_{i=1}^2 a_i z^{-i}} = \frac{1 - k_1 z^{-1} - k_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}}.$$

- 2.В. Основными факторами являются необходимость избежать: дальнейших ошибок округления или использования двойной точности, дополнительного умножителя, необходимость расположить нули цепи обратной связи по ошибке максимально близко к полюсам фильтра, чтобы компенсировать их влияние на шум округления. По этим причинам значения коэффициентов обратной связи по ошибке k_1 и k_2 выбираются из простых целых чисел $0, \pm 1, \pm 2$.
- 3.А. Используя выражение для корней полинома, находим, что полюса фильтра расположены на z -плоскости с координатами $r = \sqrt{0,81} = 0,9$ и $\theta = \pm \arccos(1,75/2 \times 0,9) = \pm 13,5^\circ$; двойной нуль имеет координаты $r = 1$ и $\theta = 180^\circ$ (т.е. на частоте $F_s/2$). В результате получаем фильтр нижних частот. Коэффициенты обратной связи по ошибке должны давать двойной нуль на $r = 1$ и $\theta = 0$, чтобы компенсировать влияние полюсов на шум округления, так что имеем следующие значения: $k_1 = 2$ и $k_2 = -1$. Можно также использовать коэффициенты обратной связи по ошибке первого порядка со значениями $k_1 = 1$ и $k_2 = 0$ (чтобы получить нуль на постоянной составляющей).
- 3.Б. У второго фильтра полюса расположены в точке с координатами $r = 0,9$ и $\theta = \pm \arccos(-1,75/2 \times 0,9) = \pm 166,4^\circ$ с двойным нулем на постоянной составляющей, так что фильтр, очевидно, является фильтром верхних частот. Нули цепи обратной связи, ближайшие к полюсам фильтра, получаются при выборе $k_1 = -2$ и $k_2 = -1$ (при этом возникает двойной нуль, расположенный в точке с координатами $r = 1$ и $\theta = 180^\circ$).
- 3.В. Третий фильтр имеет два комплексно-сопряженных полюса, расположенных в точках с координатами $r = 0,9$ и $\theta = \pm 90^\circ$ на z -плоскости, и пару нулей, расположенных в точках с координатами 0° и 180° . Чтобы компенсировать, насколько это возможно, влияние полюсов, следует выбрать целые коэффициенты цепи обратной связи — $k_1 = 0$ и $k_2 = -1$. Как видно из табл. 13.6, данный выбор даст пару комплексно-сопряженных нулей в точках с координатами $r = 1$ и $\theta = \pm 90^\circ$.

13.4.15. Граничный цикл вследствие ошибок округления произведений

Помимо снижения отношения сигнал-шум ошибки, вызванные округлением, могут привести к тому, что выход фильтра будет осциллировать или застынет на одном значении даже при отсутствии сигнала на входе. Данное явление называется *низкоуровневым граничным циклом*, и ниже оно иллюстрируется на примере.

Пример 13.17

БИХ-фильтр первого порядка характеризуется разностным уравнением

$$y(n) = x(n) + \alpha y(n-1) \quad n > 0.$$

Для данного начального условия $y(0) = 6$ и нулевого входа, т.е. $x(n) = 0$, $n = 0, 1, \dots$ выполните следующие задания:

- получите и изобразите (предполагая вычисления с бесконечной точностью) первые 10 выходных значений при 1) $\alpha = -0,75$; 2) $\alpha = 0,75$;
- повторите п. а, но с предположением, что данные и регистры имеют размер по 4 бит (т.е. 3 бит данных и 1 бит знака), и что произведения округляются;
- повторите п. а и б, предполагая усечение произведений непосредственно после умножения.

Решение

Значения выходных выборок для указанных случаев изображены на рис. 13.27 и перечислены в табл. 13.7.

Видно, что если вход $x(n)$ равен нулю бесконечно, выход $y(n)$ при бесконечной точности экспоненциально затухает до нуля, вне зависимости от знака α . В то же время, если используется арифметика конечной точности с выходом, который округляется до ближайшего целого, то при положительных α выход остается фиксированным на данном уровне. Диапазон выходных уровней, в пределах которого удерживается выход, называется *мертвой зоной* (deadband). В данном примере мертвая зона — это интервал $[-2, 2]$. В общем случае мертвая зона фильтра первого порядка определяется как [11]

$$k = \text{int} \left[\frac{0,5}{1 - ||\alpha||} \right],$$

где $\text{int}[]$ обозначает целую часть числа в квадратных скобках. Если α отрицательное, выход осциллирует с частотой $F_s/2$ между двумя фиксированными уровнями разного знака. Это вызвано тем, что, когда вход исчезает, выход фильтра затухает до величины, меньше уровня квантования, которая при округлении переходит на следующий уровень, и этот процесс продолжается, порождая низкоуровневые осцилляции. В некоторых ситуациях данные осцилляции нежелательны. Например, они порождают неприятный

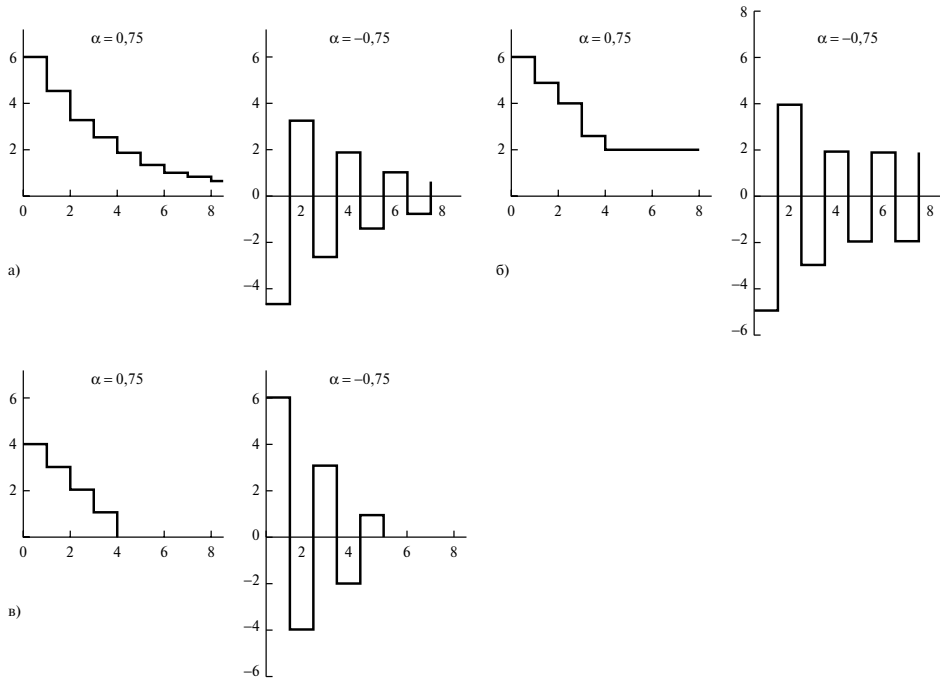


Рис. 13.27. Иллюстрация низкоуровневого граничного цикла, порожденного квантованием произведений БИХ-фильтра первого порядка: *а* — бесконечная точность; *б* — квантование через округление; *в* — квантование через усечение

шум в телефонной системе при незанятом канале (когда абонент молчит). Для уменьшения граничных циклов обычно увеличивается длина слова процессора или к выходу перед округлением прибавляется псевдослучайный сигнал. Рассмотренная ранее схема ESS также позволяет снизить амплитуду граничных циклов и в некоторых случаях полностью их устранить.

Вообще, граничные циклы округления не возникают в фильтре второго порядка, если коэффициенты находятся внутри треугольника устойчивости.

Таблица 13.7. Результаты примера 13.17

<i>n</i>	<i>y</i> (<i>n</i>), ($\alpha = 0,75$)			<i>y</i> (<i>n</i>), ($\alpha = -0,75$)		
	<i>Бесконечная точность</i>	<i>Округление</i>	<i>Усечение</i>	<i>Бесконечная точность</i>	<i>Округление</i>	<i>Усечение</i>
0	6	6	6	6	6	6
1	4,5	5	4	-4,5	-5	-4
2	3,38	4	3	3,38	4	3
3	2,53	3	2	-2,53	-3	-2
4	1,90	2	1	1,90	2	1
5	1,42	2	0	-1,42	-2	0
6	1,07	2	0	1,07	2	0
7	0,80	2	0	-0,80	-2	0
8	0,60	2	0	0,60	2	0
9	0,45	2	0	-0,45	-2	0

13.4.16. Другие нелинейные явления

Помимо переполнения и граничных циклов произведений существуют и другие нелинейные эффекты, которые могут повлиять на поведение цифрового БИХ-фильтра. Перечислим их.

1. *Явление скачка*. При подаче на фильтр синусоидального сигнала при одинаковых входных сигналах возможно два выходных уровня. Небольшое изменение амплитуды или частоты входного сигнала приводит к скачку с одного уровня на другой. Определены несколько областей внутри треугольника устойчивости, где возможны такие явления. В этих областях коэффициенты фильтра удовлетворяют условию $|a_1|a_2 < -1$. Доказано, что последствия указанных нелинейных эффектов снижает схема ESS.
2. *Субгармонический отклик*. При синусоидальном входе выход может содержать субгармоники входа [4]. Следовательно, при одном и том же входном сигнале, но разных начальных условиях можно получить существенно отличающиеся выходы. Данные эффекты проявляются более значительно, если полюса фильтра лежат близко к единичной окружности.

13.5. Эффекты конечной разрядности в алгоритмах БПФ

В большинстве алгоритмов ЦОС основными ошибками, возникающими при реализации алгоритмов БПФ с использованием арифметики с фиксированной запятой, являются:

- ошибки округления, которые возникают при усечении или округлении произведения $W^k B$ до системной длины слова;
- ошибки переполнения, возникающие, когда выход “бабочки” превышает приемлемую длину слова;
- ошибки квантования коэффициентов — следствие представления настроечных параметров ограниченным числом битов.

Рассмотрим влияние этих ошибок на примере выхода двоичного БПФ.

13.5.1. Ошибки округления в БПФ

Основной операцией во всех алгоритмах БПФ является вычисление по схеме “бабочка”, которая в двоичном БПФ с децимацией во времени характеризуется следующими уравнениями:

$$A' = A + W^k B,$$

$$B' = A - W^k B,$$

где A и B — входы “бабочки”, A' и B' — ее выходы. В общем случае настроечный параметр W^k , а также входы и выходы — комплексные. В реализации с фиксированной

запятой вычисление “бабочки” выполняется с использованием действительной арифметики, поэтому A' и B' нужно выразить в прямоугольной форме (см. главу 12):

$$\begin{aligned} A' &= A_{re} + B_{re} \cos(X) + B_{im} \sin(X) + i[A_{im} + B_{im} \cos(X) - B_{re} \sin(X)] = \\ &= A_{re} + B_{re} W_{re} + B_{im} W_{im} + i[A_{im} + B_{im} W_{re} - B_{re} W_{im}] \end{aligned} \quad (13.40, a)$$

$$\begin{aligned} B' &= A_{re} - [B_{re} \cos(X) + B_{im} \sin(X)] + i[A_{im} - \{B_{im} \cos(X) - B_{re} \sin(X)\}] = \\ &= A_{re} - (B_{re} W_{re} + B_{im} W_{im}) + i[A_{im} - (B_{im} W_{re} - B_{re} W_{im})] \end{aligned} \quad (13.40, б)$$

где индексом re обозначена действительная, а индексом im — мнимая часть переменной, $X = 2\pi k/N$. Таким образом, вычисление по схеме “бабочка” требует четырех умножений и шести действительных сложений (вычитание мы также относим к операциям сложения). В реализации с фиксированной запятой каждое из приведенных выше произведений требует для представления приблизительно вдвое большего числа битов, чем требуется для записи каждого операнда по отдельности. Например, если переменные B_{re} , B_{im} , W_{re} и W_{im} представлены как 16-битовые числа, то после умножения представление каждого произведения потребует 32 бит. Усечение или округление каждого произведения до прежних 16 бит порождает *ошибку округления*.

Таким образом, с каждой “бабочкой” можно связать четыре источника шума округления, по одному для каждого произведения, так что мощность шума округления (т.е. дисперсия) на выходе каждой “бабочки” равна

$$\sigma_B^2 = 4 \times \frac{q^2}{12}, \quad (13.41)$$

где $q = 2^{-(B-1)}$, а длина слова системы — B бит.

Шум, порождаемый “бабочкой” на каждом этапе, поступает на следующие этапы. Если исследовать граф-схему БПФ, например, рис. 13.28, видно, что каждый выход БПФ $X(k)$ можно проследить до предыдущих $(N-1)$ “бабочки”. На рис. 13.29 показаны “бабочки”, которые вносят вклад в выход $X(2)$ при $N = 7$. Вообще, каждый выход БПФ соединяется со всеми $N/2$ “бабочками” этапа 2, $N/4$ “бабочками” этапа 2, $N/8$ “бабочками” этапа 3 и т.д. Предполагая, что все “бабочки” порождают идентичные, но некоррелирующие ошибки, получим максимальную мощность шума в каждом выходе БПФ $X(k)$ [14]:

$$\sigma_0^2 = (N-1)\sigma_B^2 \approx N\sigma_B^2 = \frac{N}{3} 2^{-2(B-1)} \quad (\text{при большом } N).$$

Таким образом, мощность шума прямо пропорциональна размеру преобразования. Если удвоить N , что равносильно прибавлению этапа БПФ, то мощность шума удваивается. Поскольку мощность шума пропорциональна N и $2^{-2(B-1)}$, то, чтобы удержать ее без изменений, можно увеличить длину слова на 1.

В этом случае отношение сигнал-шум приблизительно равно

$$SNR = \frac{1/3}{N 2^{-2(B-1)}/3} = \frac{2^{2(B-1)}}{N}.$$

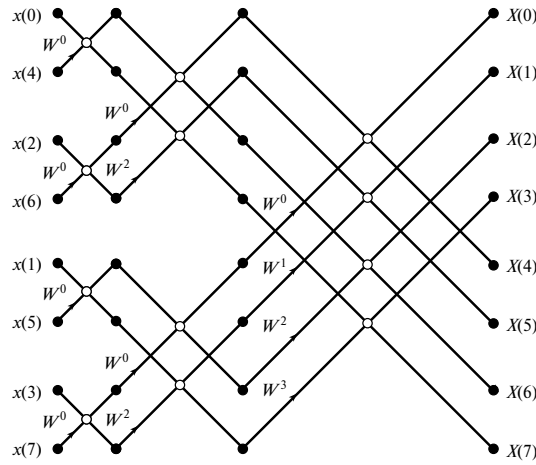


Рис. 13.28. Граф-схема 8-точечного двоичного алгоритма БПФ с децимацией во времени

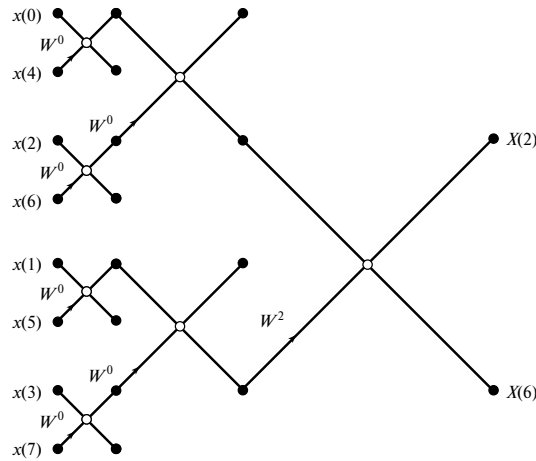


Рис. 13.29. Граф-схема, на которой показаны “бабочки”, вносящие вклад в шум округления, присутствующий в выходах $X(2)$ и $X(6)$

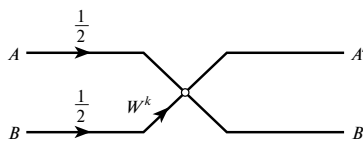


Рис. 13.30. Схема масштабирования для снижения переполнения в каждой “бабочке”

Если рассмотреть только вклад шума в “бабочки”, имеющие нетривиальные настроечные параметры (“бабочки” с настроечными параметрами $W^k = \pm 1$ или $\pm i$ дадут точные произведения без ошибок), мощность порождаемого шума будет ниже из-за ошибок округления. Фактически, если использовать эту информацию, можно найти, что некоторые выходы БПФ вообще не дают ошибок. Поэтому очевидно, что приведенные выше выражения — это верхние границы ошибок.

13.5.2. Ошибки переполнения и масштабирование в БПФ

Поскольку размер данных растет после расчета каждой “бабочки”, при вычислении БПФ необходимо масштабирование, так как оно позволяет избежать ошибок переполнения (после сложения по формулам (13.40, *a* и *b*)). Существует несколько способов масштабирования данных для предотвращения переполнения при вычислении БПФ. Одна из наиболее популярных схем масштабирования основана на наблюдении, что выход каждой “бабочки” удовлетворяет следующему соотношению [14]:

$$\max[|A'|, |B'|] \leq 2 \max[|A|, |B|]. \quad (13.42)$$

Данная запись означает, что максимальный модуль выхода “бабочки” увеличивается от этапа к этапу в два раза. Следовательно, если входы “бабочки” умножить на коэффициент 0,5, в выходах не должно возникнуть переполнения, при условии, что амплитуда входных данных принадлежит диапазону разрешенных длин слов (см. рис. 13.30).

Фактически в некоторых случаях масштабирования с коэффициентом 0,5 недостаточно для предотвращения переполнения, даже если вход меньше единицы. Для иллюстрации рассмотрим явные выражения, приведенные в формулах (13.40):

$$\begin{aligned} A' &= A_{re} + B_{re}W_{re} + B_{im}W_{im} + i[A_{im} + B_{im}W_{re} - B_{re}W_{im}], \\ B' &= A_{re} - (B_{re}W_{re} + B_{im}W_{im}) + i[A_{im} - (B_{im}W_{re} - B_{re}W_{im})], \end{aligned}$$

Если $X = 2\pi k/N = 45^\circ$, то $\cos(45^\circ) = \sin(45^\circ) = \sqrt{2}/2$. Без масштабирования и с единичными действительными и мнимыми частями входных данных (граничный случай) из приведенного выше уравнения получаем:

$$A' = 2,4142 + i; \quad B' = -0,4142 + i.$$

Если каждый вход масштабирован на 0,5, то:

$$A' = 1,2071 + 0,5i; \quad B' = -0,2071 + 0,5i.$$

Очевидно, что действительная часть A' по-прежнему вызовет переполнение, даже при указанном масштабировании, так как ее амплитуда больше единицы.

Несмотря на возможность переполнения, в большинстве реализаций все же используется масштабный коэффициент 0,5, поскольку его легко реализовать — простой сдвиг на одну позицию вправо (или ничего не делать, сыграв на 2 бит знака, возникающих после умножения). Чтобы избежать переполнения во всех возможных случаях, выход

должен масштабироваться на 1, 2071 (2, 41421/2), после чего на каждом этапе результат должен масштабироваться на 0,5. После БПФ выход приводится к прежнему масштабу. Для большинства действительных данных дополнительное масштабирование входа может быть необязательным, так как максимальное значение невозможно.

Масштабирование входов “бабочек” меняет характеристики шума округления БПФ. Выходное отношение сигнал-шум приблизительно равно

$$SNR = \frac{1}{2N} 2^{2(B-1)}. \quad (13.43)$$

Пример 13.18

Аппаратный процессор БПФ использует при расчете “бабочек” арифметику с фиксированной запятой. Оцените максимальную длину слова, необходимую для выполнения 1024-точечного БПФ с выходным отношением сигнал-шум 40 дБ. Предполагайте, что входе всего БПФ вход каждой “бабочки” масштабируется с коэффициентом 0,5.

Решение

$$40 = 10 \lg \left(\frac{1}{2N} s^{2(B-1)} \right); \quad 10^{\frac{40}{10}} = \frac{1}{2N} 2^{2(B-1)};$$

$$B - 1 = \frac{1}{2} \log_2(2N \times 10^4) / \lg(2) = 12, 14 = 13 \text{ (бит) (приблизительно)}.$$

Длина слова системы — $B = 14$ бит.

13.5.3. Квантование коэффициентов в БПФ

Во многих аппаратных реализациях БПФ действительная и мнимая части настроечного множителя W^k обычно вычисляются заранее, квантуются до B бит и заносятся в таблицу соответствий (здесь B — длина слова системы). В результате получаем обычные ошибки квантования.

13.6. Резюме

Производительность системы ЦОС ограничена числом битов, используемых в ее реализации. Четырьмя основными источниками ошибок являются: 1) входное квантование, 2) квантование коэффициентов, 3) округление произведений и 4) переполнение при сложении. В данной главе представлены методы анализа влияния этих ошибок на производительность системы ЦОС и, где это возможно, устранения или минимизации этого влияния. Для иллюстрации освещаемых концепций использовался БИХ-фильтр. Отмечено, что длины слова коэффициентов должно быть достаточно, чтобы минимизировать

влияние квантования коэффициентов на частотную характеристику и предотвратить возможную неустойчивость. Вообще, устойчивость БИХ-фильтра следует проверять всегда. Явно устойчивый БИХ-фильтр при реализации с конечной точностью может стать неустойчивым. При высококачественной работе с аудио, например, необходимы минимум 24-битовые коэффициенты, поскольку лишь они могут обеспечить обработку низкочастотных аудиосигналов. В большинстве других случаев представления коэффициентов 16 или большим числом бит и выполнения арифметических операций в накопителях двойной длины достаточно для минимизации эффектов конечной разрядности.

Ошибки усечения или округления, вызванные операциями в арифметике конечной точности, создают нелинейные эффекты в фильтре, такие как граничные циклы, когда выход фильтра осциллирует даже при отсутствии входа или при постоянном входе. Влияние ошибок округления на производительность фильтра можно выразить через отношения сигнал-шум на выходе фильтра. Снижение отношения сигнал-шум вследствие ошибок округления можно предотвратить с помощью схем формирования спектра шума (Error Spectral Shaping — ESS). Основным результатом использования подобных схем является устранение эффекта усиления полюсами фильтра ошибок округления. Цена этого — увеличение числа умножений и сложений, хотя схемы ESS первого порядка с целыми коэффициентами довольно эффективно вычислительно.

Задачи

13.1. На рис. 13.31 показано стандартное звено фильтра второго порядка.

1. Объясните, почему переполнение допустимо в узлах 1 и 3, но не в узле 2.
2. Найдите подходящие масштабные коэффициенты, позволяющие снизить возможность переполнения в узле 2.
3. Предположив, что фильтр реализован в 8-битовой системе с квантованием перед накопителем, определите число дополнительных битов, требуемых, чтобы снизить шум округления по крайней мере на 20 дБ.

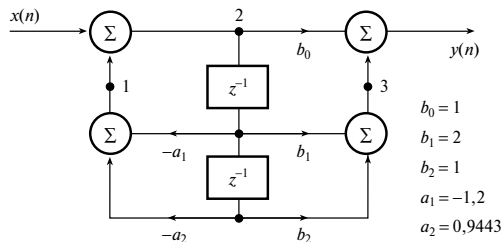


Рис. 13.31. Стандартное звено второго порядка (задача 13.1)

13.2. БИХ-фильтр характеризуется следующей передаточной функцией:

$$H(z) = \frac{0,1436 + 0,2872z^{-1} + 0,1436z^{-2}}{1 - 1,8353z^{-1} + 0,9748z^{-2}}.$$

1. Определите положения полюсов и нулей и изобразите диаграмму нулей и полюсов фильтра.
2. Определите радиальное расстояние полюсов от начала координат.
3. Определите число битов, требуемых для представления каждого коэффициента:
 - а) чтобы обеспечить устойчивость;
 - б) чтобы амплитудная характеристика в полосе пропускания менялась не более, чем на 1%.

13.3. Дана следующая передаточная функция:

$$H(z) = \frac{1 - 0,9631z^{-1} + z^{-2}}{1 - 1,5763z^{-1} + 0,9413z^{-2}}.$$

1. Определите подходящие масштабные коэффициенты, позволяющие избежать переполнения, если для реализации фильтра используется каноническое звено второго порядка.
2. Определите минимальную длину слова, требуемую для получения выходного отношения сигнал-шум 60 дБ. Укажите сделанные предположения.

13.4. Даны следующие полюса и нули БИХ-фильтра восьмого порядка:

полюса	нули
$0,2870 \pm 0,9075i$	$0,0553 \pm 0,9985i$
$0,7882 \pm 0,5658i$	$0,8828 \pm 0,4698i$
$0,4098 \pm 0,7447i$	$-0,4816 \pm 0,8764i$
$0,6479 \pm 0,5975i$	$0,9617 \pm 0,2740i$

1. Изобразите диаграмму нулей и полюсов и разбейте на пары полюса и нули.
2. По диаграмме запишите передаточную функцию фильтра. Предполагая, что фильтр нужно реализовать в каскадной форме, выберите подходящую схему упорядочения каскадов.
3. Определите подходящие масштабные множители звеньев фильтра, используя программу анализа эффектов конечной длины слова.
4. Пусть входные данные оцифрованы до 8 бит и снижение отношения сигнал-шум вследствие ошибок округления не превышает 0,5 дБ. Определите подходящие длины слов для представления промежуточных данных, коэффициентов и переменных.

13.5. Дана следующая передаточная функция:

$$H(z) = \frac{1 - 1,4890z^{-1} + z^{-2}}{1 - 0,3724z^{-1} + 0,5119z^{-2}} \times \frac{1 - 1,9020z^{-1} + z^{-2}}{1 - 0,3779z^{-1} + 0,0851z^{-2}}.$$

1. Определите и изобразите положения полюсов и нулей.
2. Изобразите амплитудную и фазовую характеристику фильтра, используя частоту дискретизации 48 кГц.

3. Запишите представление коэффициентов фильтра в формате дополнения до двух с фиксированной запятой, используя 8 бит (включая бит знака).
 4. Повторите п. 1 и 2 для квантованного фильтра и сравните полученные результаты.
- 13.6. Дана следующая передаточная функция:

$$H(z) = 0,1436 \frac{1 + 2z^{-1} + z^{-2}}{1 - 0,67993z^{-1} + 0,49133z^{-2}}.$$

1. Определите подходящие масштабные множители, чтобы на выходе сумматора 1 не было переполнения и общий масштабный коэффициент был равен 1.
2. Запишите коэффициенты фильтра и масштабные коэффициенты в 8-битовой арифметике с фиксированной запятой
3. Определите общий шум округления

Литература

1. Abu-el-Hajja A. and Al-Ibrahim M. M. (1986) Improving performance of digital sinusoidal oscillators by means of error feedback circuits. *IEEE Trans. Circuits and Systems*, **33**(4), 373–380.
2. Antoniou A. (1979) *Digital Filters Analysis and Design*. New York: McGraw-Hill.
3. Chen W. (1996) Performance of cascade and parallel IIR filters. *J. Audio Eng. Soc.*, **44**(3), 148–158.
4. Claasen T. (1974) Improvement of overflow behaviour of 2nd-order digital filters by means of error feedback. *Electronics Lett.*, **10**(12), 240–241.
5. Dattorro J. (1988) The implementation of recursive digital filters for high-fidelity audio. *J. Audio Eng. Soc.*, **36**(11), 851–878.
6. Flores I. (1963) *The Logic of Computer Arithmetic*. Englewood Cliffs NJ: Prentice-Hall.
7. Higgins W. E. and Munson D. (1982) Noise reduction strategies for digital filters: error spectrum shaping versus the optimal linear state-space formulation. *IEEE Trans. Acoustics, Speech and Signal Processing*, **30**(6), 963–973.
8. IEEE (1979) *Programs for Digital Signal Processing*. New York: IEEE Press.
9. IEEE (1985) IEEE Standard for Binary Floating Point Arithmetic. *SIGPLAN Notices*, **22**(2), 9–25.
10. Ifeachor E. C. (2001) *A Practical Guide for MATLAB and C Language Implementation of DSP Algorithms*. Harlow: Pearson Education.
11. Jackson L. B. (1986) *Digital Filters and Signal Processing*. Boston MA: Kluwer.
12. Mitra S. K., Hirano K. and Sakaguchi H. (1974) A simple method of computing the input quantization and multiplication roundoff errors in a digital filter. *IEEE Trans. Acoustics, Speech and Signal Processing*, **22**(5), 326–329.
13. Mullis C. T. and Roberts R. A. (1982) An interpretation of error spectrum shaping in digital filters. *IEEE Trans. Acoustics, Speech and Signal Processing*, **30**(6), 1013–1015.
14. Oppenheim A. V. and Weinstein C. J. (1972) Effects of finite register length in digital filtering and the fast Fourier transform. *Proc. IEEE*, **60**, 957–976.
15. Patterson D. A. and Hennessy J. L. (1990) *Computer Architecture: A Quantitative Approach*. San Mateo CA: Morgan Kaufmann.
16. Rabiner L. R. and Gold B. (1975) *Theory and Applications of Digital Signal Processing*. Englewood Cliffs NJ: Prentice-Hall.
17. Rader C. M. and Gold B. (1967) Effects of parameter quantization on the poles of a digital filter. *Proc. IEEE*, **55**, 688, 689.
18. Texas Instruments (1986) *Digital Signal Processing Applications with the TMS320 Family: Theory, Algorithms and Implementations*. Texas Instruments.

19. Tomarakos J. and Ledger D. (1998) Using the Low-cost, High-performance ADSP-21065L Digital Signal Processor for Digital Audio Applications. Analog Devices DSP Application. Подробности — на сайте www.analog.com.
20. Weitek (1984) High Speed Digital Arithmetic VLS Application Seminar Notes. Sunnyvale CA: Weitek.
21. Wilson R. (1993) Filter topologies. J. Audio Eng. Soc., **41**(9), 667–678.

Дополнительная литература

- Abu-el-Haija A. I. and Peterson A. M. (1979) An approach to eliminate roundoff errors in digital filters. *IEEE Trans. Acoustics, Speech and Signal Processing*, **27**, 195–198.
- Ahmed N. and Natarajan T. (1983) *Discrete-time Signals and Systems*. Reston VA: Reston Publishing Inc.
- Avenhaus E. (1972) Filters with coefficients of limited wordlength. *IEEE Trans. Audio Electroacoustics*, **20**, 206–212.
- Barnes C. W., Tran B. N. and Leung S. H. (1985) On the statistics of fixed-point roundoff error. *IEEE Trans. Acoustics, Speech and Signal Processing*, **33**, 595–606.
- Chang T. L. (1978) A low roundoff noise digital filter structure. In *Proc. IEEE Int. Symp. on Circuits and Systems*, May 1978, pp. 1004–1008.
- Chang T. L. (1979) Error-feedback digital filters. *Electronics Lett.* 348–349.
- Chang T. L. (1980) Comments on “An approach to eliminate roundoff errors in digital filters”. *IEEE Trans. Acoustics, Speech and Signal Processing*, **28**(2), 244, 245.
- Chang T. L. (1981) Suppression of limit cycles in digital filters designed with one magnitude-truncation quantizer. *IEEE Trans. Circuits and Systems*, **28**(2), 107–111.
- Chang T. L. (1981) On low-roundoff noise and low-sensitivity digital filter structures. *IEEE Trans. Acoustics, Speech and Signal Processing*, **29**(5), 1077–1080.
- Chang T. L. and White S. A. (1981) An error cancellation digital-filter structure and its distributed-arithmetic implementation. *IEEE Trans. Circuits and Systems*, **28**(4), 339–342.
- Charalambous C. and Best M. J. (1974) Optimization of recursive digital filters with finite wordlengths. *IEEE Trans. Acoustics, Speech and Signal Processing*, **22**(6), 424–431.
- Claasen T. A. C. M. and Kristiansson L. O. G. (1975) Necessary and sufficient conditions for the absence of overflow phenomena in a second order recursive digital filter. *IEEE Trans. Acoustics, Speech and Signal Processing*, **23**(6), 509–515.
- Claasen T. A. C. M., Mecklenbrauker W. F. G. and Peek J. B. H. (1973) Second-order digital filter with only one magnitude-truncation quantiser and having practically no limit cycles. *Electronics Lett.*, **9**, 531, 532.
- Claasen T. A. C. M., Mecklenbrauker W. F. G. and Peek J. B. H. (1973) Some remarks on the classification of limit cycles in digital filters. *Philips Research Rep.*, **28**, 297–305.
- Claasen T., Mecklenbrauker W. F. G. and Peek J. B. H. (1975) Frequency domain criteria for the absence of zero-input limit cycles in nonlinear discrete-time systems, with applications to digital filters. *IEEE Trans. Circuits and Systems*, **22**, 232–239.
- Claasen T. A. C. M., Mecklenbrauker W. F. G. and Peek J. B. H. (1976) Effects of quantization and overflow in recursive digital filters. *IEEE Trans. Acoustics, Speech and Signal Processing*, **24**(6), 517–528.
- Crochiere R. E. (1975) A new statistical approach to the coefficient wordlength problem for digital filters. *IEEE Trans. Circuits and Systems*, **22**, 190–196.
- Crochiere R. E. and Oppenheim A. V. (1975) Analysis of linear digital networks. *Proc. IEEE*, **63**(4), 581–594.
- Diniz P. S. R. and Antoniou A. (1985) Low-sensitivity digital filter structures which are amenable to error-spectrum shaping. *IEEE Trans. Circuits and Systems*, **32**(10), 1000–1007.
- Elliot D. F. (ed.) (1987) *Handbook of Digital Signal Processing*. London: Academic Press.
- IEEE (1978) *Digital Signal Processing II*. Institute of Electrical and Electronics Engineers.
- Jackson L. B. (1970) On the interaction of roundoff noise and dynamic range in digital filters. *BSTJ*, **49**(2), 159–184.
- Jackson L. B. (1976) Roundoff noise bounds derived from coefficient sensitivities for digital filters. *IEEE Trans. Circuits and Systems*, **23**(8), 481–485.

- Knowles J. B. and Olcayto E. M. (1968) Coefficient accuracy and digital filter response. *IEEE Trans. Circuit Theory*, **15**, 31–41.
- Liu B. (1971) Effect of finite wordlength on the accuracy of digital filters P a review. *IEEE Trans. Circuit Theory*, **18**, 670–767.
- Liu B. and Kaneko T. (1969) Error analysis of digital filters realized with floating-point arithmetic. *Proc. IEEE*, **57**(10), 1735–1747.
- Markel J. D. and Gray A. H. (1975) Fixed-point implementation algorithms for a class of orthogonal polynomial filter structures. *IEEE Trans. Acoustics, Speech and Signal Processing*, **23**(5), 486–494.
- Markel J. D. and Gray A. H. (1975) Roundoff noise characteristics of a class of orthogonal polynomial structures. *IEEE Trans. Acoustics, Speech and Signal Processing*, **23**(5), 473–486.
- Motorola (1988) *Digital Stereo 10-band Graphic Equalizer Using the DSP56001*. Motorola Application Note.
- Mullis C. T. and Roberts R. A. (1976) Round-off noise in digital filters: frequency transformations and invariants. *IEEE Trans. Acoustics, Speech and Signal Processing*, **24**(6), 538–550.
- Munson D. C. and Liu B. (1980) Low-noise realization for narrow-band recursive digital filters. *IEEE Trans. Acoustics, Speech and Signal Processing*, **28**, 41–54.
- Nagle H. T. and Nelson V. P. (1981) Digital filter implementation on 16 bit microcomputers. *IEEE Micro*, **1**, 23–41.
- Oppenheim A. V. and Schaffer R. W. (1975) *Digital Signal Processing*. Englewood Cliffs NJ: Prentice-Hall.
- Peled A., Liu B. and Steiglitz K. (1974) A new hardware realization of digital filters. *IEEE Trans. Acoustics, Speech and Signal Processing*, **22**, 456–462.
- Peled A., Liu B. and Steiglitz K. (1975) A note on implementation of digital filters. *IEEE Trans. Acoustics, Speech and Signal Processing*, **23**, 387–389.
- Rabiner L. R., Cooley J. W., Helms H. D., Jackson L. B., Kaiser L. F., Rader C. M., Schaffer R. W., Steiglitz K. and Weinstein C. J. (1972) Terminology in digital signal processing. *IEEE Trans. Audio and Electroacoustics*, **20**, 322–337.
- Sandberg I. W. and Kaiser J. F. (1972) A bound on limit cycles in fixed-point implementations of digital filters. *IEEE Trans. Audio and Electroacoustics*, **20**, 110–112.
- Sim P. K. and Pang K. K. (1985) Effects of input-scaling on the asymptotic overflow-stability properties of second recursive digital filters. *IEEE Trans. Circuits and Systems*, **32**(10), 1008–1015.
- Steiglitz K. (1971) Designing short-word recursive digital. *Proc. 9th Ann. Allerton Conf. on Circuit and System Theory*, 6–8 October, pp. 778–788.
- Steiglitz K., Bede L. and Liu B. (1976) An improved algorithm for ordering poles and zeros of fixed-point recursive digital filters. *IEEE Trans. Acoustics, Speech and Signal Processing*, **24**, 341–343.
- Taylor F. J. (1983) *Digital Filter Design Handbook*. New York: Marcel Dekker.
- Thong T. (1976) Finite wordlength effects in the ROM digital filter. *IEEE Trans. Acoustics, Speech and Signal Processing*, **24**, 436, 437.
- Thong T. and Liu B. (1977) Error spectrum shaping in narrowband recursive digital filters. *IEEE Trans. Acoustics, Speech and Signal Processing*, **25**, 200–203.
- Williamson D. and Sridharan S. (1985) An approach to coefficient wordlength reduction in digital filters. *IEEE Trans. Circuits and Systems*, **32**(9), 893–903.

Приложения

13.A. Программа анализа эффектов конечной разрядности для БИХ-фильтров

Программу анализа эффектов конечной разрядности для БИХ-фильтров с разобранными примерами можно найти в книге [10] (подробности — в предисловии).

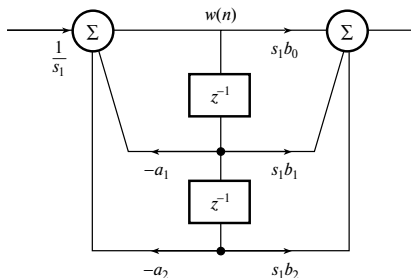


Рис. 13Б.1. Стандартное каноническое звено фильтра

13.Б. Уравнения масштабных множителей L_2

На рис. 13Б.1 изображено стандартное каноническое звено фильтра. Передаточная функция этого звена записывается следующим образом:

$$H(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}}. \quad (13B.1)$$

Масштабный множитель L_2 , который выбирается для снижения вероятности переполнения, в узле, помеченном $w(n)$, записывается следующим образом:

$$s_1^2 = \frac{1}{2\pi i} \oint \frac{z^{-1} dz}{D(z)D(z^{-1})} = \frac{1}{2\pi i} \oint F(z) dz, \quad (13B.2)$$

где

$$D(z) = 1 + a_1 z^{-1} + a_2 z^{-2}$$

$$F(z) = z^{-1}/D(z)D(z^{-1}),$$

а $\oint$ — контурный интеграл по окружности $|z| = 1$.

Используя значение $D(z)$ из уравнения (13Б.2), получаем

$$\begin{aligned} s_1^2 &= \frac{1}{2\pi i} \oint \frac{z^{-1} dz}{(1 + a_1 z^{-1} + a_2 z^{-2})(1 + a_1 z + a_2 z^2)} = \\ &= \frac{1}{2\pi i} \oint \frac{z dz}{(z^2 + a_1 z + a_2)(1 + a_1 z + a_2 z^2)}. \end{aligned}$$

Полосы z_1 и z_2 подынтегрального выражения расположены внутри единичной окружности и находятся следующим образом:

$$z^2 + a_1 z + a_2 = (z - z_1)(z - z_2) = 0. \quad (13\text{Б.3})$$

Вычисляя вычеты, получаем выражение для суммы вычетов функции $F(z)$, s_1^2 :

$$\begin{aligned}
 s_1^2 &= \lim_{z \rightarrow z_1} \frac{(z - z_1)z}{(z^2 + a_1z + a_2)(1 + a_1z + a_2z^2)} + \lim_{z \rightarrow z_2} \frac{(z - z_2)z}{(z^2 + a_1z + a_2)(1 + a_1z + a_2z^2)} = \\
 &= \frac{z_1}{(z_1 - z_2)(1 + a_1z_1 + a_2z_1^2)} - \frac{z_2}{(z_1 - z_2)(1 + a_1z_2 + a_2z_2^2)} = \\
 &= \frac{z_1(1 + a_1z_2 + a_2z_2^2) - z_2(1 + a_1z_1 + a_2z_1^2)}{(z_1 - z_2)(1 + a_1z_1 + a_2z_1^2)(1 + a_1z_2 + a_2z_2^2)} = \\
 &= \frac{1 - a_2z_1z_2}{(1 + a_1z_1 + a_2z_1^2)(1 + a_1z_2 + a_2z_2^2)} = \quad (13Б.4) \\
 &= \frac{1 - a_2z_1z_2}{1 + a_1(z_1 + z_2) + a_2(z_1^2 + z_2^2) + a_1^2z_1z_2 + a_1a_2z_1z_2(z_1 + z_2) + a_2^2z_1^2z_2^2} = \\
 &= \frac{1 - a_2z_1z_2}{1 + a_1(z_1 + z_2) + a_2[(z_1 + z_2)^2 - 2z_1z_2] + a_1^2z_1z_2 + a_1a_2z_1z_2(z_1 + z_2) + a_2^2(z_1z_2)^2}
 \end{aligned}$$

Далее из уравнения (13Б.3) получаем:

$$z^2 + a_1z + a_2 = (z - z_1)(z - z_2) = z^2 - (z_1 + z_2)z + z_1z_2.$$

Следовательно,

$$\begin{aligned}
 a_1 &= -(z_1 + z_2) \\
 a_2 &= z_1z_2.
 \end{aligned}$$

Подставляя значения a_1 и a_2 в уравнение (13Б.4), получаем

$$\begin{aligned}
 s_1^2 &= \frac{1 - a_2^2}{1 - a_1^2 + a_2(a_1^2 - 2a_2) + a_1^2a_2 - a_1^2a_2^2 + a_2^4} = \\
 &= \frac{1 - a_2^2}{1 - a_1^2 - 2a_2^2 + 2a_1^2a_2 - a_1^2a_2^2 + a_2^4} = \\
 &= \frac{1 - a_2^2}{(1 - a_2^2)^2 - a_1^2(1 - 2a_2 + a_2^2)} = \\
 &= \frac{1 - a_2^2}{(1 - a_2^2)^2 - a_1^2(1 - a_2)^2} = \\
 &= \frac{1}{(1 - a_2^2) - a_1^2(1 - a_2)/(1 + a_2)}.
 \end{aligned}$$

Итак,

$$s_1^2 = \frac{1}{(1 - a_2^2) - a_1^2(1 - a_2)/(1 + a_2)}. \quad (13Б.5)$$

Приложения и разобранные примеры

14

14.1. Платы для демонстрации обработки сигналов в реальном времени	935
14.2. Приложения ЦОС	939
14.3. Модельные задачи	967
14.4. Вопросы по ЦОС для компьютерных тестов	975
14.5. Резюме	983
Задачи	983
Литература	984
Дополнительная литература	986
Приложение	986

При написании этой главы ставилось четыре цели. Первая — описать некоторые дешевые платы, которые можно использовать для реализации алгоритмов ЦОС, рассмотренных в предыдущих главах. Представлены две платы для процессоров ЦОС с фиксированной запятой первого и второго поколений, которые авторы использовали для демонстрации студентам принципов цифровой обработки сигналов. Кроме того, приводится обзор дешевых плат для процессоров ЦОС с фиксированной запятой третьего порядка.

Вторая цель — описать несколько реальных приложений ЦОС в форме разобранных примеров. Ниже рассмотрены такие приложения, как обработка аудиосигналов в реальном времени, адаптивная фильтрация артефактов на электроэнцефалограмме (фиксация электрической активности мозга) человека и детектирование сердцебиения плода на электрокардиограмме (фиксация электрической активности сердца) плода, что необходимо для оценки состояния ребенка во время родов. Рассматриваемые аспекты логически продолжают круг вопросов, освещенных в предыдущих главах.

Третья цель — представить несколько перспективных практических задач, обсуждение которых можно продолжить в группах на практических занятиях. Последняя цель — изложить набор вопросов с несколькими вариантами ответов, позволяющих подробнее рассмотреть различные аспекты цифровой обработки сигналов.

14.1. Платы для демонстрации обработки сигналов в реальном времени

14.1.1. Основы

Как и в других инженерных сферах, для правильной работы с ЦОС, необходим практический опыт разработки и реализации ее алгоритмов. Студентам инженерных специ-

альностей, имеющим только базовые знания из сферы аналоговой обработки сигналов, сложно разобраться в методах ЦОС, особенно не обладая математическими знаниями, необходимыми для понимания концепций с теоретической точки зрения. Довольно часто такие студенты не могут понять, например, как численные операции, используемые в БИХ- или КИХ-фильтрах, могут приводить к фильтрации данных. Они чувствуют себя комфортно с аналоговыми фильтрами и понимают, как фильтрация достигается на основе сопротивлений и частотно-зависимых характеристик конденсаторов и катушек индуктивности. Часто у студентов возникает вопрос: “А как, собственно, работает цифровой фильтр?”.

Исходя из таких наблюдений, авторы пришли к мысли о необходимости простого, самодостаточного аппаратного блока, на основе которого студенты смогут реализовывать простые функции ЦОС. Мы также хотели поднять такие практические вопросы, возникающие при реализации цифровой обработки сигналов в реальном времени, как наложение, создание зеркальных изображений, затухание по закону $\sin(x)/x$, переполнение и т.д. В результате было разработано несколько простых целевых плат для процессоров ЦОС с фиксированной запятой первого и второго поколений. Данные платы оказались полезными и недорогими платформами для демонстрации алгоритмов ЦОС в реальном времени. В настоящее время налажено промышленное производство нескольких типов дешевых демонстрационных плат для процессоров ЦОС с фиксированной запятой второго и третьего поколений. Некоторые из подобных плат будут описаны ниже.

14.1.2. Целевая плата TMS320C10

Нашей первой платой была целевая плата TMS320C10. На настоящий момент она по-прежнему используется для демонстрации простых алгоритмов ЦОС реального времени. Отметим основные ее особенности:

- самодостаточность и возможность выполнения простых алгоритмов ЦОС в реальном времени;
- простой аналоговый вход-выход, квантованный до 8 бит;
- легкая модификация кодов алгоритмов ЦОС;
- возможность работы с двумя различными частотами дискретизации;
- возможность изучать наложение и создание зеркальных изображений.

Блок-схема системы, показанная на рис. 14.1, состоит из четырех основных устройств: цифрового процессора сигналов первого поколения TMS320C10 (сердце системы), блока 8-битового АЦП/ЦАП, схемы синхронизации и ячеек памяти. Для систем, требующих высокого качества, 8 бит недостаточно, но для демонстрации принципов ЦОС плата вполне подходит. Устройство памяти состоит из ключа выбора программы и пары СППЗУ (стираемое программируемое постоянное запоминающее устройство), смонтированных для облегчения использования на разъемах с нулевым усилием сочленения (Zero Insertion Force — ZIF). СППЗУ разбито на восемь блоков по 1 Кбайт каждый, обращаться к которым можно посредством ключа выбора программы. Такое решение позволяет хранить в СППЗУ до восьми различных программ. Для автономной работы использование СППЗУ обязательно. Пользователь может выбирать из двух доступных частот дискретизации: 7,5 и 15 кГц.

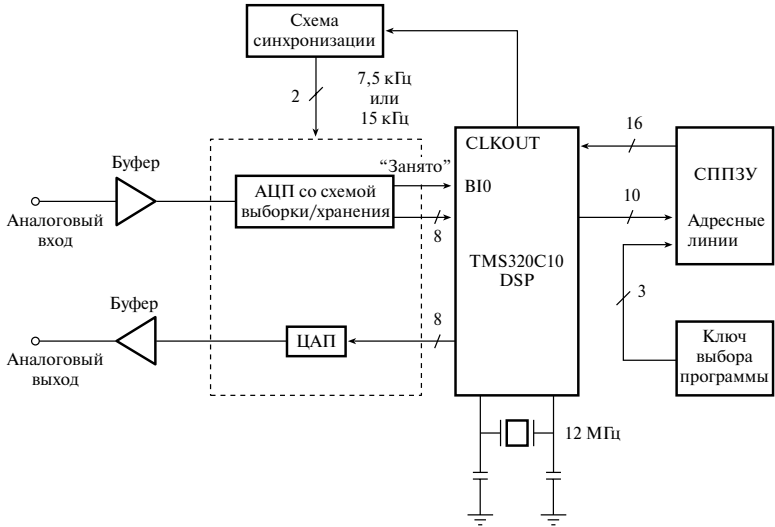


Рис. 14.1. Простая блок-схема целевой платы TMS320C10

Таблица 14.1. Выбор программы из памяти

Блок	Адрес	Программы ЦОС
0	0000–03FF	Цикл входа-выхода
1	0400–07FF	Генератор шума
2	0800–0BFF	Генератор прямоугольных сигналов
3	0C00–0FFF	41-точечный полосовой КИХ-фильтр
4	1000–13FF	61-точечный режекторный БИХ-фильтр
5	1400–17FF	Каскадная реализация БИХ-фильтра нижних частот четвертого порядка
6	1800–1BFF	Параллельная реализация БИХ-фильтра нижних частот четвертого порядка
7	1C00–1FFF	Каскадная реализация полосового БИХ-фильтра четвертого порядка

На данной целевой плате реализованы многие алгоритмы ЦОС, например, КИХ- и БИХ-фильтры, генераторы шума и сигналов прямоугольной формы (см. табл. 14.1).

14.1.3. Демонстрационный модуль для ЦОС реального времени

Плата TMS320C10 полезна при демонстрации простых функций ЦОС в автономном режиме, но в серьезных задачах разработки ее пригодность ограничена. Для быстрой разработки и демонстрации систем ЦОС реального времени лучше подходит дешевый демонстрационный модуль Motorola DSP56002EVM. За последние шесть лет мы использовали данный модуль в одном из наших курсов по цифровой обработке сигналов, в основном из-за того, что он великолепно подходит для обработки реальных аудиосигналов. Отметим основные особенности данного модуля:

- 24-битовый процессор ЦОС DSP56002;
- для автономной работы предусмотрены 32 Кбайт статического ОЗУ и (необязательно) 32 Кбайт флэш-ППЗУ;

- аудиокодек CD-качества (16-битовые стереоаналогоцифровой и цифроаналоговый преобразователи);
- частота дискретизации 48, 32, 16, 9,6 или 8 кГц;
- ассемблер и программы отладки.

Процессор ЦОС имеет два 48-битовых регистра X и Y , которые также можно использовать как четыре 24-битовых регистра ($X0$, $X1$, $Y0$ и $Y1$), два 56-битовых накопителя и аппаратный умножитель, бесценные в обработке сигналов. Отладчик позволяет вносить изменения в регистры и исходный код с помощью простого экранного редактирования. В данном разделе для иллюстрации использования платы DSP56002 рассмотрена задача разработки фильтра.

14.1.4. Демонстрационные платы TMS320C54 и DSP56300

В настоящее время существуют и более сложные программные и аппаратные средства, предназначенные для новых поколений процессоров ЦОС с фиксированной или плавающей запятой (например, Texas Instruments Code Composer Studio), подробное описание этих средств можно найти на Web-сайтах ведущих производителей. В данном разделе кратко описаны два дешевых демонстрационных модуля, которые хорошо подходят для изучения концепций ЦОС и разработки сравнительно сложных систем ЦОС.

Демонстрационный модуль TMS320C54x [35] — это компьютерная сменная плата, которую можно использовать для выполнения алгоритмов ЦОС в реальном времени. Основные особенности этого модуля:

- 16-битовый процессор ЦОС с фиксированной запятой TMS320C541 с 5 Кбайт внутреннего ОЗУ и 28 Кбайт внутреннего ПЗУ;
- графическая программа отладки под среду Windows;
- встроенная поддержка эмуляции отладчика кода на C;
- аналоговый интерфейс ввода-вывода.

Аналоговый интерфейс ввода-вывода поддерживает программируемую фильтрацию с целью защиты от наложения и подавления зеркальных частот, управления амплитудой и частотой дискретизации (до 43,2 кГц). Предлагается также одноканальный 14-битовый аналого-цифровой/цифроаналоговый преобразователь.

Великолепной дешевой автономной (с возможностью подключения к компьютеру) платформой для разработки систем ЦОС является демонстрационный модуль DSP56302 [27]. Разработанную пользователем программу можно загрузить с компьютера во встроенную память для отладки и выполнения. Основные особенности модуля DSP56302 таковы:

- 24-битовый процессор ЦОС с фиксированной запятой DSP56302;
- 32 Кбайт внутренней программной памяти/кэша и ОЗУ для хранения данных;
- два канала аудиокодек CD-качества (16-битовые АЦП/ЦАП);
- кросс-ассемблер и программа отладки под среду Windows.

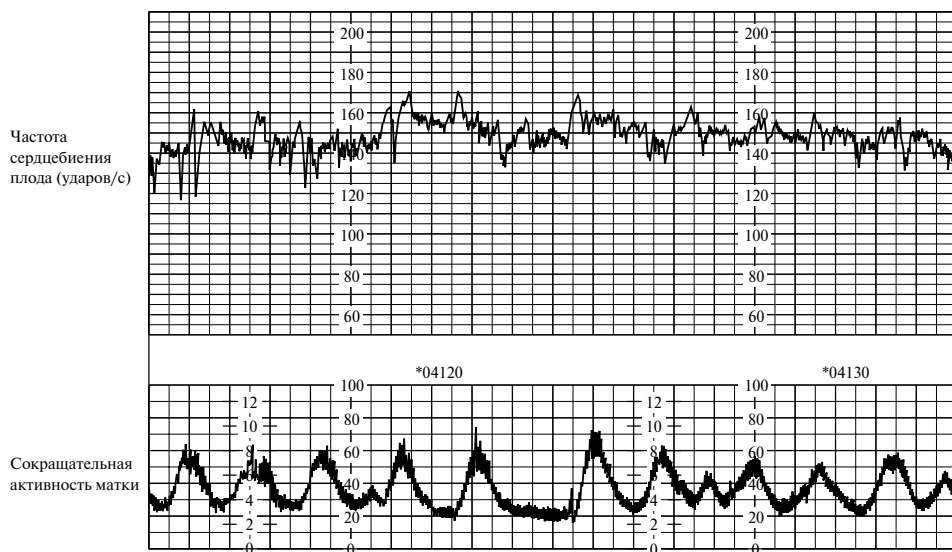


Рис. 14.2. Пример кардиотокограммы (КТГ). КТГ состоит из сигнала сердцебиения плода и сигнала маточной активности

14.2. Приложения ЦОС

14.2.1. Детектирование сердцебиения плода во время родов¹

Во всем мире стандартным методом наблюдения за плодом во время родов является отображение непрерывной частоты сердцебиения плода (Fetal Heart Rate — FHR) и маточной активности, что в сумме составляет сигнал кардиотокограммы (КТГ) (рис. 14.2). Анализируя и соответствующим образом интерпретируя изменения КТГ, акушеры надеются предотвратить появление на свет мертвого младенца или младенца с задержкой в развитии из-за недостатка кислорода во время родов.

14.2.1.1. Электрокардиограмма плода

Частота сердцебиения плода во время родов обычно определяется с помощью электрокардиограммы (ЭКГ), записи электрической активности сердца (рис. 14.3) или с помощью ультразвуковых методов. Подобно ЭКГ взрослого человека нормальная ЭКГ плода характеризуется пятью максимумами и долинами, которые отмечаются последовательными буквами алфавита P, Q, R, S и T [16]. Таким образом, говорят, что ЭКГ состоит из волны P, комплекса QRS и волны T [16].

Как показано на рис. 14.3, величина, обратная периоду сердечного ритма, т.е. интервалу времени между пиками R-R (в миллисекундах), умноженная на 60 000, дает

¹Данный раздел частично основан на выпускном проекте одного из наших бывших студентов, Яда Резека (Lead Rezek).

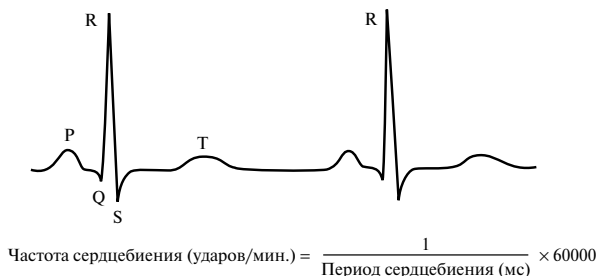


Рис. 14.3. Электрокардиограмма

мгновенный сердечный ритм. Частота сердцебиения плода в верхней половине рис. 14.2 представляется графиком последовательных мгновенных сердечных ритмов.

На практике для измерения сердечного ритма плода к детектированию (аппаратному или программному) последовательных комплексов QRS применяется подходящий алгоритм ЦОС, а по полученным данным вычисляется интервал R-R и соответствующий сердечный ритм. В большинстве методов детектирования комплексов QRS предполагается, что форма комплекса QRS плода известна *априори*, но время его появления неизвестно. Это предположение разумно, хотя и не всегда справедливо, поскольку форма комплекса QRS различна для разных пациентов, и даже один пациент может давать QRS-комплексы разной формы. Таким образом, сравнивая сигнал ЭКГ с известным шаблоном QRS, можно определить положение комплексов QRS в ЭКГ, используя некоторую меру сходства, например, большое значение взаимной корреляции.

Основная проблема здесь — надежное детектирование комплексов QRS. Ухудшение сигнала, например, в связи со смещением базовой линии, интерференцией с сигналом питающей сети, маточными сокращениями, насыщением АЦП и движением ребенка или матери, может привести к ложному детектированию или пропущенным комплексам QRS. Цель данного примера заключается в исследовании и сравнении двух методов детектирования комплексов QRS, которые могут иметь практическую ценность в наблюдении за плодом в реальном времени. Данная работа является малой частью текущего исследовательского проекта, инициированного местными госпиталями с целью разработки интеллектуальных систем помощи медицинским работникам в принятии родов [22].

Данные ЭКГ плода, использованные в рассматриваемом примере, взяты из базы данных, собранной авторами во время работы над проектом. Для получения сигнала ЭКГ измеряется разность сигналов от электрода на голове плода и обычного электрода, размещенного на бедре матери, в качестве заземления использовался второй электрод, расположенный на коже матери (рис. 14.4). Вектор чувствительности данной направленной системы расположен в продольной плоскости плода (в отличие от сагиттальной плоскости соединения стандартного электрода на черепе плода), что должно уменьшить изменение вектора ЭКГ, вызванное вращением плода [25]. ЭКГ плода подавалась через пациента, изолированный усилитель, аналоговый полосовой фильтр (полоса пропускания — 0,07–100 Гц) и оцифровывалась с частотой 500 выборок/с с разрешением 8 бит.

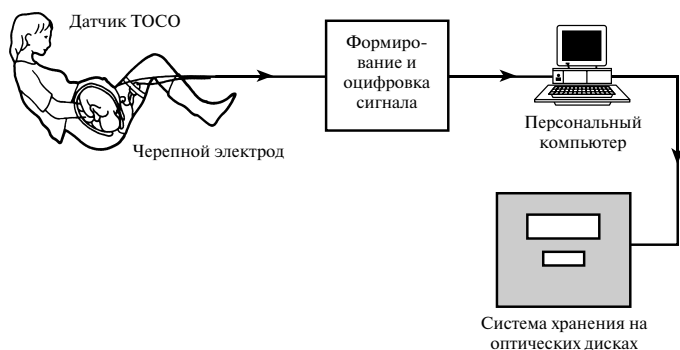


Рис. 14.4. Измерение электрокардиограммы плода

Примеры измеренных ЭКГ плода представлены на рис. 14.5, *а–в*. На этих рисунках показано, что данные на рис. 14.5, *а* имеют относительно большое отношение сигнал-шум по сравнению с рис. 14.5, *б* и *в*, с большими амплитудами волн R (выглядят, как пики). С другой стороны, для данных на рис. 14.5, *б* характерно относительно высокое содержание шума и значительные смещения базовой линии, хотя волны R по-прежнему различимы. Данные на рис. 14.5, *в* содержат ошибки АЦП, которые выглядят как большие перепады амплитуды между максимальным и минимальным значениями АЦП в начале записи (возможно, вызваны насыщением АЦП), а также значительные смещения базовой линии и высокочастотный шум (включая загрязнение сигналом питающей сети). Три набора данных на рис. 14.5, *а–в* классифицированы как ЭКГ класса 1 (хорошая), 2 (средняя) и 3 (плохая) соответственно.

14.2.1.2. Предварительная обработка сигнала ЭКГ плода

Для данных класса 2 и 3 уровни шума и смещения базовой линии затрудняют детектирование комплексов QRS в необработанных ЭКГ. Поэтому для надежного детектирования комплексов QRS необходимо предварительно обработать ЭКГ, чтобы минимизировать влияние этих источников ухудшения сигнала, а лишь затем следует пытаться детектировать комплексы QRS. Известно, что значительные частотные компоненты комплекса QRS расположены между 4 и 45 Гц. Смещения базовой линии в ЭКГ обычно низкочастотные, как правило, меньше 3 Гц, хотя для данных класса 3 частота базовой линии может достигать до 15 Гц и даже превышать эту величину.

Для предварительной обработки грубой ЭКГ перед детектированием комплексов QRS можно использовать цифровой полосовой КИХ- или БИХ-фильтр. Мы предпочитаем использовать КИХ-фильтр, поскольку БИХ-фильтр высокого (например, восьмого) порядка иногда входит в резонанс при возбуждении узкополосными комплексами QRS, что может затруднить точную локализацию волны R. В нашем примере используются следующие спецификации фильтра:

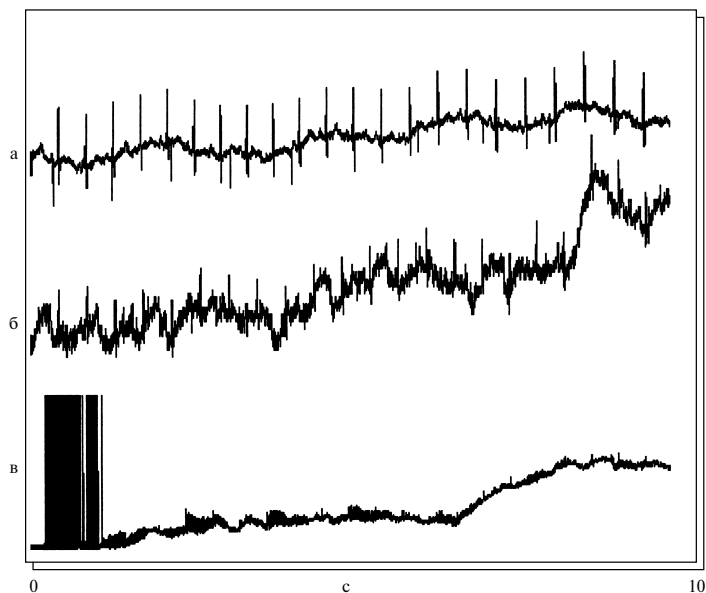


Рис. 14.5. Примеры классов данных ЭКГ: *а* — степень 1 (хорошая);
б — степень 2 (средняя); *в* — степень 3 (плохая)

длина фильтра	75;
частота дискретизации	500 Гц;
полосы подавления	0–1, 47–250 Гц;
полоса пропускания	9–39 Гц;
неравномерность в полосе пропускания	0,5 дБ;
затухание в полосе подавления	30 дБ.

Коэффициенты фильтра получены с помощью оптимального метода, описанного в главе 7. На рис. 14.6, *а–в* показаны отфильтрованные данные ЭКГ. По сравнению с соответствующими нефильтрованными данными (рис. 14.6, *а–в*) смещения базовой линии, а также высокочастотный шум в фильтрованных данных снижены (исключая исходные импульсные помехи в фильтрованных данных). В данных класса 3 ошибка АЦП выглядит как пакетная помеха, которая несомненно разрушает большинство алгоритмов детектирования комплексов QRS (см. рис. 14.6, *в*).

14.2.1.3. Шаблон QRS

Большинство методов детектирования комплексов QRS полагаются на доступность характерного шаблона QRS, с которым сравнивается входной сигнал ЭКГ. Шаблон может генерироваться по необработанным ЭКГ-данным путем детектирования и усреднения нескольких хороших комплексов QRS. Это можно сделать автоматически или полупручным образом путем визуального изучения ЭКГ класса 1 и определения хороших однозначных комплексов QRS. Затем волны R синхронизируются и комплексы

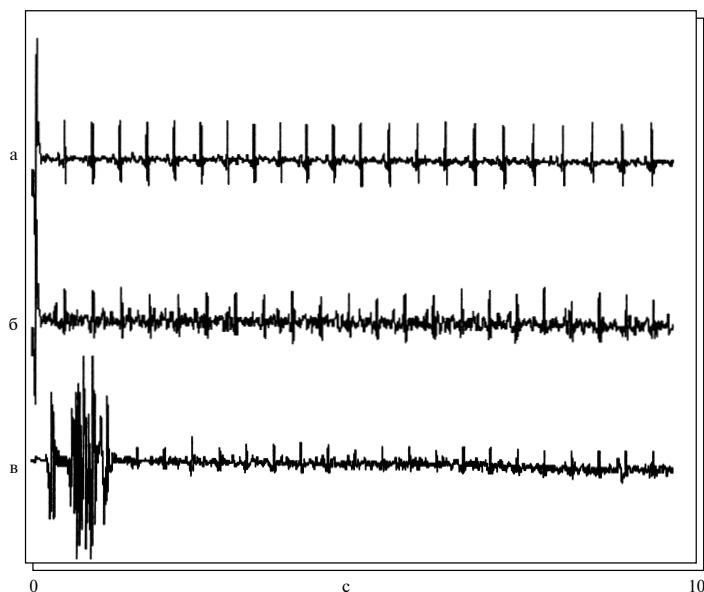


Рис. 14.6. Фильтрованные данные ЭКГ: *а* — класс 1 (хороший); *б* — класс 2 (средний); *в* — класс 3 (плохой)

QRS усредняются. Для детектирования комплексов QRS может использоваться фиксированный шаблон QRS, или же в начале каждой новой записи ЭКГ генерироваться новый шаблон. Пример шаблона QRS, полученного путем усреднения 69 комплексов данных класса 1 с последующим взятием 31 выборки (15 выборок с каждой стороны волны R) усредненных комплексов QRS, показан на рис. 14.7.

В данном примере пробовались шаблоны различной длины. Обычно длина шаблона N составляла от 11 до 31 выборки, т.е. ширина при частоте дискретизации 500 выборок/с равна 20–60 мс. Ниже представлены результаты, полученные для двух шаблонов длиной 11 и 31 выборки.

14.2.1.4. Методы детектирования комплексов QRS

Общая блок-схема процесса детектирования комплексов QRS изображена на рис. 14.8. Необработанные ЭКГ-данные вначале предварительно обрабатываются для снижения влияния шума. Обработанные выборки данных подаются в буфер по одной выборке за такт. Для каждой новой точки данные подаются в буфер, старейшая выборка данных удаляется, и содержимое буфера сравнивается с шаблоном в детекторе QRS. Затем выход детектора QRS сравнивается с порогом. Если выход превышает пороговое значение, считается, что комплекс QRS присутствует. В данном примере сравниваются два удобных метода детектирования, выбранные с точки зрения практической пригодности или потенциальной практической пригодности. Отметим, что существует множество иных методов детектирования комплексов QRS.

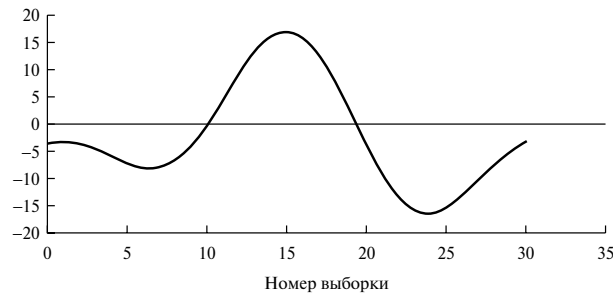


Рис. 14.7. Пример схемы комплекса QRS. Получен путем усреднения по 69 комплексам QRS в ЭКГ класса 1 с синхронизированными R-волнами. Комплексы QRS детектировались с уровнем порога 13

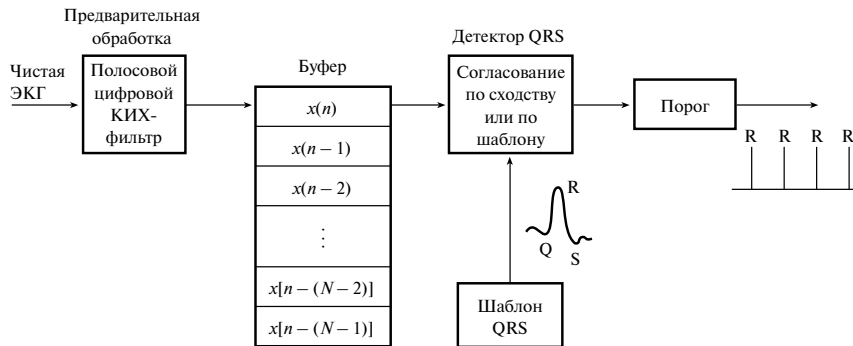


Рис. 14.8. Концепции детектирования комплексов QRS по необработанной ЭКГ

Итак, используются следующие методы:

- 1) усреднение амплитуд перекрестных разностей (Average Magnitude Cross-Difference — AMCD), в настоящее время используемое в новом мониторе наблюдения за плодом [25];
- 2) согласованная фильтрация — распространенный метод детектирования комплексов QRS, который исследовался многими авторами [1, 10]: он тесно связан с методом корреляции.

Усреднение амплитуд перекрестных разностей

В данном методе блоки предварительно обработанных данных ЭКГ-плода сравниваются с шаблонным комплексом QRS, как описано выше. Далее вычисляются разности между соответствующими выборками в ЭКГ и шаблоном. Затем высчитывается сумма $y(i)$ абсолютных значений разностей:

$$y(i) = \sum_{k=0}^{N-1} |x_t(k) - x_t - [x(k+i) - x_i]|, \quad i = 0, 1, \dots, \quad (14.1)$$

где $x_t(k)$ — выборки шаблонного комплекса QRS, $x(k+i)$ — выборки сигнала ЭКГ, N — длина шаблона, i — параметр смещения времени. Величина x_t — среднее значение шаблона QRS, а x_i — среднее значение i -го блока данных для сигнала ЭКГ, которое находится следующим образом:

$$x_t = \frac{1}{N} \sum_{k=0}^{N-1} x_t(i),$$

$$x_i = \frac{1}{N} \sum_{k=0}^{N-1} x(k+i).$$

Если сигнал ЭКГ и шаблон QRS весьма похожи по форме, т.е. принадлежат окрестности комплекса QRS, значение AMCD $y(i)$ становится минимальным (теоретически — равным нулю).

Цифровая согласованная фильтрация

Согласованная фильтрация широко используется для детектирования периодических сигналов, скрытых в шуме. При данном подходе делаются следующие предположения: сигнал ограничен во времени и имеет известную форму. Таким образом, задача формулируется как определение момента прохождения сигнала. Известно, что импульсная характеристика цифрового согласованного фильтра $h(k)$ является обращенной во времени копией детектируемого сигнала. Итак, в нашем случае, если $x_t(k)$ — шаблонный комплекс QRS, коэффициенты фильтра записываются следующим образом:

$$h(k) = x_t(N - k - 1), \quad k = 0, 1, \dots, N - 1. \quad (14.2)$$

Цифровой согласованный фильтр можно представить как КИХ-фильтр с обычной трансверсальной структурой, вход и выход которого связаны следующими соотношениями:

$$\begin{aligned} y(i) &= \sum_{k=0}^{N-1} h(k)x(i-k) = \\ &= \sum_{k=0}^{N-1} x_t(N-k-1)x(i-k), \end{aligned}$$

где $x(i)$ — выборки входного сигнала ЭКГ, $x_t(k)$ — выборки шаблона QRS, N — длина фильтра, $h(k)$ — коэффициенты согласованного фильтра, i — коэффициент сдвига времени. Очевидно, что при совмещении шаблона и комплекса QRS выход согласованного фильтра будет максимальным. Следовательно, исследуя выход фильтра на предмет значений, превышающих порог, можно определить начало QRS-комплекса.

14.2.1.5. Мера эффективности схемы детектирования QRS

Чтобы оценивать и сравнивать алгоритмы, требуется мера эффективности. Определим (согласно [1]) меру эффективности как

$$\frac{(\text{общее число волн R} - \text{число пропусков} - \text{число случаев ложного детектирования}) \times 100\%}{\text{общее число волн R плода}}. \quad (14.3)$$

Для данной записи ЭКГ мера эффективности достигает значения 100%, если только волны R в записи корректно обнаружены, нет пропусков (необнаруженных волн R), случаев ложного детектирования (ложных тревог). Для данного метода детектирования комплексов QRS число пропусков или ложных тревог можно определить, визуально сравнивая выход детектора и предварительно обработанную ЭКГ. Другая альтернатива — использовать так называемое “правило 28 ударов” (28-beat rule), которое медицинские работники применяют, чтобы различать истинные изменения сердечного ритма плода и ложные изменения, вызванные, например, ошибками приборов. Согласно этому правилу волна R, изменяющая частоту сердцебиения плода на более чем ± 28 ударов в минуту, указывает на пропущенный или ложный комплекс QRS. Чтобы облегчить применение этого правила, к шаблону сердечного ритма плода можно подобрать базовую линию.

14.2.1.6. Результаты

На рис. 14.9 и 14.10 изображена зависимость эффективности методов AMCD и согласованной фильтрации соответственно от порогов для данных класса 1 и 2.

Эффективность обоих методов зависит от используемого порогового уровня (каждый порог выражается как доля максимального входного сигнала) и длины шаблона QRS. В обоих методах наилучшая производительность достигается с порогом порядка 50%. Более широкие шаблоны обычно эффективнее узких, если качество данных хорошее, но основная характеристика метода — его чувствительность к уровням порога.

В целом, с точки зрения эффективности методы AMCD и согласованной фильтрации не сильно отличаются. При подходящих уровнях порога оба метода достигают следующей эффективности:

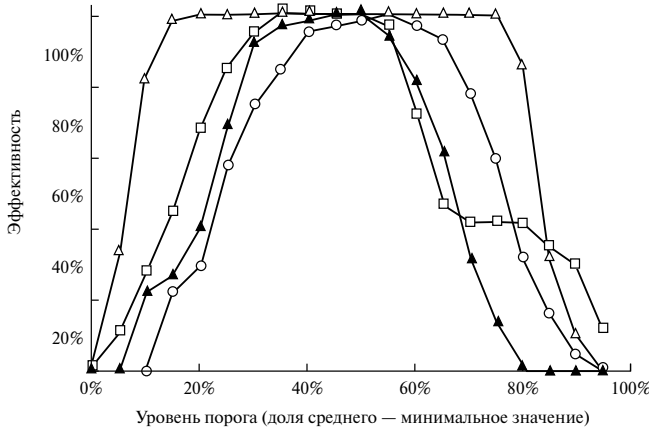


Рис. 14.9. Эффективность метода среднего перекрестной разности амплитуд для данных класса 1 и 2 и шаблонов длины 11 и 31: □ — класс 1, длина 11; △ — класс 1, длина 31; ○ — класс 2, длина 11; ▲ — класс 2, длина 31

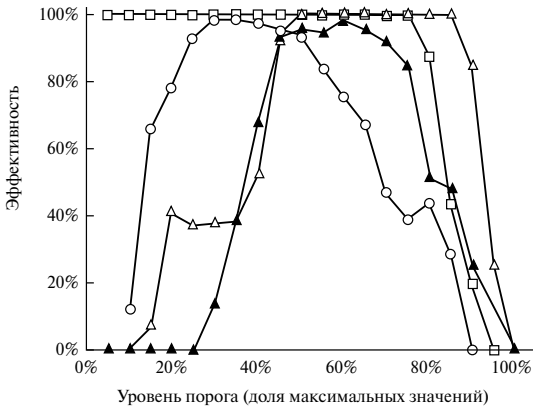


Рис. 14.10. Эффективность метода согласованной фильтрации для данных класса 1 и 2 и шаблонов длины 11 и 31: □ — класс 1, длина 11; △ — класс 1, длина 31; ○ — класс 2, длина 11; ▲ — класс 2, длина 31

- 100%-ное детектирование для всех ЭКГ класса 1;
- детектирование больше 90% для данных класса 2;
- детектирование больше 60% для данных класса 3.

14.2.2. Адаптивное устранение окулярных артефактов с ЭКГ человека²

14.2.2.1. Вступление

Работа, описанная в данном разделе, касается оперативного устранения окулярных артефактов с электрокардиограммы (ЭКГ) человека. ЭКГ широко используется в клинических и психологических ситуациях, но часто она зашумляется окулярными артефактами (ОА), порожденными движением окулярных систем (глазных яблок, век и т.д.). В некоторых случаях, например, у младенцев с повреждениями головного мозга и пациентов с опухолями лобных долей мозга, трудно различить патологические медленные волны в ЭКГ и окулярные артефакты. Сходство ОА и сигналов, представляющих интерес, также затрудняет автоматизацию анализа ЭКГ с помощью компьютера. Реакция на раздражитель, известный как условная отрицательная вариация (Contingent Negative Variation — CNV), которая используется при диагностике пациентов с хореей Хантингтона (Huntingdon's chorea) [23], очень чувствительна к окулярным артефактам. Следовательно, необходимо удалить ОА с ЭКГ, чтобы анализировать можно было запись истинной ЭКГ.

Хотя устранить ОА в настоящее время можно в автономном режиме, до сих пор нет удовлетворительной схемы оперативного устранения ОА. Ранее заявленные методы оперативного устранения артефактов требуют совместной работы с пациентом (обеспечить которую можно не всегда), включают трудоемкую ручную калибровку и позволяют обработать ОА максимум одного типа, поскольку в них предполагается постоянный поправочный коэффициент. В данном разделе описана новая оперативная система устранения окулярных артефактов из сигналов ЭКГ, которая не имеет указанных недостатков и предлагает такие дополнительные преимущества, как гибкость. Система основана на микропроцессоре Motorola 68000 и использует численно устойчивый алгоритм UD-факторизации, который допускает непрерывное адаптивное устранение окулярных артефактов. Рассматривается также оперативный алгоритм, аппаратное и программное обеспечение системы.

Методы устранения окулярных артефактов и контроля над ними

Проблема устранения окулярных артефактов с ЭКГ усложняется их сходством с определенными церебральными волнами, представляющими практический интерес, и их спектральным перекрытием. Из различных методов, предложенных для устранения окулярных артефактов из ЭКГ и контроля над ними, наилучшими, пожалуй, являются методы вычитания электроокулограмм (ЭОГ). В данной главе термином *электроокулограмма* называется электрический потенциал, порожденный окулярными движениями, измеренный между двумя электродами на коже, расположенными близко к глазам. Впрочем, заявленные на сегодняшний день различные методы вычитания ЭОГ не решают за-

²Адаптировано из [21].

дачу полностью, поэтому непрерывно развиваются новые подходы. Все существующие методы основаны на предположении, что окулярный артефакт аддитивно прибавляется к фоновой ЭКГ. Следовательно, в декретной форме записи получаем:

$$y(i) = \sum_j^n \theta_j x_j(i) + e(i) = \mathbf{x}^T(i)\theta + e(i), \quad (14.4)$$

где

$$\begin{aligned} \mathbf{x}^T &= [x_1(i) \quad x_2(i) \quad \dots \quad x_n(i)], \\ \theta &= [\theta_1(i) \quad \theta_2(i) \quad \dots \quad \theta_n(i)]^T, \end{aligned}$$

$y(i)$ и $x_j(i)$ — выборки измеренных ЭЭГ и ЭОГ соответственно, e_i — “истинная ЭЭГ”, которую можно рассматривать как сигнал ошибки, а i — номер выборки. Величины θ_j — коэффициенты пропорциональности, которые далее будем называть *параметрами окулярного артефакта*, n — число параметров модели; θ_j также называется *коэффициентами передачи*. Величины $\mathbf{x}^T(i)$ и θ — векторы параметров ЭОГ и окулярного артефакта соответственно, а ^{T} обозначает транспонирование. Если значение θ_j можно оценить, можно получить и оценку $e(i)$:

$$\hat{e}(i) = y(i) - \sum_j^n \hat{\theta}_j x_j(i), \quad i = 1, 2, \dots, m, \quad (14.5)$$

где $\hat{\theta}_j$ — оценки θ_j , а $\hat{e}(i)$ и m — номера выборок, используемых в оценке. Таким образом, задача превращается в поиск оценки θ_j . Ее иллюстрация приведена на рис. 14.11. Для данного типа окулярного движения $\hat{\theta}_j$ практически константа, но эта константа значительно меняется для разных типов окулярных артефактов, хотя можно показать, что $\hat{\theta}_j$ меняется медленно, по крайней мере для данного типа окулярных артефактов. Вообще, тип окулярного артефакта, который появится в данное время, определить невозможно, поскольку во многих случаях одновременно проявляется несколько типов окулярных артефактов, $\hat{\theta}_j$ нельзя считать константой, так что лучшей стратегией является адаптивная оценка θ_j . Термин “адаптивная” использован для того, чтобы подчеркнуть, что оценки параметров окулярного артефакта, а следовательно, и алгоритм устранения ОА должны автоматически подстраиваться под изменения окулярного артефакта. Различные методы вычитания ЭОГ отличаются преимущественно способом оценки θ_j , числом используемых сигналов ЭОГ и способом их измерения.

Метод вычитания ЭОГ можно реализовать либо оперативно, т.е. по мере получения данных, либо автономно. Основным преимуществом автономного метода является возможность использования более сложных методов удаления артефактов. Впрочем, в приложениях, требующих обработки и анализа в реальном времени, задержка, предполагаемая при использовании автономных методов, неприемлема. В настоящее время в обработке сигналов ЭЭГ наблюдается тенденция к обработке в реальном времени, поэтому устранение артефактов нужно проводить оперативно.

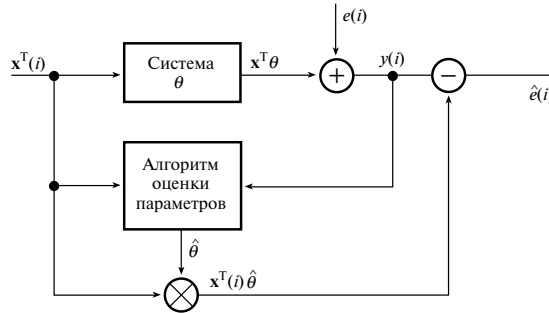


Рис. 14.11. Блок-схема устранения окулярных артефактов

В автономных методах оценки θ получаются при минимизации J , суммы квадратов члена ошибки, т.е. $J = \sum_{i=1}^m e^2(i)$. Такой минимум дает

$$\hat{\theta}_m = [\mathbf{X}_m^T \mathbf{X}_m]^{-1} \mathbf{X}_m^T \mathbf{Y}_m, \quad (14.6)$$

где

$$\mathbf{Y}_m = [y(1) \ y(2) \ \dots \ y(m)]^T, \mathbf{X}_m = [\mathbf{x}^T(1) \ \mathbf{x}^T(2) \ \dots \ \mathbf{x}^T(m)]^T$$

$$\hat{\theta}_m = [\hat{\theta}_1 \ \hat{\theta}_2 \ \dots \ \hat{\theta}_m]^T, \mathbf{E}_m = [e(1) \ e(2) \ \dots \ e(m)]^T.$$

Данное уравнение, основа автономных методов устранения окулярных артефактов, — это просто оценка по схеме наименьших квадратов величины θ , которую можно получить, используя любой подходящий метод обращения матриц. После получения $\hat{\theta}_m$ оценивается окулярный артефакт, а следовательно, из уравнения (14.5) можно получить фоновую ЭЭГ $\hat{e}(i)$.

Описанный метод наименьших квадратов можно расширить на многоканальную систему, где корректируется более одного сигнала ЭЭГ. Таким образом, систему с n входами ЭОГ и q измеренными выходами ЭЭГ можно рассматривать как q отдельных подсистем с одним выходом. Общая система определяется q различными способами.

Типичный пример ранее описанных оперативных методов изображен на рис. 14.12. В данном метод (согласно [13]) исходная калибровка выполнена путем настройки потенциометров, когда пациент двигал глазами в вертикальной или горизонтальной плоскости, пока в ЭЭГ наблюдался минимальный окулярный артефакт. Затем устройство с такими настройками применялось при записи. Данный метод был использован несколькими авторами, которые нашли его громоздким и недостаточным для устранения ОА (см., например, [15]).

В ранее заявленных оперативных методах устранения окулярных артефактов используются аналоговые методы [3, 13], требующие совместной работы с пациентами, гарантировать которую можно не всегда. В общем случае такие методы хуже автономных с точки зрения устранения окулярных артефактов [5, 24], требуют от пользователя знания метода и включают трудоемкую ручную калибровку каждого канала ЭЭГ. Кроме того, данные методы можно настроить только для одного типа ОА.

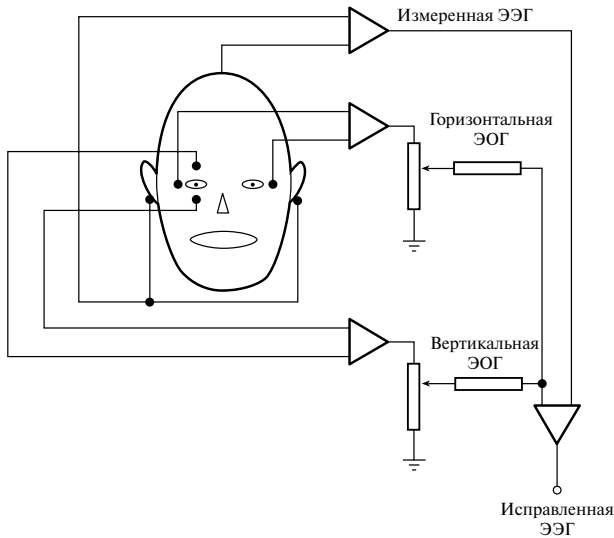


Рис. 14.12. Типичный пример автономных методов устранения артефактов [13]

В данном разделе описана оперативная система устранения окулярных артефактов, не имеющая указанных недостатков. Это микрокомпьютерная система, в которой используется численно устойчивый оперативный алгоритм, основанный на эффективной рекурсивной схеме наименьших квадратов.

14.2.2.2. Алгоритмы оперативного устранения окулярных артефактов

Расчет $\hat{\theta}_m$ по формуле (14.6) требует трудоемкого вычисления обратной матрицы. Очевидно, что обычная схема наименьших квадратов не подходит для оперативной (в реальном времени) оценки, тогда как предпочтителен оперативный алгоритм, включающий ограниченное фиксированное число арифметических операций и не содержащий прямого обращения матриц. При использовании этих оперативных алгоритмов $\hat{\theta}$ так обновляется с поступлением каждой новой выборки, что изменения окулярных движений могут отражаться в $\hat{\theta}$, и устранение окулярных артефактов в этом случае можно рассматривать как адаптивную фильтрацию окулярных артефактов с ЭЭГ [38].

Существует два метода оценки параметров, подходящих для оперативной оценки θ , — метод наименьших квадратов [38] и рекурсивный метод наименьших квадратов [7, 30]. С точки зрения требований к вычислениям и памяти алгоритм, построенный на схеме наименьших квадратов, более эффективен. Кроме того, для него несущественна проблема численной неустойчивости, присущая алгоритму, построенному на рекурсивной схеме (см. ниже). В то же время рекурсивная схема наименьших квадратов дает лучшую сходимость и по этой причине предпочтительнее.

Обычно используется следующая форма схемы наименьших квадратов:

$$\hat{\theta}(m+1) = \hat{\theta}(m) + 2\mu \mathbf{x}(m+1)[y(m+1) - \mathbf{x}^T(m+1)\hat{\theta}(m)], \quad (14.7)$$

где $\hat{\theta}(m)$ и $\hat{\theta}(m+1)$ — оценки θ в m -й и $(m+1)$ -й точке выборки соответственно, μ — константа, отвечающая за скорость сходимости и устойчивость алгоритма. Для обеспечения сходимости μ должно входить в диапазон

$$0 < \mu < 1/\lambda_{\max},$$

где $\lambda_{\max}$ — максимальное собственное значение матрицы $(\mathbf{X}_m^T \mathbf{X}_m)$ в формуле (14.6). Впрочем, время сходимости алгоритма прямо пропорционально отношению максимального и минимального собственных значений матрицы $(\mathbf{X}_m^T \mathbf{X}_m)$, которое может быть очень большим, если входные переменные близки к коллинеарным, что обычно для ЭОГ. В то же время схема наименьших квадратов широко используется в биомедицинских приложениях для снижения шума артефактов в основном из-за своей простоты.

Подходящий рекурсивный алгоритм, построенный на схеме наименьших квадратов, получается путем экспоненциального взвешивания данных с целью постепенного устранения влияния на оценку старых данных. Таким образом,

$$J = \sum_{i=1}^m \gamma^{m-i} e^2(i), \quad 0 < \gamma < 1. \quad (14.8)$$

Минимизация J относительно значений θ приводит к следующему рекурсивному алгоритму:

$$\hat{\theta}(m+1) = \hat{\theta}(m) + \mathbf{G}[y(m+1) - \mathbf{x}^T(m+1)\hat{\theta}(m)] \quad (14.9, a)$$

$$\mathbf{P}(m+1) = \frac{1}{\gamma} \left[\mathbf{P}(m) - \frac{1}{\alpha} \mathbf{P}(m) \mathbf{x}(m+1) \mathbf{x}^T(m+1) \mathbf{P}(m) \right], \quad (14.9, б)$$

где

$$\alpha = \gamma + \mathbf{x}^T(m+1) \mathbf{P}(m) \mathbf{x}(m+1),$$

$$\mathbf{x}^T = [x_1(m+1) \quad x_2(m+1) \quad \dots \quad x_n(m+1)],$$

$$\mathbf{G} = \mathbf{P}(m+1) \mathbf{x}(m+1) = \mathbf{P}(m) \mathbf{x}(m+1) / \alpha.$$

Здесь аргумент m используется, чтобы подчеркнуть тот факт, что величины получаются в каждой точке выборки, γ называется *коэффициентом забывания* и не дает матрице $\mathbf{P}(m+1)$ устремиться к нулю (а $\hat{\theta}(m+1)$ — к константе) с увеличением m , таким образом становится возможным отслеживание медленно меняющегося параметра. Обычно γ выбирается между 0,98 и 1. При меньших значениях самым свежим данным присваиваются слишком большие весовые коэффициенты, что приводит к сильно флуктуирующим оценкам.

Существует, впрочем, две основные проблемы, которые могут возникнуть при непосредственной реализации рекурсивного алгоритма наименьших квадратов. Первая, имеваемая “раздуванием” (“blow-up”) возникает, когда сигнал не иницируется непрерыв-

но, например, когда нет окулярных движений, что приводит к экспоненциальному росту элементов $\mathbf{P}$ в формуле (14.9, б):

$$\lim_{m \rightarrow \infty} [P_{ij}(m+1)] = \lim_{m \rightarrow \infty} \left[\frac{P_{ij}(m)}{\gamma^m} \right] \rightarrow \infty. \quad (14.10)$$

Впрочем, из-за появления миниатюрных окулярных движений (которые есть всегда) и других сигналов, обычно встречающихся в каналах ЭОГ, данная проблема не настолько серьезна.

Вторая проблема — это чувствительность алгоритма к компьютерным ошибкам округления, которые дают отрицательно определенную матрицу $\mathbf{P}$ и в конечном итоге — неустойчивость. Для успешных расчетов необходимо, чтобы матрица $\mathbf{P}$ была положительно квазиопределенной, что эквивалентно следующему требованию: в автономном алгоритме матрица $\mathbf{X}_m^T \mathbf{X}_m$ имеет обратную, но из-за наличия в уравнении (14.9, б) разности членов положительную определенность $\mathbf{P}$ гарантировать можно не всегда [4, 7, 30]. Данная проблема острее в многопараметрических моделях, особенно если переменные (в данном случае — электроокулограммы) линейно зависимы [30], и если алгоритм реализован в малой системе с конечной длиной слова [7].

Проблему численной неустойчивости можно решить путем подходящей факторизации матрицы $\mathbf{P}$, чтобы избежать вычитания в формуле (14.9, б). Подобные алгоритмы факторизации лучше подходят для численных расчетов, а их точность сравнима с точностью алгоритмов, построенных на рекурсивной схеме наименьших квадратов, в которых используется двойная точность [4, 5]. Два таких алгоритма — алгоритм квадратного корня и алгоритм UD-факторизации. С точки зрения памяти и сложности вычислений UD-алгоритм эффективнее. Фактически UD-алгоритм — это обычный алгоритм квадратного корня, в котором не используется квадратный корень, а значит, свойства этих алгоритмов аналогичны.

В UD-алгоритме матрица $\mathbf{P}(m+1)$ факторизуется следующим образом:

$$\mathbf{P}(m+1) = \mathbf{U}(m+1) \mathbf{D}(m+1) \mathbf{U}^T(m+1), \quad (14.11)$$

где $\mathbf{U}(m+1)$ — единичная верхняя треугольная матрица, $\mathbf{U}^T(m+1)$ — обратная к ней, $\mathbf{D}(m+1)$ — диагональная матрица. Таким образом, вместо обновления матрицы $\mathbf{P}$ обновляются ее факторы $\mathbf{U}$ и $\mathbf{D}$.

Используя уравнение (14.11), формулу (14.9, б) можно переписать следующим образом:

$$\mathbf{P}(m+1) = \frac{1}{\gamma} \mathbf{U}(m) \left[\mathbf{D}(m) - \frac{1}{\alpha} \mathbf{v} \mathbf{v}^T \right] \mathbf{U}^T(m), \quad (14.12)$$

где

$$\mathbf{v} = \mathbf{D}(m) \mathbf{U}^T(m) \mathbf{x}(m+1).$$

Если выражение в квадратных скобках факторизовать на верхнюю треугольную и диагональную матрицы, получим

$$\bar{\mathbf{U}}(m) \bar{\mathbf{D}}(m) \bar{\mathbf{U}}^T(m) = \mathbf{D}(m) - \frac{1}{\alpha} \mathbf{v} \mathbf{v}^T, \quad (14.13)$$

где черта используется, чтобы отличить факторы $\mathbf{U}$ и $\mathbf{D}$ матрицы $\mathbf{D}(m) - (1/\alpha)\mathbf{v}\mathbf{v}^T$ от факторов матрицы $\mathbf{P}$. Далее получаем

$$\mathbf{P}(m+1) = \frac{1}{\gamma} \mathbf{U}(m) \bar{\mathbf{U}}(m) \bar{\mathbf{D}}(m) \bar{\mathbf{U}}^T(m) \mathbf{U}^T(m). \quad (14.14)$$

Сравнивая уравнения (14.11) и (14.14) и заметив, что произведение верхних треугольных матриц также является верхней треугольной матрицей, с помощью симметрии уравнения (14.14) получаем

$$\mathbf{U}(m+1) = \mathbf{U}(m) \bar{\mathbf{U}}(m) \quad (14.15, a)$$

$$\mathbf{D}(m+1) = \frac{1}{\gamma} \bar{\mathbf{D}}(m). \quad (14.15, б)$$

Таким образом, для обновления $\mathbf{U}(m+1)$ и $\mathbf{D}(m+1)$ необходимы подходящие рекурсивные формулы для $\mathbf{U}$ и $\mathbf{D}(m)$. В работе [4] приводится алгоритм рекурсивного обновления $\mathbf{U}(m+1)$ и $\mathbf{D}(m+1)$ для фильтра Калмана, в котором используется дисперсия члена ошибки $e(i)$, но не γ . Данный алгоритм тривиально приводится к виду, удобному для использования в задаче устранения окулярных артефактов. Данный модифицированный алгоритм, в котором фигурирует величина γ , приведен в приложении 14А.

Вектор усиления $\mathbf{G}$, получаемый на этапе 10 указанного алгоритма, используется для обновления оценок параметров, как следует из формулы (14.9, а). Так, хотя $\mathbf{P}(m+1)$ можно получить, обновляя UD-элементы согласно формуле (14.11), непосредственно вычислять $\mathbf{P}(m+1)$ необязательно.

14.2.2.3. Некоторые свойства UD-алгоритма

Чтобы лучше понять UD-алгоритм, полезно записать его явно. Таким образом, для двухпараметрической модели алгоритм, приведенный в приложении, выглядит так:

- этап 1: $v_1 = x_1(m+1)$; $v_2 = x_2(m+1) + U_{12}(m)x_1(m+1)$;
- этап 2: $b_1 = d_1(m)x_1(m+1)$; $b_2 = d_2(m)v_2$;
- этап 3: $\alpha_1 = \gamma + b_1v_1 = \gamma + d_1(m)x_1^2(m+1)$;
- этап 4: $d_1(m+1) = d_1(m)/\alpha_1$;
- этап 5: $\alpha_2 = \alpha_1 + b_2v_2 = \alpha_1 + d_2(m)v_2^2$;
- этап 6: $U_{12}(m+1) = U_{12}(m) - b_1v_2/\alpha_1$;
- этап 7: $b_1 = b_1 + b_2U_{12}(m) = d_1(m)x_1(m+1) + d_2(m)v_2U_{12}(m)$;
- этап 8: $d_2(m+1) = d_2\alpha_1/\gamma\alpha_2$.

Из этапа 3 видно, что, если начальные значения диагональных элементов (т.е. $d_1(0)$ и $d_2(0)$) положительны, α_1 , а следовательно, $d_1(m+1)$ всегда будет больше нуля. То же справедливо для α_2 (α в формуле (14.9)) и $d_2(m+1)$.

1. *Положительная определенность $\mathbf{P}$* . Матрица $\mathbf{P}$ положительно определена тогда и только тогда, когда $\mathbf{x}^T \mathbf{P} \mathbf{x} > 0$, исключая случай $x_1 = x_2 = \dots = x_n = 0$ [2]. Из формул (14.9) и (14.11) получаем, что $\alpha = \gamma + \mathbf{x}^T \mathbf{P} \mathbf{x} = \gamma + \mathbf{x}^T \mathbf{U} \mathbf{D} \mathbf{U}^T \mathbf{x}$, откуда для двухпараметрической модели вычисляем (см. выше этапы 3 и 5):

$$\alpha_2 = \alpha_1 + d_2v_2^2 = \gamma + d_1(m)x_1^2(m+1) + d_2(m)v_2^2,$$

так что в этом случае

$$\mathbf{x}^T \mathbf{P} \mathbf{x} = d_1(m)x_1^2(m+1) + d_2(m)v_2^2.$$

Следовательно, знак $\mathbf{x}^T \mathbf{P} \mathbf{x}$ зависит от знаков диагональных элементов (т.е. d_1 и d_2), которые, как отмечалось ранее, всегда положительны, поэтому матрица $\mathbf{P}$ является положительно определенной. Значит, алгоритм UD-факторизации гарантирует положительную определенность $\mathbf{P}$.

2. *Проблема раздувания.* При отсутствии данных (т.е. когда $x_1 = x_2 = \dots = x_n = 0$) $\alpha_1 = \alpha_2 = \gamma$ (этапы 3 и 5) и обе величины d_1 и d_2 постоянно масштабируются на γ , которое меньше единицы, в результате диагональные элементы экспоненциально растут. Таким образом, похоже, что проблема “раздувания” не исчезает при факторизации, как утверждают некоторые авторы. Для уменьшения эффекта “раздувания” можно использовать другие рекурсивные схемы наименьших квадратов (см., например, [14]), но в задаче устранения окулярных артефактов минимальные окулярные движения и другие присущие системе шумы гарантируют, что значения x никогда не станут равными нулю.

Оперативный алгоритм устранения окулярных артефактов с помощью UD-факторизации и алгоритма квадратного корня моделировался на компьютере. Было показано, что полученные результаты подобны результатам, которые позволяют получить автономные эквивалентные средства.

14.2.2.4. Аппаратное обеспечение для оперативной системы устранения окулярных артефактов

В данном и следующем разделах описывается оперативная система устранения окулярных артефактов, в которой используется UD-алгоритм. Вначале устанавливается целевая спецификация системы. Затем подходящая система описывается на уровне блок-схемы.

Система устранения окулярных артефактов разрабатывалась с учетом следующих требований:

- 1) совместимость со стандартными аппаратами ЭЭГ;
- 2) возможность непрерывного устранения окулярных артефактов в реальном времени из многоканальных сигналов ЭЭГ (устранение окулярных артефактов теперь основывается на субъективных критериях и является адаптивным);
- 3) возможность подачи скорректированных ЭЭГ и/или нескорректированных ЭЭГ и ЭОГ на аппарат ЭЭГ, чтобы можно было мгновенно сравнить скорректированную и нескорректированную ЭЭГ;
- 4) возможность избежать насыщения, которое снизит эффективность схемы коррекции, кроме того, система должна иметь некоторые средства автоматического выбора диапазона;
- 5) инструмент может использовать необученный человек.

Данным требованиям удовлетворяет подходящая микропроцессорная система, на которой реализован UD-алгоритм. Использование микропроцессорной системы дополнительно имеет следующие преимущества.

1. Управляемая программно структура дает очень эффективную систему. В одной системе можно реализовать несколько алгоритмов и моделей устранения окулярных артефактов, выбирать из которых сможет пользователь.
2. Для исследования новых моделей или идей требуется всего лишь модифицировать программное обеспечение, а не создавать новый инструмент. Следовательно, система с программным управлением может значительно облегчить исследования.
3. Программируемый инструмент позволяет внедрить обслуживающие программы для самопроверки, автоматической калибровки, снижения проблем переполнения и т.д. Обработка данных на удаленной системе может включать цифровую фильтрацию электроокулограмм с целью снижения влияния вторичных артефактов [17].

На настоящий момент система устранения окулярных артефактов может обрабатывать только шесть каналов сигналов ЭЭГ и ЭОГ, но ее можно расширить до 20 каналов (см. далее). Блок-схема системы приведена на рис. 14.13. Каждый сигнал ЭЭГ/ЭОГ со вспомогательного выхода аппарата ЭЭГ вначале усиливается, а затем ограничивается по полосе до 30 Гц с помощью фильтра нижних частот, выход которого соединен со схемой выборки-хранения. Затем сигналы ЭЭГ/ЭОГ одновременно дискретизируются в моменты положительных переходов сигнала дискретизации FS. Одновременная дискретизация позволяет избежать задержки между соответствующими временными точками. Отрицательные переходы сигнала FS прерывают процессор и сигнализируют о начале цикла, в течение которого 20 выборок последовательно выбираются уплотнителем (MUX) и оцифровываются аналоговым преобразователем (АЦП) под управлением микропроцессора.

Усилитель с программируемым усилением и детектор диапазона используются для расширения динамического диапазона АЦП и предотвращения переполнения. Переполнение АЦП является существенной проблемой в работе схемы и может привести к ложной оценке параметров. Чтобы избежать этого, обычно используется только часть динамического диапазона АЦП, поэтому переполнение возникает редко, а при переполнении все данные в области, где оно произошло, отбрасываются, хотя это может привести к утрате данных. Использование усилителей с программируемым усилением и детектора диапазона позволяет динамически менять усиление. Таким образом, если выход усилителя превышает предопределенный предел детектора, усиление автоматически снижается, при этом значения выборок уменьшаются в два раза, и они переносятся в динамический диапазон АЦП, устраняя таким образом насыщение.

Далее оцифрованные выборки обрабатываются согласно алгоритму устранения окулярных артефактов, и получаются скорректированные ЭЭГ. Выборки скорректированных ЭЭГ плюс необработанные ЭЭГ/ЭОГ при желании можно подать на вспомогательный вход аппарата ЭЭГ, используя аналого-цифровой преобразователь (АЦП) и соответствующую цепь.

Мультиплексор и демультиплексоры требуются для совместного использования ресурсов входным и выходным каналами. Альтернативное решение — предоставить отдельные АЦП и ЦАП для каждого канала. Такой подход снижает шум системы, вызванный перекрестными помехами каналов, до минимума, но признан чересчур дорогим.

Каждый канал имеет собственную выходную схему выборки-хранения и отдельную шину дискретизованных сигналов. Схема выборки-хранения используется для хранения

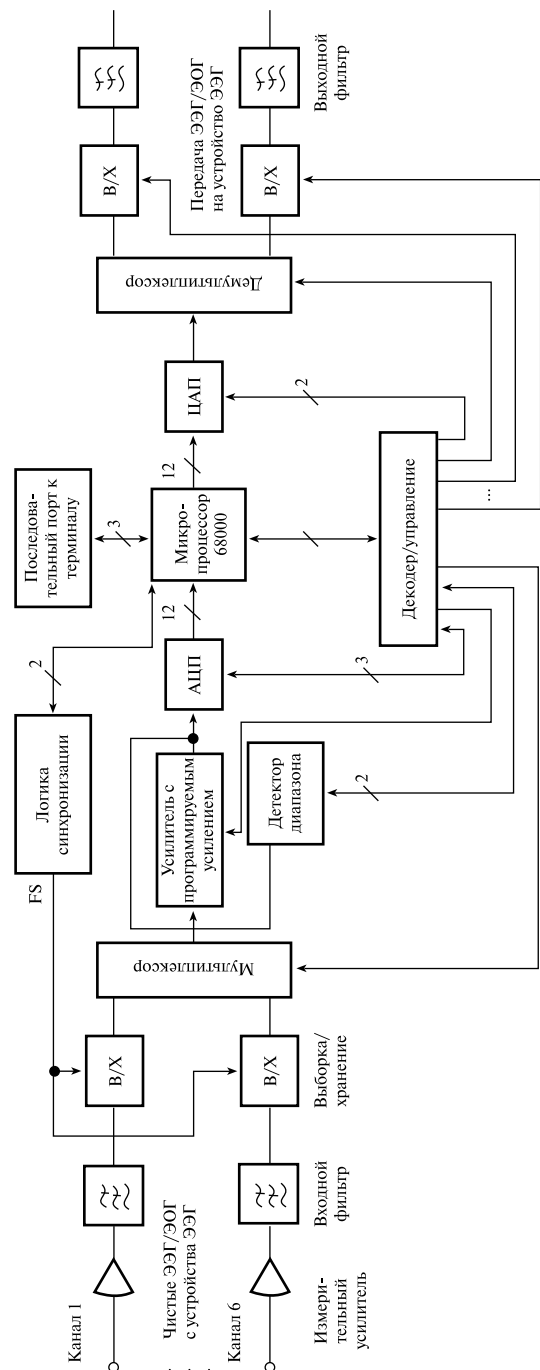


Рис. 14.13. Блок-схема микропроцессорной системы устранения окулярных артефактов

аналоговых выборок до получения новой аналоговой выборки. Эта схема “растягивает” дискретные импульсы и увеличивает мощность сигнала, но вводит апертурное искажение, которое, впрочем, в данном случае мало.

14.2.2.5. Программное обеспечение для системы оперативного устранения окулярных артефактов

Программное обеспечение для системы устранения окулярных артефактов состоит из программ получения и распределения данных, программы оперативного устранения окулярных артефактов, программ арифметики с плавающей запятой и управляющей основной программы. Все программное обеспечение занимает 3 Кбайт памяти. Сердцем программного обеспечения, записанным на языке ассемблера микропроцессора 68000, является UD-алгоритм, описанный в подразделе 14.2.2.2. Ниже предлагается краткий обзор данного программного обеспечения.

Система устранения окулярных артефактов управляется прерываниями. Сигнал прерывания исходит от программируемого датчика времени, встроенного в контроллер системы, и имеет частоту 128 Гц (иногда используется частота 95 Гц, дающая больше времени на вычисления; измерение частоты сигнала легко выполняется с помощью программного обеспечения). Во время прерывания программное обеспечение системы используется для получения данных ЭОГ/ЭЭГ, устранения окулярных артефактов из выборок ЭЭГ с помощью UD-алгоритма и подачи скорректированных ЭЭГ и/или необработанных данных на аппарат ЭЭГ, чтобы можно было получить график на бумаге. Схематически работа системы показана на рис. 14.14.

В фазе инициализации пользователю посредством дисплея предлагается выбрать различные константы системы, а именно — число каналов ЭЭГ, число параметров модели и, следовательно, модель, которая будет использоваться в алгоритме устранения, и число скорректированных и/или необработанных сигналов ЭЭГ, которые будут поданы на аппарат ЭЭГ. Некоторые сигналы ЭОГ и оценки параметров могут также подаваться на аппарат ЭЭГ. Эти константы проверяются и, если они приемлемы, используются для инициализации системы. Если константа неприемлема, используется значение по умолчанию. После инициализации программа повторяется циклично до тех пор, пока поступают приемлемые данные (рис. 14.14, *а*). Отметим, что описанная процедура применима только к системе-прототипу. В будущих системах устранения окулярных артефактов пользователю не понадобится дисплей, выбор необходимых параметров выполняется простым нажатием клавиш, и “фоновая” программа заменяется более полезной обслуживающей программой (см. подраздел 14.2.2.3). Флаг (флаг данных) устанавливается в служебной программе прерывания (рис. 14.14, *б*) при каждом прерывании и свидетельствует о том, что доступны приемлемые данные. После получения данных обновляются элементы UD-алгоритма, с ЭЭГ удаляются окулярные артефакты, и скорректированные ЭЭГ и/или необработанные данные подаются на аппарат ЭЭГ. В завершение флаг данных очищается, чтобы указать, что текущие выборки данных были успешно обработаны.

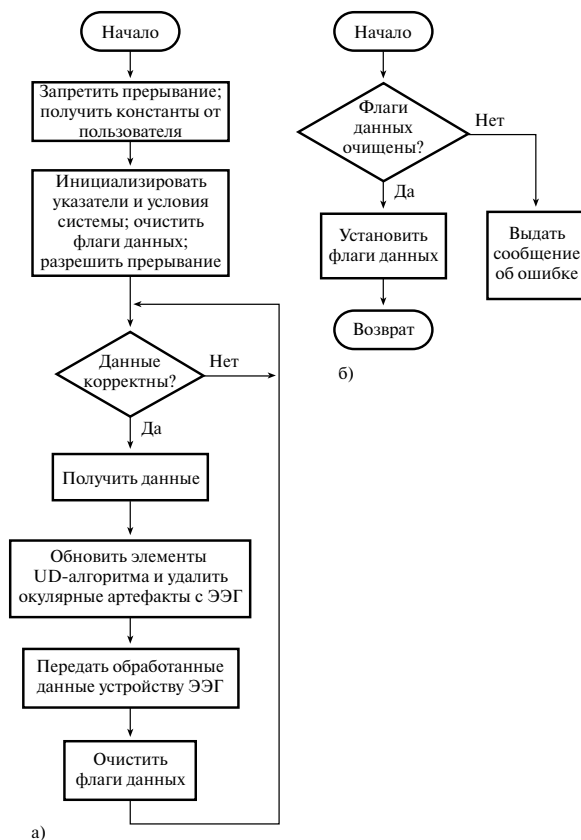


Рис. 14.14. Программное обеспечение системы устранения окулярных артефактов: а) основная программа; б) служебная программа прерывания

Если новое прерывание происходит до завершения обработки предыдущего, на дисплей выводится сообщение об ошибке, и программа останавливается. Обычно это происходит, если задано больше параметров и/или каналов ЭЭГ, чем система может обработать за интервал дискретизации 8 мс. Остановка предотвращает накопление прерываний, которые нельзя обработать, и последующий сбой системы.

Арифметические операции в системе устранения окулярных артефактов выполняются с использованием формата с плавающей запятой, что подразумевает большой динамический диапазон чисел и отсутствие проблем масштабирования, присущих арифметике с фиксированной запятой.

В данном приложении скорость критична, и наилучшим кажется аппаратное решение с плавающей запятой, однако существующие устройства с плавающей запятой и слишком дороги, и слишком медленны. Следовательно, число каналов ЭЭГ, которые можно обработать, было уменьшено, чтобы можно было использовать программные средства с плавающей запятой.

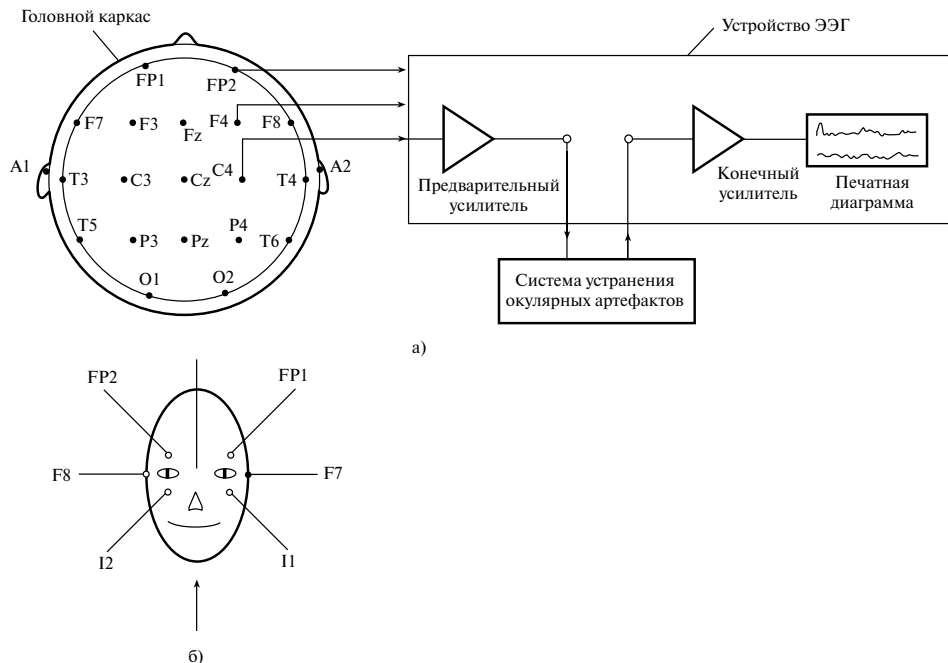


Рис. 14.15. Используемая экспериментальная установка: а) соединение с аппаратом ЭЭГ; б) размещение электродов для снятия ЭОГ

14.2.2.6. Тестирование системы и экспериментальные результаты

Предклиническая проверка системы устранения окулярных артефактов выполнялась в Freedom Fields Hospital (Плимут). В первой фазе для расширенного тестирования надежности системы использовались шесть здоровых пациентов. Данная фаза являлась также настроочной, в которой изучалось поведение системы и определялись незначительные погрешности. Во второй фазе для оценки влияния процесса устранения окулярных артефактов на пики и медленные волны использовался один неконтактный больной, ЭЭГ которого содержала пики и разрывы волны, и два пациента, один из которых был неконтактным, ЭЭГ которых содержали медленные волны. Во всех проведенных тестах пациентов просили выполнить определенные движения глазами — периодическое и случайное моргание, вертикальное и горизонтальное движение глаз.

Используя размещение электродов, как показано на рис. 14.15, было получено девять сигналов ЭЭГ: FP2–F4, F4–C4, C4–P4, FP1–F7, F3–C3, C3–P3, Fz–Cz, Cz–мочка правого уха (A2) и Cz–Pz. Затем с электродов, расположенных возле глаз, как показано на рис. 14.15, б, были получены сигналы ЭОГ.

Сигналы ЭЭГ и ЭОГ подавались на восьмиканальный аппарат ЭЭГ через головной каркас и после усиления переносились в систему устранения окулярных артефактов с помощью 37-контактного D-соединителя. После устранения окулярных артефактов из сигналов ЭЭГ на финальный усилитель аппарата ЭЭГ подавались скорректированная и

необработанная ЭЭГ (с удаленными средними) и/или ЭОГ, откуда для изучения сигналы переносились на бумагу (см. рис. 14.15, *а*).

В тестах использовалось несколько моделей, но ниже описаны только три, давшие наилучшие результаты. В двух из этих моделей применялись электроокулограммы, полученные с электродов, размещенных так, как показано на рис. 14.15, *б* (в предварительных исследованиях это давало наилучшие результаты). Итак, две использованные модели записываются следующим образом:

$$\begin{aligned} 3D \quad y(i) &= \theta_1 VR(i) + \theta_2 HR(i) + \theta_3 HL(i) + e(i) \\ 4D \quad y(i) &= \theta_1 VR(i) + \theta_2 HR(i) + \theta_3 HL(i) + \theta_4 HL(i)HR(i) + e(i). \end{aligned} \quad (14.16)$$

В третьей модели, далее именуемой *моделью 2H*, использовались пары электродов FP1–F7 и FP2–F8:

$$2H \quad y(i) = \theta_i EOGR(i) + \theta_2 EOGL(i) + e(i). \quad (14.17)$$

Следует также отметить, что ключ выбора режима в аппарате ЭЭГ можно использовать для принудительного перехода к разнообразным моделям путем выбора различных пар электродов ЭОГ, сигналы с которых подаются в каналы 1–4 (зарезервированные для ЭОГ).

Система устранения окулярных артефактов использовалась для устранения ОА из сигналов, поступающих с других электродов ЭЭГ (см. рис. 14.15, *а*), с помощью модели 2H (иногда моделей 3D и 4D). Было найдено, что во всех случаях окулярные артефакты успешно устранялись, в том числе во всех фронтальных каналах ЭЭГ, где они наиболее значительны. На рис. 14.16 приведены результаты для четырех различных электродов (Fz–Cz, Cz–A2, F4–C4 и F3–C3) в экспериментах с морганием глаз. На рис. 14.16, *а* и *б* окулярные артефакты устранялись одновременно из двух различных сигналов ЭЭГ. Из сравнения скорректированной и необработанной ЭЭГ видно, что система успешно удаляет окулярные артефакты (сравните кривые на панелях *д* и *е* с кривыми на панелях *в* и *з*).

Отметим, что для электродов ЭЭГ, расположенных ближе к задней части головы (например, Cz–Pz и C4–P4), наблюдалось незначительное засорение сигнала окулярными артефактами. В этих случаях все модели показывали равно хорошие результаты.

На рис. 14.17 показана ЭЭГ-запись неконтактного пациента, содержащая помимо окулярных артефактов эпилептический пик и всплески волн. (В данном и последующих случаях, если ЭЭГ содержит патологические волны, необработанная ЭЭГ подается непосредственно на конечный усилитель аппарата ЭЭГ и на вход системы устранения окулярных артефактов, как описывалось ранее, что разрешает использовать несмещенный анализ записей.) Сравнивая скорректированную и необработанную ЭЭГ (рис. 14.17, *з* и *д*), видим, что окулярные артефакты были удалены, но пики и волны остались.

Следует также отметить получение хороших результатов от неконтактного психически больного, ЭЭГ которого содержали только медленные волны малой амплитуды. Заслуживает внимания тот факт, что существовавшие ранее оперативные методы не подходят для работы с неконтактными пациентами.

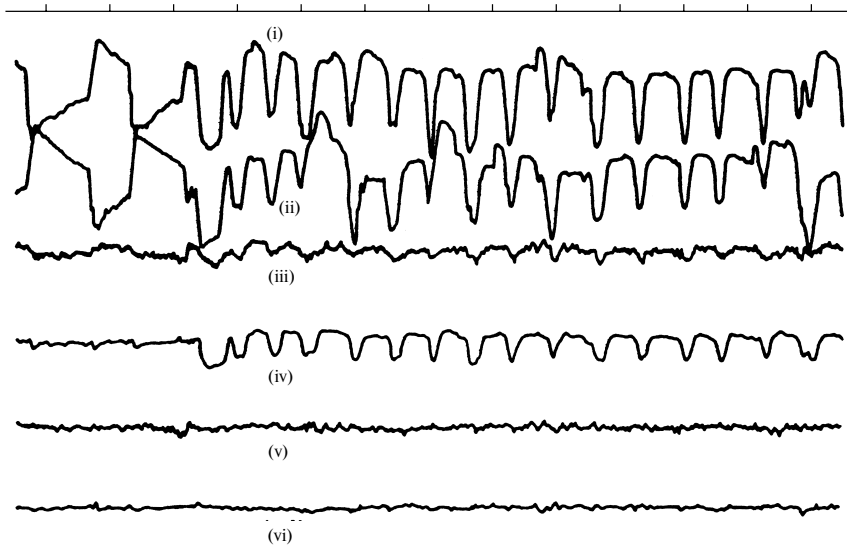


Рис. 14.16. Одновременное адаптивное удаление окулярных артефактов в паре электродов ЭЭГ в эксперименте с морганием: *a* и *б* — измеренные сигналы ЭОГ от правого и левого глаза; *в* и *г* — ЭЭГ, измеренные с электродов Cz-A2 и Fz-Cz, *д* и *е* — соответствующие ЭЭГ с удаленными артефактами

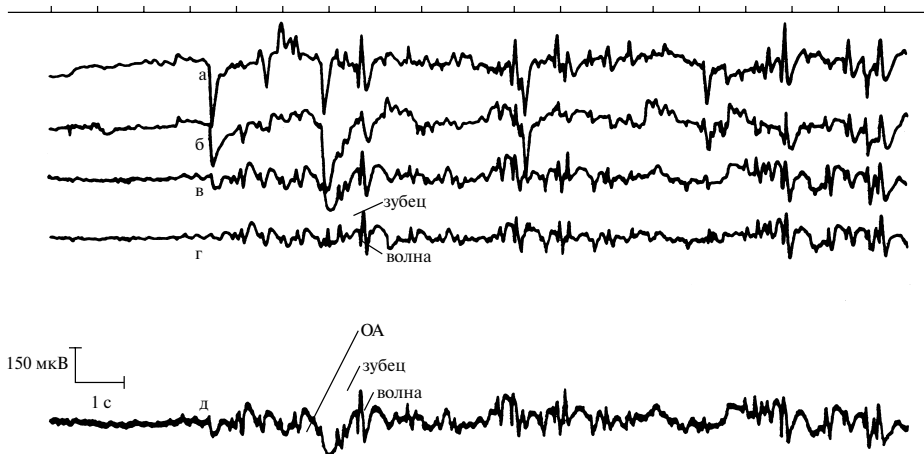


Рис. 14.17. Устранение окулярных артефактов при наличии эпилептических пиков и всплесков волн: *a* и *б* — измеренные ЭОГ для левого и правого глаза; *в* — измеренная ЭЭГ; *г* — ЭЭГ с удаленными артефактами; *д* — необработанная ЭЭГ

14.2.2.7. Обсуждение

Тесты, в которых использовались различные типы окулярных движений (моргание, вертикальное движение, горизонтальное движение), показали, что достичь приемлемого устранения окулярных артефактов, вызванных указанными движениями, во всех электродах ЭЭГ можно, используя численно устойчивый UD-алгоритм. Было отмечено, что при размещении электродов ближе в затылку наблюдается крайне незначительное загрязнение ЭЭГ со стороны окулярных артефактов, и в этих случаях все модели показывают хорошие результаты. Также отмечалось, что, хотя удовлетворительное устранение ОА достигается при вертикальном движении глаза, окулярные артефакты, вызванные другими артефактами, не устраняются целиком. Подобные результаты были получены в предварительных исследованиях, и проведенные эксперименты их подтвердили. Неспособность системы устранения окулярных артефактов полностью избавиться от рейтерных артефактов, возможно, отчасти объясняется тем, что в использованных моделях учитываются только одновременные изменения ЭОГ и ЭЭГ. Если требуется получить улучшенные результаты, можно использовать динамическую модель следующего вида (единственный вход или ЭОГ):

$$y(m) = \sum_{n=0}^{N-1} h(n)x(m-n) + e(n).$$

Результаты, полученные для больных пациентов, показывают, что при появлении патологических медленных волн и пиков при отсутствии окулярных артефактов процесс устранения ОА в общем случае влияет на них незначительно. В то же время, когда они появляются одновременно с ОА, их амплитуда может снижаться, хотя и не до нуля. Уменьшение амплитуды происходит в основном во фронтальных каналах ЭЭГ. Таким образом, может потребоваться различать ОА и медленные волны. Для решения данной проблемы нужны дальнейшие исследования.

Система устранения окулярных артефактов ограничена тем, что даже при простейшей модели (2Н) она позволяет одновременно устранять окулярные артефакты максимум из четырех сигналов ЭЭГ, что вызвано малой скоростью программ, в которых используется арифметика с плавающей запятой, — для выполнения арифметической операции обычно требуется 70 мкс. Решение этой проблемы состоит в использовании скоростных аппаратных устройств с плавающей точкой, способных выполнять арифметическую операцию за 1–2 мкс.

14.2.2.8. Выводы

Предварительные результаты, полученные на здоровых и больных пациентах, показали, что система устранения окулярных артефактов хорошо справляется со своей задачей при моргании, вертикальном и горизонтальном движении глаз и биполярной установке электродов ЭЭГ. Использование алгоритма UD-факторизации и системы с программным управлением позволяет избавиться от недостатков существовавших ранее оперативных методов устранения окулярных артефактов. Таким образом, описанная система может обрабатывать множественные артефакты, не требует действий со стороны пациента на стадии предварительной калибровки и использует критерий

устранения, основанный на чисто объективном методе. Данная система, первая в своем роде, совместима со стандартными аппаратами ЭЭГ, поэтому ее можно производить как вспомогательное оборудование. В то же время полностью оценить полезность инструмента можно только после расширенных клинических испытаний.

Хотя система УОА разработана специально для устранения окулярных артефактов с ЭЭГ, ее можно использовать как универсальную систему устранения артефактов (или шума) в большинстве физиологических ситуаций, где можно по отдельности померять зашумленный сигнал и сигнал шума. Пример — задача измерения электрокардиограммы (ЭКГ) плода в присутствии значительной зашумляющей ЭКГ матери [38]. Другой пример — устранение ОА и артефактов ЭКГ с ЭЭГ [12]. В обоих приложениях систему УОА можно использовать для устранения артефактов, возможно, после незначительных модификаций на программном и аппаратном уровнях. Должным образом запрограммированная система УОА может также использоваться в других задачах обработки сигналов, например, цифровой фильтрации.

Со времени разработки системы ОАУ 10 лет назад в сфере ЦОС произошли существенные изменения, особенно в сфере аппаратного обеспечения ЦОС. Очевидно, что если описанную систему реализовать сейчас, можно будет использовать хороший чип ЦОС. В настоящее время для срочных приложений, подобных описанным выше, хорошо подходят такие цифровые процессоры сигналов с плавающей запятой, как TMS320C30 или TMS320C40. Напомним, что, несмотря на изменения в ЦОС, общие принципы разработки и возникающие вопросы остались прежними. Надеемся, что наш опыт будет полезен читателям.

14.2.3. Выравнивание цифровых аудиосигналов³

Выравнивание аудиосигналов является важным функциональным требованием микшерных пультов, используемых во многих профессиональных и полупрофессиональных аудиоприложениях, например, при студийной записи, усилении звука в системах местного радиовещания и ширококвещательных станциях. Аудиоэквалайзер — это, по сути, набор фильтров с регулируемыми частотными характеристиками, который используется для формирования требуемого спектра аудиосигналов. В традиционном микшерном пульте выравнивание аудиосигналов выполняется за счет аналоговой фильтрации, но наблюдается тенденция к переходу на чисто цифровые микшерные пульта, поскольку это предлагает улучшенное качество звука и потенциальное снижение стоимости производства в будущем. В полностью цифровом микшере аналоговые фильтры заменяются эквивалентными цифровыми фильтрами реального времени. Ниже описан полупараметрический эквалайзер реального времени, реализованный с использованием высокоскоростного цифрового процессора с плавающей запятой TMS320C30.

Стандартный параметрический эквалайзер позволяет пользователю выбирать конкретную частоту в полосе аудиочастот и настраивать уровень аудиосигнала на одной частоте или в диапазоне частот. Используются фильтры трех основных типов.

- *Звонковый фильтр.* Позволяет усиливать или подавлять определенную частоту в полосе аудиочастот. Звонковый фильтр — это, по сути, полосовой фильтр с настра-

³ Данный проект основан на выпускной работе одного из наших бывших студентов — Робина Кларка (Robin Clark).

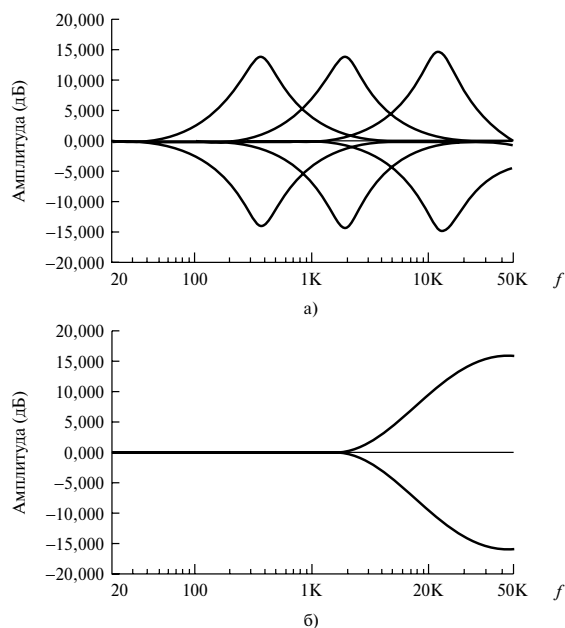


Рис. 14.18. Амплитудная характеристика звонкового фильтра для различных центральных частот и шириин полос (панель *а*). Амплитудная характеристика полочного фильтра (панель *б*)

иваемым усилением, добротностью и центральной частотой. Центральная частота может меняться в диапазоне 20–16 кГц, добротность — в диапазоне 0,5–3, а усиление — в интервале ± 15 дБ. Пример характеристики звонкового фильтра приведен на рис. 14.18, *а*.

- **Полочный фильтр.** Позволяет регулировать усиление и частоту среза эквалайзера в интервале, принадлежащем либо диапазону верхних частот, либо диапазону нижних частот аудиополосы. Полочковые фильтры нижних частот используются для усиления или подавления полосы низких частот, например, между 20 и 500 Гц, а полочковые фильтры верхних частот — для усиления или подавления полосы высоких частот, например, между 1,6 и 16 кГц. Обычная регулировка верхних и нижних звуковых частот в домашних аудиосистемах фактически выполняется с помощью полочковых фильтров с фиксированными характеристиками. Типичная характеристика указанного фильтра приведена на рис. 14.18, *б*.
- **Пропускающие фильтры.** Это, по сути, фильтры нижних частот и верхних частот с фиксированными частотами среза, которые используются для удаления низко- и высокочастотного шума из аудиосигнала.

Полное выравнивание выполняется за счет совместного использования основных фильтров. В аналоговых эквалайзерах характеристики фильтра (усиление, центральная частота, добротность и т.д.) регулируются пользователем посредством интерактивного управления (переменные резисторы, *реостаты*). В цифровой реализации настройка

выполняется путем замены в реальном времени коэффициентов цифрового фильтра в ответ на изменение параметров эквалайзера.

Анализ типичного аналогового параметрического эквалайзера показывает, что фильтры всех указанных выше типов можно рассматривать как фильтр Баттерворта, передаточные функции которых имеют следующую форму [6]: для звонкового фильтра —

$$H(s) = \frac{s^2 + A\omega_n s + \omega_n^2}{s^2 + B\omega_n s + \omega_n^2}, \quad (14.18, a)$$

для низкочастотного полочкового фильтра —

$$H(s) = \frac{s^2 + 2A\omega_n s + A^2\omega_n^2}{s^2 + 2B\omega_n s + B^2\omega_n^2}, \quad (14.18, б)$$

для высокочастотного полочкового фильтра —

$$H(s) = \frac{A^2 s^2 + 2A\omega_n s + \omega_n^2}{B^2 s^2 + 2B\omega_n s + \omega_n^2}, \quad (14.18, в)$$

где

$$A = C/Q,$$

$$B = 1/Q.$$

Чтобы достичь производительности, сравнимой с производительностью традиционного аналогового эквалайзера, указанные выше аналоговые фильтры заменяются эквивалентными цифровыми путем преобразования приведенных передаточных функций, записанных на s -плоскости, с помощью билинейного z -преобразования (см. главу 8). Все получающиеся передаточные функции на z -плоскости можно записать в форме

$$H(z) = \frac{az^2 + bz + c}{dz^2 + ez + f}, \quad (14.19)$$

где

$$a = P^2 + AP + \omega_n^2, b = 2\omega_n^2 - 2P^2, c = P^2 - AP + \omega_n^2,$$

$$d = P^2 + BP + \omega_n^2, e = 2\omega_n^2 - 2P^2, f = P^2 - BP + \omega_n^2,$$

$$P = \frac{\omega_n}{\omega_p}, \omega_p = \frac{2}{T} \operatorname{tg} \left(\frac{\omega_n T}{2} \right).$$

Модельные задачи показывают, что, поскольку параметры эквалайзера настраиваются в диапазоне аудиочастот, для учета большого разброса в диапазоне значений коэффициентов фильтра должна использоваться арифметика с плавающей запятой. Такая арифметика также привлекательна тем, что позволяет легче пересчитывать коэффициенты фильтра с помощью билинейного z -преобразования в “реальном времени”, чтобы можно было выполнять оперативную настройку характеристик эквалайзера.

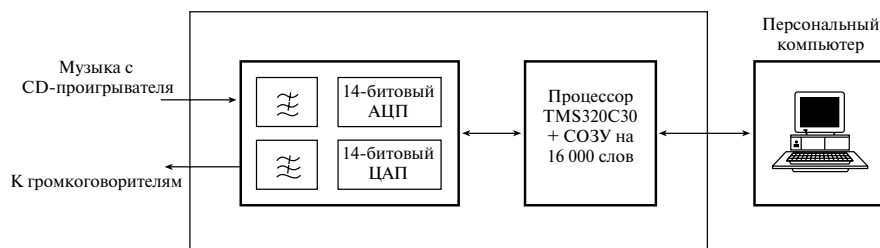


Рис. 14.19. Упрощенная блок-схема демонстрационного модуля TMS320C30

Эквалайзер был реализован на компьютерном демонстрационном модуле TMS320C30, содержащем процессор TMS320C30, 14-битовый модуль АЦП/ЦАП и пакет программной разработки. Упрощенная блок-схема параметрического эквалайзера на основе TMS320C30 показана на рис. 14.19. Аналоговый аудиосигнал (например, поступающий с проигрывателя компакт-дисков) оцифровывается до 14 бит с частотой порядка 18,9 кГц и передается на процессор, где он проходит цифровую фильтрацию с помощью звонкового, полочкового и/или пропускного фильтра. Для настройки параметров эквалайзера (усиление, частота, тип фильтра) используется клавиатура, а на дисплее динамически отображается частотная характеристика эквалайзера.

В демонстрационном модуле и персональном компьютере проходит несколько процессов. Процессор TMS320C30 выполняет фильтрацию, необходимую для выравнивания, и пересчитывает коэффициенты фильтра для эквалайзера, когда пользователь настраивает его параметры. Пересчет коэффициентов происходит в фоновом режиме, а фильтрация управляется прерываниями. Программа написана на языке ANSI C, а арифметические операции выполняются с использованием плавающей запятой (24-битовая мантисса и 8-битовая экспонента). Фрагменты кодов C для фильтрации и расчета коэффициентов можно найти в книге [20] (подробности см. в предисловии).

Производительность эквалайзера оценивалась для различных выбранных параметров с использованием музыки от проигрывателя компакт-дисков. Сделан вывод: эквалайзер достаточно эффективен. На настоящее время ограничением является только то, что демонстрационный модуль допускает частоту дискретизации максимум 18,9 кГц, тогда как для профессиональной работы требуется частота 44,1 кГц.

14.3. Модельные задачи

В большинстве курсов по ЦОС выигрышной стратегией часто является постановка одной или нескольких серьезных практических задач, работать над решением которых можно индивидуально или группами. Предназначение подобных задач — дать студенту возможность получить более глубокие знания ЦОС. Данные задачи построены на нескольких аспектах концепций ЦОС и могут ставиться ближе к концу курса или ранее (в последнем случае студент овладеет концепциями как раз тогда, когда они ему станут необходимы).

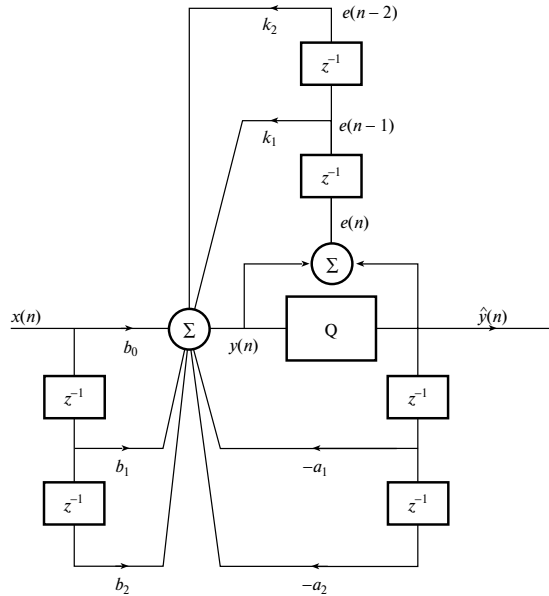


Рис. 14.20. Схема снижения шума второго порядка

В данном разделе представлено несколько подобных серьезных задач, которые авторы нашли полезными в некоторых прочитанных ими курсах. Указано также, какие обучающие цели ставились, и какие подзадачи предполагалось использовать.

1. Схемы снижения шума округления для цифровых БИХ-фильтров с фиксированной запятой

На рис. 14.20 показана структура цифрового БИХ-фильтра второго порядка со схемой обратной связи по ошибке, предназначенной для уменьшения ошибок округления произведений. Фильтр требуется реализовать с использованием арифметики с дополнением до двух на процессоре ЦОС с архитектурой “ $B/2B$ бит”. Коэффициенты и другие переменные хранятся как B -битовые слова, и произведения квантуются после суммирования. Коэффициенты схемы обратной связи по ошибке могут принимать одно из указанных значений: $0, \pm 0,25, \pm 0,5, \pm 0,75, \pm 1, \pm 1,25, \pm 1,5, \pm 1,75, \pm 2$.

А. Выведите общую формулу для следующих величин:

- 1) передаточная функция, связывающая источник шума округления $e(n)$ и выход фильтра $y(n)$;
- 2) выходная мощность шума, порожденного ошибками округления;
- 3) отношение сигнал-шум на выходе фильтра; предполагается, что входной сигнал $x(n)$ — синусоида с известной частотой, квантованная до B бит;
- 4) положения нулей цепи обратной связи.

Б. Разработайте

- 1) алгоритм вычисления входной мощности шума, порожденного ошибкой округления, отношения сигнал-шум на выходе фильтра и положения нулей цепи об-

ратной связи по данным значениям некантованных коэффициентов фильтра, коэффициентов обратной связи и длины слова системы B ;

- 2) программу MATLAB, реализующую алгоритм Б.1.
- В. Используйте написанную программу MATLAB плюс другие утилиты MATLAB, чтобы исследовать влияние схемы обратной связи по ошибке на уровень шума округления произведений для каждого из указанных ниже фильтров (задачу следует повторить для двух ситуаций : $B = 16$ бит и $B = 24$ бит) [9].
 1. Режекторный фильтр для обработки аудиосигналов, который характеризуется следующими параметрами:

коэффициенты фильтра	$b_0 = 0,996\ 450\ 761\ 790,$
	$b_1 = -1,992\ 821\ 454\ 486,$
	$b_2 = 0,996\ 443\ 656\ 208,$
	$a_1 = -1,992\ 821\ 454\ 486,$
	$a_2 = 0,992\ 894\ 417\ 998;$
частота дискретизации	48 кГц.
 2. “Усиливающий” фильтр для обработки аудиосигналов, который характеризуется следующими параметрами:

коэффициенты фильтра	$b_0 = 0,996\ 450\ 761\ 790,$
	$b_1 = -1,992\ 821\ 454\ 486,$
	$b_2 = 0,996\ 443\ 656\ 208,$
	$a_1 = 1,992\ 821\ 454\ 486,$
	$a_2 = -0,992\ 894\ 417\ 998;$
частота дискретизации	48 кГц;
центральная частота	15 кГц.
 3. Фильтр верхних частот, который характеризуется следующими параметрами:

коэффициенты фильтра	$b_0 = 0,292\ 893,$	$b_1 = -0,585\ 786,$
	$b_2 = 0,292\ 893,$	
	$a_1 = 0,$	$a_2 = 0,171\ 572\ 8;$
частота дискретизации	8 кГц;	
граница полосы подавления	500 кГц;	
граница полосы пропускания	2 кГц.	
- Г. Исследование должно включать следующие элементы:
 - 1) для каждого фильтра и длины слова B следует рассчитать и занести в таблицу выходную мощность шума, отношение сигнал-шум и положение нулей цепи обратной связи для каждой пары коэффициентов обратной связи, приведенной в табл. 14.2;
 - 2) расчет пары коэффициентов обратной связи, которая дает наименьшую выходную мощность шума (из всех возможных пар и разрешенных коэффициентов обратной связи, указанных в п. А);
 - 3) графики частотной характеристики, соответствующей передаточной функции шума для каждой пары коэффициентов обратной связи в табл. 14.2 и для лучшей пары коэффициентов, найденной в п. 2;

Таблица 14.2. Значение коэффициентов обратной связи по ошибке

№	$k1'$	$k2'$
1	0	-1
2	0	1
3	-1	0
4	-1	-1
5	-1	1
6	1	0
7	1	-1
8	-2	-1
9	2	-1
10	0	0

4) краткий отчет (максимум 3–4 страницы, включая диаграммы и программы) об исследовании и полученных результатах. Ниже перечислены необходимые элементы отчета.

- Вступление с формулировкой проблемы шума округления в системах ЦОС.
- Детали ответов по п. А–Г.
- Критическое обсуждение полученных результатов и рекомендации по выбору коэффициентов обратной связи по ошибке на практике.
- Критическое обсуждение ограничений схем обратной связи по ошибке при использовании для снижения шума округления.
- Обсуждение других преимуществ схем обратной связи по ошибке помимо снижения шума округления.
- Резюме по оценке специфических параметров, выполненной в ходе решения задачи.
- Используемые программы MATLAB.

2. Пример разработки фильтра с обработкой при нескольких скоростях

Исходные данные

Данный пример основан на одной из тем фильтрации на базе MATLAB, разработанной одним из наших бывших студентов, Ником Отрамом [28], для одного из наших курсов. Ставятся такие обучающие цели:

- глубже понять концепции обработки сигналов при нескольких скоростях;
- продемонстрировать некоторые ограничения КИХ-фильтров в случаях, когда требуется узкая амплитудная характеристика;
- разобраться на собственном опыте, как создаются и внедряются фильтры с обработкой на нескольких скоростях.

В данном примере, чтобы облегчить процесс обучения, используется электрокардиограмма (ЭКГ) плода. Форма ЭКГ плода, представляющая медицинский интерес, искажается низкочастотным артефактом, известным как *смещение базовой линии* (baseline shift) (см. рис. 14.21). Смещения базовой линии препятствуют точному измерению и анализу

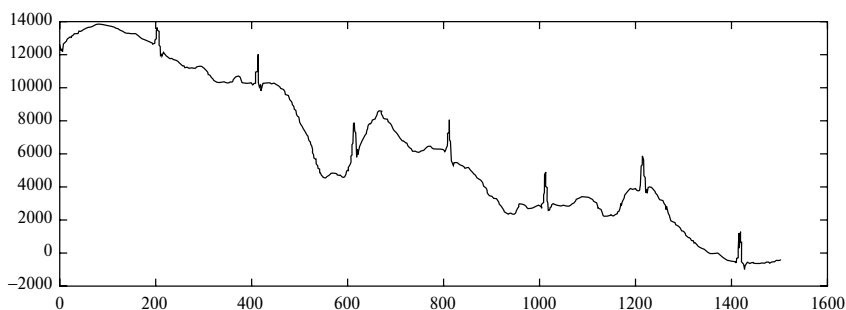


Рис. 14.21. Острые смещения базовой линии

особенностей ЭКГ плода. Сложность состоит в том, что очистка сигнала не должна искажать низкочастотную информацию, представляющую медицинский интерес. К сожалению, значительная часть энергии артефакта приходится на область ниже 3 Гц.

В данном исследовании даны два набора данных: `good.ecg`, содержащий качественные записи ЭКГ плода практически без смещения базовой линии, и `poor.ecg`, содержащий ЭКГ плода со значительными смещениями базовой линии. Наборы состоят из 12-битовых значений, дискретизованных с частотой 500 Гц.

Задача

Основная задача состоит в разработке схемы обработки сигналов с помощью КИХ-фильтров с линейной фазовой характеристикой, устраняющей смещение базовой линии без искажения формы ЭКГ.

Для решения основной задачи используются такие подзадачи:

- а) написать программу MATLAB для считывания файлов данных ЭКГ в два отдельных массива и отображения на некоторой оси первых 4000 выборок каждого сигнала;
- б) оценить, используя функцию MATLAB `remezord`, число коэффициентов, требуемых для соответствия спецификациям, приведенным на рис. 14.22; прокомментировать полученные результаты и задачу разработки узкополосных КИХ-фильтров;
- в) разработать и реализовать фильтр с обработкой при нескольких скоростях в MATLAB, используя при необходимости методы децимации и интерполяции;
- г) протестировать фильтр, используя данные, проверить, не искажается ли сигнал ЭКГ, и определить задержку прохождения сигнала через систему с обработкой при нескольких скоростях;
- д) составить краткий отчет о выполненном задании, в котором должны быть описаны разработка и тестирование фильтра с обработкой при нескольких скоростях.

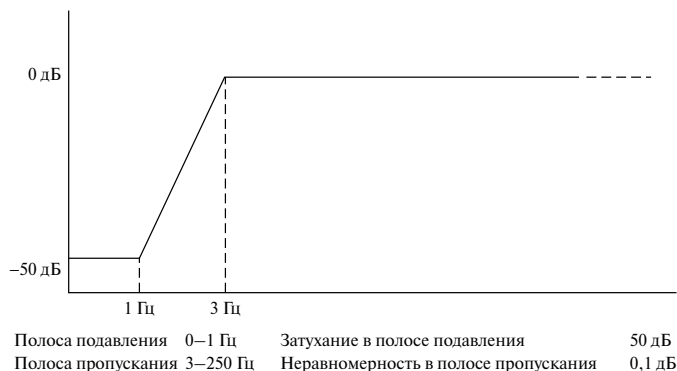


Рис. 14.22. Спецификация фильтра для устранения смещений базовой линии

3. Разработка и реализация цифрового фильтра с использованием MATLAB и процессора DSP56002

Исходные данные

Цель данной задачи — позволить студенту самому получить необходимые навыки разработки и реализации КИХ-фильтров с помощью MATLAB и процессора ЦОС с фиксированной запятой. Задача основана на лабораторной работе наших студентов Брагима Хамадичарефа (Brahim Hamadicharef), Робина Кларка (Robin Clark) и Эдди Риддингтона (Eddie Riddington) в одном из наших курсов по ЦОС.

Специальные обучающие цели:

- 1) использовать MATLAB для
 - расчета коэффициентов и частотной характеристики КИХ-фильтров для данного набора спецификаций;
 - исследования влияния квантования коэффициентов на частотную характеристику;
- 2) реализовать КИХ-фильтры на базе процессора ЦОС с фиксированной запятой для
 - ознакомления со средствами разработки ЦОС;
 - разработки простой программы на языке ассемблера для КИХ-фильтра с фиксированной запятой;
 - тестирования и демонстрации фильтрации в реальном времени.

Задача

Требуется КИХ-фильтр с линейной фазовой характеристикой, устраняющий помеху с частотой 1 кГц из аудиосигнала. Фильтр должен удовлетворять следующим требованиям:

неравномерность в полосе пропускания	0,5 дБ,
затухание в полосе подавления	25 дБ,
полоса пропускания	900–1100 Гц,
границы полосы подавления	990 и 1010 Гц,
частота дискретизации	8 кГц.

Фильтр требуется разработать и реализовать, используя MATLAB и процессор ЦОС с фиксированной запятой Motorola DSP56002. Ниже перечислены основные подзадачи.

- А. Вычислить коэффициенты подходящего КИХ-фильтра, используя MATLAB и оптимальный метод. *Примечание:* нужно использовать функцию `remesord` для оценки длины фильтра N , рассчитать коэффициенты фильтра, используя функцию `remez`, и нарисовать частотную характеристику фильтра, применяя функцию `freqz`. При определении параметров неравномерности могут быть полезны следующие соотношения:

$$R_p = -20 \lg \frac{1 - \delta_p}{1 + \delta_p} - \text{неравномерность в полосе пропускания в дБ,}$$

$$R_s = -20 \lg \frac{\delta_s}{1 + \delta_s} - \text{затухание в полосе подавления в дБ,}$$

где

δ_p — неравномерность в полосе пропускания
 δ_s — неравномерность в полосе подавления

- Б. Определить подходящие длины слов коэффициентов фильтра, проанализировав эффекты квантования коэффициентов до 4, 8, 16 и 24 бит на частотную характеристику фильтра. *Примечание:* для квантования коэффициентов фильтра до B -битовых чисел в арифметике с дополнением до двух с фиксированной запятой каждый коэффициент просто умножается на 2^{B-1} , а затем округляется до ближайшего целого. Разность между реальными коэффициентами и представлением в форме с плавающей запятой является мерой ошибки квантования коэффициентов.
- В. Реализовать фильтр на демонстрационном модуле DSP56002 и проверить корректность его работы. *Примечание:* в качестве введения в цифровую КИХ-фильтрацию с использованием DSP56002 рассмотрите простой трехточечный КИХ-фильтр, изображенный на рис. 14.23. КИХ-фильтр характеризуется следующим уравнением:

$$y(n) = h(0)x(n) + h(1)x(n-1) + h(2)x(n-2).$$

В табл. 14.3 представлена исходная карта памяти с коэффициентами и данными фильтра. Регистры `r0` и `r4` используются как указатели на коэффициенты и ячейки памяти соответственно. Используется круговая адресация. В программе 14.1 приведен код DSP 56002 для реализации простого фильтра. В столбце комментариев объясняется действие программы 14.1.

Для КИХ-фильтра с большим числом коэффициентов эффективнее использовать команду повтора DSP56002 вместе с командой `mac`. Ниже приводится фрагмент соответствующего кода DSP560002.

```
clr    a    x(r0)+, x0 y(r4)+, y0
rep    #N-1
mac    x0, y0, a    x(r0)+, x0 y(r4)+, y0
macr   x0, y0, a    (r0)-
move   a, y:Output
```

Г. Напишите краткий отчет по разработке и реализации фильтра, включая листинги программ MATLAB и DSP56002.

Таблица 14.3. Исходная карта памяти

Память X		Память H	
$h(2)$	$\leftarrow r0$	$x(n-2)$	$\leftarrow r4$
$h(1)$		$x(n-1)$	
$h(0)$		$x(n)$	

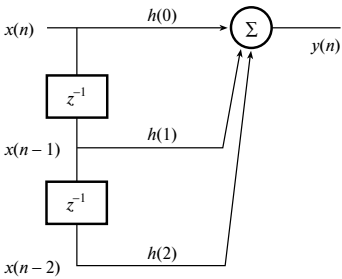


Рис. 14.23. Простой трехточечный КИХ-фильтр

Программа 14.1. Код DSP56002 для простого трехточечного КИХ-фильтра

```
move   x: (r0)+, x0    ; скопировать h(2) в регистр x0
                        ; и увеличить указатель r0.
move   y: (r4)+, y0    ; скопировать x(n-1) в регистр y0
                        ; и увеличить указатель r4.
mpy    x0, y0, a x: (r0)+, x0 y: (r4)+, y0 ; перемножить h(2) и x(n-2),
                        ; результат занести в накопитель a.
                        ; скопировать h(1) в регистр x0 и увеличить r0.
                        ; скопировать x(n-1) в регистр y0 и увеличить r4.
mac    x0, y0, a x: (r0)+, x0 y: (r4)+, y0 ; перемножить h(1) и x(n-1),
                        ; результат прибавить к значению в накопителе a.
                        ; скопировать a0 в регистр x0 и увеличить r0.
                        ; скопировать x(n) в регистр y0 и увеличить r4.
mac    x0, y0, a y1, y: (r4)+ ; перемножить h(0) и x(n),
                        ; результат прибавить к значению в накопителе a.
                        ; накопитель a содержит выходную выборку y(n).
                        ; x(n-2), x(n-1) и x(n) теперь можно рассматривать
                        ; как x(n-3), x(n-2) and x(n-1). Поверх x(n-3)
                        ; записывается следующая входная выборка, которая
                        ; хранится в y(1).
```

14.4. Вопросы по ЦОС для компьютерных тестов

В данном разделе описан подход, который авторы нашли полезным для быстрой оценки знаний большой группы студентов и определения областей, требующих дополнительного изучения. Важной особенностью данного подхода является то, что он позволяет студентам пересмотреть материалы по ЦОС и проверить их понимание специфических вопросов.

По сути, мы разработали две группы вопросов с вариантами ответов, охватывающих основные и углубленные (прикладные) темы ЦОС. Каждый год группы вопросов обновлялись и увеличивались. Группа с вопросами по базовым темам ЦОС используется для тестирования студентов в первом семестре, а группа с вопросами углубленной (практической) направленности — во втором. В каждом случае примерно за две недели до тестирования студентам раздавались копии всех возможных вопросов без ответов. Никакая прямая помощь при подготовке студентам не предоставлялась. Для проверки и автоматической оценки теста использовалось средство Questionmark. Каждый тест рассчитан на полчаса (при запуске теста автоматически запускается таймер) и состоит примерно из 30 вопросов, которые случайным образом выбираются из группы.

Все вопросы относятся к классу выбора одного или нескольких правильных ответов из предложенных. При оценке результатов тестирования использовалась следующая балльная система (которую, впрочем, легко изменить):

Выбор одного ответа	Правильный ответ	+1
	Неправильный ответ	-1
	Без ответа	0
Выбор нескольких ответов	Правильный ответ	+1
	Неправильный ответ	-1
	Без ответа	0

Ниже приводятся примеры вопросов с одним правильным ответом. Более подробно все вопросы рассмотрены в книге [20] (подробности см. в предисловии).

14.4.1. Примеры вопросов по базовым темам ЦОС

Вопрос 1

Аналоговый сигнал с целью анализа спектра дискретизируется с частотой 125 Гц и получаются 1024 выборки данных. Чему равно расстояние по частоте между спектральными выборками?

1. 0,04142 Гц.
2. 0,12207 Гц.
3. 0,00098 Гц.
4. 0,0080 Гц.

Вопрос 2

Цифровой фильтр называется БИХ-фильтром, если

- 1) его полюса располагаются вне единичной окружности;
- 2) один или несколько коэффициентов знаменателя не равны нулю;
- 3) текущий выход зависит от предыдущего выхода;
- 4) его выход осциллирует.

Вопрос 3

Передаточная функция фильтра дискретного времени $H(z)$ равна

$$H(z) = 1/(1 + 0,454456z^{-1} + 0,269259z^{-2}).$$

Найдите полюса фильтра.

1. Полюса: $0,45 + 0,75i$ и $0,45 - 0,75i$.
2. Полюса: $-0,2272 + 0,64664i$ и $-0,2272 - 0,4664i$.
3. Полюса: $-0,9452 + 0,5i$ и $-0,9452 - 0,5i$

Вопрос 4

Цифровой полосовой фильтр можно разработать, преобразовав подходящий аналоговый фильтр-прототип нижних частот в полосовой фильтр. Данное преобразование

- 1) в два раза уменьшает порядок аналогового фильтра нижних частот;
- 2) удваивает порядок аналогового фильтра нижних частот;
- 3) отображает частоты среза фильтра нижних частот F_c и $-F_c$ в частоты среза полосового фильтра F_{c2} и F_{c1} ;
- 4) отображает частоты среза фильтра нижних частот F_c и $-F_c$ в центральную частоту полосового фильтра.

Вопрос 5

Полюса дискретного фильтра расположены в точках $z = 0,75i$ и $z = -0,75i$, а нули — в точках $z = -1$ и $z = 1$. Чему равна амплитудная характеристика на четверти частоты дискретизации?

1. 1,52.
2. 4,57.
3. 7,15.

Вопрос 6

Квантование коэффициента $-0,1743$ до 8 бит дает

- 1) -22 ;
- 2) -23 ;
- 3) $+22$;
- 4) 23×10^{-7} .

Вопрос 7

Какое из приведенных утверждение справедливо в сфере ЦОС?

1. Частота Найквиста равна половине частоты дискретизации.
2. Частота Найквиста равна частоте дискретизации.
3. Частота Найквиста равна максимальной частоте сигнала.
4. Частота Найквиста равна половине частоты передискретизации.

Вопрос 8

Передаточная функция фильтра дискретного времени $H(z)$ равна

$$H(z) = (1 - 1,6z^{-1} + z^{-2}) / (1 - 1,5z^{-1} + 0,8z^{-2}).$$

Чему равна амплитудная характеристика на 0 Гц?

1. 0.
2. 1,33.
3. 1.
4. 1,6.

Вопрос 9

Передаточная функция фильтра дискретного времени $H(z)$ равна

$$H(z) = (1 - 1,6z^{-1} + z^{-2}) / (1 - 1,5z^{-1} + 0,8z^{-2}).$$

Чему равны радиусы полюсов и нулей?

1. 0,9; 1.
2. 0,81; 1.
3. 1; 0,81.
4. 1; 1.
5. 0,5; 0,5.

Вопрос 10

Какое из приведенных ниже высказываний справедливо для разработки БИХ-фильтров методом билинейного z -преобразования (БЗП)?

1. Деформирующий эффект БЗП проявляется сильнее по мере приближения к частоте Найквиста.
2. Предварительная деформация компенсирует искажение, которое вводит БЗП, во всей полосе частот.
3. Предварительная деформация компенсирует искажение, которое вводит БЗП, на определенных частотах.
4. Частотное масштабирование устраняет искажающие влияние БЗП на частотную характеристику.

Вопрос 11

Требуется, чтобы цифровой БИХ-фильтр верхних частот удовлетворял следующим спецификациям:

граница полосы подавления	1 кГц,
граница полосы пропускания	2 кГц,
частота дискретизации	16 кГц,
неравномерность в полосе пропускания	3 дБ,
затухание в полосе подавления	30 дБ.

Чему равны частоты полосы пропускания и подавления подходящего НЧ-фильтра-прототипа? Предполагается, что используется метод билинейного z -преобразования.

1. 0,0414; 0,0198.
2. 1; 2.
3. 1; 2,0813.
4. 2; 1.
5. 0,0198; 0,0414.

Вопрос 12

Аналоговый сигнал на входе системы ЦОС ограничивается по полосе фильтром Баттерворта четвертого порядка с частотой среза 10 кГц, после чего дискретизируется с частотой 50 кГц. Предполагается, что сигнал широкополосный. Чему равен уровень ошибки наложения на 10 кГц?

1. 0,156.
2. 0,026.
3. 0,04.
4. 0,707.

Вопрос 13

Аналоговый сигнал на входе системы ЦОС ограничивается по полосе фильтром Баттерворта четвертого порядка с частотой среза 10 кГц, после чего дискретизируется. Предполагается, что сигнал широкополосный. Определите минимальную частоту дискретизации, при которой на 10 кГц отношение сигнала к ошибке наложения составляет 20 дБ.

1. 10 кГц.
2. 20 кГц.
3. 29,39 кГц.
4. 19,39 кГц.
5. 50 кГц.

Вопрос 14

Компоненты аналогового аудиосигнала принадлежат диапазону 0–24 кГц. Сигнал дискретизируется с частотой 3,072 МГц. Чему равен коэффициент передискретизации?

1. 32.
2. 64.

3. 128.

4. 256.

Вопрос 15

Аналоговый полосовой сигнал дискретизируется согласно теореме о дискретном представлении. Пусть сигнал занимает полосу 48–52 кГц. Чему равна минимальная теоретическая частота дискретизации, позволяющая избежать наложения?

1. 104 кГц.

2. 96 кГц.

3. 8 кГц.

4. 16 кГц.

Вопрос 16

Последовательность данных, состоящая из 32 выборок, записана в памяти в естественном порядке. Если последовательность данных последовательно скремблируется с помощью стандартного алгоритма обращения битов, какими будут новые индексы выборок $x(7)$ и $x(13)$? Предполагается, что используется 32-точечный процессор БПФ.

1. $x(11)$ и $x(14)$.

2. $x(22)$ и $x(28)$.

3. $x(7)$ и $x(13)$.

4. Ничего из вышеперечисленного.

Вопрос 17

Даны коэффициенты 4-точечного ДПФ.

1. $X(0) = 1, 5$.

2. $X(1) = 1 - 0, 5i$.

3. $X(2) = 0, 5$.

4. $X(3) = 1 + 0, 5i$.

Чему равна эквивалентная последовательность дискретного времени?

1. $x(0) = 1, x(1) = 0, x(2) = 1, x(3) = 0$.

2. $x(0) = 0, 5, x(1) = 1, x(2) = 0, x(3) = 0$.

3. $x(0) = 1, x(1) = 0, 5, x(2) = 0, x(3) = 0$.

4. $x(0) = 0, 5, x(1) = 0, 5, x(2) = 0, x(3) = 0, 5$.

5. Ничего из вышеперечисленного.

Вопрос 18

Дано четыре первых выходных выборок 8-точечного двоичного БПФ.

1. $X(0) = 27$.

2. $X(1) = -4 + 3i$.

3. $X(2) = 4 + i$.

4. $X(3) = 0 - 5i$.

Какое из приведенных ниже выражений справедливо?

1. Значение $X(7) = 0 + 5i$.
2. Значение $X(7) = -4 - i$.
3. Значение $X(7) = -3i$.
4. Значение выходной последовательности на 0 Гц равно 27.
5. Ничего из вышеперечисленного.

14.4.2. Примеры вопросов по углубленным (практическим) темам ЦОС

Вопрос 1

Какое из приведенных ниже утверждений справедливо? В цифровой связи минимально устойчивые БИХ-фильтры используются для восстановления тактовой частоты, потому что

- 1) их полюса и нули близки к единичной окружности;
- 2) их импульсная характеристика медленно затухает со временем;
- 3) при этом тактовый сигнал присутствует даже при отсутствии переходов во входном потоке данных в течение относительно длительных периодов времени;
- 4) их характеристики не искажаются при старении или изменении температуры;
- 5) они подходят для приложений, включающих пакетную передачу.

Вопрос 2

Скорость передачи данных системы цифровой связи равна 56 Кбод, а частота дискретизации — 444 000 выборок/с. В системе используется БИХ-фильтр с полосой 25 Гц, задача которого — восстановление тактовой частоты. Какое из приведенных ниже утверждений справедливо?

1. Полюса фильтра расположены в точках с координатами $r = 0,9998$, $\phi = \pm 45^\circ$.
2. Полюса фильтра расположены в точках с координатами $r = 0,9998$, $\phi = 45^\circ$.
3. Полюса фильтра расположены в точках с координатами $r = 0,9998$, $\phi = \pm 0,785^\circ$.
4. Полюса фильтра описываются коэффициентами $a_1 = -1,413965$, $a_2 = 0,999825$.
5. Полюса фильтра описываются коэффициентами $a_1 = -1,413965$, $a_2 = 0,999649$.
6. Фильтр восстановления тактовой частоты — полосовой.
7. Фильтр восстановления тактовой частоты — фильтр нижних частот.

Вопрос 3

Скорость передачи данных системы цифровой связи равна 28 Кбод, а частота дискретизации — 224 000 выборок/с. В системе используется БИХ-фильтр с полосой 100 Гц и коэффициентами знаменателя $a_1 = -1,412230$, $a_2 = 0,997196$, задача которого — восстановление тактовой частоты. Какое из приведенных ниже утверждений справедливо, если коэффициенты фильтра квантованы до 8-битовых чисел с фиксированной запятой?

1. Коэффициенты с фиксированной запятой равны $a'_1 = -90$, $a'_2 = 63$.
2. Коэффициенты с фиксированной запятой равны $a'_1 = -181$, $a'_2 = 127$.

3. В дробной форме записи квантованные коэффициенты равны $a'_1 = -1,40625$ и $a'_2 = 0,984375$.
4. В дробной форме записи квантованные коэффициенты равны $a'_1 = -1,414062$ и $a'_2 = 0,992187$.
5. Ничего из вышеперечисленного.

Вопрос 4

БИХ-фильтры с фиксированной запятой реализуются с использованием канонических звеньев второго порядка. Укажите истинные выражения.

1. Ошибки переполнения возникают из-за того, что результат сложения превышает разрешенную длину слова.
2. Ошибки переполнения могут привести к самоподдерживающимся осцилляциям.
3. В общем случае масштабирование в частотной области (норма Чебышева) устраняет ошибки переполнения.
4. Методы обратной связи по ошибке или формирования спектра шума позволяют уменьшить ошибки переполнения.
5. Масштабирование для предотвращения переполнения улучшает отношение сигнал-шум на выходе фильтра.

Вопрос 5

Цифровой узкополосный режекторный БИХ-фильтр с частотой режекции, равной четверти частоты дискретизации, имеет следующие коэффициенты: $b_0 = 1$, $b_1 = 0$, $b_2 = 1$; $a_1 = 0$, $a_2 = 0,81$. Каким должен быть коэффициент масштабирования в частотной области на входе фильтра? Предполагается, что фильтр реализован с использованием звеньев второго порядка.

1. 1,81.
2. 0,55.
3. 5,26.
4. 0,19.
5. Ничего из вышеперечисленного.

Вопрос 6

Какие из приведенных ниже выражений справедливы? В декодировании ДТМЧ для каждого фильтра Горцеля требуется только один действительный коэффициент, потому что

- 1) для декодирования сигналов ДТМЧ требуются только амплитуды тональных частот;
- 2) в декодировании ДТМЧ не требуется информация о фазе;
- 3) фильтр Горцеля обрабатывает каждую выборку данных при ее поступлении и не ждет набора из N выборок, как к БПФ;
- 4) фильтр Горцеля, адаптированный для декодирования ДТМЧ, работает быстро и требует мало памяти;
- 5) алгоритм Горцеля состоит из ряда БИХ-фильтров второго порядка.

Вопрос 7

Схема детектирования тонов ДТМЧ для цифрового телефона использует ряд фильтров Горцеля второго порядка для извлечения тонов ДТМЧ и их вторых гармоник. Предположим, что тоны ДТМЧ для цифры “0” — 941 Гц и 1336 Гц. Определите значения коэффициентов в тракте обратной связи двух фильтров для низкочастотного тона (941 Гц), если для основной и второй гармоник использованы длины последовательности $N = 205$ и $N = 210$, а соответствующие дискретные частотные каналы — 24 и 47.

1. Для низкочастотного тона (т.е. 941 Гц) значения коэффициентов обратной связи — $a_1 = 0,99971$, $a_2 = -1$.
2. Для низкочастотного тона (т.е. 941 Гц) значения коэффициентов обратной связи — $a_1 = 1,482867$, $a_2 = -1$.
3. Для низкочастотного тона (т.е. 941 Гц) значения коэффициентов обратной связи — $a_1 = 1,345621$, $a_2 = -1$.
4. Для второй гармоники низкочастотного тона (т.е. 2×941 Гц) значения коэффициентов обратной связи — $a_1 = 0,463812$, $a_2 = -1$.
5. Для второй гармоники низкочастотного тона (т.е. 2×941 Гц) значения коэффициентов обратной связи — $a_1 = 0,488851$, $a_2 = -1$.
6. Для второй гармоники низкочастотного тона (т.е. 2×941 Гц) значения коэффициентов обратной связи — $a_1 = 0,327635$, $a_2 = -1$.

Вопрос 8

Какие из приведенных ниже утверждений справедливы?

1. Чтобы различить речь и тоны ДТМЧ, требуется знать амплитуды вторых гармоник, а также амплитуды основных частот.
2. Чтобы различить речь и тоны ДТМЧ, не требуется знать амплитуды вторых гармоник.
3. Частотная характеристика телефонной системы такова, что высокочастотные тоны ДТМЧ поглощаются сильнее низкочастотных.
4. Частотная характеристика телефонной системы такова, что низкочастотные тоны ДТМЧ поглощаются сильнее высокочастотных.
5. Речь имеет значительные гармоники четного порядка, а сигналы ДТМЧ — нет.
6. Сигналы ДТМЧ имеют значительные гармоники четного порядка, а речь — нет.

Вопрос 9

Двухкаскадный дециматор используется для снижения частоты дискретизации с 240 000 до 8 000 выборок/с. Коэффициенты децимации каскадов — $M_1 = 15$ и $M_2 = 2$, а длины фильтров — $N_1 = 45$ и $N_2 = 43$. Какие из приведенных ниже выражений справедливы?

1. Частоты дискретизации на выходах двух каскадов равны 16 000 и 8 000 выборок/с.
2. Частоты дискретизации на выходах двух каскадов равны 8 000 и 16 000 выборок/с.

3. Меры сложности дециматора: УВС = 1 064 000, ОТП = 88.
4. Меры сложности дециматора: УВС = 1 064, ОТП = 88.
5. Ничего из вышеперечисленного.

Вопрос 10

Двухкаскадный дециматор используется для снижения частоты дискретизации с 500 до 12,5 Гц. Коэффициенты децимации каскадов — 10 и 4, а длины фильтров — $N_1 = 55$ и $N_2 = 97$. Чему равны индексы сложности дециматора?

1. УВС = 32 350, 5, ОТП = 152.
2. УВС = 3962, 5, ОТП = 152.
3. УВС = 500, ОТП = 55.
4. УВС = 28 712, 5, ОТП = 152.
5. Ничего из вышеперечисленного.

Вопрос 11

Двухкаскадный дециматор используется для снижения частоты дискретизации с 500 до 12,5 Гц. Коэффициенты децимации каскадов — 10 и 4. Интерес представляет полоса частот 0–4 Гц. Чему равны граничные частоты фильтра защиты от наложения спектров первого каскада?

1. 0; 4; 6,25 Гц.
2. 0; 4; 43,75 Гц.
3. 0; 4; 12,5 Гц.
4. 0; 4; 50; 6,25 Гц.
5. Ничего из вышеперечисленного.

14.5. Резюме

В данной главе рассмотрено несколько примеров и плат, которые можно использовать для реализации алгоритмов ЦОС, изложенных в книге. Разобраны реальные приложения ЦОС, что позволяет читателю получить представление о практической ценности описанных концепций. Кроме того, приводится пример тестов, которые авторы используют для быстрой оценки знаний по курсу ЦОС.

Задачи

- 14.1. Покажите, указав сделанные предположения, что автокорреляционная функция сигнала, загрязненного случайным шумом, равна АКФ чистого сигнала. Объясните, как этот результат можно использовать для детектирования скрытой периодичности.

- 14.2. Докажите в рамках разумных предположений, что максимальное отношение сигнал-шум на выходе цифрового согласованного фильтра не зависит от формы входного сигнала.
- 14.3. Периодический сигнал, скрытый в шуме, требуется детектировать с помощью цифровой согласованной фильтрации. Даны последовательные выборки чистого и зашумленных сигналов.
- | | |
|--------------------|---|
| чистый сигнал | $\{-0, 51; -0, 35; -0, 29; -0, 25; -0, 29; -0, 39; -0, 47\}$ |
| зашумленный сигнал | $\{-0, 18; -0, 06; 0, 27; 0, 69; -0, 50; -0, 44; -0, 20; -1, 46; 0, 93; -1, 46; -0, 91; -0, 39; -1, 70\}$ |

Определите

- коэффициенты цифрового согласованного фильтра;
- выход цифрового согласованного фильтра;
- улучшение отношения сигнал-шум, выраженное в децибелах, получаемое за счет согласованного фильтра.

Примечание: дисперсия σ_0^2 шума на выходе фильтре записывается следующим образом:

$$\sigma_0^2 = \sigma^2 \sum_{m=0}^{\infty} h^2(m),$$

где σ^2 — дисперсия шума на входе фильтра, $\{h(m)\}$ — коэффициенты фильтра.

Литература

- Azevedo S. and Longini R. L. (1980) Abdominal-lead fetal electrocardiographic R-wave enhancement for heart rate determination. *IEEE Trans. Biomedical Engineering*, **27**(5), 255–260.
- Bajpai, A. C., Calus I. M. and Fairley J. A. (1973) *Mathematics for Engineers and Scientists*, Volume 2. New York: Wiley.
- Barlow J. S. and Rémond A. (1981) Eye movement artifact nulling in EEGs by multichannel on-line EOG subtraction. *Electroencephalography and Clinical Neurophysiology*, **52**, 418–423.
- Bierman G. J. (1976) Measurement updating using the U-D factorization. *Automatica*, **12**, 375–382.
- Bierman G. J. (1977) *Factorization Methods for Discrete Sequential Estimation*. New York: Academic Press.
- Clark R. J., Ifeachor E. C., Van Eetvelt P. W. J. and Rogers G. M. (2000) Techniques for generating digital equalizer coefficients. *J. Audio Eng. Soc.*, **48**(4), April, 281–298.
- Clarke D. W. (1981) Implementation of self-tuning controllers. In *Self-Tuning and Adaptive Control* (Harris C. J. and Billings S. A. (eds)), pp. 144–165. Stevenage, UK: Peter Peregrinus.
- Cowan C. F. N. and Grant P. M. (1984) Adaptive processing — an overview. In *IEE Colloq. Adaptive Processing and Biomedical Applications*, October 1984, paper 1.
- Dattorro J. (1988) The implementation of recursive digital filters for high-fidelity audio. *J. Audio Eng. Soc.*, **36**(11), 851–878.
- Favret A. G. (1968) Computer matched filter location of fetal R-waves. *Medical and Biological Engineering*, **6**, 467–475.
- Flores I. (1963) *The Logic of Computer Arithmetic*. Englewood Cliffs NJ: Prentice-Hall.
- Fortgens C. and De Bruin M. P. (1983) Removal of eye movement and ECG artifacts from the non-cephalic reference EEG. *Electroencephalography and Clinical Neurophysiology*, **56**, 90–96.

13. Girtan D. G. and Kamiya A. J. (1973) A simple on-line technique for removing eye movement artifacts from the EEG. *Electroencephalography and Clinical Neurophysiology*, **34**, 212–218.
14. Goodman G. C. and Sin K. S. (1984) *Adaptive Filtering, Prediction and Control*. Englewood Cliffs NJ: Prentice-Hall.
15. Gotman J., Gloor P. and Ray W. F. (1975) A quantitative comparison of traditional reading of the EEG and interpretation of computer-extracted features in patients with supratentorial brain lesions. *Electroencephalography and Clinical Neurophysiology*, **38**, 623–639.
16. Greene K. R. (1987) The ECG waveform. In *Balliere's Clinical Obstetrics and Gynaecology* (Whittle M. (ed.)), Volume 1, pp. 131–155.
17. Hamer C. F., Ifeachor E. C. and Jervis B. W. (1985) Digital filtering of physiological signals with minimal distortion. *Medical and Biological Engineering and Computation*, **23**, 274–278.
18. Harris C. J. (1983) Brainwaves appear on T. V. in real-time. *Electronics*, February, 47–48.
19. IEEE (1985) IEEE Standard for Binary Floating Point Arithmetic. *SIGPLAN Notices*, **22**(2), 9–25.
20. Ifeachor E. C. (2001) *A Practical Guide for MATLAB and C Language Implementations of DSP Algorithms*. Harlow: Pearson Education.
21. Ifeachor E. C., Jervis B. W., Morris E. L., Allen E. M. and Hudson N. R. (1986) A new microcomputer-based online ocular artefact removal (OAR) system. *Proc. IEE*, **133**(5), 291–300.
22. Ifeachor E. C., Keith R. D. F., Westgate J. and Greene K. R. (1991) An expert system to assist in the management of labour. In *Proc. World Congress on Expert Systems* (Liebowitz J. (ed.)), Volume 4, pp. 2615–2622. New York: Pergamon.
23. Jervis B. W., Allen E., Johnson T. E., Nichols M. J. and Hudson N. R. (1984) The application of pattern recognition techniques to the contingent negative variation for the differentiation of subject categories. *IEEE Trans. Biomedical Engineering*, **31**, 342–349.
24. Jervis B. W., Nichols M. J., Allen E., Hudson N. R. and Johnson T. E. (1985) The assessment of two methods for removing eye movement artefact from the EEG. *Electroencephalography and Clinical Neurophysiology*, **61**, 444–452.
25. Lindecrantz K. G., Lilja H. and Rosen K. G. (1988) New software QRS detector algorithm suitable for realtime applications with low signal to noise ratios. *J. Biomedical Engineering*, **10**, 280–283.
26. Motorola (1995) DSP56000 Digital Signal Processor Family Manual. Austin TX: Motorola.
27. Motorola (1996) DSP56302 Evaluation Module. Motorola Inc. www1.motoroladsp.com/docs/docs.html
28. Outram N. J., Ifeachor E. C., Van Eetvelt P. W. J. and Curnow J. S. H. (1995) Techniques for optimal enhancement and feature extraction of the fetal electrocardiogram. *IEE Proc.-Sci. Meas. Technol.*, **142**(6), 482–489.
29. Patterson D. A. and Hennessy J. L. (1990) *Computer Architecture: A Quantitative Approach*. San Mateo CA: Morgan Kaufmann.
30. Peterka V. (1975) A square root filter for real-time multivariate regression. *Kybernetika*, **11**, 53–67.
31. Quilter P. M., Macgillivray B. B. and Wadbrook D. G. (1977) The removal of eye movement artefact from EEG signals using correlation techniques. *IEE Conf. Publ.*, **159**, 93–100.
32. Rabiner L. R. and Gold B. (1975) *Theory and Application of Digital Signal Processing*. Englewood Cliffs NJ: Prentice-Hall.
33. Takeda H. and Hata S. (1985) Development of micro-computerized topographic EEG analyzer and its application to real time display. *Electroencephalography and Clinical Neurophysiology*, **61**, 98.
34. Texas Instruments (1986) *Digital Signal Processing Applications with the TMS320 Family: Theory, Algorithms and Implementations*. Texas Instruments.
35. Texas Instruments (1995) TMS320C54x evaluation module technical reference. Texas Instruments. www.ti.com/sc/docs/psheets/abstract/apps/spru135.html
36. Tomè A. M., Principe J. C. and Da Silva A. M. (1985) Micro analysis of spike and wave bursts in children's EEG. *Electroencephalography and Clinical Neurophysiology*, **61**, 113.
37. Weitek (1984) *High Speed Digital Arithmetic VLS Application Seminar Notes*. Sunnyvale CA: Weitek.
38. Widrow B., Glover J. R., McCool J. M., Kaunitz J., Williams C. S., Hearn R. H., Zeidler J. R., Dong E. and Goodlin R. C. (1975) Adaptive noise cancelling: principles and applications. *Proc. IEEE*, **63**, 1692–1716.
39. Young P. (1974) *Recursive approaches to time series analysis*. Bull. IMA, 10, 209–224.

Дополнительная литература

- Clarke D. W. (1980) Some implementation considerations of self-tuning controllers. In *Numerical Techniques for Stochastic Systems* (Archetti F. and Cugiani M. (eds)), pp. 81–101. Amsterdam: North-Holland.
- Clarke D. W., Cope S. N. and Gawthrop P. J. (1975) *Feasibility Study of the Application of Microprocessors to Self-tuning Controllers*. OUEL Report 1137y75.
- Dattorro J. (1988) The implementation of recursive digital filters for high fidelity audio. *J. Audio Eng. Soc.*, **36**(11), 851–878.
- Kay S. M. (1987) *Modern Spectrum Estimation*. Englewood Cliffs NJ: Prentice-Hall.
- Marple S. L. Jr (1987) *Digital Spectral Analysis with Applications*. Englewood Cliffs, NJ: Prentice-Hall.
- Motorola (1980) *16-bit Microprocessor User's Manual*. Austin TX: Motorola Semiconductor.
- Otnes R. K. and Enochson L. (1978) *Applied Time Series Analysis*, Volume 1. New York: Wiley.
- Rosen K. G. and Lindecrantz K. G. (1989) STAN, the Gothenburg model for fetal surveillance during labour by ST analysis of the fetal electrocardiogram. *Clinical Physiology and Physiological Measurement*, Suppl. B, **10**, 51–56.
- Stanley W. D., Dougherty G. R. and Dougherty R. (1984) *Digital Signal Processing*, 2nd edn. Reston VA: Reston Publications.
- Verleger R., Gasser T. and Möcks J. (1982) Correction of EOG artifacts in event-related potentials of the EEG: aspects of reliability and validity. *Psychophysiology*, **19**, 472–480.

Приложение

14.A. Модифицированный алгоритм UD-факторизации

- Этап 1: $\mathbf{v} = \mathbf{U}^T(m)\mathbf{x}$.
 - Этап 2: $b_i = d_i(m)v_i$, $i = 1, \dots, n$.
 - Этап 3: $\alpha_1 = \gamma + n_1v_1$.
 - Этап 4: $d_1(m+1) = d_1(m)/\alpha_1$.
Для $j = 2, \dots, n$ рекурсивно проводятся вычисления по формулам (14.9)–(14.13).
 - Этап 5: $\alpha_j = \alpha_{j-1} + v_jb_j$.
 - Этап 6: $\rho_j = -v_j/\alpha_{j-1}$.
Для $j = 2, \dots, n$ рекурсивно проводятся вычисления по формуле (14.11).
 - Этап 7: $U_{kj}(m+1) = U_{kj}(m) + b_k\rho_j$.
 - Этап 8: $b_k = b_k + b_jU_{kj}(m)$.
 - Этап 9: $d_j(m+1) = d_j(m)\alpha_{j-1}/\alpha_j\gamma$.
 - Этап 10: $G = b/\alpha_n$ ($g_i = b_i/\alpha_n$, $i = 1, \dots, n$)
Следует отметить два момента.
1. Величина α_n на этапе 10 — это значение α_j (этап 5) после n -й итерации. Это значение также равно α в формуле (14.9) алгоритма наименьших квадратов.
 2. Элементы $\mathbf{D}$ (т.е. d_i на этапах 4 и 9) можно хранить в ячейках памяти, выделенных для диагональных элементов $\mathbf{U}$, поскольку $U_{jj} = 1$. Кроме того, для экономии памяти и облегчения программирования элементы $\mathbf{U}$, включая элементы $\mathbf{D}$, можно записать как вектор, хотя из индексов (k, j) следует, что $\mathbf{U}$ — двумерный массив.

Предметный указатель

Символы

n -е гармоники, 137

G

GSM 900, 49

S

SIMD, 799

V

VLIW, 852

A

Автокорреляционная функция, 33

Адаптивный процесс, 846

Амплитудный спектр, 141; 740

Анализ влияния конечной разрядности,
373, 388

Апертурное искажение, 111

Апертурный эффект, 80

Аппаратная реализация, 573

Арифметическое переполнение, 451

Архитектурные особенности, 818

B

Вейвлет Морле, 174

Взаимно-корреляционная функция, 33

Взвешивание, 319

Воспроизводимость, 28

Выигрыш от обработки, 747

Вычет, 217; 223

Вычисление с замещением, 839

D

Дельта-функция, 150

Децимация, 630

Диаграмма нулей и полюсов, 501

Диаграмма реализации, 248

Дискретное преобразование Фурье, 32,
150

Дискретный сигнал, 65

Дисперсия, 305

Дифференцирование, 230

Длина слова, 819

Доказательство теоремы о корреляции,
306

Долгоиграющая пластинка, 45

Z

Заграждающий фильтр, 542

Закон ассоциативности, 320

Закон дистрибутивности, 320

Закон коммутативности, 319

Звонковый фильтр, 964

I

Идентификация системы, 282; 320

Импульсный отклик, 312

Инвариантное преобразование
импульсной характеристики,
511

Интеграл Фурье, 141

Интеграл свертки, 314

Интегрирующий фильтр, 110

Интерполяция, 112; 630; 636

Искажение по закону $\sin x/x$, 111

К

Канал, 49
Каноническое звено, 562
Кардиотокограмма, 56
Каскадная реализация, 830
Квантование после накопления, 909
Компакт-диск, 45
Компандирование, 94
Корреляция, 32; 282
Коэффициент забывания, 952
Коэффициент перевыборки, 96
Коэффициентами передачи, 949
Коэффициент забывания, 718
Кратномасштабный анализ, 176
Круговая корреляция, 151

Л

Линейность, 229
Линия задержки с отводами, 366
Линия максимумов, 179

М

Масштабные граничные циклы
переполнения, 890
Метод взвешивания, 438
Метод размещения нулей и полюсов, 504
Метод частной выборки, 438
Методы устранения окулярных
артефактов, 948
Многофазная структура, 663
Многофазный фильтр, 664
Модулирующий сигнал, 38

Н

Наложение, 745
Наложением-сложением, 447
Наложение, 67
Несущая частота, 38
Нормировка краевых частот, 412
Норма L_1 , 892
Норма L_2 , 893
Нуль (функции), 212; 232; 574

О

Область сходимости, 210
Обработка с избыточностью, 747
Обращение свертки, 282; 322
Ограничение скорости, 357
Окулярный артефакт, 720
Оптимизационные методы, 439
Отбеливание, 324
Отношение T/QRS, 57
Ошибка квантования коэффициентов,
450; 872
Ошибка округления, 369; 899; 924

П

Параллельная реализация, 834
Положительная определенность, 954
Полосовой фильтр, 540
Полочный фильтр, 965
Полюса, 574
Потери при обработке в наихудшем
случае, 748
Потоковая обработка, 783
Проблема раздувания, 955
Прореживание, 630
Просачивание спектральных
составляющих, 745

Р

Раздувание, 952
Размывание, 745
Разрешение по частоте, 740
Реостат, 965

С

Свертка, 31; 229; 311
Сверточная сумма, 314
Свойство цикличности, 147
Сигнал, 27
Система дискретного времени, 208
Система инициализации, 827
Скорость выполнения, 818
Слепое обращение свертки, 282
Смещение базовой линии, 970

Соты, 49
Спектр, 740
Статическая суперскалярная
 архитектура, 852
Субгармонический отклик, 923

Т

Теорема вычетов, 223
Теорема Котельникова, 67
Теорема Найквиста, 67
Теорема о корреляции, 151
Теорема о свертке, 325
Теорема отсчетов, 67
Теорема Парсеваля, 150
Тип арифметики, 819
Точность, 28
Трансверсальный фильтр, 35; 366

У

Упорядочение по Уолшу, 168
Условие фазового искажения, 70

Ф

Фазовый спектр, 740
Фильтр верхних частот, 521; 538
Фильтр нижних частот, 520; 535
Фильтр сглаживания, 110
Фильтрация, 32
Формирование спектра ошибки, 910
Функции Уолша, 168
Функция отсчетов (sinc), 140
Фурье-образ, 141

Ц

Целополосной фильтр, 677
Циклическая частота, 137
Цифровая согласованная фильтрация,
 945
Цифровая фильтрация, 845
Цифровой сигнал $x(n)$ ($n = 0, 1, \dots$), 65
Цифровой фильтр, 377

Ч

Частота Найквиста, 68; 70; 143

Ш

Ширина полосы, 424
Ширина полосы минимального
 разрешения, 748
Ширина полосы шумового
 (прямоугольного) эквивалента,
 746
Шум АЦП, 450

Э

Электроокулограмма, 720
Эффект Гиббса, 391
Эффект Найквиста, 555
Эффекты конечной разрядности, 450

Я

Явление скачка, 923
Ядро Дирихле, 745
Ячейка, 49

Научно-популярное издание

Эммануил С. Айфичер, Барри У. Джервис

**Цифровая обработка сигналов:
практический подход
2-е издание**

Литературный редактор	<i>О.Ю. Белозовская</i>
Верстка	<i>А.В. Назаренко</i>
Художественный редактор	<i>С.А. Чернокозинский</i>

Издательский дом “Вильямс”
127055, г. Москва, ул. Лесная, д. 43, стр. 1

Подписано в печать 19.12.2007. Формат 70х100/16.
Гарнитура Times. Печать офсетная.
Усл. печ. л. 80,0. Уч.-изд. л. 74,78
Тираж 1000 экз. Заказ № 0000.

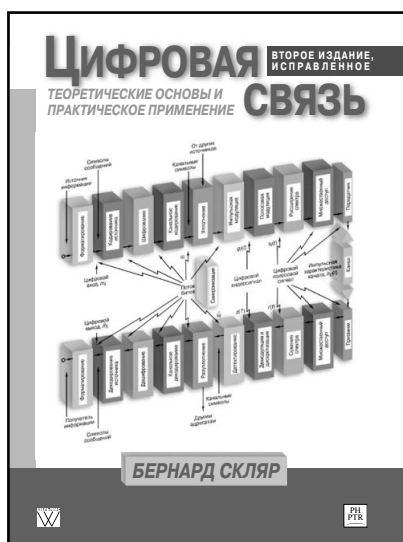
Отпечатано по технологии СtP
в ОАО “Печатный двор” им. А.М. Горького
197110, Санкт-Петербург, Чкаловский пр., 15.

ЦИФРОВАЯ СВЯЗЬ

ТЕОРЕТИЧЕСКИЕ ОСНОВЫ И ПРИМЕНЕНИЕ

2-е издание

Бернард Скляр



www.williamspublishing.com

Предлагаемую книгу стоит прочесть всем, кто интересуется цифровой связью. Это учебник, в котором математически строго описаны все преобразования, которым подвергается информация на пути от источника к адресату. Это также и справочник, в котором дано описание схем, необходимых для практической реализации соответствующих математических абстракций. И, наконец, это просто хорошая и интересная книга для всех тех, кто хочет узнать *все* о цифровой связи, прочитав всего одну серьезную и, одновременно, доступную работу.

ISBN 978-5-8459-0497-3 **в продаже**

ДИФФЕРЕНЦИАЛЬНЫЕ УРАВНЕНИЯ И КРАЕВЫЕ ЗАДАЧИ

МОДЕЛИРОВАНИЕ И ВЫЧИСЛЕНИЕ С ПОМОЩЬЮ MATHEMATICA, MAPLE И MATLAB

3-Е ИЗДАНИЕ

Чарльз Генри Эдвардс
Дэвид Э. Пенни



www.williamspublishing.com

Данный учебник представляет собой весьма полный современный вводный курс обыкновенных дифференциальных уравнений. Довольно подробно освещены все темы, затрагиваемые в классических вводных курсах, включая и такие как применение матричных методов, операционного исчисления, а также степенных рядов и рядов Фурье. Не обойдены вниманием и современные исследования в области дифференциальных уравнений, такие как, например, хаос в динамических системах и нелинейные явления и системы. Особое внимание авторы уделяют численным методам и обучению построения математических моделей самых разнообразных (например, экологических, физических, инженерных) систем. Для изучения таких моделей авторы используют самые современные математические пакеты: MATLAB, Maple и Mathematica. Кроме того, для каждого раздела имеются задачи различной сложности, а также проекты для самостоятельной разработки студентами.

ISBN 978-5-8459-1166-7

в продаже