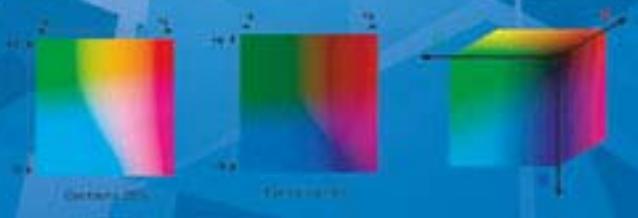


В.П. Дворкович
А.В. Дворкович

Цифровые видеоинформационные
системы (теория и практика)





М И Р цифровой обработки

В.П. Дворкович,
А.В. Дворкович

Цифровые
видеоинформационные
системы
(теория и практика)

ТЕХНОСФЕРА
Москва
2012



*Издание осуществлено при финансовой
поддержке Российского фонда фундаментальных
исследований по проекту № 12-07-07104*

УДК 004.421, 004.932, 519.722, 621.317.799, 621.397

ББК 32.81

Д24

*Рецензенты: д.т.н., профессор Калмыков В.В.;
д.т.н., профессор Сагдулаев Ю.С.*

Д24 Дворкович В.П., Дворкович А.В.

Цифровые видеонформационные системы (теория и практика)

Москва: Техносфера, 2012. – 1008с. ISBN 978-5-94836-336-3

Книга посвящена теоретическим основам и практической реализации цифровых видеонформационных систем. Материал содержит девять частей, состоящих из 23 глав.

В Части I изложены проблемы цифрового преобразования статических и динамических изображений, оцениваются объемы информации при преобразовании изображений в цифровую форму. Часть II посвящена исследованиям проблем статистической и визуальной избыточности изображений, излагаются методы эффективного энтропийного кодирования, оцениваются возможности использования визуальной избыточности статических и динамических изображений. В Части III рассматриваются непосредственное кодирование, кодирование с предсказанием и групповое кодирование изображений, методы блочного и фрактального кодирования, кратномасштабная обработка изображений. В Части IV излагаются методы повышения качества внутрикадрового кодирования статических изображений, принципы анализа движения в динамических изображениях, идеология обработки изображений при кратномасштабном анализе. Часть V посвящена анализу методов и стандартов кодирования статических и динамических изображений, рассмотрены некоторые проблемы реализации кодирующих устройств и визуализации их параметров. В Части VI анализируются основные характеристики и цифровое представление речевой и звуковой информации, приводится описание основных стандартов кодирования речи и звука. Часть VII посвящена анализу основных методов канального кодирования и модуляции цифровой информации. В Части VIII приводится описание основных стандартов цифрового ТВ вещания. Часть IX посвящена изложению стандартов мультимедийного телерадиовещания, а также проблемам реализации систем видеоконференцсвязи на базе использования рекомендаций ИТУ-Т.

Книга адресована студентам и специалистам телекоммуникационной отрасли, работающим в области обработки и передачи видео- и звуковой информации.

УДК 004.421, 004.932, 519.722, 621.317.799, 621.397

ББК 32.81

© 2012, Дворкович В.П., Дворкович А.В.

© 2012, ЗАО «РИЦ «Техносфера»,
оригинал-макет, оформление.

ISBN 978-5-94836-336-3

Содержание

Вступительное слово академика РАН А.С. Бугаева	13
Предисловие	16
Часть I. Видеоинформационные приложения и объем цифровой информации	
Введение	30
Глава 1	
Форматы изображений. Статические и динамические эталонные изображения	33
1.1. Аналоговые и цифровые растровые изображения	33
1.2. Цветовые системы	39
1.3. Статические и динамические эталонные изображения	47
Глава 2	
Системы формирования и передачи видеоинформации	52
2.1. Монохромное телевидение	52
2.2. Цветное телевидение	55
2.2.1. Система NTSC	57
2.2.2. Система PAL	59
2.2.3. Система SECAM	62
2.2.4. Цифровые компонентные телевизионные сигналы	67
2.3. Телевидение повышенной, высокой и сверхвысокой четкости, «цифровое кино»	70
2.4. Видеотелефония и видеоконференцсвязь, домашнее видео	71
2.5. Графические изображения, факсимильные системы и интерактивные системы передачи изображений	72
Глава 3	
Цифровое представление видеоинформации	75
3.1. Представление изображения конечным объемом данных	75
3.2. Фурье-преобразование, теорема о дискретизации	79
3.3. Изображение, изменяющееся во времени	82
3.4. Квантование. Импульсно-кодовая модуляция	84
Литература к части I	89



Часть II. Статистическая и визуальная избыточность изображений

Введение	93
----------------	----

Глава 4

Статистическая избыточность дискретизированных данных	95
4.1. Информационная теорема	95
4.2. Виды статистического кодирования	96
4.2.1. Методы представления целых чисел	97
4.2.2. Алгоритм Шеннона–Фано	100
4.2.3. Алгоритм Хаффмана	101
4.2.4. Блочное и условное кодирование	105
4.2.5. Арифметическое кодирование	107
4.2.6. Словарные методы кодирования дискретной информации	109
4.2.7. Статистические методы моделирования дискретной информации	115
4.2.8. Контекстные методы энтропийного кодирования	127
4.3. Кодирование с преобразованием	136
4.3.1. Кодирование с предсказанием	136
4.3.2. Кодирование с дискретным преобразованием	138
4.4. Статистика монохромных и цветных изображений	142
4.4.1. Статистика монохромных изображений без преобразования	142
4.4.2. Статистика монохромных изображений с преобразованием	145
4.4.3. Статистика цветных изображений	146
4.5. Кодирование с сокращением объема передаваемой информации	147

Глава 5

Визуальная избыточность изображений	150
5.1. Заметность пространственных изменений яркости изображений	150
5.2. Заметность временных изменений яркости изображений	157
5.3. Влияние помех	158
5.4. Специфические особенности восприятия цветов	159
5.5. Методы субъективной оценки качества изображений	162
5.6. Компьютерные методы хранения изображений	166

Литература к части II	178
------------------------------------	------------

Часть III. Основные методы сжатия изображений

Введение	185
----------------	-----

Глава 6

Непосредственное кодирование изображений и кодирование с предсказанием	188
6.1. Импульсно-кодовая модуляция	188
6.2. Кодирование изображений с предсказанием, дифференциальная импульсно-кодовая модуляция	191

6.3.	Основные виды предсказания.....	197
6.3.1.	Внутрикадровое предсказание	197
6.3.2.	Межкадровое предсказание.....	199
6.4.	Квантование сигналов, ошибки предсказания	201
6.5.	Помехоустойчивость кодирования с предсказанием	206

Глава 7

Групповое кодирование изображений

7.1.	Дискретные линейные ортогональные преобразования.....	208
7.1.1.	Дискретное преобразование Фурье.....	210
7.1.2.	Преобразование Хаара	213
7.1.3.	Преобразование Уолша–Адамара	214
7.1.4.	Дискретное синусное преобразование	217
7.1.5.	Дискретные косинусные преобразования	217
7.1.6.	Преобразование Кархунена–Лоэва	221
7.1.7.	Другие виды преобразований	226
7.2.	Квантование коэффициентов преобразования.....	228
7.3.	Кодирование коэффициентов преобразования	231
7.4.	Межкадровое или трехмерное кодирование	233

Глава 8

Другие методы кодирования изображений

8.1.	Кодирование с использованием гибридных преобразований	235
8.2.	Другие методы блочного и интерполяционного кодирования	236
8.3.	Фрактальные методы кодирования изображений.....	244

Глава 9

Вейвлеты и кратномасштабная обработка изображений

9.1.	Непрерывное вейвлет-преобразование.....	253
9.2.	Дискретное вейвлет-преобразование.....	256
9.3.	Кратномасштабный вейвлет-анализ	258
9.4.	Основные сведения о субполосном кодировании.....	262
9.4.1.	КИХ-фильтры с нечетным числом отсчетов цифровой решетки и линейной (нулевой) ФЧХ.....	265
9.4.2.	КИХ-фильтры с четным числом отсчетов цифровой решетки....	269
9.4.3.	Квадратурно-зеркальные КИХ-фильтры	273
9.4.4.	Кратномасштабное дискретное вейвлет-преобразование изображений.....	274
9.4.5.	Использование банков трехполосных КИХ-фильтров при обработке изображений	275

Литература к части III

281



Часть IV. Внутрикадровая и межкадровая обработка изображений	
Введение	287
Глава 10	
Внутрикадровое кодирование изображений	290
10.1. Изменения коэффициентов квантования блоков в JPEG-подобных алгоритмах видеокомпрессии	290
10.2. Направленное пространственное внутрикадровое предсказание	295
10.2.1. Режим предсказания пикселов яркости блоков 4×4	297
10.2.2. Режим предсказания пикселов яркости блоков 8×8	301
10.2.3. Режим предсказания пикселов яркости блоков 16×16	304
10.2.4. Режим предсказания пикселов блоков цветности	305
10.3. Уменьшение влияния блокинг-эффекта	308
Глава 11	
Обработка изображений при кратномасштабном анализе	314
11.1. Оценка энергетических характеристик изображения в зависимости от вида вейвлет-базиса разложения	315
11.2. Выбор маски квантования высокочастотных компонент вейвлет-преобразования изображений	317
11.3. Обработка тестовых изображений двухполосными вейвлет-фильтрами	322
11.4. Обработка тестовых изображений трехполосными вейвлет-фильтрами	323
Глава 12	
Методы анализа и компенсации движения в динамических изображениях	329
12.1. Основные методы анализа движения	331
12.1.1. Метод полного перебора	332
12.1.2. Логарифмический, комбинированный по двум направлениям, трехшаговый, иерархический методы	332
12.1.3. Методы, основанные на оптическом уравнении	335
12.1.4. Использование временной и пространственной корреляции для поиска векторов движения	336
12.1.5. Использование временной и пространственной корреляции векторов с оптимизацией возмущения битового потока	337
12.1.6. Метод полного перебора с преобразованием Фурье	338
12.1.7. Методы фазовой корреляции	339
12.1.8. Оптимизация поиска векторов движения по стандарту MPEG-4 (метод быстрого поиска MVFAST)	340
12.2. Повышение эффективности анализа движения по опорным точкам	344
12.3. Дополнительные возможности компенсации движения деталей в динамических изображениях	355
12.4. Анализ смены сюжета в динамических изображениях	361
Литература к части IV	366

Часть V. Основные стандарты цифрового кодирования видеинформации

Введение	369
-----------------------	-----

Глава 13

Стандарты кодирования статических изображений	371
--	-----

13.1. Стандарт JPEG и его модификация	372
13.2. Стандарт JPEG-2000.....	377

Глава 14

Стандарты кодирования динамических изображений	387
---	-----

14.1. H.261 — первый стандарт видеокодирования МСЭ	390
14.2. H.263. Видеокодирование для систем связи с низкой пропускной способностью	395
14.3. MPEG-1 — кодирование видео- и аудиоинформации для цифровых систем с объемом данных приблизительно до 1,5 Мбитов/с	402
14.4. MPEG-2. Кодирование динамических изображений и связанной с ними звуковой информации	406
14.4.1. Общие положения	407
14.4.2. Совместимость стандартов MPEG-1 и MPEG-2	410
14.4.3. Масштабируемость в MPEG-2	411
14.4.4. Профили и уровни	414
14.4.5. Транспортный и программный потоки MPEG-2	415
14.4.6. Видеопоток стандарта MPEG-2	421
14.4.7. Структура битового потока видеопоследовательности стандарта MPEG-2	429
14.5. MPEG-4. Обобщенное кодирование аудиовизуальных объектов	454
14.5.1. Кодированное представление медийных объектов	455
14.5.2. Аудиообъекты	457
14.5.3. Видеообъекты	458
14.6. H.264/MPEG-4 часть 10 AVC — улучшенное видеокодирование	467
14.6.1. Профили и уровни стандарта H.264/AVC	468
14.6.2. Эффективность сжатия в стандарте H.264/AVC	471
14.6.3. Уровень сетевой абстракции	476
14.6.4. Уровень видеокодирования	478
14.6.5. Преобразование, масштабирование и квантование	490
14.6.6. Деблокирующая фильтрация	497
14.6.7. Энтропийное кодирование	498
14.7. Некоторые проблемы реализации видеокодирующих систем	503
14.7.1. Методика построения промежуточных кадров видеопоследовательности	503
14.7.2. Методика регулировки потока при сжатии динамических изображений	506
14.7.3. Анализ смены сюжета	516
14.7.4. Оценка быстродействия кодирования изображений	519



14.7.5. Возможности кодирования стереоскопических изображений в рамках совместимости со стандартами MPEG	521
14.8. Визуализация параметров видеокодирования	528
Литература к части V	537

Часть VI. Основные методы кодирования речевой и звуковой информации

Введение	543
-----------------------	------------

Глава 15

Основные характеристики и цифровое представление звуковой информации

15.1. Статистика мгновенных значений и уровней звуковых сигналов.....	548
15.2. Спектральные характеристики звуковых сигналов	553
15.3. Огибающая и мгновенная частота звуковых сигналов.....	558
15.4. Пространственное восприятие звуковых сигналов	561
15.5. Основные принципы цифровых преобразований звуковых сигналов	564
15.6. Аналого-цифровое и цифро-аналоговое преобразование звуковых сигналов.....	568
15.6.1. Импульсно-кодовая модуляция (ИКМ)	568
15.6.2. Дифференциальная импульсно-кодовая модуляция (ДИКМ)	569
15.6.3. Сигма-дельта-модуляция (СДМ).....	571
15.7. Принципы кодирования речевой и звуковой информации	574
15.7.1. Принципы кодирования речевой информации.....	574
15.7.2. Принципы кодирования звуковой информации	577
15.8. Предыскажения звуковых сигналов при цифровом кодировании	591

Глава 16

Основные стандарты кодирования речевой информации

16.1. G.711 — стандарт кодирования речи с применением ИКМ.....	593
16.2. G.722 — кодирование речи со спектром до 7 кГц в цифровой поток 64 кбитов/с	593
16.3. G.723.1 — речевой кодек с двумя скоростями потока — 5,3 и 6,3 кбитов/с	596
16.4. G.726 — речевой кодек АДИКМ со скоростями 40, 32, 24, 16 кбитов/с.....	602
16.5. G.727 — речевой кодек с алгоритмом вложенной АДИКМ.....	603
16.6. G.728 — речевой кодек с выходным потоком 16 кбитов/с и малой задержкой	605
16.7. G.729 — речевой кодек со скоростью 8 кбитов/с	607
16.8. GSM — речевой кодек для систем сотовой связи	611
16.9. GIPS iLBC — кодек, предназначенный для передачи речи по IP-сетям.....	616
16.10. Speex — свободный кодек, предназначенный для передачи речи по IP-сетям.....	621



Глава 17	
Стандарты кодирования звуковой информации	624
17.1. MPEG-1 Audio — кодирование аудиоинформации для цифровых систем с потоком до 1,5 Мбит/с.....	625
17.1.1. MPEG-1 Audio Layer-1 — профессиональная система кодирования аудиоинформации с высоким качеством.....	626
17.1.2. MPEG-1 Audio Layer-2 — система кодирования аудиоинформации для высококачественного телерадиовещания	632
17.1.3. MPEG-1 Audio Layer-3 — система кодирования аудиоинформации для Интернет-вещания	634
17.2. MPEG-2 Audio — стандарт кодирования аудиоинформации для многоканального стереофонического вещания.....	640
17.3. MPEG-2 AAC — стандарт усовершенствованного аудиокодирования.	643
17.4. MPEG-4 Audio — стандарт аудиокодера для мультимедийных приложений.....	645
17.5. Dolby AC3 — стандарт аудиокодера для телевидения и других применений	647
17.6. Совершенствование алгоритмов сжатия звука на основе стандарта AAC	651
17.6.1. Общие алгоритмы кодека AAC.....	654
17.6.2. Метод спектрального дублирования SBR.....	658
17.6.3. Параметрическое стереокодирование PS	663
17.6.4. Конфигурации режимов работы кодеков AAC и HE-AAC	664
Глава 18	
Подавление акустического эха в системах телеконференцсвязи.....	667
18.1. Основные положения	667
18.2. Алгоритмы подавления акустического эха	670
18.3. Принципы линейной адаптивной фильтрации.....	674
18.4. Детектор двойного разговора	676
18.5. Подавление акустического эха на базе фильтра Калмана	679
18.6. Особые случаи	681
18.7. Метрологические требования к подавлению акустического эха.....	682
Литература к части VI.....	684
Часть VII. Основные методы и системы канального кодирования и модуляции цифровой информации	
Введение	688
Глава 19	
Помехоустойчивое кодирование в системах передачи цифровой информации.....	691
19.1. Принципы кодирования источника сообщений	691
19.1.1. Теоретические предпосылки помехоустойчивого кодирования	692

19.1.2. Классификация помехоустойчивых кодов	696
19.2. Линейные блочные коды	698
19.2.1. Математическое описание линейных блочных кодов	698
19.2.2. Свойства линейных блочных кодов	701
19.2.3. Практические примеры блочных кодов	704
19.3. Циклические коды	708
19.3.1. Понятие циклического кода	708
19.3.2. Порождающая и проверочная матрицы циклического кода	710
19.3.3. Систематическое кодирование циклическими кодами	712
19.3.4. Обнаружение ошибок	714
19.3.5. Исправление однократных ошибок	714
19.3.6. Общий принцип декодирования циклических кодов	715
19.3.7. Арифметика полей Галуа	716
19.3.8. Циклические коды Хэмминга	722
19.3.9. CRC-коды	723
19.3.10. Двоичные БЧХ-коды	724
19.3.11. Коды Рида–Соломона	726
19.4. Сверточные коды	728
19.4.1. Систематические сверточные коды	730
19.4.2. Управление скоростью кодирования	732
19.4.3. Представление сверточных кодеров в виде цифровых фильтров	733
19.4.4. Диаграммы состояний и кодовые решетки	734
19.4.5. Декодирование сверточных кодов	736
19.5. Турбокодирование	741
19.5.1. Основные положения	741
19.5.2. Обобщенная схема турбокодера ТСС с параллельным каскадированием	742
19.5.3. Сверточные турбокоды	742
19.5.4. Декодирование турбокодов	744
19.5.5. Характеристики помехоустойчивости сверточных турбокодов ТСС	749
19.5.6. Блочные турбокоды	751
19.6. Низкоплотностные коды	753
19.6.1. Основные положения	753
19.6.2. Классификация LDPC-кодов	754
19.6.3. Методы построения проверочных матриц	755
19.6.4. Алгоритмы декодирования низкоплотностных кодов	761
19.6.5. Оценка сложности алгоритмов декодирования	764
Глава 20 Системы модуляции и сигнального кодирования цифровой информации	767
20.1. Созвездия дискретной модуляции	768
20.1.1. Одномерные и двумерные созвездия	768
20.1.2. Сигнальные созвездия квадратурно-амплитудной модуляции	769
20.1.3. Фильтрация сигналов в передающих и приемных устройствах	776



20.2. Система многочастотной модуляции	778
20.2.1. Структурные схемы OFDM-модуляторов и демодуляторов	779
20.2.2. Защитный интервал, устранение интерференции между несущими частотами	782
20.2.3. Иерархические режимы OFDM-модуляции	784
20.3. Алгоритмы синхронизации OFDM-сигналов в приемных устройствах	786
20.3.1. Методы демодуляции OFDM-сигнала	786
20.3.2. Оценка частотного смещения при приеме OFDM-сигнала	789
Литература к части VII	791

Часть VIII. Основные стандарты систем цифрового телевизионного вещания

Введение	794
-----------------------	------------

Глава 21

Стандарты цифрового телевизионного вещания	797
---	------------

21.1. Стандарт цифрового телевидения ATSC	799
21.2. Система цифрового наземного телевизионного вещания DVB-T	808
21.3. Усовершенствованная система цифрового наземного ТВ-вещания DVB-T2	826
21.4. Система цифрового спутникового телевизионного вещания DVB-S	867
21.5. Система высокоскоростного цифрового спутникового ТВ-вещания DVB-S2	870
21.6. Система цифрового кабельного телевизионного вещания DVB-C	881
21.7. Система высокоскоростного цифрового кабельного ТВ-вещания DVB-C2	883
21.8. Система наземного цифрового телевизионного вещания ISDB-T	893
21.9. Система цифрового спутникового многопрограммного ТВ-вещания ISDB-S	897
21.10. Система цифрового кабельного многопрограммного ТВ-вещания ISDB-C	902
21.11. Система цифрового наземного телевизионного вещания DTMB	905
21.12. Эффективность использования систем цифрового ТВ-вещания в России	909
Литература к части VIII	918

Часть IX. Стандарты систем мобильного радиовещания и видеоконференцсвязи

Введение	921
-----------------------	------------

Глава 22

Стандарты цифрового мобильного радиовещания	930
--	------------

22.1. Система мобильного телевизионного вещания DVB-H	930
22.2. Система мобильного телевизионного вещания MediaFLO	937
22.3. Система мобильного мультимедийного вещания T-DMB	940



22.4. Аудиовизуальная информационная система реального времени РАВИС ...	943
22.5. Цифровое радиовещание DRM+ в ОВЧ-диапазоне частот.....	963
Глава 23	
Системы видеоконференцсвязи	969
23.1. Основные стандарты систем видеоконференцсвязи.....	969
23.2. Возможности технологии видеоконференцсвязи.....	985
Литература к части IX.....	999

Вступительное слово академика РАН А.С. Бугаева

Внедрение информационных технологий в начале XXI века в основном определяется научными и практическими результатами развития техники цифровой обработки сигналов, обеспечивающей решение задач обработки, сокращения избыточности сигнала и передачи и приема информации в реальном масштабе времени.

Целью данной книги является подробное изложение современных достижений теории и практики видеоинформационных систем, реализующих цифровое кодирование и передачу динамических изображений, речи, звука, иных данных по каналам с различной пропускной способностью (видеотелефон, стационарная и мобильная видеоконференцсвязь, многофункциональные интерактивные телевизионные системы, радиолокационные системы и пр.).

Авторы книги — доктор технических наук, профессор Дворкович Виктор Павлович и его сын, доктор технических наук, профессор Дворкович Александр Викторович — хорошо известны специалистам большим вкладом в разработку систем и средств цифровой и компьютерной обработки изображений, а также метрологического обеспечения аналогового и цифрового телевизионного вещания.

С 2000 г. деятельность авторов связана с созданием принципиально новой системы мультимедийного вещания, получившей название РАВИС (RAVIS — Realtime AudioVisual Information System). Результаты разработки системы РАВИС отражены во многих научных статьях, в 18 вкладах России в Международные организации ITU и СЕРТ. В ходе разработок высокоеффективных систем кодирования изображений было сделано более 10 изобретений, защищенных патентами. Наряду с научной работой Дворковичи ведут активную научно-преподавательскую деятельность (Виктор Павлович — профессор МГТУ им. Н.Э. Баумана, Александр Викторович — профессор МЭИ), они авторы и соавторы ряда учебных пособий.

Авторы книги определяют в качестве важнейшей проблемы создания в России современных эффективных систем цифровой обработки, кодирования, передачи, приема изображений различного разрешения, звука и других видов информации полное овладение теорией и математическими методами обработки этих видов информации при использовании имеющихся отечественных и зарубежных материалов в качестве данных, задающих общее направление работ.

Внедрение видеоинформационных систем и их практическое использование послужит началом реализации планов обучения высококлассных специалистов данного профиля. В области организации науки и подготовки кадров важнейшим является использование научно-исследовательских центров по видеоинформационным проблемам, обладающих необходимым научным и техническим потенциалом для решения как фундаментальных проблем обработки сигналов, так и широкого круга прикладных задач.

Предлагаемый авторами материал книги дает возможность понять, с какими проблемами сталкиваются специалисты — разработчики цифровых систем обработки и передачи аудиовизуальной информации.

Следует отметить главное, что подтверждает актуальность издания этого материала, — важность мировой практики перехода к цифровым технологиям

в телекоммуникационных средствах передачи сигналов по каналам связи, наземного и спутникового вещания.

Книга содержит девять разделов, изложенных в 23 главах.

В первых двух разделах последовательно и подробно рассмотрены вопросы цифрового преобразования статических и динамических изображений, оцениваются объемы информации при преобразовании в цифровую форму изображений различного разрешения, приводятся результаты исследований статистической и визуальной избыточности изображений, излагаются методы и алгоритмы эффективного энтропийного кодирования различных видов информации, оцениваются возможности использования визуальной избыточности изображений для сокращения объемов передаваемой информации.

Следующие четыре раздела посвящены изложению методов кодирования изображений, включая их кратномасштабную обработку. Особое внимание уделяется оптимизации методов поиска векторов движения, дополнительным возможностям компенсации движения деталей в динамических изображениях, подробно изложены методы кодирования статических изображений JPEG и JPEG-2000, систем кодирования динамических изображений MPEG-1, MPEG-2, MPEG-4, а также наиболее эффективного стандарта кодирования — H.264/AVC, анализируются основные характеристики и основные стандарты цифрового кодирования речи, алгоритмы кодирования высококачественного звука, приведенные в стандартах MPEG-1, MPEG-2, MPEG-4, Dolby AC-3. Рассмотрены также проблемы подавления акустического эха в системах телеконференций.

Следующие три раздела посвящены анализу основных методов модуляции и помехоустойчивого кодирования цифровой информации, приводится достаточно подробное описание систем многочастотной OFDM-модуляции, основных современных стандартов вещания видео- и аудиоинформации. Основное внимание уделено изложению европейских стандартов цифрового телевидения DVB-T/DVB-T2, DVB-S/DVB-S2, DVB-C/DVB-C2, а также отечественной мультимедийной системе РАВИС и проблемам реализации систем видеоконференций.

Следует подчеркнуть, что отличительной особенностью данной книги является то, что она написана не сторонними исследователями приведенного объемного материала, а непосредственными разработчиками математических методов и аппаратных реализаций соответствующих систем.

Естественно, что участие в решении этих сложных задач и накопленный при этом опыт не могли не вызвать интереса к их анализу и систематизации, поскольку очевидно большое практическое значение таких работ. Это особенно важно в настоящее время — время неотвратимости слияния телекоммуникационных и компьютерных технологий.

Стремление авторов помочь читателям разобраться в весьма сложных проблемах цифровой обработки информации, значимости и эффективности стандартов кодирования и передачи аудиовизуальной информации является особенностью данной книги.

В этом фундаментальном труде авторов удивительно сочетается множество факторов, придающих ему исключительную значимость для дальнейших исследований и практического использования.

Книга, безусловно, будет способствовать активному внедрению цифровых телекоммуникационных систем и, с учетом широкого использования при этом ком-

пьютерных технологий, заинтересует не только специалистов в областях кодирования и передачи аудиовизуальной информации, но и в других областях знаний.

Поскольку в книге весьма полно отражены проблемы, обсуждению которых она посвящена, анализируется обширная литература по данным вопросам, предлагаются оригинальные решения, способствующие развитию науки и достижению эффективных результатов развитии новых перспективных технологий, она может быть отнесена к категории монографий.



*Председатель экспертного совета
по электронике, измерительной технике,
радиотехнике и связи, академик РАН
А.С. Бугаев*

Предисловие

Технической основой создаваемой в России информационной среды становятся современные видеонформационные системы. Они обеспечивают цифровую передачу динамических изображений, речи, звука, иных данных по каналам с различной пропускной способностью (видеотелефон, стационарная и мобильная видеоконференцсвязь, интерактивные телевизионные системы и пр.).

Состояние и перспективы развития информационных технологий в начале XXI века характеризуются становлением и широким практическим использованием техники цифровой обработки сигналов — одной из самых динамичных и быстро развивающихся технологий в мире телекоммуникаций и информатизации общества. Цифровая обработка сигналов (ЦОС) — это информатика реального времени, призванная решать задачи приема, обработки, сокращения избыточности сигнала и передачи информации в реальном времени.

Внедрение видеонформационных систем различного назначения решает проблемы:

- создания высококачественных систем интерактивного цифрового телевизионного вещания при удовлетворении постоянно возрастающих запросов на частотные присвоения систем связи без пересмотра частотных планов;
- разработки и внедрения принципиально новых систем мобильного телерадиовещания;
- создания принципиально новых интерактивных систем опроса общественного мнения;
- обеспечения деятельности органов государственной власти;
- создания мобильной видеоконференцсвязи между центральными учреждениями с удаленными центрами и районами, а также удаленных районов между собой;
- обеспечения сбора и распространения информации различного экономического и политического характера, распространяемой органами власти среди населения, популяризации проводимых властями программ, акций, мероприятий;
- реализации оперативного контроля объектов и дистанционного управления по устранению аварий и чрезвычайных ситуаций;
- оптимизации лечебной и профилактической деятельности, создания систем мобильной телемедицины;
- создания систем дистанционного обучения на базе ведущих вузов, расширения системы подготовки абитуриентов и пр.;
- поддержки малого бизнеса, проведения рекламных мероприятий и др.

Проводимые во всех технически развитых странах разработки алгоритмов и аппаратуры сокращения объема, рационального пакетирования и передачи по каналам связи с различной пропускной способностью видео-, аудио- и сопутствующей информации являются основой эффективного использования телекоммуникационных систем и радиочастотного спектра, сохранения действующих частотных планов, высвобождения значительной части частотного пространства для передачи потребителям дополнительных видов услуг — видеотелефонии,

мобильной и стационарной видеоконференцсвязи, многопрограммного интерактивного телевидения, телевидения повышенной и высокой четкости, технологий трехмерного ТВ (3D-TV), телевидения ультравысокой четкости (ТУВЧ), многопрограммного звукового вещания, а также систем ТВ со многими (в будущем — с произвольным числом) точками наблюдения.

Внедрение видеонформационных систем в России связано с решением ряда проблем:

- эффективного использования мирового опыта по созданию и применению систем цифровой обработки и передачи информации и разработке соответствующих стандартов России;
- разработки и внедрения собственных мультимедийных систем и соответствующей аппаратуры;
- подготовки специалистов для обеспечения разработок, производства и эффективной эксплуатации таких систем;
- создания средств метрологического обеспечения цифровых видеонформационных систем;
- интегрирования в общемировую систему телекоммуникаций с учетом тенденций перехода к системам многопрограммного телевидения и телевидения высокой четкости в глобальном масштабе, передачи информации по наземным, спутниковым, кабельным и Интернет-сетям.

Размер кадров изображений, их частота и формат существенно изменяются в зависимости от вида приложений. Величина компрессии цифрового потока составляет от сотен (для систем с одной точкой наблюдения) до десятков тысяч раз (для многомерных систем).

В настоящее время наряду с улучшением и созданием новых систем сжатия традиционных видеосигналов, снимаемых с одной точки наблюдения, бурно развивается как стереоскопическое телевидение (бинокулярное), так и автостереоскопическое (или многоракурсное, с несколькими точками наблюдения — до 9–16 видов). Также создано и развивается такое направление, как телевизионные системы с произвольной точкой наблюдения (ТСПН или Free Viewpoint TV — FTV).

На рис. П.1 поясняются основные тенденции развития современных видеонформационных систем. Наряду с ростом числа пикселов в системах ТВ с одной точкой наблюдения, выражаящемся в переходе от систем ТВ к ТВЧ и затем к ТУВЧ, наблюдается рост числа точек наблюдения — два вида для стереоскопических систем, до 16 видов для систем со многими видами и практически бесконечное число видов для систем ТСПН (FTV).

Кодирование видеонформации можно охарактеризовать тремя факторами:

- оптимизированные методы сжатия, обладающие высокой эффективностью, но связанные с появлением артефактов на определенных типах изображений; при этом тратятся существенные усилия на улучшение кодирования с целью уменьшения их визуальной зависимости;
- вычислительная сложность кодирования зачастую становится важной проблемой, в том числе и из-за роста мобильных маломощных приложений, в то время как в прошлом это считалось не столь существенным фактором и приводило лишь к необходимости учета ограничений на сложность алгоритма обработки;

- активно рассматриваются новые типы контента (трехмерное телевидение, мультивидовое видео и т. д.), при этом технологии сжатия развиваются так, что особое внимание уделяется общей эффективности кодирования, но зачастую оказывается неясным, как пользователи смогут получить доступ к этим наборам данных и насколько хорошо сжатые форматы подходят к типичным сценариям их использования.

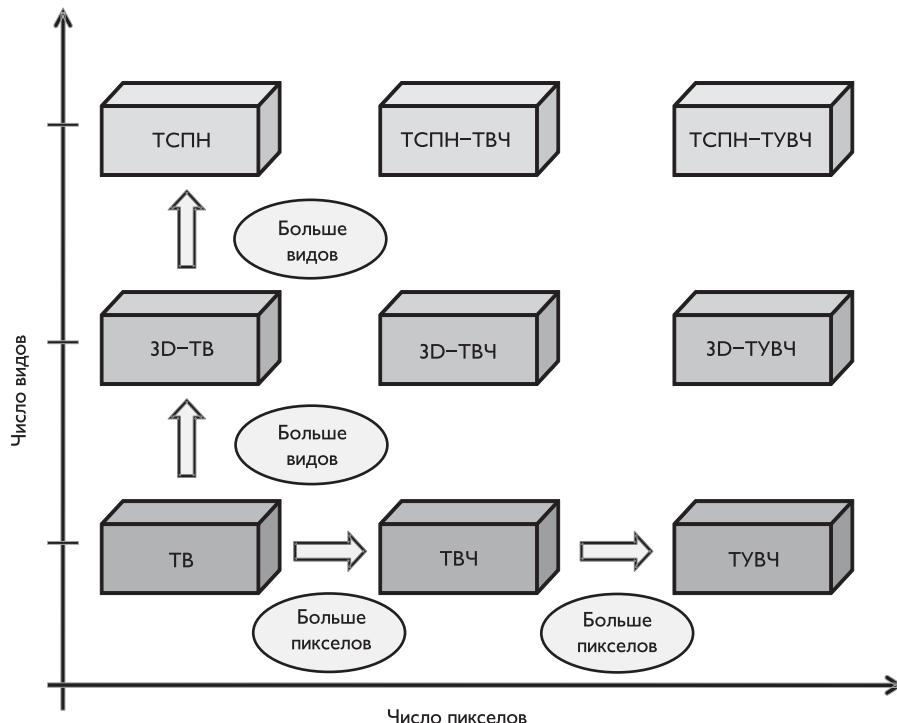


Рис. П.1. Тенденции развития современных видеоинформационных систем

Помимо устранения пространственной избыточности, эффективное временное предсказание движения деталей изображения и его компенсация всегда было ключевым фактором, определяющим общий коэффициент сжатия при кодировании видеоинформации.

Практически реализованные системы кодирования используют блочную оценку движения и его компенсацию, сопровождающую блочным преобразованием изображения, квантованием и энтропийным кодированием. Даже после замены дискретного косинусного преобразования (ДКП) на его целочисленный вариант блочные технологии кодирования видеоинформации остаются подобными первоначально предложенным. Хотя очевидно, что «истинное» движение в кадре не является ни поступательным, ни блочно-постоянным, блочная природа этих алгоритмов весьма полезна в вычислительном отношении. Последние разработ-

ки существенно не изменили способов оценки движения. Расширение коснулось изменения размеров блока, повышения пиксельной точности и новых режимов предсказания при внутрикадровом кодировании.

Значительные успехи были достигнуты в создании метрик качества, которые лучше оценивают субъективно воспринимаемое (перцепционное) качество. Однако в большинстве случаев эти новые метрики были разработаны при кодировании статических изображений. Сопоставимые результаты для динамических изображений не так развиты. Развитие перцепционно ориентированных инструментов, которые включали бы временные качественные критерии, является ключевой задачей по улучшению рабочих характеристик систем кодирования видео. Заметим, что эти виды инструментов достаточно хорошо развиты для кодирования аудио.

Теоретически кодер будет более эффективным при применении соответствующих методов оптимизации, реализация которых приводит к использованию множества способов выбора режимов его работы в зависимости от структуры кодируемых динамических изображений.

В связи с резким увеличением числа возможных комбинаций режимов и его влиянием на вычислительную сложность таких кодеков анализируются различные возможности построения пространства решений, исследуются алгоритмы и синтаксис кодирования, которые понизят общее количество допустимых режимов (мод), ограничивают выбор режимов блоков предсказания и компенсации движения деталей в нескольких кадрах динамических изображений.

В системах с произвольной точкой наблюдения пользователи практически всегда имеют неполный доступ к набору данных, что приводит к созданию кодеров, позволяющих декодерам обладать некоторой гибкостью. Предположим, сигналы от многих камер совместно кодировались, используя инструменты, развитые в контексте многоракурсного кодирования (MVC — Multiview Coding). Тогда пользователи могут изменять угол наблюдения экрана, выбирая только один из видов в потоке битов. Такие наборы данных должны позволять более комплексный доступ к видеинформации, разрешая пользователям выбор точки наблюдения и управления произвольным образом. При этом декодеру нужно позволить использовать различные пути декодирования, каждый с различным порядком просмотра данных, таким образом соответствуя различным определениям того, что является будущим, а что — прошлым с точки зрения цели декодирования.

Основными организациями по стандартизации кодирования видео являются:

- ITU-T — Группа экспертов кодирования видео (Video Coding Experts Group — VCEG) в Международном союзе электросвязи — Телекоммуникационный сектор стандартизации (International Telecommunications Union — Telecommunications Standardization Sector — ITU-T, организация ООН, ранее МККТТ — CCITT), Исследовательская группа 16, Вопрос 6 (Study Group 16, Question 6);
- ISO/IEC — Группа экспертов подвижных изображений (Moving Picture Experts Group — MPEG) в Международной организации по стандартизации и Международной электротехнической комиссии, Объединенный технический комитет 1, Подкомиссия 29, Рабочая группа 11 (International Standardization Organization and International Electrotechnical Commission, Joint Technical Committee Number 1, Subcommittee 29, Working Group 11).

Движущей силой создания стандартов кодирования видеоинформации являются различные приложения и развитие их аппаратного обеспечения. Блочное гибридное кодирование является, по сути, ядром всех стандартов сжатия видео. Среди прошлых стандартов можно перечислить H.261, MPEG-1 часть 2, MPEG-2 часть 2/H.262, H.263 и MPEG-4 часть 2 (см. рис. П.2). Наиболее эффективным является стандарт MPEG 4 часть 10 AVC/H.264. К этому стандарту были приняты поправки по кодированию масштабируемого видео (SVC — Scalable Video Coding), кодированию многоракурсного видео (MVC — Multiview Video Coding) и реконфигурируемого кодирования видео.

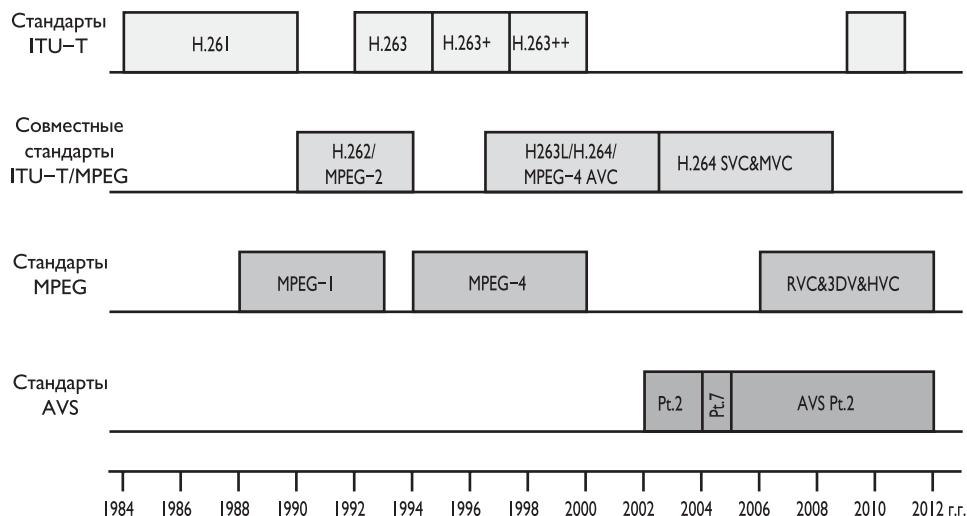


Рис. П.2. История развития методов кодирования видео

Гибридное кодирование комбинирует два метода: движение от кадра к кадру оценивается и компенсируется с помощью предсказания, основанного на ранее за- кодированных кадрах; остаточная разность после предсказания кодируется, данные декоррелируются в пространственной области посредством преобразования в 2-D-частотную (или вейвлет) область. Преобразованные данные квантуются, после чего данные энтропийно кодируются с помощью метода Хаффмана, арифметического кодера или иного метода.

Стандартный ряд гибридного кодирования H.26x (x = 1 … 4) рекомендован группой VCEG (Video Coding Experts Group) ITU-T. MPEG-x (x = 1, 2, 4) разрабатывается группой MPEG (Moving Picture Experts Group) и стандартизируется ISO/IEC.

Группа VCEG ответственна также за работу ITU-T над стандартами кодирования неподвижных изображений, включая JPEG (ITU-T T.80, T.81, T.83, T.84, и T.86), JBIG-1 (ITU-T T.80, T.82 и T.85), JBIG-2 (ITU-T T.88 и T.89), LS JPEG (ITU-T T.87 и T.870), JPEG 2000 (ITU-T T.800 — T.812), подобный JPEG ITU-T T.851, и JPEG XR (ITU-T T.832, T.834 и T.835). VCEG работает над большинством этих стандартов совместно с ISO/IEC JTC 1/SC 29/WG 1 (Совместная экспертная группа по фотографии/Объединенная группа экспертов по бинарным изображениям — Joint Photographic Experts Group/Joint Bi-level Image experts

Group). Исследовательская группа ответственна за исследования, касающиеся возможностей мультимедийной службы и возможностей приложений (включая поддержку сетей будущего поколения — NGN). Это охватывает мультимедийные терминалы, системы (сетевое оборудование обработки сигналов, многоточечные модули конференций, маршрутизаторы, гейткеперы, модемы, факсы), протоколы и обработку сигналов (кодирование источника).

Несмотря на широкое внедрение цифровых видеоинформационных систем, процесс разработки более эффективных алгоритмов сжатия визуальной и звуковой информации и их передачи по каналам с различной пропускной способностью продолжается. Поскольку имеющиеся стандарты определяют лишь цифровой поток кодируемых данных и не регламентируют принципы его формирования, имеется возможность более эффективной обработки сигналов в рамках этих стандартов.

В 2004 г. организация ITU-T VCEG начала изучение технологических решений, которые могли бы позволить создать новый стандарт сжатия видео, существенно превосходящий H.264/AVC, который получил условное название H.265.

В 2007 г. ISO/IEC (MPEG) запустила проект с экспериментальным названием HVC.

К июлю 2009 г. результаты экспериментов показали среднее понижение потока битов приблизительно на 20% по сравнению с AVC (Высокий профиль); эти результаты побудили MPEG инициировать сотрудничество с VCEG. Заявка на создание технологии сжатия видео была подготовлена VCEG и MPEG в январе 2010 г. Представленные предложения были рассмотрены на первой встрече Объединенной команды по кодированию видео (Joint Collaborative Team on Video Coding — JCT-VC) MPEG и VCEG, которая прошла в апреле 2010 г. JCT-VC была создана как группа экспертов по видеокодированию от Исследовательской группы ITU-T 16 (VCEG) и ISO/IEC JTC 1/SC 29/WG 11 (MPEG) с целью разработки стандарта видеокодирования нового поколения. Оценки показали, что некоторые из представленных 27 предложений могли обеспечить такое же визуальное качество, как у AVC, при уменьшении в два раза потока битов на многих тестовых последовательностях за счет увеличения вычислительной сложности в 2–10 раз. Некоторые предложения обеспечивали хорошее субъективное качество и значения потока битов с более низкой вычислительной сложностью, чем у AVC. Для совместного проекта было принято название HEVC (High Efficiency Video Coding — высокоэффективное видеокодирование). В настоящее время JCT-VC работает над объединением основных технологий лучших предложений в одно программное обеспечение. График работ нацелен на получение проекта конечного стандарта для HEVC приблизительно к январю 2013 г.

На рис. П.3 приведены зависимости отношения сигнал/шум MPEG-кодеров от реализуемых ими цифровых потоков. Из этих зависимостей следует, что стандарт H.264/AVC по эффективности существенно превосходит ранее разработанные кодеры стандартов MPEG-1, MPEG-2 и MPEG-4.

Мультивидовое (или многоракурсное) видео (MVC — Multiview Video Coding) обеспечивает обширную информацию о сцене и расширяет зрительские впечатления по сравнению с традиционным видео. Благодаря усовершенствованиям технологий захвата и отображения мультивидовое видео станет востребованным в потребительской сфере, включая 3DTV и телевидение с произвольной точкой

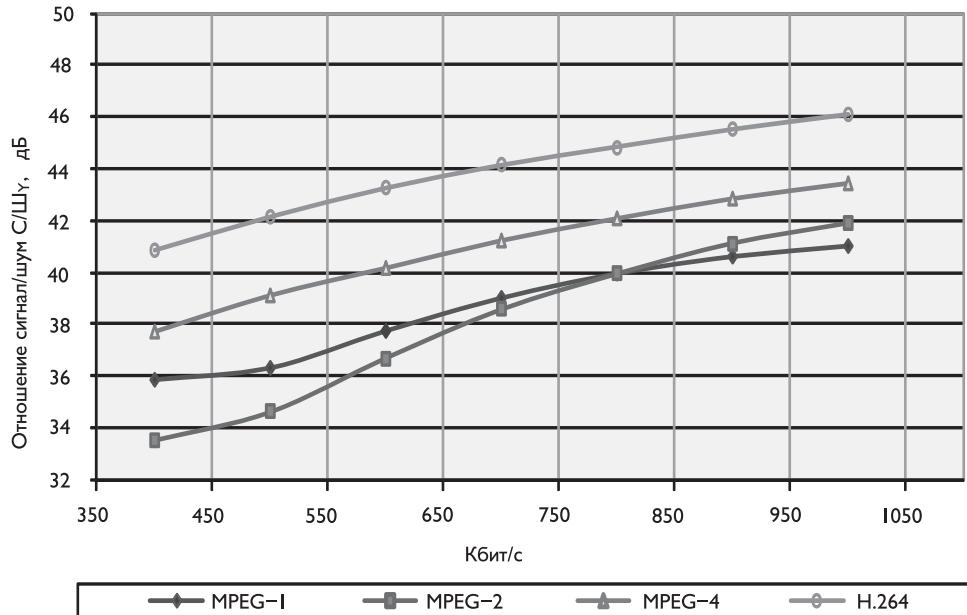


Рис. П.3. Сравнение характеристик MPEG-кодеров

наблюдения (ТСПН — FTV). Среди реализаций 3D-приложений MVC является одной из наиболее многообещающих технологий, поскольку огромный объем данных, пропорциональный числу камер, должен быть сжат до такой степени, чтобы его можно было передавать в пределах возможностей сетей связи.

Поправки MVC к структуре кодера H.264/AVC допускают широкий диапазон совместно применяемых структур временного и межвидового предсказания, что делает возможным достижение компромисса между эффективностью кодирования и сложностью декодирования, включая управление задержкой и буфером декодированного изображения. Все возможные структуры совместно применяемого временного и межвидового предсказания в основном относятся к особым случаям, определяемым поправкой масштабируемого кодирования видео (SVC).

Тенденции развития алгоритмов кодирования и полученные степени сжатия в их историческом развитии иллюстрируются рис. П.4, из которого следует, что стандарт H.264/AVC при высоком качестве кодирования обеспечивает формирование информационного потока при величине 0,17 бита на пиксель изображения.

Это означает, что при кодировании стандартного ТВ-сигнала (8×3 бит/пиксель, 720×576 пикселов/кадр, 25 кадров/с) удается сформировать потоковое изображение менее 2 Мбит/с, а при кодировании сигнала ТВЧ — порядка 5 Мбит/с.

Предполагается, что новый стандарт H.265 обеспечит до 20% улучшения объективных критериев и, кроме того, дополнительное уменьшение потока будет связано с изменением субъективной оценки, т. е. в стандарте H.265 будут допускаться большие потери, чем в H.264, но эти потери будут не так очевидны для восприятия.

После успешной разработки множества стандартов кодирования видео группа MPEG приступила к деятельности по стандартизации реконфигурируемого

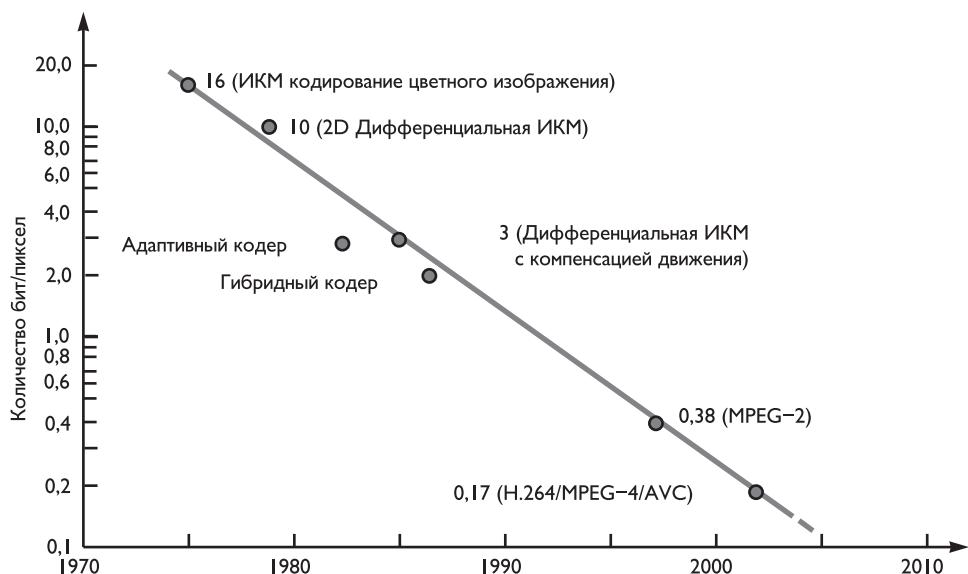


Рис. П.4. Достижения в кодировании видеосигналов

кодирования видео (RVC — Reconfigurable Video Coding). Стандарт RVC предлагает базовую структуру, подходящую для построения видеокодека с помощью конфигурации средств кодирования, таким образом способствуя динамической разработке, реализации и принятию решений по стандартизованному кодированию видео.

Совершенствование и развитие современных мультимедийных систем связано не только с реализацией эффективных методов сжатия различных видов информации, но и ее передачи в реальных сетях связи и вещания, предназначенных для доставки контента телерадиовещания, видеоконференцсвязи и ряда дополнительных услуг передачи данных. Такие сети, как правило, предназначены для достижения заданных качества обслуживания и режимов передачи/приема (например, с использованием наземных, спутниковых и кабельных сетей, Интернета).

В частности, цифровое телевизионное вещание в Европе основано на очень успешной серии стандартов DVB. Частотные распределения согласованы в рамках международного союза электросвязи ITU, в то время как сами сети вещания реализуются в соответствии с национальными спецификациями.

Как показано на рис. П.5, внедрение цифрового формата осуществляется в трех основных направлениях — спутникового, наземного и кабельного вещания. Первыми, последовательно внедряемыми, были спутниковые стандарты DVB-S, а затем DVB-S2. В дополнение к этим стандартам следует добавить стандарт DVB-RCS/RCS2 (RCS — Return Channel via Satellite), обеспечивающий реализацию обратного канала на основе режима многочастотного доступа с разделением по времени (MF-TDMA) для интерактивных спутниковых услуг, а также стандарт DVB-DSNG, используемый для сбора новостей через спутники.

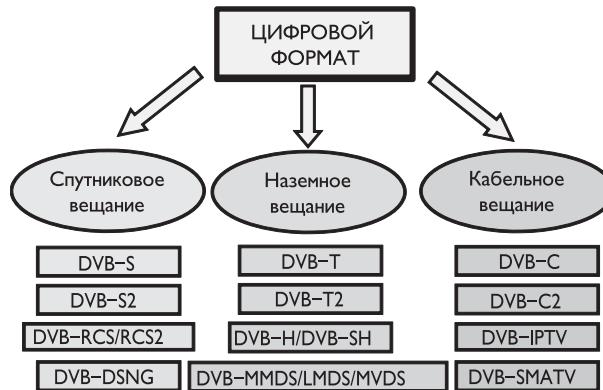


Рис. П.5. Научно-технический прогресс в сфере вещательных технологий

Стандарты DVB-T, а затем DVB-T2 определяют принципы трансляции эфирного цифрового телевизионного вещания. Стандарты DVB-H/DVB-SH используют эфирные и спутниковые каналы для передачи узкополосного телевизионного вещания на малогабаритные приемники с малыми экранами. Традиционно MMDS (Microwave Multipoint Distribution Service) — системы беспроводного многоканального телевидения ориентированы на микроволновые частоты в диапазоне 2,1 ГГц и от 2,5 ГГц до 2,7 ГГц. В последнее время все большее внимание уделяется системам беспроводной передачи на частотах выше 20 ГГц. В этой области стандартно используются диапазоны 25–32 ГГц и 40,5–42,5 ГГц. Нижний диапазон был первоначально освоен в США и Канаде, где системы высокочастотной передачи использовались вместо витой пары при построении телекоммуникационных сетей. В соответствии с назначением системы получили название LMDS (Local Multipoint Distribution Service). Сейчас технология LMDS при полосе каждого канала от 950 до 2150 МГц начала использоваться и в Европе, для построения телекоммуникационных, в том числе телевизионных сетей. Верхний диапазон первоначально был выделен в Европе для аналогового ТВ-вещания. В связи с этим системы, работающие на частотах 40,5–42,5 ГГц, получили название MVDS (Multipoint Video Distribution Systems). Наиболее привлекательным качеством систем MVDS является колоссальная ширина предоставляемого диапазона — 2 ГГц. Это в два с лишним раза превышает диапазон наземного вещания и в 10 раз — частотную полосу систем MMDS.

Системы DVB-C и DVB-C2 регулируют цифровые стандарты кабельного телевизионного вещания, использующие VHF и UHF-диапазоны частот на вторичной основе. К кабельным системам примыкают также стандарт телевизионного вещания по IP-сетям DVB-IPTV и стандарт DVB-SMATV (SMATV — Satellite Master Antenna Television), подразумевающий непосредственное распределение спутниковых сигналов по сетям кабельного телевидения. Действие европейского стандарта DVB-SMATV распространяется на кабельные сети до 3000 МГц, т. е. полностью — на весь диапазон промежуточных частот 950–2400 МГц.

С 2010 года внедряются стандарты цифрового вещания второго поколения для систем спутникового, эфирного и кабельного вещания — DVB-S2/DVB-T2/DVB-C2.



Эти стандарты применяются в службах вещания, распределения контента, электронного сбора новостей и подачи сигнала в студию, интерактивных службах. По сравнению с первым поколением стандартов DVB-S/DVB-T/DVB-C были изменены параметры обработки сигналов: использовано новое поколение прямой коррекции ошибок, что позволило вплотную приблизиться к пределу Шеннона; применяются более высокие созвездия, что привело к существенному повышению эффективности; увеличено возможное количество несущих OFDM; вводятся новые защитные интервалы; осуществлена минимизация количества рассеянных пилотов в зависимости от защитного интервала, в результате чего сокращены накладные расходы; расширена полоса пропускания; повышена эффективность обработки информации за счет расширенного временного и частотного перемещения битов и ячеек.

В табл. П.1 приведено сравнение достигаемых скоростей цифрового потока различных систем цифрового вещания первого и второго поколений для стандартных сетей связи.

Таблица П.1. Скорости цифрового потока, Мбит/с

Полоса канала	26 МГц		54 МГц		8 МГц			
	DVB-S	DVB-S2	DVB-S	DVB-S2	DVB-T	DVB-T2	DVB-C	DVB-C2
Минимальная	18,7	12,9	38,9	26,2	4,9	7,4	6,4	25,0
Максимальная	32,8	116,7	68,8	242,4	31,6	50,5	38,1	78,6

Из сравнения возможных максимальных скоростей цифрового потока, передаваемого по каналам цифрового телевизионного вещания систем первого и второго поколений стандартов, вытекает существенное увеличение эффективности использования стандартных каналов связи.

Так, например, если при использовании стандарта эфирного вещания DVB-T через канал с полосой пропускания 8 МГц возможно максимально передать до 14 программ стандартного телевидения или не более 6 программ телевидения высокой четкости, то при переходе на стандарт DVB-T2 эти данные могут быть увеличены примерно в 1,5 раза — до 25 и 10 соответственно.

Следует упомянуть несколько форматов мультимедийного вещания, ориентированных на прием с помощью мобильных терминалов. К ним относятся упомянутый при рассмотрении эфирных систем формат DVB-H, формат системы MediaFLO и группу форматов, базирующихся на системе радиовещания DAB (T-DAB, DAB+, T-DMB), японский стандарт эфирного вещания ISDB-T и китайский формат DTMB. Для сжатия информации стандартно в настоящее время используется технология MPEG-4 (H.264/AVC и HE-AAC), цифровые аудио- и видеосигналы мультиплексируются в один мультимедийный высокоскоростной транспортный поток.

Разработанная в России аудиовизуальная информационная система реального времени РАВИС (RAVIS) обеспечивает решение проблемы цифровизации сети мобильного радиовещания взамен аналоговых систем ЧМ-вещания (с полярной модуляцией и с пилот-тоном) в выделенных для этих целей в России диапазонах частот (66–74 МГц и 87,5–108 МГц соответственно) и внедрение собственных мультимедийных систем и соответствующей аппаратуры.

Система РАВИС позволяет в стандартном канале (200–250 кГц) передавать либо много программ стереозвука, включая поддержку объемного звукового ве-

щания, либо качественное цифровое телевизионное изображение с разрешением 352×288 пикселов на кадр и соответствующий звуковой стереосигнал.

Полностью отечественная разработка РАВИС имеет ряд ключевых преимуществ:

- возможность в стандартном канале ЧМ-вещания передачи нескольких стереофонических и многоканальных звуковых программ;
- трансляцию видеопрограмм для мобильного потребителя;
- реализацию телевизионного вещания в малонаселенных пунктах при ущемлении системы кодирования и трансляции;
- обеспечение устойчивого мобильного приема сигнала в условиях многолучевого распространения в среде с городской застройкой, в горной и лесистой местности, в водных акваториях;
- радиус покрытия в десятки раз больше, а затраты на создание инфраструктуры мобильного телевещания в десятки раз ниже (к примеру, покрытие мобильным телевещанием всей территории Москвы может быть обеспечено одним либо максимум тремя передатчиками, работающими в одночастотном режиме);
- возможность создания одночастотных сетей мобильного вещания вдоль железнодорожных и автомобильных трасс;
- реализацию эфирных систем оповещения населения и организаций в условиях ЧС;
- организацию систем для использования различными спецслужбами и др.

В конце XX века одной из бурно развивающихся технологий в области телекоммуникаций стала видеоконференцсвязь (ВКС). Появление видеоконференцсвязи явилось технологическим развитием цифровой телефонии и видеотелефонии. Новые возможности, предоставляемые ВКС, привели к бурному росту количества пользователей этой технологии. Естественно, что развитие ВКС не могло идти без использования предшествующих стандартов ITU и создания новых рекомендаций, позволяющих унифицировать подходы к передаче мультимедийного контента и внедрения дополнительных сервисов.

Серьезной проблемой при внедрении видеоинформационных систем в России является подготовка специалистов данного профиля. Отечественная литература, посвященная изложению современных способов цифровой обработки мультимедийной информации, практически отсутствует, эта важнейшая область не отражена в ныне существующих программах обучения специалистов. Функцией обучения специалистов теории и практике мультимедийных систем, помимо создания представления о весьма сложных преобразованиях сигналов в процессе сокращения объема информации, должно стать привлечение молодых специалистов России к исследованиям еще не решенных проблем в этой перспективной области.

Хотелось бы надеяться, что начавшееся широкое внедрение в практику видеоинформационных систем послужит началом реализации планов обучения специалистов данного профиля. В области организации науки и подготовки кадров важнейшим является формирование сети научно-исследовательских центров по видеоинформационным проблемам, обладающих необходимым научным и тех-



ническим потенциалом для решения как фундаментальных проблем обработки сигналов, так и широкого круга прикладных задач.

Хотя проблемы метрологического обеспечения разработки и внедрения видеоинформационных систем в данной работе почти не затрагиваются, следует указать, что в России сложилась крайне тяжелая ситуация, вызванная повсеместным недостатком, а часто и отсутствием измерительной аппаратуры, необходимой для разработки, настройки и поддержания технических характеристик систем передачи информации в состоянии, обеспечивающем их качественное функционирование.

Особые проблемы возникают в связи с началом широкого внедрения систем и аппаратуры цифровой обработки и передачи компрессированных сигналов по каналам связи. Разрабатываемые системы требуют создания соответствующих контрольных и измерительных средств. С внедрением видеоинформационных систем потребность в использовании измерительной аппаратуры будет возрастать, поскольку требуется создание принципиально новых способов и средств контроля и измерений. При этом необходимо, чтобы эти средства были совместимы и с традиционными аналоговыми системами.

Метрологическая безопасность России — залог создания высококачественной видеоинформационной аппаратуры и ее эффективного использования.

На современном этапе развития техники выполнение требований метрологии связано с созданием виртуальных измерительных систем на базе использования персональных компьютеров в качестве устройств анализа и организации структуры систем формирования и обработки измерительной информации.

Для оценки соответствия качества передаваемой видеоинформации требованиям, предъявляемым к ней в различных системах и условиях передачи, а также для сравнения эффективности работы различных устройств видеокодирования и передачи видеоинформации, требуются методики и средства субъективной и объективной оценки качества видео, наблюдаемого потребителем. Эти методики должны учитывать характерные особенности и искажения как при аналоговой, так и при цифровой и смешанной системах обработки и передачи видеоинформации.

В настоящее время широко дебатируются вопросы, связанные с проблемами реализации и использования различных цифровых абонентских устройств. Самым рациональным решением проблемы приема является применение компьютерных систем, содержащих соответствующие тюнеры. Эффективность использования компьютеров связана также с возможностью постепенного наращивания мультимедийных функций путем замены плат компьютеров на более совершенные и дополнения их новыми платами и программами. При этом весьма привлекательным является использование программных декодеров, что обеспечивает возможность совершенствования стандартов кодирования, не изменяющих аппаратные средства абонентов.

Главным преимуществом применения компьютерной технологии является использование различных дисплеев: от настольных до настенных вариантов, от стандартного разрешения до разрешения высокой четкости и др. При этом изменение типа дисплея практически не требует замены аппаратной части.

В связи с этим важнейшей проблемой проведения фундаментальных работ в России по данной тематике является задача полного овладения теорией и ма-

тематическими методами обработки информации, используя имеющиеся зарубежные материалы в качестве информации, задающей общее направление работ. А далее с учетом полученного опыта по освоению стандартных алгоритмов следует разработка более эффективных систем.

Материал, изложенный в данной книге, состоит из девяти частей, изложенных в 22 главах.

В части I (главы 1–3) «Видеинформационные приложения и объем цифровой информации» сжато изложены проблемы цифрового преобразования статических и динамических изображений, оцениваются объемы информации при преобразовании в цифровую форму изображений различного разрешения — от видеотелефонии до «цифрового кино», изложены принципы эффективного квантования различных видов информации.

Часть II (главы 4, 5) «Статистическая и визуальная избыточность изображений» посвящена исследованиям проблем статистической и визуальной избыточности монохромных и цветных изображений, излагаются методы и алгоритмы эффективного энтропийного кодирования различных видов информации об изображениях, оцениваются возможности использования визуальной избыточности статических и динамических изображений для сокращения объемов передаваемой информации.

В части III (главы 6–9) «Основные методы сжатия изображений» рассматриваются методы непосредственного кодирования и кодирования с предсказанием, групповое кодирование изображений, методы блочного и фрактального кодирования, кратномасштабная обработка изображений. Основное внимание уделено анализу дискретных линейных ортогональных преобразований и, в частности, широко используемым дискретному преобразованию Фурье, косинусному преобразованию, преобразованию Кархунена–Лоэва и др. Подробно анализируются методы обработки изображений с применением вейвлет-преобразований и реализации на их основе кратномасштабных преобразований изображений.

В части IV (главы 10–12) «Внутрикадровая и межкадровая обработка изображений» достаточно подробно излагаются методы повышения качества внутрикадрового кодирования статических изображений, принципы анализа движения деталей и его компенсации в динамических изображениях, идеология обработки изображений при кратномасштабном анализе. Особое внимание уделяется оптимизации методов поиска векторов движения, дополнительным возможностям компенсации движения деталей в динамических изображениях.

Часть V (главы 13, 14) «Основные стандарты цифрового кодирования видеинформации» посвящена анализу методов кодирования статических изображений JPEG и JPEG-2000, системам кодирования динамических изображений MPEG-1, MPEG-2, обобщенному кодированию аудиовизуальных объектов MPEG-4, а также наиболее эффективному стандарту кодирования таких изображений H.264/AVC. Кроме того, в этой части рассмотрены некоторые проблемы реализации кодирующих устройств и визуализации их параметров.

В части VI (главы 15–18) «Основные методы кодирования речевой и звуковой информации» анализируются основные характеристики и цифровое представление речевой и звуковой информации, приводится достаточно подробное описание основных стандартов цифрового кодирования речи: G.711, G.722, G.723.1, G.728, G.729 и др., алгоритмы кодирования высококачественного звука, приведенные



в стандартах MPEG-1, MPEG-2, MPEG-4, Dolby AC-3. Рассмотрены также проблемы подавления акустического эха в системах телеконференцсвязи. Глава 18, посвященная подавлению акустического эха, написана совместно с Сараной Д.В.

Часть VII (главы 19, 20) «Основные методы и системы канального кодирования и модуляции цифровой информации» посвящена анализу основных методов модуляции и помехоустойчивого кодирования цифровой информации, приводится достаточно подробное описание систем многочастотной OFDM-модуляции, алгоритмы синхронизации OFDM-сигналов в приемных устройствах. Глава 19, посвященная канальному кодированию информации, составлена совместно с к. т. н. Бакке А.В.

В части VIII (глава 21) «Основные стандарты систем цифрового телевизионного вещания» приводится достаточно подробное описание основных современных стандартов передачи видео- и аудиоинформации: ATSC, DVB-S/S2, DVB-T/T2, DVB-C/C2, ISDB-S, ISDB-T, DTMB (DMB-T/H). Проводится сравнение европейских стандартов первого и второго поколений. Показано, сколь важен быстрый переход в России на системы вещания второго поколения DVB-S2, DVB-T2, DVB-C2.

Последняя часть IX (главы 22, 23) «Стандарты систем мобильного радиовещания и видеоконференцсвязи» посвящена изложению стандартов телерадиовещания DVB-H, MediaFLO, T-DMB/T-DAB, РАВИС и DRM+, а также проблемам реализации систем видеоконференцсвязи на базе использования трех разделов рекомендаций ITU-T:

- аудиовизуальные и мультимедийные системы (серия H);
- передающие системы и средства, цифровые системы и цепи (серия G);
- терминалы для телематических сервисов (серия T).

Основное внимание удалено изложению принципов построения российской аудиовизуальной системы реального времени — РАВИС, предназначенной для создания цифровых систем взамен аналоговых сетей УВЧ/FM радиовещания, а также реализации разработки российских программных решений видеоконференцсвязи VPhone и «ДАВ телекон». Глава 23, в которой рассмотрены проблемы построения систем видеоконференцсвязи, написана совместно с к. м. н. Федоровым В.Ф.

ЧАСТЬ I

ВИДЕОИНФОРМАЦИОННЫЕ ПРИЛОЖЕНИЯ И ОБЪЕМ ЦИФРОВОЙ ИНФОРМАЦИИ

Введение

Процесс цифровой обработки визуальной информации охватил различные виды видеоинформационных приложений и существенно расширил их спектр: кроме широко распространенного традиционного телевидения и различных экзотических разновидностей ТВ специального назначения появились и повсеместно внедряются самые различные видеоприложения — от видеотелефонии до телевидения высокой четкости (ТВЧ) и «цифрового кино» [1.1, 1.2].

Цифровые системы кодирования изображений, их передачи и приема позволяют достичь невиданного доселе уровня качества и предоставляют пользователю массу новых возможностей и новых видов услуг [1.1–1.4].

Неудивительно в связи с этим то, что цифровая обработка телевизионных и компьютерных изображений ввиду ее особой важности выделилась в самостоятельную область техники, в которую входят [1.5–1.7]:

- коррекция изображений, их «препарирование», т. е. сознательное разделение на части цифровыми средствами, видоизменение этих частей и их обратная «сборка»;
- оценка параметров изображений с целью контроля качества их передачи и приема;
- преобразование и кодирование изображений для хранения или передачи по каналам связи;

- компьютерная графика, а также визуализация информации, т. е. представление массивов данных в виде различных изображений, что очень эффективно, так как облегчает решение многих задач, сложных именно своей абстрактностью.

Эта область включает также моделирование систем обработки, хранения и передачи визуальной информации по каналам связи, т. е. набор компьютерно-математических задач, необходимых для разработки новой цифровой телевизионной техники.

Разумеется, все перечисленные задачи связаны между собой — и по методам решения, и по используемым для этого техническим средствам.

Цифровая обработка изображений является, по сути, основной базой для создания нового поколения видеинформационной техники. Без нее немыслима реализация систем видеотелефонии, видеоконференцсвязи, цифрового интерактивного телевизионного вещания, цифрового кино. Работы по созданию таких систем уже полным ходом ведутся сегодня в технически развитых странах, и привлеченные финансовые, технические и интеллектуальные ресурсы таковы, что становится совершенно ясно: переход к цифровым видеинформационным системам в общемировом масштабе неизбежен и является делом близкого будущего.

Именно по всем этим причинам исследования методов цифровой обработки видеинформации должны стать одной из важнейших частей курса обучения специалистов широкого профиля, и в частности, телевидения.

Однако и сама цифровая обработка изображений в связи с особенностями проблем, стоящих на пути создания видеинформационных систем будущего, имеет свое ядро, т. е. свою главную задачу, без решения которой немыслимо решение всех остальных.

Это — задача так называемого сжатия изображений, т. е. уменьшения объема информации, необходимого для передачи информации по каналам связи путем сокращения содержащейся в составе изображения информационной избыточности. Таким образом, ключевыми вопросами, решение которых определяет эффективность развивающихся и вновь создаваемых систем, являются задачи компрессии видеинформации и оценки ее качества.

Решение именно этой задачи имеет, в частности, важное значение для внедрения телевидения высокой четкости. Упомянутая «высокая четкость» в этих системах достигается увеличением как количества строк в кадре, так и количества элементов в каждой строке, а значит, резко возрастает и соответствующий каждому изображению (кадру) объем информации.

Между тем передавать эти изображения предстоит с использованием существующих каналов связи, пропускная способность которых, естественно, остается прежней. Становится очевидным, что как только будет создан эффективный и достаточно просто реализуемый алгоритм сжатия телевизионных изображений (а такие работы ведутся давно, и варианты решения этой задачи практически уже найдены), внедрение телевидения высокой четкости станет реальностью.

Однако не только для телевидения высокой четкости нужно сжатие изображений. Максимально эффективное использование существующих каналов связи является сегодня для телевидения проблемой номер один.

Частотное пространство телевидения во многих случаях ограничено, а, кроме того, крайне привлекательным представляется внедрять новые системы телеви-

дения, не меняя при этом существующую линейную каналообразующую аппаратуру. Поэтому решение такой задачи, как обеспечение возможности передачи хотя бы 4–6 программ телевидения в стандартном радиоканале, является сегодня исключительно важным, прежде всего с точки зрения экономической.

Таким образом, разработка эффективного способа сжатия изображений позволит решить сразу две задачи, равные по своей важности.

Главной функцией данного материала является создание ясного представления о сложных цифровых преобразованиях сигналов в процессе сокращения объема передаваемой визуальной информации.

Прежде чем перейти к изложению методов цифровой обработки видеоинформации, необходимо оценить ее объем для различных систем формирования и передачи изображений. Этому и посвящен первый раздел настоящего материала. В нем кратко изложены сведения об особенностях существующих стандартов телевидения, а также систем кодирования цвета, а кроме того, рассмотрены некоторые вопросы, связанные с представлением в цифровом виде сигналов различных систем передачи изображений. Следует отметить, что везде в этом разделе, где будет говориться о способах уменьшения объема информации при цифровой передаче изображений, имеются в виду широко известные тривиальные методы предварительного снижения объема цифрового потока, которые сами по себе не в состоянии обеспечить решения изложенных выше задач и не имеют ничего общего с собственно алгоритмами сжатия изображений, содержание которых изложено в последующих разделах.

ГЛАВА I

ФОРМАТЫ ИЗОБРАЖЕНИЙ. СТАТИЧЕСКИЕ И ДИНАМИЧЕСКИЕ ЭТАЛОННЫЕ ИЗОБРАЖЕНИЯ

1.1. Аналоговые и цифровые растровые изображения

Телевизионные и компьютерные изображения обычно воспроизводятся в виде растра, представимого непрерывной или дискретной функцией вдоль строк по горизонтали и конечным числом строк по вертикали. В аналоговом телевидении каждый кадр динамического изображения формируется путем чересстрочной развертки в виде двух последовательно передаваемых полей [1.8, 1.9].

На рис. 1.1 приведено схематичное представление чересстрочной развертки изображения. Сплошные черные линии характеризуют движение луча электронной трубы в одном поле, а пунктирные — в другом поле кадра. Тонкие серые штриховые линии характеризуют обратный ход луча по строкам и между полями.

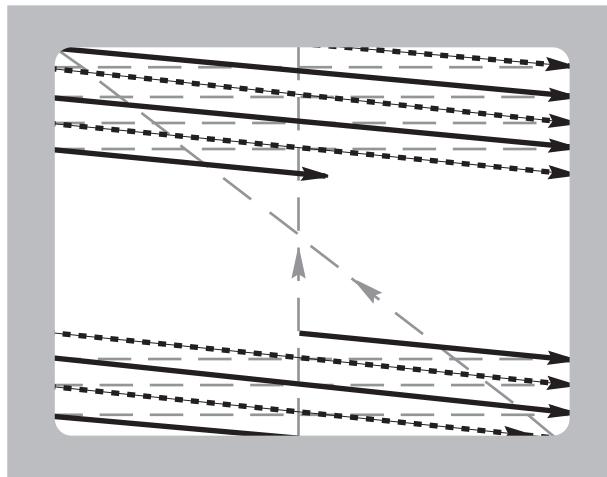


Рис. 1.1. Представление растра аналогового телевидения

Цифровые изображения разбиваются также и вдоль строк на дискретные отсчеты, называемые **пикселями** (от сокращения английских слов — **PICTure'S Element**, элемент изображения). Количество пикселов, на которые разбито изображение, определяет его детальность. Если изображение представимо в виде M

строк, каждая из которых содержит N дискретных отсчетов, то массив кадра состоит из $N \times M$ пикселов.

В различных приложениях используются разные форматы кадров. Так, эталонные компьютерные изображения часто имеют равное количество строк и пикселов в каждой строке, т. е. NM . В стандартном телевидении используется формат кадра с соотношением $N : M = 4 : 3$, а в телевидении высокой четкости и в цифровом кино — с соотношением $N : M = 16 : 9$ [1.2]. Однако номенклатура используемых форматов кадра значительно более обширна, часть достаточно широко используемых форматов приведена в табл. 1.1 [1.10–1.18].

Таблица 1.1. Форматы кадра изображения

Наименование формата	Разрешение, $N \times M$	Соотношение сторон кадра	Применение
1	2	3	4
Общий формат обмена CIF и его производные			
QQCIF	88 × 72	11 : 9	Мобильные телефоны
SQCIF (Sub-QCIF)	128 × 96	4 : 3	- «-
QCIF	176 × 144	11 : 9	Видеоконференция (PAL/SECAM)
CIF	352 × 288	- «-	- «-
iCIF	352 × 576	11:18	Цифровое видео (PAL/SECAM)
2CIF	704 × 288	22:9	- «-
4CIF	704 × 576	11:9	ТВ PAL/SECAM
16CIF	1408 × 1152	- «-	Цифровое видео
Стандартный формат изображения SIF и его производные			
SQSIF	80 × 60	4:3	Мобильные телефоны
QSIF	160 × 120	- «-	- «-
SIF	320 × 240	- «-	Видеоконференция (NTSC)
iSIF	352 × 480	11:15	Цифровое видео(NTSC)
2SIF	704 × 240	44:15	-
4SIF	704 × 480	22:15	ТВ NTSC
Форматы графических подсистем			
QQQVGA	80 × 60	4:3	Мобильные телефоны
QQVGA	160 × 120	- «-	- «-
QVGA	320 × 240	- «-	Видеоконференция
VGA	640 × 480	- «-	ТВ NTSC
WVGA	800 × 480	5:3	Компьютерные мониторы
SVGA	800 × 600	4:3	- «-
XGA	1024 × 768	- «-	- «-
WXGA	1280 × 768	5:3	- «-
QuadVGA	1280 × 960	8:3	- «-
SXGA	1280 × 1024	5:4	- «-
SXGA+	1400 × 1050	4:3	- «-

Таблица 1.1 (окончание)

1	2	3	4
WSXGA	1440×900	8:5	- <<-
UGA, UVGA	1600×1200	4:3	- <<-
UXGA	1600×1280	5:4	- <<-
WSXGA+	1680×1050	8:5	- <<-
WUXGA	1920×1200	- <<-	- <<-
QXGA	2048×1536	4:3	- <<-
WQXGA	2560×1600	8:5	- <<-
QSXGA	2560×2048	5:4	- <<-
WQSXGA	3200×2048	25:16	- <<-
QUXGA	3200×2400	4:3	- <<-
WQUXGA	3840×2400	8:5	- <<-
HXGA	4096×3072	4:3	- <<-
WHXGA	5120×3200	8:5	- <<-
HSXGA	5120×4096	5:4	- <<-
WHSXGA	6400×4096	25:16	- <<-
HUXGA	6400×4800	4:3	- <<-
WHUXGA	7680×4800	8:5	- <<-
Видеоформаты и телевизионные форматы			
w288p	512×288	16:9	Широкоформатное видео
400p	528×400	4:3	Видео стандартной четкости
448p	576×448	- <<-	- <<-
w384p	672×384	16:9	Широкоформатное видео
ТВ стандарт (D1)	720×480	3:2	DVD-Video NTSC
ТВ стандарт (D1)	720×576	5:4	DVD-Video PAL/SECAM
w448p	768×448	16:9	Широкоформатное видео
w480p, ED	854×480	- <<-	LCD/PDP ТВ-дисплеи
w576p	1024×576	- <<-	Широкоформатное видео
w720p, HD	1280×720	- <<-	ТВ повышенной четкости
w768p	1366×768	- <<-	LCD/PDP ТВ дисплеи
w1080p, UHD	1920×1080	- <<-	ТВЧ ATSC
w1152p, UHD	2048×1152	- <<-	ТВЧ DVB-T
w4320p, UHDV	7680×4320	- <<-	ТВ сверхвысокой четкости

Форматы кадров в этой таблице разбиты на 4 раздела.

Первый из них характеризует **Общий Формат Обмена CIF (Common Intermediate Format)**, обычно используемый в телеконференциях, и производ-

ные от него форматы. Второй раздел определяет параметры стандартного формата изображения **SIF (Standard Image/Interchange Format)** и его производных. Третий, наибольший раздел характеризует форматы графических подсистем, используемые в основном в компьютерных дисплеях. И, наконец, в четвертом разделе приведен перечень основных форматов, используемых в телевизионных системах стандартной и высокой четкости.

В зависимости от параметров каждого пикселя изображения можно разделить на бинарные, полутонаовые, полноцветные и палитровые [1.6, 1.19].

Пиксели **бинарных изображений** имеют лишь два значения, характеризующие черные или белые точки. Такая информация используется для передачи простых изображений, например при передаче факсов.

Полутонаовые изображения характеризуются значительным количеством уровней на каждый пиксель — его величина определяет яркость воспроизведенного элемента. Существуют различные форматы полутонаовых изображений, определяющие динамический диапазон воспроизведенных яркостей элементов. Чаще всего используется формат элементов, который кодируется 8 битами на пиксель, т. е. может принимать значения от 0 до $2^8 - 1 = 255$. В ряде случаев, например в медицинских технологиях, используют полутонаовые изображения, элементы которых кодируются до 16 битов на пиксель, т. е. число возможных градаций их яркости может составлять $2^{16} = 65\,536$.

Каждый пиксель **полноцветного изображения** представим в виде тройки — красной (**Red**), зеленой (**Green**) и синей (**Blue**) составляющих, каждая из которых обычно задается 8 битами — значениями от 0 до 255. Следовательно, цвет каждого пикселя определяется 24 битами, и изображение теоретически может содержать $(2^8)^3 = 16\,777\,216$ различных цветов.

Величины красного, зеленого и синего цветов являются *координатами цвета* и обозначаются соответственно X , Y и Z . Часто описание цвета выражают *координатами цветности* — относительными величинами, которые задаются соотношениями:

$$x = \frac{X}{X + Y + Z}; \quad y = \frac{Y}{X + Y + Z}; \quad z = \frac{Z}{X + Y + Z}. \quad (1.1)$$

Из (1.1) следует, что $x + y + z = 1$.

На рис. 1.2 приведено схематическое изображение цветового куба xyz . Точки, соответствующие красному, зеленому и синему цветам, расположены на трех вершинах куба с координатами $(1, 0, 0)$, $(0, 1, 0)$ и $(0, 0, 1)$. Точки на главной диагонали представляют собой оттенки серого цвета: от черного в начале координат $(0, 0, 0)$ до белого в точке $(1, 1, 1)$.

На рис. 1.3 представлены изображения цветового куба, состоящего из $(2^8)^3$ различных цветов. Иногда 24-битовое цветное изображение называют изображением в естественных цветах, поскольку при этом возможно воспроизведение практически всех оттенков реально существующих предметов.

Палитровые изображения используются в целом ряде приложений, в которых имеет смысл использовать ограниченное число цветов. При этом используемое количество воспроизведенных цветов ограничено величиной 256 или меньшим числом. Применительно к Интернет-приложениям это подмножество цветов называется палитрой фиксированных Web цветов или набором цветов, одинаково

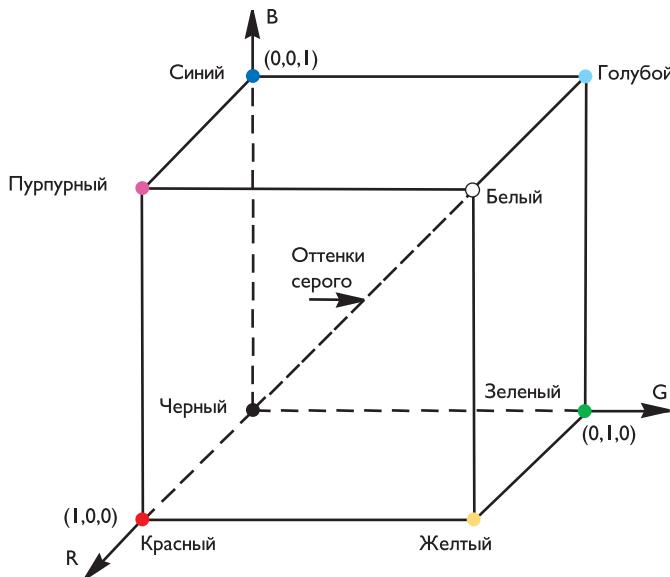


Рис. 1.2. Схематическое изображение цветового куба

воспроизводимых всеми программами просмотра Интернет-сайтов. Поскольку из 256 цветов только 216 воспроизводятся одинаково большинством операционных систем, эти $216 = 6^3$ цветов стали *de facto* стандартом фиксированных цветов (см. рис. 1.4).

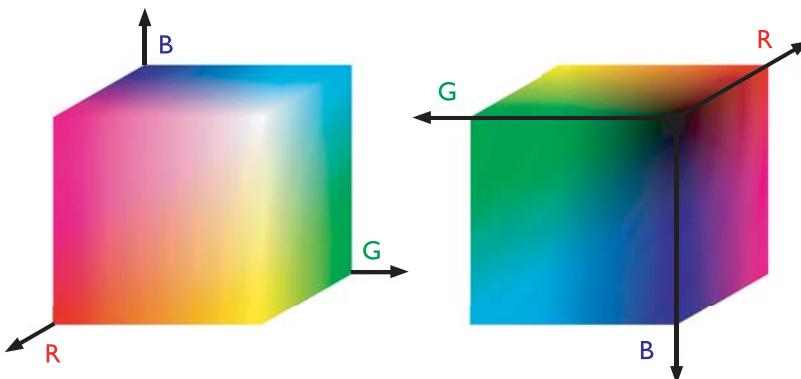


Рис. 1.3. 24-битовый полноцветный куб RGB: вид с двух сторон

В данном варианте каждая из трех RGB компонент может принимать лишь 6 возможных значений: 0, 51, 102, 153, 204 и 255. Эти значения обычно выражают в шестнадцатеричной системе счисления, как приведено в табл. 1.2.

Поскольку для формирования цвета требуется три числа, каждый цвет из палитры задается тремя двузначными числами (в шестнадцатеричной системе

счисления). Так, шесть воспроизводимых оттенков серого цвета, входящих в па-

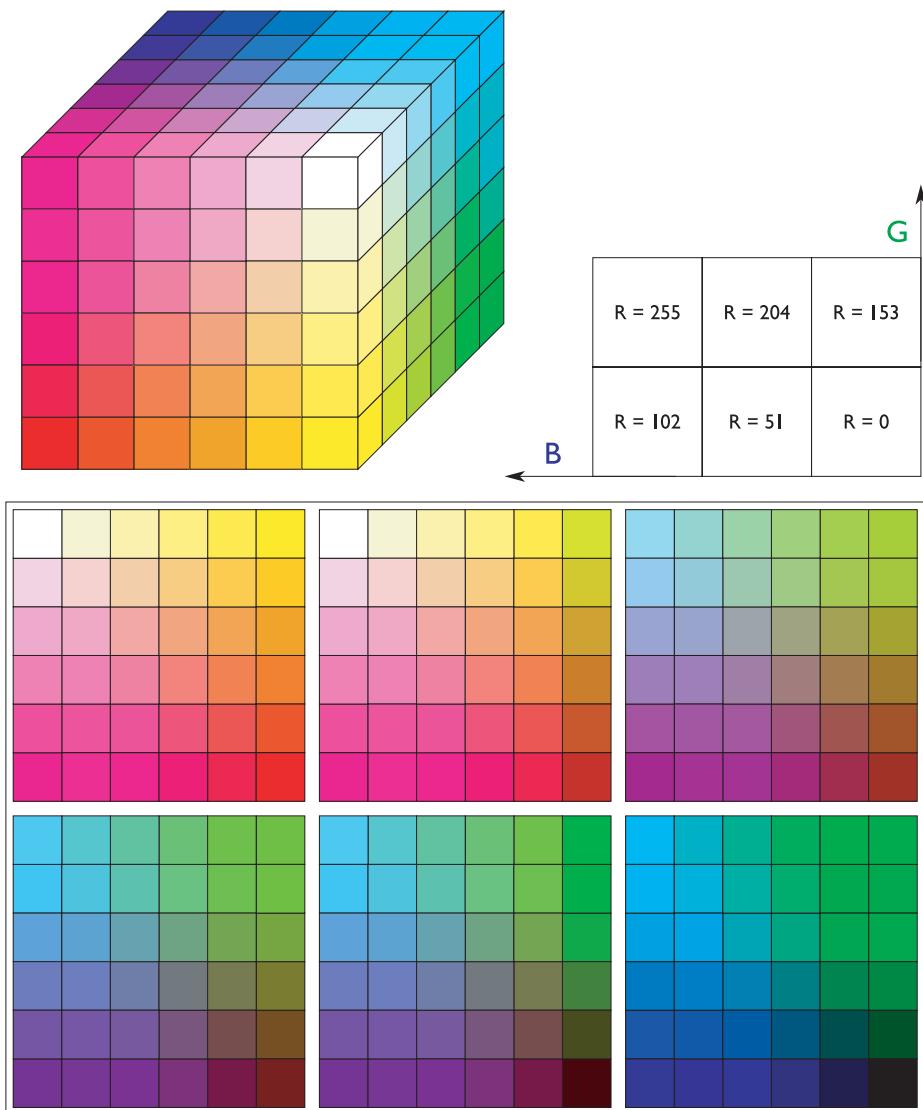


Рис. 1.4. Куб (а) и палитра (б) фиксированных RGB цветов

Таблица 1.2. Значения RGB компонент в палитре фиксированных частот

Система счисления	Значения RGB компонент					
	0	33	66	99	CC	FF
Шестнадцатеричная	0	33	66	99	CC	FF
Десятичная	0	51	102	153	204	255

литру, определяются числами (00 00 00), (33 33 33), (66 66 66), (99 99 99), (CC CC CC),

(FF FF FF). Наиболее яркие красный, зеленый и синий цвета задаются, соответственно, значениями (FF 00 00), (00 FF 00) и (00 00 FF).

Основной закон смешения цветов устанавливает трехмерность цвета и возможность выражения практически всех цветов через смеси только трех линейно независимых цветов. При этом яркость смеси цветов равна сумме яркостей составляющих смеси [1.21–1.23].

Трехмерное цветовое пространство является удобной и наглядной формой геометрического представления множества цветов и действий над ними. В цветовом пространстве можно ввести сколько угодно координатных систем. Началом координат во всех этих системах обычно устанавливается точка нулевого цвета — черного.

1.2. Цветовые системы

Цветовая система полностью определяется заданием трех координатных осей, исходящих из начала координат, и соответствующим их масштабированием. В общем случае устанавливаемые по осям масштабы могут быть различными. Основные цвета, положенные в основу цветовой координатной системы, используются для выражения остальных цветов в данной координатной системе. Важным условием выбора основных цветов является их линейная независимость, т. е. ни один из трех основных цветов не должен получаться путем смешения двух других цветов.

В системе координат с заданными основными цветами X, Y, Z любой цвет F определяется соотношением

$$F = \alpha_x X + \alpha_y Y + \alpha_z Z. \quad (1.2)$$

Яркость F_B также является одной из возможных цветовых координат:

$$F_B = \alpha_{Bx} X + \alpha_{By} Y + \alpha_{Bz} Z. \quad (1.3)$$

Коэффициенты $\alpha_{Bx}, \alpha_{By}, \alpha_{Bz}$ характеризуют вклады каждой из цветовых координат в яркость воспроизведенного цвета и являются яркостными коэффициентами в данной цветовой системе координат. Термин «яркость» обозначает количественную меру интенсивности ахроматического (неокрашенного) света, которая принимает значения в диапазоне от черного до белого с промежуточными серыми оттенками.

Хроматический (окрашенный) свет характеризуется следующими параметрами [1.21, 1.22]:

- электромагнитным спектром, охватывающим диапазон длин волн примерно от 380 до 780 нм (рис. 1.5);
- потоком лучистой энергии, излучаемой в единицу времени и измеряемой в ваттах;
- световым потоком, оцениваемым по зрительному ощущению и измеряемым в люменах;
- светлотой, цветовым тоном и насыщенностью, которые отражают уровень производимого световым потоком зрительного ощущения и определяют цветовое восприятие.

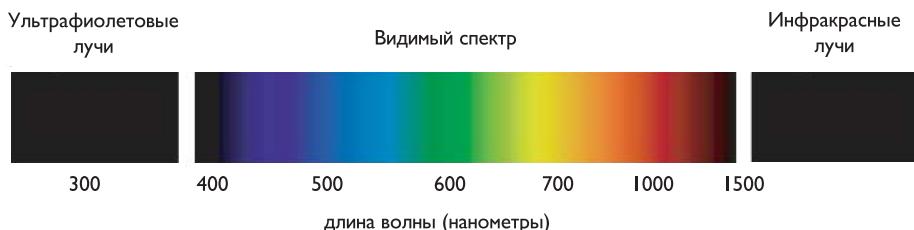


Рис. 1.5. Длины волн видимой части электромагнитного спектра

Цветовая система *RGB*

Субъективные измерения цветовых координат смешения трех чистых спектральных цветов (при одинаковой мощности монохроматических излучений с равной энергией) привели к разработке принятого Международной комиссией по освещению (МКО) стандартного набора монохроматических первичных основных цветов: красного (R) с длиной волны 700 нм, зеленого (G) — 546,1 нм и синего (B) — 435,8 нм.

В соответствии с теорией цветового зрения предполагается наличие в сетчатке глаза светочувствительных приемников (колбочек) трех видов: 65% из них воспринимают красный свет, 33% — зеленый свет и 2% — синий свет.

Совокупность реакций каждого из трех приемников на падающее излучение определяет восприятие цвета. Если обозначить спектральные чувствительности трех приемников $\bar{R}(\lambda)$, $\bar{G}(\lambda)$ и $\bar{B}(\lambda)$, то три составляющие светового потока R , G и B определяются соотношениями:

$$R = \int_{\lambda_{\min}}^{\lambda_{\max}} \varphi(\lambda) \bar{R}(\lambda) d\lambda, \quad G = \int_{\lambda_{\min}}^{\lambda_{\max}} \varphi(\lambda) \bar{G}(\lambda) d\lambda, \quad B = \int_{\lambda_{\min}}^{\lambda_{\max}} \varphi(\lambda) \bar{B}(\lambda) d\lambda, \quad (1.4)$$

где $\varphi(\lambda)$ — спектральная интенсивность лучистого потока на длине волны λ .

На рис. 1.6 представлены относительные кривые спектральной чувствительности трех светочувствительных приемников глаза [1.20]. Эти зависимости восприятия цвета показывают, что никакая монохроматическая составляющая не может характеризовать красный, зеленый или синий цвет. Следовательно, определенный набор монохроматических первичных составляющих не обеспечивает возможность получения всех цветов спектра. Разным цветовым ощущениям в одинаковых условиях наблюдения соответствуют разные спектральные распределения излучения. Однако обратное утверждение неверно, т. е. ощущение одного и того же цвета может возникать при различных спектральных распределениях излучения.

Трехцветное выражение цвета опирается на установленный опытным путем факт, что для большинства встречающихся в природе цветов можно выполнить зрительное уравнивание со смесями взятых в различных пропорциях только трех **основных** цветов. Эта связь взаимно однозначна независимо от спектрального распределения излучения данного цвета [1.23–1.29].

В основе RGB-модели лежит Декартова система координат, цветовое пространство в соответствии с соотношениями (1.1) и рис. 1.3 представляет собой

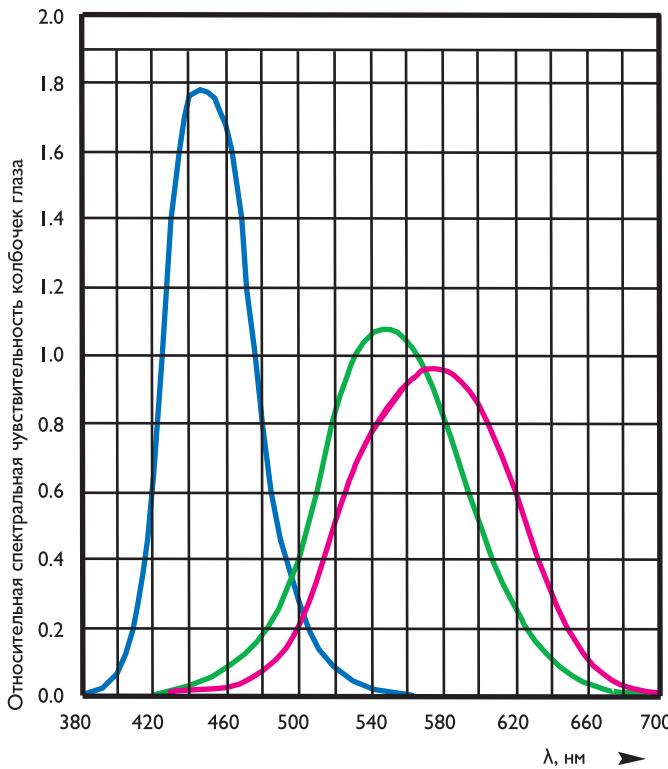


Рис. 1.6. Функции относительной спектральной чувствительности трех видов светочувствительных приемников глаза

куб. Другой способ задания цвета основан на применении двумерной диаграммы цветностей МКО с x (красной) и y (зеленой) координатами (рис. 1.7). Третья z (синяя) координата вычисляется однозначно: $z = 1 - x - y$.

На диаграмме цветностей МКО показано положение цветностей чистых спектральных цветов, образующих подковообразную кривую чистых спектральных цветностей. Линия, соединяющая точки чистых спектральных цветностей с длинами волн 380 и 780 нм, называется линией **пурпурных цветов**. На этой же диаграмме приведены точки стандартизованных величин источников света, которые воспроизводят условия освещения: А — искусственного — электрическими лампами накаливания; В — прямого солнечного света; С — рассеянного дневного; D_{65} — усредненного дневного. Точка Е — точка равной энергии при $x = y = z = 1/3$, она соответствует **цветности естественного дневного цвета**.

Источники света характеризуют также цветовой температурой, соответствующей абсолютной температуре идеального излучателя (абсолютно черного тела), цвет которого имеет ту же цветность, что и данное излучение (табл. 1.3) [1.29].

Выбранные на диаграмме МКО точки основных цветов определяют цветовую модель RGB. При этом сочетания двух из основных цветов: зеленый + синий, синий + красный, красный + зеленый — воспроизводятся в виде вторичных цветов — голубого, пурпурного и желтого цветов, как показано на рис. 1.8.

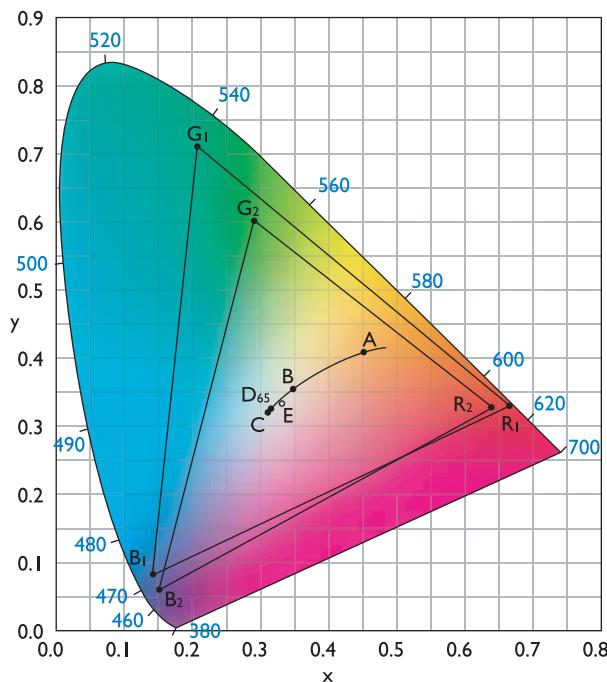


Рис. 1.7. Диаграмма цветностей МКО

Таблица 1.3. Координаты цветности и цветовая температура стандартных источников света

Источник излучения	Координаты цветности		Цветовая температура, К
	x	Y	
A	0,4512	0,4059	2856
B	0,3498	0,3527	4874
C	0,3104	0,3191	6774
D ₆₅	0,3138	0,3310	6504
E	1/3	1/3	5400

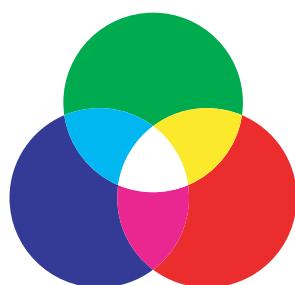


Рис. 1.8. Первичные и вторичные основные цвета световых источников

Таким образом, использование трех основных цветов световых источников — RGB, в которых каждый элемент (пиксель) имеет глубину представления 24 бита (по 8 битов на красную, зеленую и синюю составляющие), позволяет создавать на экране цветного монитора полноцветное (True Color) изображение, как показано на рис. 1.9.

Цветовые системы CMY и CMYK

Система CMY используется в цветных принтерах и копировальных устройствах, в которых цветные красители наносятся на бумагу.



Рис. 1.9. Формирование полноцветного изображения

Если в системе RGB голубой (Cyan), пурпурный (Magenta) и желтый (Yellow) являются вторичными цветами, то в системе CMY они образуют первичные цвета красителей, реализуя субстрективный цветовой синтез.

Очевидно, если поверхность, покрытая голубой краской, освещается белым светом, то красный цвет от нее не отражается. От поверхностей, покрытых пурпурной или желтой краской, не отражаются, соответственно, зеленый и синий цвета.

Преобразование данных из RGB в CMY осуществляется простой операцией:

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix}, \quad (1.5)$$

где значения цветов нормированы и находятся в диапазоне $[0, 1]$.

В связи с тем, что смешивание в равном количестве голубого, пурпурного и желтого цветов приводит при печати к появлению не черного, а грязно-серого цвета, система CMY расширяется до системы CMYK, содержащей дополнительно ключевой (Key) — черный цвет (рис. 1.10).

Цветовая система HSI

Цветовые системы RGB и CMYK не приспособлены для словесного описания цветов. Цветной объект обычно описывается такими понятиями, как цветовой тон, его насыщенность и светлота, соответствующая понятию интенсивности (полутоновой яркости).

В цветовой системе HSI (Hue — цветовой тон, Saturation — насыщенность, Intensity — интенсивность) используется естественное описание цвета. Специфи-

кой системы HSI является тот факт, что яркостная информация (интенсивность) отделена от цветовой информации, характеризуемой цветовым тоном и насыщенностью.

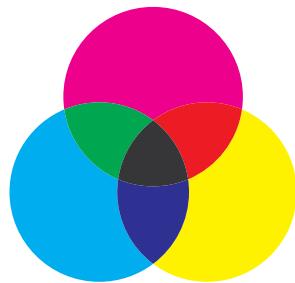


Рис. 1.10. Первичные и вторичные основные цвета красителей

Существуют различные формы геометрической интерпретации системы HSI. Одна из них представляет все цветовое пространство в виде фигуры из двух одинаковых и соединенных своими основаниями конусов, как показано на рис. 1.11.

Расстояние между вершинами этих конусов нормировано и равно 1, так что высота каждого конуса равна 0,5. Вершина нижнего конуса соответствует уровню черного (0,0), а верхнего — уровню белого (1,0). Таким образом, интенсивность (яркость) объекта определяется расстоянием от точки уровня черного до горизонтального сечения, характеризующего цветовую информацию. На рис. 1.11 приведено два сечения, соответствующие $I = 0,5$ (в соединении оснований двух конусов) и $I = 0,75$.

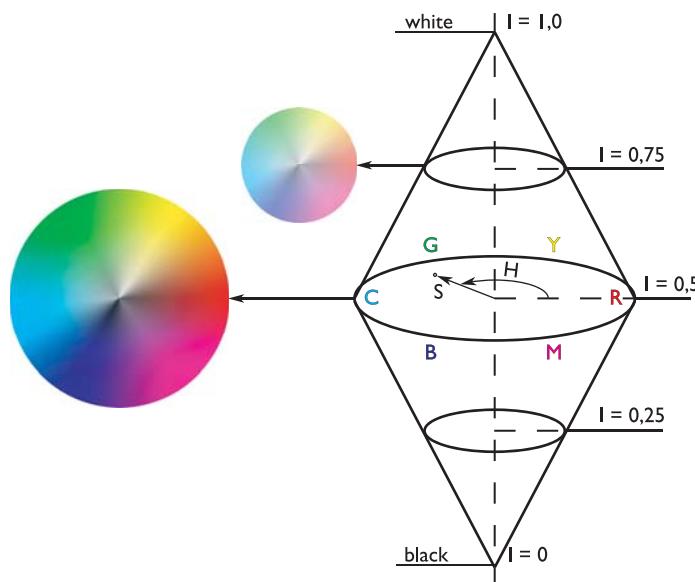


Рис. 1.11. Геометрическая интерпретация цветовой системы HSI

Цветовые параметры объекта характеризуются параметрами вектора, соединяющего точку с центром круга, образованного соединением оснований двух конусов, т. е. сечением фигуры на уровне интенсивности $I = 0,5$. Радиус этого круга равен 1. При этом цветовой тон определяется углом H° наклона вектора относительно ординаты, проведенной в направлении красного цвета. В представленной на рис. 1.11 плоскости сечения фигуры первичные основные (красный, зеленый и синий) цвета расположены относительно друг друга под углом 120° .

Вторичные (голубой, пурпурный и желтый) цвета смешены относительно друг друга также на углы 120° , а относительно первичных цветов — на углы 60° . Насыщенность цвета S имеет значения в диапазоне $[0, 1]$ и определяется длиной вектора, соединяющего центр круга с точкой, определяющей цветовые параметры воспроизведимой детали.

Преобразование цветов из системы RGB в систему HSI

Если координаты RGB нормированы таким образом, что их значения ограничены диапазоном $[0, 1]$, и угол H° отсчитывается от красной оси пространства HSI , то интенсивность (яркость), насыщенность и цветовой тон определяются соотношениями:

$$I = \frac{1}{3}(R + G + B), \quad (1.6)$$

$$S = \frac{\min(R, G, B)}{I}, \quad (1.7)$$

$$H^\circ = \begin{cases} \theta^\circ, & \text{при } B \leq G, \\ 360^\circ - \theta^\circ, & \text{при } B > G, \end{cases} \quad (1.8)$$

где $\theta^\circ = \left(\frac{180}{\pi}\right) \cdot \arccos \left\{ \frac{R - \frac{1}{2}(G+B)}{[(R-G)^2 + (R-B)(G-B)]^{1/2}} \right\}$.

Преобразование цветов из системы HSI в систему RGB

Формулы преобразования различны для трех секторов параметра H° .

RG сектор ($0^\circ \leq H^\circ < 120^\circ$):

$$\begin{cases} R = I \left[1 + \frac{S \cos H^\circ}{\cos(60^\circ - H^\circ)} \right], \\ B = I(1 - S), \\ G = 3I - (R + B). \end{cases} \quad (1.9)$$

GB сектор ($120^\circ \leq H^\circ < 240^\circ$):

$$\begin{cases} G = I \left[1 + \frac{S \cos H^\circ}{\cos(180^\circ - H^\circ)} \right], \\ R = I(1 - S), \\ B = 3I - (R + G). \end{cases} \quad (1.10)$$

BR сектор ($240^\circ \leq H^\circ < 360^\circ$):

$$\begin{cases} B = I \left[1 + \frac{S \cos H^\circ}{\cos(300^\circ - H^\circ)} \right], \\ G = I(1 - S), \\ R = 3I - (B + G). \end{cases} \quad (1.11)$$

Цветовые системы YUV , YIQ , $YD_R D_B$, $YC_R C_B$

Эти цветовые системы чаще всего используются в аналоговом и цифровом телевидении. Составляющая Y характеризует яркость элементов телевизионного изображения, ее величина определяется линейной комбинацией основных цветов:

$$Y = \alpha_R R + \alpha_G G + \alpha_B B, \quad (1.12)$$

где коэффициенты $\alpha_R, \alpha_G, \alpha_B$ определяют параметры исходного белого цвета, и при изменяющихся основных RGB цветах в диапазоне $[0, 1]$ $\alpha_R + \alpha_G + \alpha_B = 1$.

Остальные параметры этих систем (U, V) , (I, Q) , (D_R, D_B) , (C_R, C_B) — линейно зависят от двух так называемых цветоразностных составляющих R-Y и B-Y или являются их линейной комбинацией. Более подробное описание этих систем приводится в главе 2.

Цветовая система CIELAB

Унифицированное пространство восприятия цвета CIELAB является стандартом, принятым МКО в 1976 году [1.30]. Евклидово расстояние между двумя любыми одинаково визуально различимыми цветовыми точками этого пространства имеет практически равную величину.

В цветовом пространстве CIELAB (точнее, CIE 1976 $L^*a^*b^*$) значение **светлоты** отделено от значения хроматической составляющей цвета (**тон, насыщенность**). Светлота задана координатой L (изменяется от 0 до 100, т. е. от самого темного до самого светлого), хроматическая составляющая — двумя полярными координатами a и b . Первая обозначает положение цвета в диапазоне от зеленого до пурпурного, вторая — от синего до желтого. Это позволяет эффективно применять данную систему при цветовом анализе.

На рис. 1.12 приведены изображения системы $L^*a^*b^*$ для двух значений светлоты L .

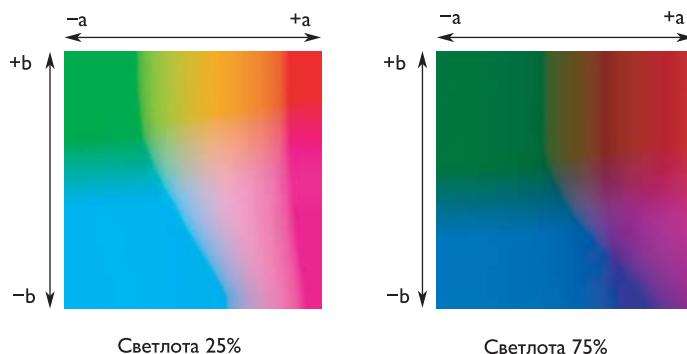


Рис. 1.12. Изображения изменений полярных координат a и b системы CIELAB при двух значениях светлоты L

В отличие от цветовых пространств RGB или CMYK, которые являются, по сути, набором аппаратных данных для воспроизведения цвета на экране монитора или на бумаге (цвет может зависеть от типа печатной машины, марки красок, влажности воздуха в цехе или производителя монитора и его настроек), CIELAB однозначно определяет цвет. Поэтому CIELAB нашел широкое применение в программном обеспечении для обработки изображений в качестве промежуточного цветового пространства, через которое происходит конвертирование данных между другими цветовыми пространствами. При этом особые свойства $L^*a^*b^*$ сделали редактирование в этом пространстве мощным инструментом **цветокоррекции**.

Благодаря характеру определения цвета в L^*a^*b появляется возможность отдельно воздействовать на яркость, контраст изображения и на его цвет. L^*a^*b предоставляет возможность избирательного воздействия на отдельные цвета в изображении, усиления цветового контраста, незаменимыми являются и возможности, которые это цветовое пространство предоставляет для борьбы с *шумом* на изображении.

Учитывая соотношения

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0,412453 & 0,357580 & 0,180423 \\ 0,212671 & 0,715150 & 0,072169 \\ 0,019334 & 0,119193 & 0,950227 \end{bmatrix} \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix}, \quad (1.13)$$

перевод RGB данных в компоненты $L^*a^*b^*$ осуществляется следующим образом:

$$\begin{cases} L* = 116 \cdot f\left(\frac{Y}{Y_n}\right) - 16, \\ a* = 500 \cdot \left[f\left(\frac{X}{X_n}\right) - f\left(\frac{Y}{Y_n}\right) \right], \\ b* = 200 \cdot \left[f\left(\frac{Y}{Y_n}\right) - f\left(\frac{Z}{Z_n}\right) \right], \end{cases} \quad (1.14)$$

где

$$f(q) = \begin{cases} q^{1/3}, & \text{при } q < 0,008856; \\ 7,787 \cdot q + 16/116, & \text{иначе;} \end{cases} \quad q \in \left\{ \frac{X}{X_n}, \frac{Y}{Y_n}, \frac{Z}{Z_n} \right\}.$$

Величины X_n , Y_n , Z_n определяют источник белого цвета D_{65} стандарта МКО и могут быть получены при $R = G = B = 100$.

1.3. Статические и динамические эталонные изображения

Цифровая обработка изображений применяется для решения различных задач:

- пространственного и частотного улучшения изображений;
- восстановления изображений, пораженных различными видами помех;
- сокращения избыточности и эффективного кодирования изображений для хранения и передачи по каналам связи;
- распознавания и обнаружения объектов в изображениях и др.

При создании систем цифровой обработки изображений используются различные статические и динамические эталонные изображения, обладающие определенными свойствами [1.31–1.35].

Статические эталонные изображения применяются, в частности, для анализа эффективности методов сжатия изображений, а также для анализа внутрикадрового кодирования динамических изображений. Динамические эталонные изображения используются, например, для анализа эффективности поиска векторов и компенсации движения деталей в изображениях, и др.

Статические эталонные изображения

На рис. 1.13а приведено широко используемое классическое цветное изображение «Лена» («Lenna»), содержащее множество плавных цветовых переходов и наклонных границ, а на рис. 1.13б — черно-белое изображение «Элина» («Elaine»).



Рис. 1.13. Изображения «Лена» (а) и «Элина» (б)



Рис. 1.14. Цветные изображения «Девушка» (а) и «Перцы» (б)

Аналогичными свойствами обладают эталонные изображения «Девушка» («Girl») и «Перцы» («Peppers»), которые даны на рис. 1.14.

На рис. 1.15 представлены три черно-белых изображения. Первое из них — «Барбара» («Barbara») — содержит множество различно ориентированных полосок на скатерти, платке и брюках; при неэффективной обработке этого изображения часто возникает муар и другие артефакты. На втором изображении — «Золотой Холм» («Goldhill») — на переднем и заднем планах содержится множество мелких деталей, третье изображение — «Кинооператор» («Cameraman») — содержит ряд протяженных разнонаправленных деталей.

Рис. 1.16 иллюстрирует два классических тестовых цветных изображения, содержащие вертикально и горизонтально ориентированные элементы, — «Гавань» («Harbour») и «Корабль» («Boat»).



Рис. 1.15. Черно-белые эталонные изображения «Барбара» (а), «Золотой Холм» (б), «Кинооператор» (в)



Рис. 1.16. Цветные изображения «Гавань» (а), «Корабль» (б)

Динамические эталонные изображения

При анализе качества преобразований динамических изображений используются различные эталоны в зависимости от вида приложений.

Так, при разработке систем видеотелефонии и видеонаблюдения применяют малоподвижные изображения.

Такие последовательности можно увидеть при воспроизведении AVI файлов, содержащих цветные динамические изображения «Клер» («Claire», 335 кадров), черно-белые изображения «Сьюзи» («Susie», 149 кадров), цветные изображения «Монитор холла» («Hall_monitor», 599 кадров).

При создании систем видеоконференцсвязи используют изображения с более подвижными деталями. В качестве примеров приведены AVI файлы трех последовательностей изображений: «Теннис» («Tennis», 223 кадра), «Прораб» («Foreman», 599 кадров) и «В автомобиле» («Carphone», 191 кадр).

Для оценки качества кодирования телевизионных изображений чаще всего используются динамические изображения «Сад цветов» («FlowerGarden», 229 кадров), «Береговая охрана» («Coastguard», 300 кадров) и «Футбол» («Football», 249 кадров).



Рис. 1.17. Кадры из видеопоследовательности «Клер»



Рис. 1.18. Кадры из видеопоследовательности «Сьюзи»



Рис. 1.19. Кадры из видеопоследовательности «Монитор холла»



Рис. 1.20. Кадры из видеопоследовательности «Теннис»



Рис. 1.21. Кадры из видеопоследовательности «Прораб»



Рис. 1.22. Кадры из видеопоследовательности «В автомобиле»



Рис. 1.23. Кадры из видеопоследовательности «Сад цветов»



Рис. 1.24. Кадры из видеопоследовательности «Береговая охрана»



Рис. 1.25. Кадры из видеопоследовательности «Футбол»

ГЛАВА 2

СИСТЕМЫ ФОРМИРОВАНИЯ И ПЕРЕДАЧИ ВИДЕОИНФОРМАЦИИ

2.1. Монохромное телевидение

Полный сигнал монохромного телевидения состоит из **сигнала яркости**, несущего информацию о яркости передаваемого изображения, и **сигнала синхронизации**, который представляет собой совокупность синхронизирующих импульсов строк и полей телевизионного кадра [1.8, 1.9, 1.35].

В составе ТВ-сигнала чередуются так называемые активные интервалы, в которых передаются собственно сигналы изображения, и пассивные интервалы, предназначенные для передачи управляющих — гасящих и синхронизирующих — импульсов.

На рис. 2.1 в качестве примера приведено стилизованное изображение так называемого линейного градационного клина (10 градаций яркости) и соответствующая ему форма полного ТВ-сигнала в интервале одной строки.

При этом сигнал изображения имеет положительную полярность, изменяясь от уровня черного (минимальный уровень) до уровня белого (максимальный уровень), а синхронизирующие и гасящие импульсы характеризуются сигналами отрицательной полярности.

За опорный уровень принят **уровень гашения**, соответствующий передаче уровня гасящих импульсов. Номинальные **уровень белого** и **уровень черного** характеризуют максимально допустимые изменения сигнала яркости.

При цифровой обработке сигнала яркости обычно предусматривается использование 8-разрядного цифрового кода, обеспечивающего точное воспроизведение 256 градаций яркости [1.36–1.39]. Как известно, максимальный контраст телевизионных изображений не превышает величины 200. Длительность активных интервалов, форма гасящих и синхронизирующих импульсов, частоты строк, полей и кадров, верхняя граничная частота сигнала яркости и др. нормируются ТВ-стандартами [1.8, 1.9, 1.35, 1.40].

В табл. 2.1 приведена классификация используемых в различных странах ТВ-стандартов, объединенных по принципу равенства верхних граничных частот спектра сигнала яркости ($f_{\text{гр}}$).

В таблице также приведена необходимая скорость передачи информации (или объем информации, передаваемой за 1 сек) при дискретизации полного ТВ-сигнала с частотой Котельникова–Найквиста и при использовании 8-разрядного кодирования ($K = 8$).

Очевидно, что объем передаваемой информации (I) прямо пропорционален граничной частоте спектра сигнала яркости [1.41]:

$$I = K \times 2 \times f_{\text{гр}}.$$

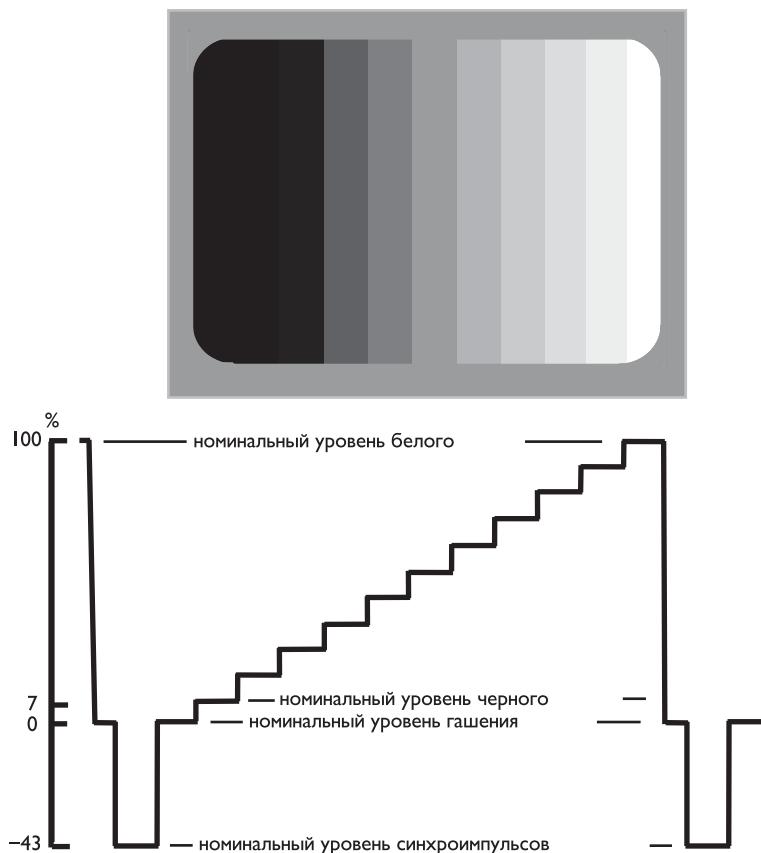


Рис. 2.1. Изображение градационного клина и форма сигнала в строчном интервале

Таблица 2.1. Стандарты аналогового ТВ-вещания

Стандарты	Число строк/полей в секунду	Границчная частота спектра сигнала яркости, МГц	Объем информации, Мбит/с
M N	525/60 625/50	4,2	67,2
B/G/H	625/50	5,0	80,0
I	625/50	5,5	88,0
D/K/K1/L	625/50	6,0	96,0

Для телевизионного стандарта, используемого в России, $f_{\text{тр}} = 6$ МГц и $I = 8 \times 2 \times 6 = 96$ Мбит/сек. При цифровой обработке сигналов весьма важными являются сведения об объеме передаваемой информации или требуемой скорости передачи цифрового потока, получаемого первоначально при переводе сигнала в цифровой вид (еще до цифровой обработки), т. к. эти сведения служат отправной точкой для всех дальнейших рассуждений.

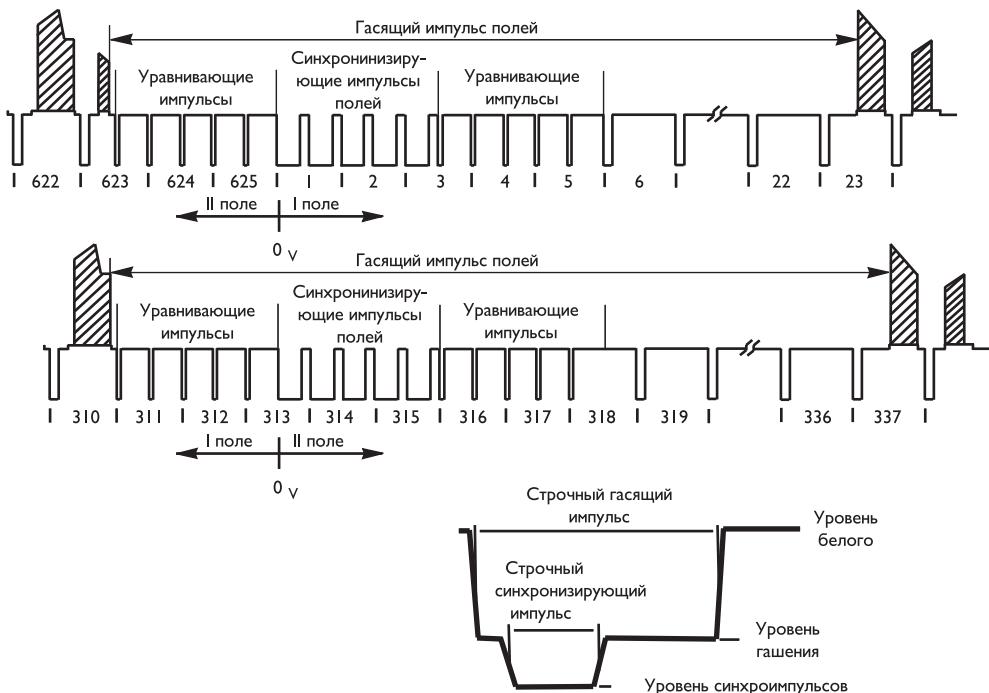


Рис. 2.2. Форма гасящих и синхронизирующих импульсов строк и полей

Стандарты *M* и *N*, нормирующие граничную частоту спектра сигнала яркости, равной $f_{\text{гр}} = 4,2 \text{ МГц}$, используются в странах Северной и Южной Америки, Японии, Южной Корее и ряде стран Юго-Восточной Азии.

Граничная частота 5 МГц (стандарты *B/G/H*) используется в системах телевидения, применяемых в большинстве стран Западной и Северной Европы, Северной и Восточной Африки, Юго-Западной и Южной Азии, в Австралии.

Стандарт *I*, используемый в Великобритании и странах Южной Африки, нормирует граничную частоту сигнала яркости в 5,5 МГц.

И, наконец, Франция, страны Восточной Европы, Россия, страны СНГ, Китай и ряд стран Центральной Африки используют стандарты *D/K/K1/L*, нормирующие частоту сигнала яркости в 6 МГц.

Учитывая, что сигнал яркости передается лишь в активных интервалах полного ТВ-сигнала, на самом деле объем информации может быть несколько уменьшен. Его величина может быть рассчитана следующим образом:

$$I_0 = I \times \frac{V_{\text{акт}}}{V} \times \frac{H_{\text{акт}}}{H} = I \times \frac{(V - V_{\text{пас}})}{V} \times \frac{(H - H_{\text{пас}})}{H},$$

где V , H — длительности кадра и строки, $V_{\text{акт}}$ и $H_{\text{акт}}$ — длительности активных частей кадра и строки, $V_{\text{пас}}$ и $H_{\text{пас}}$ — длительности пассивных частей кадра и строки. В таком случае, учитывая, что $V = 625H$, $V_{\text{пас}} = 50H$, можно рассчитать:

$$I = 96 \times (575/625) \times (64 - 12)/64 = 72 \text{ Мбит/с.}$$

На рис. 2.2 приведена форма гасящего импульса и последовательности синхронизирующих импульсов полей, а также форма пассивного строчного интервала для используемых в России стандартов D, K.

2.2. Цветное телевидение

Стандартные системы цветного телевидения NTSC, PAL и SECAM [1.8, 1.42–1.44] различаются в основном принципами формирования сигналов цветности, передаваемых на поднесущих путем уплотнения спектра полного телевизионного сигнала монохромного телевидения, так что на участке спектра, занимаемом сигналами цветности, располагаются также и спектральные составляющие сигнала яркости.

В табл. 2.2 приведен перечень государств, использующих различные системы цветного телевидения.

Таблица 2.2. Использование различных систем цветного ТВ

B,G/PAL	Австралия	Замбия	Либерия	ОАЭ	Танзания
	Австрия	Зимбабве	Люксембург	Оман	Тунис
	Албания	Израиль	Македония	Пакистан	Турция
	Алжир	Индия	Малави	Папуа — Новая	Уганда
	Бангладеш	Индонезия	Малайзия	Гвинея	Финляндия
	Бахрейн	Иордания	Мальдивы	Португалия	Хорватия
	Бельгия	Италия	Мальта	Сербия	Черногория
	Бруней	Исландия	Мозамбик	Сингапур	Швейцария
	Гана	Испания	Монако	Сирия	Швеция
	Германия	Йемен	Нигерия	Словения	Шри-Ланка
	Гибралтар	Камерун	Нидерланды	Судан	Экваториальная- ная Гвинея
	Дания	Катар	Новая	Сьерра-Леоне	
	Босния и Герцеговина	Кения	Зеландия	Таиланд	Эфиопия
		Кувейт	Норвегия		
I/PAL	Ангола	Великобритания	Гонконг	Лесото	Южная Африка
	Ботсвана		Ирландия		
N/PAL	Аргентина	Парагвай	Уругвай		
D/PAL	Китай				
M/PAL	Бразилия				
D,K/PAL	КНДР	Румыния			
B,G/SECAM	Греция	Иран	Ливия	Мали	Саудовская
	Египет	Кипр	Маврикий	Марокко	Аравия
	Ирак	Ливан	Мавритания		Тунис
D,K/SECAM	Афганистан	Венгрия	Монголия	СНГ и страны	Словакия
	Болгария	Вьетнам	Польша	Балтии	Чехия
K1/SECAM	Бенин	Габон	Заир	Мадагаскар	Того
	Буркина-Фасо	Гвинея	Конго	Нигерия	ЦАР
	Бурунди	Джибути	Кот- д'Ивуар	Сенегал	Чад
L/SECAM	Люксембург	Монако	Франция		
M/NTSC	Антильские острова	Венесуэла	Колумбия	США	Япония
	Бермуды	Виргинские ост- рова	Коста-Рика	Суринам	
	Бирма	Гаити	Куба	Филиппины	
	Боливия	Гватемала	Мексика	Чили	
	Доминиканская респ.	Гондурас	Никарагуа	Эквадор	
		Канада	Панама	Ю. Корея	
			Перу	Ямайка	

Система NTSC используется в Северной и Центральной Америке, в ряде стран восточной части Южной Америки, а также в Японии, Южной Корее и в ряде стран Юго-Восточной Азии.

Система SECAM применяется в России, в странах Восточной Европы и СНГ, во Франции, в большинстве стран Северной Африки и Юго-Западной Азии.

Система PAL применяется в большинстве стран Западной и Северной Европы, в Индии, Китае, Австралии, Бразилии, Аргентине и ряде других стран.

Телевизионные системы, как следует из правил колориметрии, должны воспроизводить все цветности, лежащие в пределах треугольника основных цветов на диаграмме международной комиссии по освещению МКО [1.24–1.27, 1.45].

На графике цветностей МКО (в обычной системе координат xy) основные цвета $R1$, $G1$ и $B1$, а также белый цвет C , или исходная цветность системы приемника стандарта NTSC, имеют следующие координаты [1.40]:

$$\begin{array}{ll} X_{R1} = 0,67; & Y_{R1} = 0,33; \\ X_{G1} = 0,21; & Y_{G1} = 0,71; \\ X_{B1} = 0,14; & Y_{B1} = 0,08; \\ X_C = 0,310; & Y_C = 0,316. \end{array}$$

Различным сочетанием (линейной комбинацией) этих трех основных цветов можно получить любую цветность в пределах треугольника основных цветов на диаграмме МКО.

При стандартизации систем PAL и SECAM были принятые измененные основные цвета ($R2$, $G2$, $B2$) и исходная цветность D_{65} , которые соответствовали созданным ко времени стандартизации люминофорам с повышенной яркостью, но с уменьшенной насыщенностью зеленого и красного цветов. Основные цвета и исходная цветность системы приемника имеют следующие координаты [1.40]:

$$\begin{array}{ll} X_{R2} = 0,64; & Y_{R2} = 0,33; \\ X_{G2} = 0,29; & Y_{G2} = 0,6; \\ X_{B2} = 0,15; & Y_{B2} = 0,06; \\ X_D = 0,313; & Y_D = 0,329. \end{array}$$

Цветовые треугольники изображения на фоне цветового локуса диаграммы цветностей МКО даны на рис. 1.7.

При формировании полного цветового телевизионного сигнала во всех системах цветного телевидения используются сигнал яркости (E'_Y) и два цветоразностных сигнала (E'_{R-Y} и E'_{B-Y}), формируемые из гамма-скорректированных сигналов основных цветов (E'_R , E'_G и E'_B) [1.8, 1.9]:

$$\begin{aligned} E'_Y &= 0,299E'_R + 0,587E'_G + 0,114E'_B; \\ E'_{R-Y} &= E'_R - E'_Y; \\ E'_{B-Y} &= E'_B - E'_Y. \end{aligned}$$

Для оценки качества передачи и воспроизведения полного цветового ТВ-сигнала обычно в качестве тестовых используют сигналы цветных полос, содержащих три полосы основных цветов (красную, зеленую и синюю), три полосы главных дополнительных цветов (желтую, голубую, пурпурную), а также белую и черную полосы [1.46].

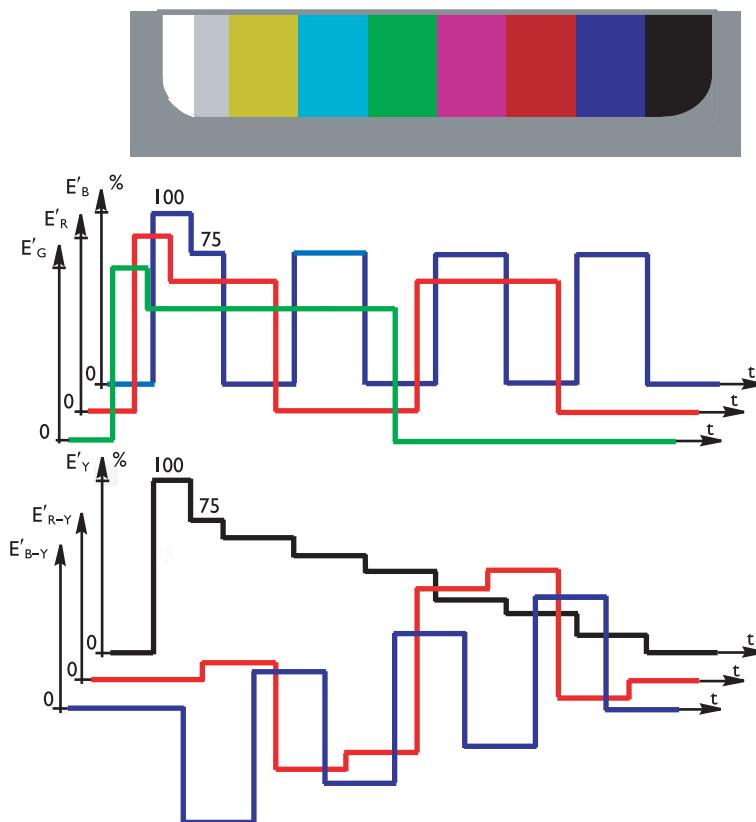


Рис. 2.3. Цветные полосы и диаграммы их видеосигналов

Наиболее часто применяют сигналы вертикальных цветных полос, соответствующих 100% яркости белой полосы, 75% яркости и 100% насыщенности цветных полос. Иногда первая полоса разбивается на два участка: 100% яркости (белая полоса) и 75% яркости (серая полоса).

На рис. 2.3 приведено стилизованное изображение и диаграммы сигналов при передаче указанных вертикальных цветных полос. Верхняя часть диаграммы характеризует формы сигналов основных цветов E'_R , E'_G и E'_B , а нижняя — формы сигнала яркости E'_Y (черная линия) и цветоразностных сигналов E'_{R-Y} и E'_{B-Y} .

2.2.1. Система NTSC

В системе NTSC используются **не непосредственно** цветоразностные сигналы, а их линейная комбинация [1.8, 1.43]:

$$E'_I = -0,27E'_{B-Y} + 0,74E'_{R-Y};$$

$$E'_Q = 0,41E'_{B-Y} + 0,48E'_{R-Y}.$$

Коэффициенты здесь выбраны экспериментально с таким расчетом, чтобы минимизировать ширину спектра одного из сигналов, а именно E'_Q . Полосу, занимае-

мую этим сигналом, удалось таким образом уменьшить примерно до 0,6 МГц. Оба сигнала E'_I и E'_Q передаются путем **квадратурной модуляции** одной поднесущей, частота которой жестко связана с частотой строчной развертки f_H и равна:

$$f_S = \frac{455}{2} \times f_H = 3,579545 \text{ МГц},$$

$$f_H = 15734,264 \text{ Гц}.$$

Система NTSC применяется только в странах (см. табл. 2.2), в которых используется телевизионный стандарт М (границная частота спектра сигнала яркости — 4,2 МГц).

В соответствии с требованиями на систему NTSC ограничение спектра сигнала E'_I должно осуществляться ФНЧ, имеющим затухание менее 3 дБ на частоте 1,3 МГц и более 20 дБ на частоте 3,6 МГц. Фильтр, обеспечивающий ограничение спектра сигнала E'_Q , должен иметь затухание менее 2 дБ на частоте 0,4 МГц, менее 6 дБ на частоте 0,5 МГц и более 6 дБ на частоте 0,6 МГц.

Сигнал цветности NTSC на поднесущей можно представить в виде:

$$E'_S = E'_I \cos(\omega_S t + 33^\circ) + E'_Q \sin(\omega_S t + 33^\circ) = E'_{S0} \cos(\omega_S t + 33^\circ - \varphi),$$

где $E'_{S0} = \sqrt{(E'_I)^2 + (E'_Q)^2}$, $\varphi = \text{Arctg}(E'_Q/E'_I)$, $\omega_S = 2\pi \cdot f_S$, f_S — частота цветовой поднесущей.

Стилизованное изображение и форма сигналов E'_I , E'_Q , E'_S и полного цветового ТВ-сигнала при передаче цветных полос 75% яркости и 100% насыщенности приведены на рис. 2.4.

Для восстановления фазы цветовой поднесущей в приемном устройстве в составе полного цветового ТВ-сигнала NTSC передается вспышка цветовой поднесущей с длительностью, равной не менее 8 периодов частоты f_S . Фаза поднесущей во вспышке повернута на 180° относительно оси В–Y.

Стилизованная векторная диаграмма сигналов цветных полос изображена на рис. 2.5, а параметры этих сигналов даны в табл. 2.3.

На рис. 2.5 вектор серого цвета характеризует вспышку цветовой поднесущей. Ввиду того, что частота цветовой поднесущей в системе NTSC кратна нечетной гармонике половины строчной частоты, полный цикл изменения фазы равен двум кадрам (табл. 2.4).

Теоретически перевод полного цветового ТВ-сигнала NTSC в цифровой вид может быть осуществлен, как было указано выше, при частоте дискретизации, равной частоте Котельникова–Найквиста, т. е. 8,4 МГц.

Однако на практике при такой частоте требуется чрезвычайно точное квантование сигнала по уровню (более 10 разрядов), а также очень точные и трудно реализуемые фильтры на передаче и приеме.

Только таким образом при этом можно избежать искажений цветовой поднесущей, приводящих к возникновению ложных движущихся деталей в составе воспроизводимого на приеме изображения.

Указанные дефекты можно также устранить путем повышения частоты дискретизации и привязки ее к частоте цветовой поднесущей [1.37, 1.47].

При использовании 9-разрядного кодирования, необходимого для обеспечения малых погрешностей восстановления фазы поднесущей, объем информации

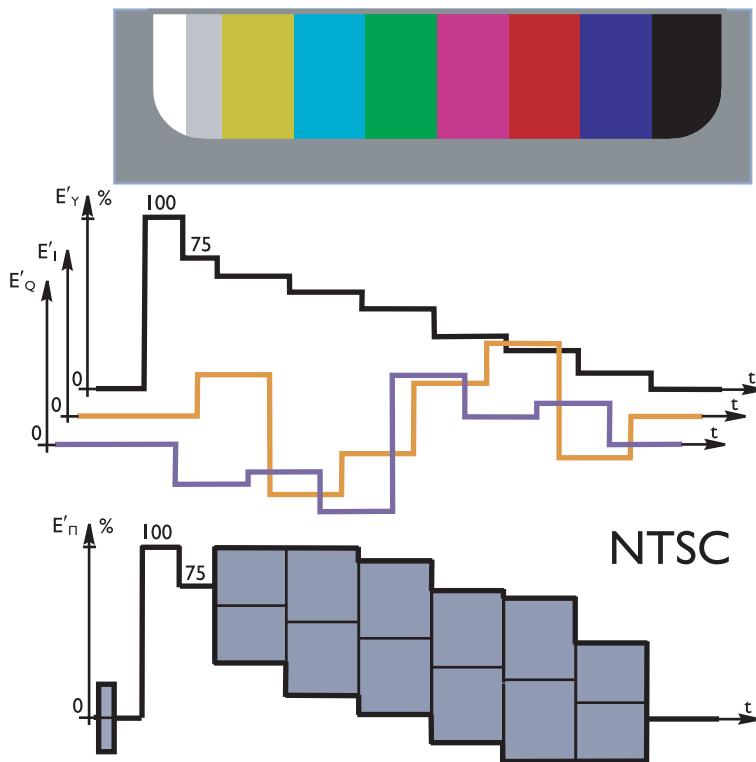


Рис. 2.4. Изображение цветных полос и форма соответствующих сигналов системы NTSC преобразованного сигнала составит:

$$I_0 = 9 \times f_D = 128,7 \text{ Мбит/с.}$$

2.2.2. Система PAL

В системе PAL [1.9, 1.37, 1.43, 1.44], широкое применение которой иллюстрируется табл. 2.2, используются два компрессированных цветоразностных сигнала:

$$E'_U = 0,493E'_{B-Y};$$

$$E'_V = 0,877E'_{R-Y}.$$

Сигнал цветности формируется путем квадратурной модуляции этими сигналами одной цветовой поднесущей, причем модуляция осуществляется таким образом, что полярность одного из цветоразностных сигналов на поднесущих коммутируется от строки к строке:

$$E'_S = E'_U \cos \omega_S t + E'_V \sin \omega_S t = E'_{S0} \cos(\omega_S t - \varphi),$$

$$E'^*_S = E'_U \cos \omega_S t - E'_V \sin \omega_S t = E'_{S0} \cos(\omega_S t + \varphi),$$

где $E'_{S0} = \sqrt{(E'_U)^2 + (E'_V)^2}$, $\varphi = \arctg(E'_V/E'_U)$, $\omega_S = 2\pi \cdot f_S$, f_S — частота цветовой поднесущей.

$$f_S = \left(\frac{1135}{4} + \frac{1}{625} \right) f_H = 4,43361875 \text{ МГц} — \text{для стандартов B, G, H, I};$$

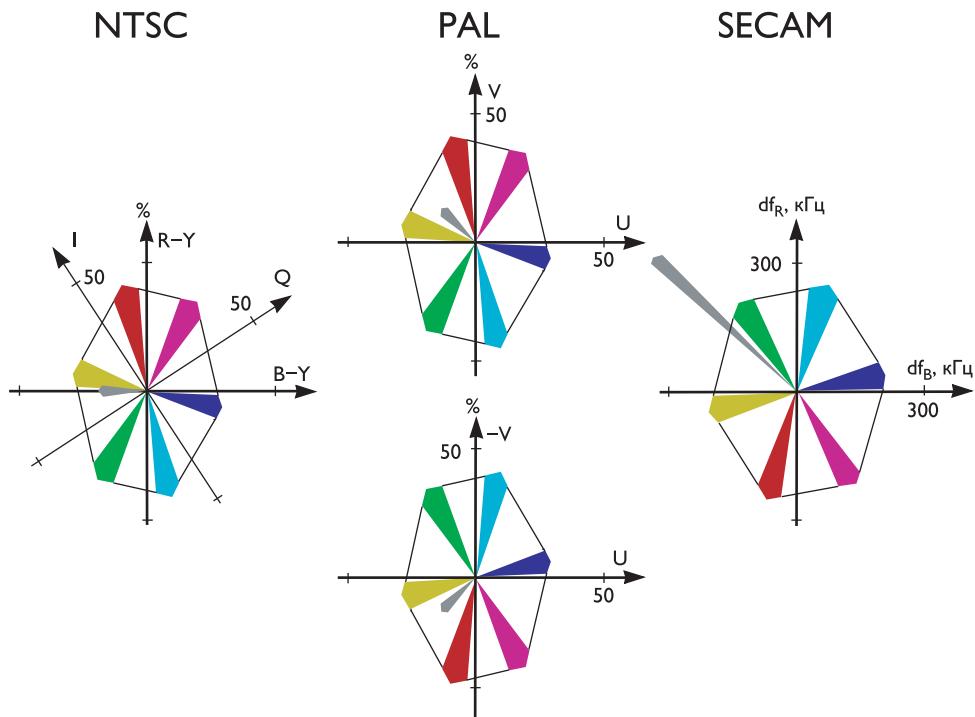


Рис. 2.5. Векторные диаграммы сигналов цветных полос различных систем цветного телевидения

Таблица 2.3. Параметры сигналов цветных полос NTSC

Цвет полосы	Сигналы основных цветов			Сигнал яркости	Сигналы цветности NTSC			
	E'_R , %	E'_G , %	E'_B , %		E'_Y , %	E'_Q , %	E'_I , %	размах, %
Белый	100	100	100	100	0	0	0	—
Серый	75	75	75	75	0	0	0	—
Желтый	75	75	0	66,6	-23,4	+24,2	66,6	167
Голубой	0	75	75	52,4	-16,0	-45,0	95,2	283,5
Зеленый	0	75	0	44,2	-39,2	-20,7	88,4	240,5
Пурпурный	75	0	75	30,7	+39,2	+20,7	88,4	60,5
Красный	75	0	0	22,5	+16,0	+45,0	95,2	103,5
Синий	0	0	75	8,2	+23,4	-24,2	66,6	347
Черный	0	0	0	0	0	0	0	—
Вспышка	—	—	—	—	-16,8	+10,9	40	180

$$f_S = \frac{909}{4} f_H = 3,57561149 \text{ МГц} — \text{для стандарта N.}$$

В соответствии со стандартом на систему PAL фильтры, осуществляющие ограничение спектра сигналов E_U и E_V , должны иметь затухание не более 3 дБ на частоте 1,3 МГц и не менее 20 дБ на частоте 4 МГц.

Таблица 2.4. Изменение фазы поднесущей по строкам сигнала NTSC

№ строки	№ кадра					
	1 кадр		2 кадр		3 кадр	
	1 поле	2 поле	1 поле	2 поле	1 поле	2 поле
n-2	0	π	π	0	0	π
n-1	π	0	0	π	π	0
n	0	π	π	0	0	π
n+1	π	0	0	π	π	0
n+2	0	π	π	0	0	π

Форма осциллограмм цветоразностных сигналов и полного цветового ТВ-сигнала PAL при передаче цветных полос 75%-й яркости и 100%-й насыщенности приведена на рис. 2.6.

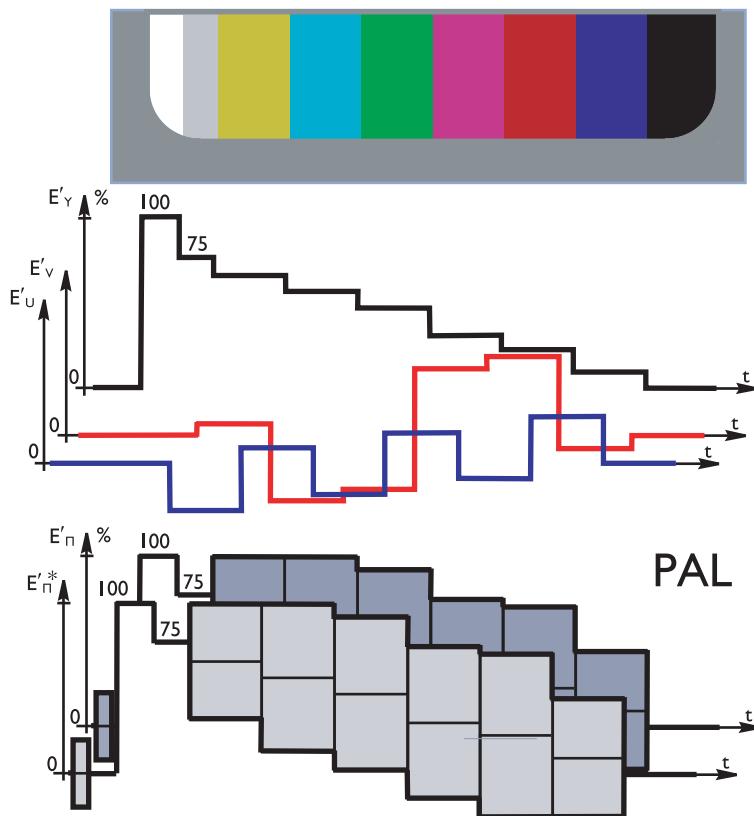


Рис. 2.6. Изображение цветных полос и форма соответствующих сигналов системы PAL

Вспышка цветовой поднесущей передается на задней площадке строчного гасящего импульса (10 ± 1 периодов колебаний), ее фаза коммутируется от строки к строке и составляет 135° в строках, где передается сигнал $+E_V$, и -135° в строках, где передается сигнал $-E_V$. Наглядное представление о сигнале цветности дает векторная диаграмма рис. 2.5. В табл. 2.5 приведены основные параметры составляющих полного цветового ТВ-сигнала цветных полос.

Ввиду того, что в системе PAL в двух смежных строках передаются различные сигналы цветности, а частота цветовой поднесущей равна сумме нечетной гармоники четверти строчной частоты и частоты кадров (кроме стандарта N), полный цикл изменения сигнала цветности равен четырем кадрам (см. табл. 2.6) [1.48].

При использовании в качестве частоты дискретизации четвертой гармоники цветовой поднесущей и 9-разрядного кодирования объем информации преобразованного в цифровой код полного цветового ТВ-сигнала системы PAL составит:

- для стандарта N — $I_0 \approx 128,7$ Мбит/с;
- для стандартов B, G, H, I — $I_0 \approx 159,6$ Мбит/с.

Таблица 2.5. Параметры сигналов цветных полос PAL

Цвет полосы	Сигналы основных цветов			Сигнал яркости	Сигналы цветности PAL			
	на видеочастоте		на поднесущей		размах, %	фаза, °	E'_S	E'^*_S
	E_R , %	E_G , %	E_B , %	E_Y , %				
Белый	100	100	100	100	0	0	0	—
Серый	75	75	75	75	0	0	0	—
Желтый	75	75	0	66,6	-32,9	+7,3	66,6	167
Голубой	0	75	75	52,4	+11,1	-46,5	95,2	283,5
Зеленый	0	75	0	44,2	-21,8	-39,2	88,4	240,5
Пурпурный	75	0	75	30,7	+21,8	+39,2	88,4	60,5
Красный	75	0	0	22,5	-11,1	+46,5	95,2	103,5
Синий	0	0	75	8,2	+32,9	-7,3	66,6	347
Черный	0	0	0	0	0	0	0	—
Вспышка	—	—	—	—	-15,1	+15,1	42,8	135
								225

Таблица 2.6. Изменение фазы поднесущей по строкам сигнала PAL

№ строки	№ кадра									
	1 кадр		2 кадр		3 кадр		4 кадр		5 кадр	
	1 поле	2 поле								
$n - 4$	F_0	$F_{\pi/2}^*$	$F_{3\pi/2}^*$	F_0	F_π	$F_{3\pi/2}^*$	$F_{\pi/2}^*$	F_π	F_0	$F_{\pi/2}^*$
$n - 3$	$F_{3\pi/2}^*$	F_0	F_π	$F_{3\pi/2}^*$	$F_{\pi/2}^*$	F_π	F_0	$F_{\pi/2}^*$	$F_{3\pi/2}^*$	F_0
$n - 2$	F_π	$F_{3\pi/2}^*$	$F_{\pi/2}^*$	F_π	F_0	$F_{\pi/2}^*$	$F_{3\pi/2}^*$	F_0	F_π	$F_{3\pi/2}^*$
$n - 1$	$F_{\pi/2}^*$	F_π	F_0	$F_{\pi/2}^*$	$F_{3\pi/2}^*$	F_0	F_π	$F_{3\pi/2}^*$	$F_{\pi/2}^*$	F_π
n	F_0	$F_{\pi/2}^*$	$F_{3\pi/2}^*$	F_0	F_π	$F_{3\pi/2}^*$	$F_{\pi/2}^*$	F_π	F_0	$F_{\pi/2}^*$
$n + 1$	$F_{3\pi/2}^*$	F_0	F_π	$F_{3\pi/2}^*$	$F_{\pi/2}^*$	F_π	F_0	$F_{\pi/2}^*$	$F_{3\pi/2}^*$	F_0
$n + 2$	F_π	$F_{3\pi/2}^*$	$F_{\pi/2}^*$	F_π	F_0	$F_{\pi/2}^*$	$F_{3\pi/2}^*$	F_0	F_π	$F_{3\pi/2}^*$
$n + 3$	$F_{\pi/2}^*$	F_π	F_0	$F_{\pi/2}^*$	$F_{3\pi/2}^*$	F_0	F_π	$F_{3\pi/2}^*$	$F_{\pi/2}^*$	F_π
$n + 4$	F_0	$F_{\pi/2}^*$	$F_{3\pi/2}^*$	F_0	F_π	$F_{3\pi/2}^*$	$F_{\pi/2}^*$	F_π	F_0	$F_{\pi/2}^*$

Примечание: F и F^* — вид передаваемого сигнала; $0, \pi/2, \pi, 3\pi/2$ — фаза поднесущей.

2.2.3. Система SECAM

В системе SECAM [1.9, 1.37, 1.43, 1.44, 1.49] (см. табл. 2.2) сигнал цветности представляет собой две цветовых поднесущих, промодулированных по частоте

цветоразностными сигналами в двух смежных строках:

$$D'_B = 1,5E_{B-Y};$$

$$D'_R = -1,9E_{R-Y}.$$

Особенностью этой системы является то, что сигналы двух частотно-модулированных цветовых поднесущих передаются в строчных интервалах поочередно, т. е. в каждой из строк передается только один из сигналов D'_B или D'_R , а на приеме в качестве второго сигнала используется сигнал из предыдущей строки, что достигается применением линии задержки на длительность строчного интервала. Такая замена реального сигнала на сигнал предыдущей строки не ведет к заметным искажениям, т. к. сигнал цветности от строки к строке изменяется очень мало.

Учитывая, что при декодировании частотно-модулированных поднесущих спектр флуктуационных помех приобретает треугольный характер, для повышения помехоустойчивости в кодирующем устройстве цветоразностные сигналы подвергаются предварительной коррекции по видеочастоте с помощью фильтра, комплексная частотная характеристика которого равна:

$$\bar{K}_B(i f) = \frac{1 + i f / f_0}{1 + i f / 3f_0},$$

где $f_0 = 85$ кГц, $f < 3$ МГц.

На рис. 2.7 приведены формы цветоразностных сигналов цветных полос после их предварительной коррекции фильтром с указанной характеристикой.

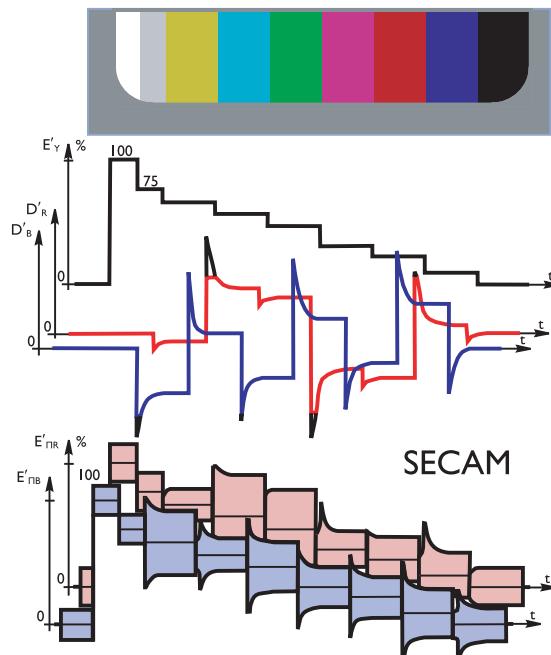


Рис. 2.7. Изображение цветных полос и форма соответствующих сигналов системы SECAM

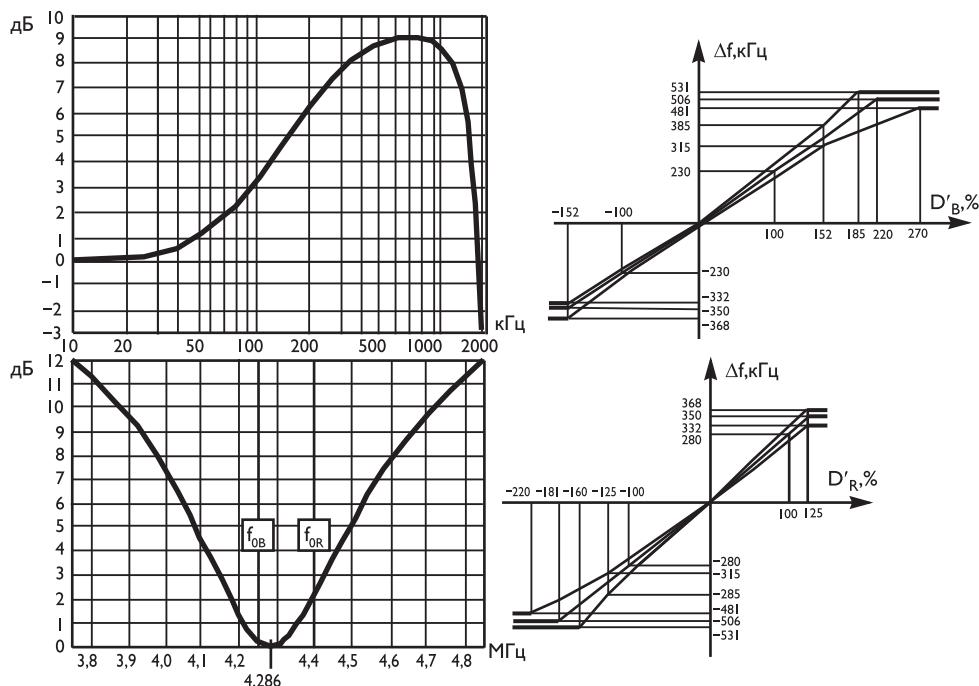


Рис. 2.8. Параметры цепей формирования сигналов цветности системы SECAM

На рис. 2.8 приведена номинальная результирующая АЧХ цепи предыскажений цветоразностных сигналов и ФНЧ, ограничивающих их спектр. Подавление спектра этих сигналов на частоте 1,3 МГц должно быть менее 3 дБ, а на частоте 3,5 МГц — более 30 дБ.

В результате предкоррекции сигналов D'_R и D'_B на их перепадах появляются выбросы (см. рис. 2.7), размах которых с учетом ограничения спектра сигналов примерно равен размаху перепадов. Для ограничения диапазона изменения поднесущих частот в частотном модуляторе производят ограничение выбросов, превышающих границы интервалов:

$$\begin{aligned}
 &+1,25 \text{ и } -1,81 \text{ для сигнала } D'_R; \\
 &+2,20 \text{ и } -1,52 \text{ для сигнала } D'_B.
 \end{aligned}$$

Номинальная девиация частот цветовых поднесущих, соответствующая уровням цветоразностных сигналов, равным 1,0, должна быть 280 кГц в строке с сигналом D'_R и 230 кГц в строке с сигналом D'_B (рис. 2.9).

Предкорректированные сигналы D'_R модулируют по частоте цветовую поднесущую $f_{0R} = 282 \cdot f_H = 4,40625$ МГц, а предкорректированные сигналы D'_B — цветовую поднесущую $f_{0B} = 272 \cdot f_H = 4,25$ МГц.

Предельная величина девиации, соответствующая передаче пиковых значений цветоразностных сигналов, равна:

- для сигнала D'_R : +350 кГц и -506 кГц;
- для сигнала D'_B : +506 кГц и -350 кГц.

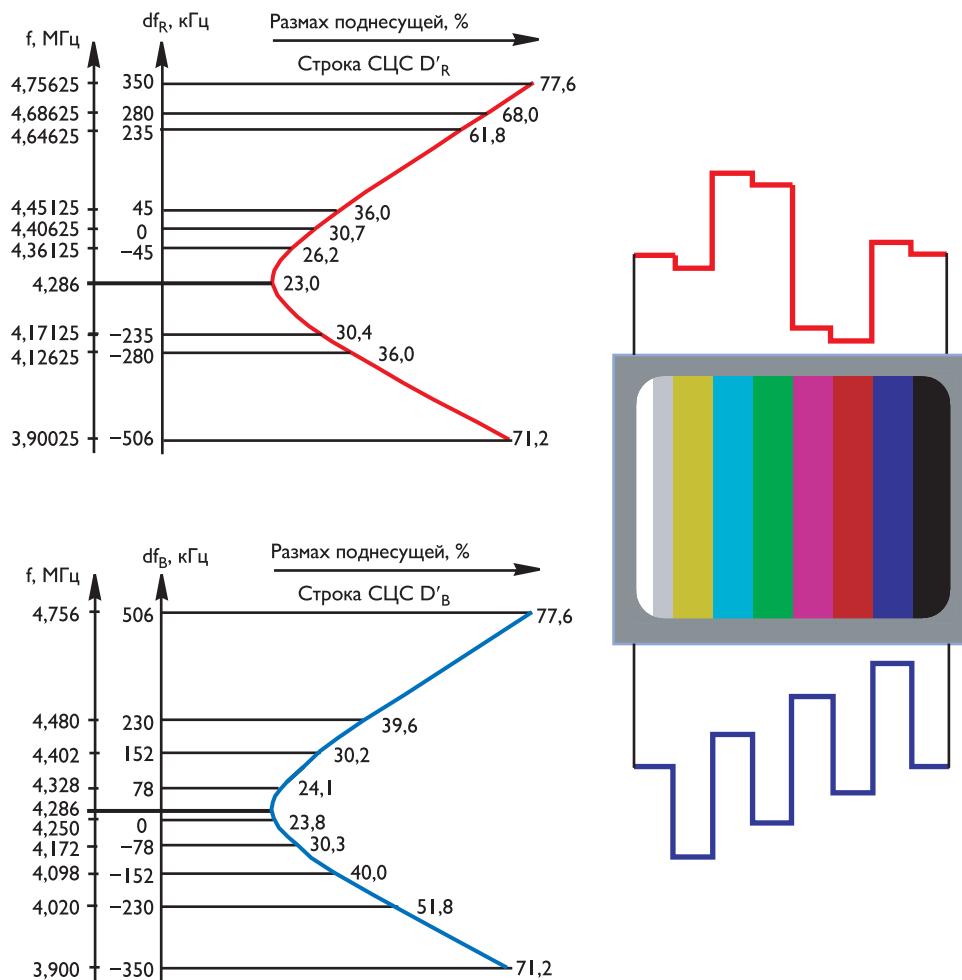


Рис. 2.9. Диаграммы, поясняющие взаимосвязь между частотами поднесущих системы SECAM и их девиацией

Характеристики модуляторов также изображены на рис. 2.8. Иллюстрацией здесь могут служить данные, приведенные на рис. 2.9 и в табл. 2.7.

Ввиду того, что частоты f_{0R} и f_{0B} кратны частоте строк, без использования специальных мер цветовые поднесущие на экране кинескопа могут создавать заметный узор из темных и светлых полос.

Для снижения заметности этого рисунка производят коммутацию фазы цветовых поднесущих по следующему закону: фаза цветовых поднесущих независимо от вида модулирующих сигналов (D'_R или D'_B) изменяется на 180° в начале и в конце каждой третьей строки и, кроме того, в начале и конце каждого поля.

В табл. 2.8 приведено изменение вида сигнала цветности и фазы цветовых поднесущих по строкам в системе SECAM (передаваемый сигнал D'_R обозначен буквой R , а D'_B — буквой B) [1.48].

Таблица 2.7. Параметры сигналов цветных полос SECAM

Цвет полосы	Сигналы основных цветов			Сигнал яркости	Сигналы цветности SECAM					
	по видеочастоте		на поднесущей							
	$E_R, \%$	$E_G, \%$	$E_B, \%$	$E_Y, \%$	$D'_R, \%$	$D'_B, \%$	строка D'_R	строка D'_B		
Белый	100	100	100	100	0	0	30,7	0	23,8	0
Серый	75	75	75	75	0	0	30,7	0	23,8	0
Желтый	75	75	0	66,6	-16,2	-100	26,2	-45	51,8	-230
Голубой	0	75	75	52,4	+100	+33,6	68,0	+280	24,1	+78
Зеленый	0	75	0	44,2	+83,6	-66,0	61,8	+235	40,0	-152
Пурпурный	75	0	75	30,7	-83,6	+66,0	30,4	-235	30,2	+152
Красный	75	0	0	22,5	-100	-33,6	36,0	-280	30,3	-78
Синий	0	0	75	8,2	+16,2	+100	36,0	+45	39,6	+230
Черный	0	0	0	0	0	0	30,7	0	23,8	0
СЦС	-	-	-	-	+125	-152	77,6	+350	71,2	-350

Примечание: СЦС —сигналы цветовой синхронизации, передаваемые в интервале кадрового гасящего импульса.

Для повышения помехоустойчивости декодирования цветоразностных сигналов дополнительно производится предварительная коррекция частотно-модулированных поднесущих с помощью цепи, комплексный коэффициент передачи которой:

$$\bar{K}_P(i f) = \frac{1 + i \cdot 16F}{1 + i \cdot 1,26F}, \quad \text{где} \quad F = \frac{f}{f_0} - \frac{f_0}{f}, \quad f_0 = 4,286 \text{ МГц}, |F| < 1.$$

Номинальная АЧХ этой цепи также приведена на рис. 2.8.

Включение в приемнике корректирующих цепей на поднесущей и по видеочастоте с характеристиками, обратными $\bar{K}_P(i f)$ и $\bar{K}_B(i f)$, обеспечивает уменьшение суммарной мощности флуктуационных помех в цветоразностных сигналах примерно в 4 раза. Форма цветовых поднесущих после предкоррекции при передаче цветных полос и форма полного цветового ТВ-сигнала в активных строках раstra приведены на рис. 2.8. Наглядное представление о взаимосвязи между частотами поднесущих, их девиациями дает рис. 2.9.

Полный цветовой ТВ-сигнал формируется путем сложения сигналов яркости, цветности и синхронизации.

Сигналы цветовой синхронизации (СЦС) передаются в строках с 7-й по 15-ю и с 320-й по 328-ю и формируются из разнополярных импульсов с пилообразным фронтом. Номинальные размахи СЦС в строках D'_R и D'_B соответственно равны 0,54 В и 0,5 В. Форма СЦС приведена на рис. 2.10. В табл. 2.7 даны основные параметры сигналов цветных полос, кодируемых по системе SECAM.

При использовании в качестве частоты дискретизации четвертой гармоники цветовой поднесущей покоя в строках D'_R и 9-разрядном кодировании объем информации преобразованного в цифровой код полного цветового ТВ-сигнала системы SECAM составит $I_0 \approx 158,6$ Мбит/с.

Таблица 2.8. Изменение сигналов цветности и фаз цветовых поднесущих по строкам сигнала SECAM

Номер строки	Номер кадра													
	1 кадр		2 кадр		3 кадр		4 кадр		5 кадр		6 кадр		7 кадр	
	1 поле	2 поле	1 поле	2 поле	1 поле	2 поле	1 поле	2 поле	1 поле	2 поле	1 поле	2 поле	1 поле	2 поле
$n - 6$	R_0	B_0	B_π	R_π	R_0	B_π	B_0	R_0	R_π	B_π	B_0	R_π	R_0	B_0
$n - 5$	B_π	R_π	R_0	B_π	B_0	R_0	R_π	B_π	B_0	R_π	R_0	B_0	B_π	R_π
$n - 4$	R_0	B_π	B_0	R_0	R_π	B_π	B_0	R_π	R_0	B_0	B_π	R_π	R_0	B_π
$n - 3$	B_0	R_0	R_π	B_π	B_0	R_π	R_0	B_0	B_π	R_π	R_0	B_π	B_0	R_0
$n - 2$	R_π	B_π	B_0	R_π	R_0	B_0	B_π	R_π	R_0	B_π	B_0	R_0	R_π	B_π
$n - 1$	B_0	R_π	R_0	B_0	B_π	R_π	R_0	B_π	B_0	R_0	R_π	B_π	B_0	R_π
n	R_0	B_0	B_π	R_π	R_0	B_π	B_0	R_0	R_π	B_π	B_0	R_π	R_0	B_0
$n + 1$	B_π	R_π	R_0	B_π	B_0	R_0	R_π	B_π	B_0	R_π	R_0	B_0	B_π	R_π
$n + 2$	R_0	B_π	B_0	R_0	R_π	B_π	B_0	R_π	R_0	B_0	B_π	R_π	R_0	B_π
$n + 3$	B_0	R_0	R_π	B_π	B_0	R_π	R_0	B_0	B_π	R_π	R_0	B_π	B_0	R_0
$n + 4$	R_π	B_π	B_0	R_π	R_0	B_0	B_π	R_π	R_0	B_π	B_0	R_0	R_π	B_π
$n + 5$	B_0	R_π	R_0	B_0	B_π	R_π	R_0	B_π	B_0	R_0	R_π	B_π	B_0	R_π
$n + 6$	R_0	B_0	B_π	R_π	R_0	B_π	B_0	R_0	R_π	B_π	B_0	R_π	R_0	B_0

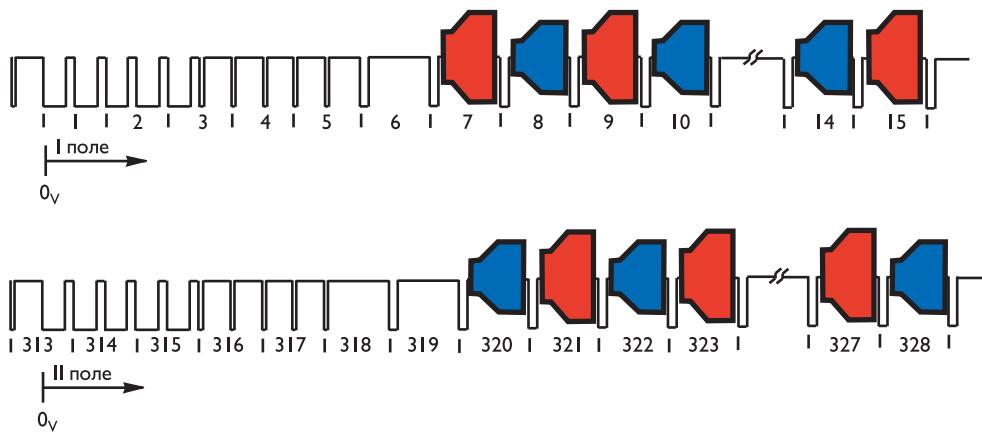


Рис. 2.10. Сигналы синхронизации полного цветового ТВ-сигнала системы SECAM

В табл. 2.9 сведены данные о требуемой скорости цифровой передачи монохромного ТВ-сигнала и сигналов различных систем цветного телевидения.

2.2.4. Цифровые компонентные телевизионные сигналы

Уменьшение объема информации при цифровой передаче сигналов цветного телевидения возможно при кодировании компонент полного цветового ТВ-сигнала.

К примеру, в системе NTSC сигналы E_Y , E_I и E_Q могут быть продискретизированы с соответствующими частотами Котельникова–Найквиста 8,4 МГц, 3 МГц и 1,2 МГц. При 8-битовом квантовании сигнала яркости и 7-битовом —

Таблица 2.9. Сравнение объемов информации для различных систем цветного телевидения

Тип изображения	Сигналы	Границные частоты сигналов яркости и цветности, МГц	Сигналы цветовой поднесущей, МГц	Скорость передачи цифровой информации, Мбит/с
Монохромное	E_Y	6,0	—	72
NTSC	$E_Y/E_I/E_Q$	4,2/1,5/0,6	\approx 3,58	128,7
PAL	$E_Y/E_U/E_V$	5,0/1,5/1,5	\approx 4,43	159,6
SECAM	$E_Y/D'_R/D'_B$ постстрочно	6,0/1,5/1,5	$4,406 - D'_R$, $4,25 - D'_B$	158,6

сигналов цветности, получим поток 96,6 Мбит/с, что соответствует дискретизации полного ТВ-сигнала 9-битовым ИКМ.

Если снизить цветовую четкость изображения по вертикали до цветовой четкости по горизонтали, т. е. уменьшить количество строк для сигнала E_I в два раза, а для сигнала E_Q — в четыре раза, то цифровой поток станет равным 79,8 Мбит/с.

Дополнительное сокращение цифрового потока связано с устранением пассивных участков строк и полей. В этом случае цифровой поток станет равным 60 Мбит/с.

Кодирование составляющих сигналов PAL и SECAM может осуществляться аналогичным образом, хотя потребуется несколько большая скорость передачи ввиду расширения полосы частот используемых сигналов яркости и цветности.

Так, для телевизионного стандарта, используемого в России (границная частота спектра сигнала яркости 6 МГц и граничные частоты спектров цветоразностных сигналов — 1,5 МГц), требуется использование цифрового потока порядка 80 Мбит/с (с учетом сокращения цветовой четкости по вертикали в 4 раза и устранения гасящих интервалов).

Попытка создать цифровой сигнал, совместимый с компонентами сигналов NTSC, PAL и SECAM привела к созданию международного стандарта покомпонентного кодирования полного цветового ТВ-сигнала [1.36, 1.37, 1.50].

Этот стандарт предусматривает передачу сигналов яркости E_Y и цветоразностных сигналов E_{R-Y} и E_{B-Y} , преобразованных в 8-разрядный цифровой код с частотой дискретизации:

$f_{DY} = 858 \cdot f_H$ NTSC = $864 \cdot f_H$ PAL / SECAM = 13,5 МГц — для яркостного сигнала, $f_{DR-Y/B-Y} = 6,75$ МГц — для цветоразностных сигналов.

Здесь f_H NTSC — частота строк для стандартов на 525/60 строк/полей в секунду, f_H PAL / SECAM — частота строк для стандартов на 625/50 строк/полей в секунду.

Для передачи диапазона размахов сигналов в пределах от 0 до 1,0 (1,0 — максимальный размах сигнала яркости) из общего числа 256 уровней — квантования отведено 220 уровней для сигнала яркости и 225 уровней для цветоразностных сигналов.

Максимальный и минимальный (255 и 0) уровни зарезервированы для сигналов синхронизации, диапазон изменений информационных сигналов 1...254.

Цифровой сигнал яркости определяется как:

$$Y_D = \text{round}(219E_Y + 16),$$

а цветоразностных сигналов:

$$C_R = \text{round}(112E_{R-Y}/0,701 + 128),$$

$$C_B = \text{round}(112E_{B-Y}/0,886 + 128),$$

где $\text{round}(x)$ —округление x до ближайшего целого.

Амплитудный диапазон сигнала E_{R-Y} лежит в пределах $-0,701\dots+0,701$, а сигнала E_{B-Y} — в пределах $-0,886\dots+0,886$.

Крайние числа этих пределов соответствуют максимальным и минимальным уровням цветоразностных сигналов при передаче цветных полос 100%-й яркости и насыщенности.

Дискретизирующая структура должна быть ортогональной и статической в пространстве и во времени. Отсчеты сигналов цветности в строке должны позиционно совпадать с нечетными отсчетами сигнала яркости.

В студийном варианте обычно производится непрерывная дискретизация сигнала яркости и цветоразностных сигналов без устранения пассивных участков этих сигналов. В этом случае при 8-битовом квантовании возникает цифровой поток объемом 216 Мбит/с.

При устранении пассивных интервалов и обеспечении равной четкости изображения по горизонтали и вертикали объем информации может быть существенно сокращен.

При частоте дискретизации 13,5 МГц количество отсчетов в строчном интервале ТВ-стандарта 625/25 равно 864. В активной части строки укладывается 720 отсчетов сигнала яркости.

Если формат изображения определяется соотношением 4 : 3, то в кадре должно быть использовано: $720 \cdot 3/4 = 540$ строк (при равной четкости по горизонтали и вертикали). Однако в стандартном телевизионном вещании используется 576 активных строк.

При 360 отсчетах каждого из цветоразностных сигналов в активной части строчного интервала количество используемых строк составляет 288. Таким образом, в случае 8-разрядного кодирования (см. рис. 2.11) сигнала яркости и двух цветоразностных сигналов общая скорость цифрового потока составит:

$$I_0 = 8 \cdot 25 \cdot [720 \cdot 576 + 2 \cdot 360 \cdot 288] = 124,416 \text{ Мбит/с.}$$

Здесь число 25 — количество кадров в секунду.

Для обеспечения качества вещания, соответствующего стандартным системам цветного телевидения, учитывая, что спектры цветоразностных сигналов примерно в четыре раза уже спектра сигнала яркости, частота дискретизации сигналов цветности может быть уменьшена еще в два раза, а количество используемых строк сокращено до 144. В этом случае объем информации о цветоразностных сигналах может быть дополнительно сокращен в четыре раза, и общая скорость цифрового потока составит:

$$I_0 = 103,68 \text{ Мбит/с.}$$

В американском стандарте используется 640 отсчетов в активной части строк и 480 активных строк при частоте кадров, равной 30 кадров в секунду. Следо-

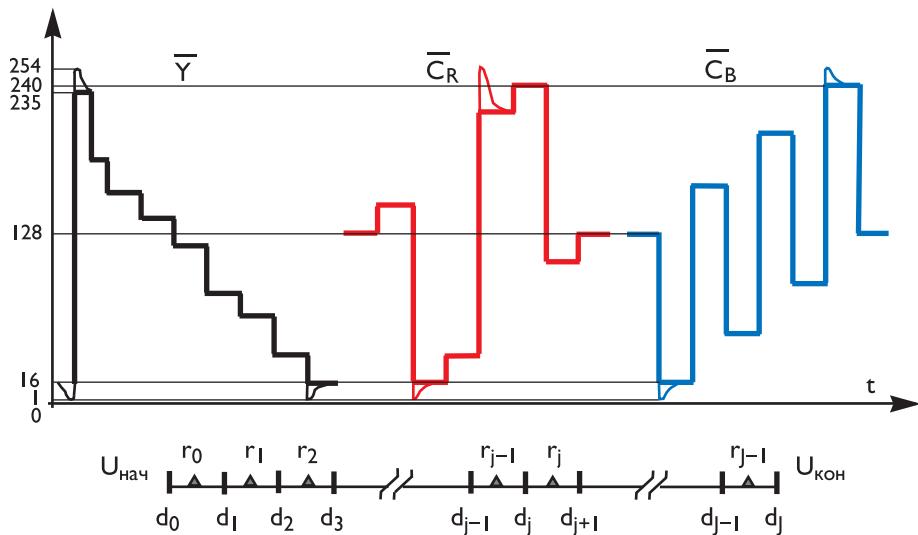


Рис. 2.11. Шкалы сигнала яркости и цветоразностных сигналов при компонентном кодировании

вательно, в случае 8-разрядного кодирования сигнала яркости и двух цветоразностных сигналов общая скорость цифрового потока составит:

$$I_0 = 8 \cdot 30 \cdot [640 \cdot 480 + 2 \cdot 320 \cdot 240] = 110,592 \text{ Мбит/с.}$$

2.3. Телевидение повышенной, высокой и сверхвысокой четкости, «цифровое кино»

Телевидение высокой четкости, рассматривавшееся в первых вариантах реализации как новая аналоговая система, в настоящее время развивается только в цифровом виде.

Телевидение повышенной четкости имеет иной формат кадра, определяемый соотношением 16 : 9 его ширины к высоте, примерно в два раза большее пространственное разрешение (1280×720) по сравнению со стандартным ТВ и может иметь в два раза большее временное разрешение (50, 60 кадров/с) при использовании прогрессивной развертки. Цветоразностные сигналы имеют в два раза меньшее разрешение по горизонтали по отношению к сигналу яркости и такое же или в два раза меньшее разрешение по вертикали (так называемые модели цветности 4 : 2 : 2 и 4 : 2 : 0) [1.2, 1.51–1.55]. При этом скорости формируемого цифрового потока для систем ТВ повышенной четкости лежат в диапазоне 264–844 Мбит/с.

Телевидение высокой четкости имеет еще более высокое пространственное разрешение (1920×1080) [1.2, 1.51–1.55]. При воспроизведении на большом экране повышается заметность мельканий и погрешностей цветности, поэтому в системах ТВЧ, как правило, рекомендуется использовать прогрессивную развертку

с частотой кадров 50 или 60 Гц и модель яркость/цветность 4 : 2 : 2, при которой на каждый пиксель приходится 16 битов информации (точнее, 32 бита на 2 пикселя — 2 восьмибитных значения яркости и по одному восьмибитному значению каждого из цветоразностных сигналов). Результирующий цифровой поток видеоданных, таким образом может составить до 2 Гбит/с.

Еще большие значения скорости цифрового потока получаются для перспективных систем ТВ сверхвысокой четкости и «цифрового кино» [1.52]. Здесь предполагается использовать пространственные разрешения от 1920×1080 (уровень Н0) до 7680×4320 (уровень Н3). Для представления отсчетов яркости и цветности предполагается использовать 10 (Н0, Н1) или 12 (Н2, Н3) битов. Скорость цифрового потока, таким образом, составит от 2,3 (Н0) до 66,7 (Н3) Гбит/с.

Наряду с улучшением и созданием новых систем сжатия традиционных видеосигналов, снимаемых с одной точки наблюдения, бурно развивается как стереоскопическое телевидение (бинокулярное), так и автостереоскопическое (с несколькими точками наблюдения, обычно до 9–16 видов). Кроме того, создано и развивается такое направление, как телевизионные системы с произвольной точкой наблюдения (ТСПТН) [1.54].

Мультивидовое видео, записанное синхронизированными фото- или видеокамерами с различных точек наблюдения, дает обширную 3D-информацию о сцене и расширяет зрительские впечатления по сравнению с традиционным видео. 3D-телевидение обеспечивает трехмерное ощущение глубины наблюдаемого вида, а ТСПТН дает возможность интерактивного выбора точки наблюдения и направления в пределах определенного рабочего диапазона углов обзора. Реализация 3D-приложений зависит от полного цикла обработки, включая захват, сжатие, передачу, отображение и интерактивное представление. Кодирование нескольких видов динамических изображений является одной из наиболее многообещающих технологий, поскольку огромный объем данных, пропорциональный числу камер, должен быть сжат до такой степени, чтобы его можно было передавать в пределах возможностей сетей связи.

2.4. Видеотелефония и видеоконференцсвязь, домашнее видео

В системах видеотелефонии и видеоконференцсвязи предъявляются пониженные (по сравнению с обычным ТВ) требования как к четкости изображения (пространственному разрешению), так и к его динамичности (временному разрешению).

В видеотелефонии обычно используются разрешения до 176×144 пикселя при скорости смены кадров до 5 кадров/с. Соответственно, скорость результирующего цифрового потока сравнительно невелика — до 1,5 Мбит/с.

В видеоконференцсвязи может использоваться большая четкость изображения — до четверти телевизионного при кадровой скорости до 25–30 кадров/с, скорость цифрового потока доходит до 35 Мбит/с.

В системах домашнего видео и видео по запросу используется телевизионное временное разрешение (25–30 кадров/с), но снижены требования к простран-

ственному разрешению, поэтому скорость цифрового потока в этих системах примерно в 4 раза ниже, чем в ТВ-системах.

Цифровое представление видеосигнала требует больших объемов памяти для хранения и высоких пропускных способностей каналов связи (см. табл. 2.10). Использование каналов связи для передачи цифровых видеосигналов без специальной обработки было бы малоэффективным, так как цифровое представление видеоинформации обладает, как правило, существенной избыточностью.

Для снижения потока видеоданных в различных системах передачи информации чаще всего используются стандартизованные методы, позволяющие производить совместимое оборудование множеству компаний, работающих в этой области. Наибольшее распространение получили стандарты семейства MPEG [1.56–1.60], использующие общую схему видеокомпрессии.

В табл. 2.10 приведены параметры исходных видеосигналов для различных приложений, обеспечивающих передачу движущихся изображений, а в табл. 2.11 — параметры исходных сигналов для некоторых видеоинформационных систем со многими точками наблюдения.

2.5. Графические изображения, факсимильные системы и интерактивные системы передачи изображений

В ряде случаев такие изображения статические и двухуровневые. Так как изображение обычно состоит из текста и рисунков, требуется более высокое пространственное разрешение по сравнению со стандартным телевидением.

В большинстве случаев такие изображения можно представить в виде сочетания мозаичных или геометрических элементов, которые хранятся в памяти приемного устройства.

Имеются стандарты МККТТ по передаче графических изображений. Факсимильные системы обеспечивают передачу сигналов по телефонным аналоговым или цифровым каналам (общего пользования или выделенным). Цифровая передача обычно ведется со скоростями 1200, 2400, 4800 или 9600 бит/с. В новейших системах скорости достигают 56 Кбит/с при помехоустойчивом кодировании.

Интерактивные системы передачи изображений — это так называемые системы Телетекст и Видеотекс [1.61, 1.62]. Первая из них предполагает передачу информации в интервалах строк гасящих импульсов полей ТВ-сигнала, вторая использует телефонные линии общего пользования. Обе имеют один стандарт, определяющий набор используемых символов:

- литеры и мозаичные элементы;
- прямые;
- точки;
- дуги;
- многоугольники и т. д.

Они объединены в кодовые таблицы, составляющие иерархическую структуру.

Таблица 2.10. Параметры видео для различных приложений

Приложения		Размер кадра	Модель цветности, количество битов на пикセル	Формат экрана	Частота кадров, кадров/с, развертка	Скорость цифрового потока, Мбит/с		
Видеотелефония		subQCIF, QSIF, QCIF	4 : 2 : 0, 12	4 : 3	1–5 прогрессивная	0,141–1,45		
Видеоконференц-связь		QSIF, QCIF, SIF, CIF	4 : 2 : 0, 12	4 : 3	5–30 прогрессивная	1,10–34,8		
Домашнее видео, видео по требованию		SIF	4 : 2 : 0, 12	4 : 3	30 прогрессивная	26,4		
		CIF			25 прогрессивная	29,0		
Стандартное ТВ		640 × 480 (NTSC)	4:2:0, 12 4:2:2, 16	4:3	30 чересстрочная	105–141		
		720 × 576 (PAL, SECAM)			25 чересстрочная	119–158		
ТВ-повышенной четкости		1280 × 720	4 : 2 : 0, 12 4 : 2 : 2, 16	16 : 9	25, 30, 50, 60 прогрессивная	264–844		
ТВЧ		1920 × 1080	4:2:2, 16	16:9	25, 30 чересстрочная	791–949		
					25, 30, 50, 60 прогрессивная	791–1900		
Цифровое кино	H0	1920 × 1080	4:2:2, 20	16 : 9	60 прогрессивная	2370		
	H1	3840 × 2160				9490		
	H2	5760 × 3240	4 : 4 : 4, 36			38400		
	H3	7680 × 4320				68300		

Таблица 2.11. Параметры видео для некоторых информационных систем со многими точками наблюдения

Многомерные телевизионные системы	Формат и размер кадра	Модель цветности и количество битов на пиксел	Число кадров в секунду, тип развертки	Скорость цифрового потока, Гбит/с
Трехмерное ТВ — $2 \times 1080i$	1920 : 1080; (2×960) : 1800; 16 : 9	4 : 2 : 0; 12	50, 60; чересстрочная	0,8–0,95
Трехмерное ТВ — Blu-Ray; 3D(кино); $2 \times 1080p$	1920 : 1080; 16 : 9	4 : 2 : 2; 32	24; прогрессивная	1,58
Трехмерное ТВ — HD-3D; $2 \times 1080p$	1920 : 1080; 16 : 9	4 : 2 : 0; 24	50, 60; прогрессивная	3,2–3,6
Трехмерное ТВ — MultiView 3D-TV, несколько видов; $n \times 1080p$	1920 : 1080; 16 : 9	4 : 2 : 0; 12	50, 60; прогрессивная	$n \times 1,5 \sim 100$ –1000
Трехмерное ТВ — Free Viewpoint TV (FTV) — произвольное к-во видов; $n \times 1080p$	1920 : 1080; 16 : 9	4 : 4 : 4; 24	50, 60; прогрессивная	$n \times 1,5 \sim 100$ –1000
Трехмерное ТВ — 3D-Integral Imaging, на основе интегральной фотографии; $n^2 \times 4320p$	7680 : 4320	4 : 4 : 4	60; прогрессивная	$n^2 \times 1,5 \sim 15$ 000
Голографические системы ТВ	4320 : 8192		60; прогрессивная	$40 \times 1,5 \sim 10$ 000–1 000 000

ГЛАВА 3

ЦИФРОВОЕ ПРЕДСТАВЛЕНИЕ ВИДЕОИНФОРМАЦИИ

3.1. Представление изображения конечным объемом данных

Телевизионное и компьютерное изображение в его цифровом представлении является набором значений интенсивностей светового потока, распределенных на конечной площади, имеющей обычно прямоугольную форму.

Для простоты рассмотрим монохромные изображения. Тогда интенсивность излучаемой световой энергии с единицы поверхности в точке с координатами (x, y) изображения можно представить неким числом $B(x, y)$, характеризующим яркость пикселя с координатами (x, y) .

Задача состоит в удобном для цифровой обработки представлении $B(x, y)$ конечным количеством двоичных цифр (битов).

Если даже $B(x, y)$ изменяется во времени, предполагается, что объем информации в единицу времени, требуемый для представления $B(x, y)$, конечен, т. е. конечна скорость ее передачи.

Таким образом, поскольку движение непрерывно, а скорость передачи конечна, то имеет место дискретное представление непрерывного процесса, и часть визуальной информации безвозвратно теряется.

Однако если даже значения $B(x, y)$ могут быть получены с большой точностью (например, для не меняющегося во времени $B(x, y)$ или для меняющегося во времени $B(x, y)$ при очень высокой скорости передачи информации), то, как будет показано ниже, далеко не всегда необходимо сохранять всю информацию.

Плотность размещения пикселов связана с пространственным разрешением, которое в телевидении обычно определяется количеством различимых линий на единицу длины. На рис. 3.1 в качестве примера приведено изображение вертикальных и горизонтальных черно-белых линий, изменяющихся по ширине скачками от минимальной величины. Очевидно: если плотность черно-белых линий достигает некоторой критической величины, то зрительное их восприятие нарушается — визуально они воспринимаются в виде непрерывного серого поля.

Обычно значение уровня сигнала в каждом из пикселов определяют путем взвешенного усреднения $B(x, y)$ в малой окрестности δ [1.37, 1.63]:

$$\hat{B}(x, y) = \int_{y-\delta}^{y+\delta} \int_{x-\delta}^{x+\delta} B(w, z) \cdot H_C(x - w, y - z) dw dz, \quad (3.1)$$

где $H_C(u, v)$ — взвешивающая функция или импульсный отклик фильтра.

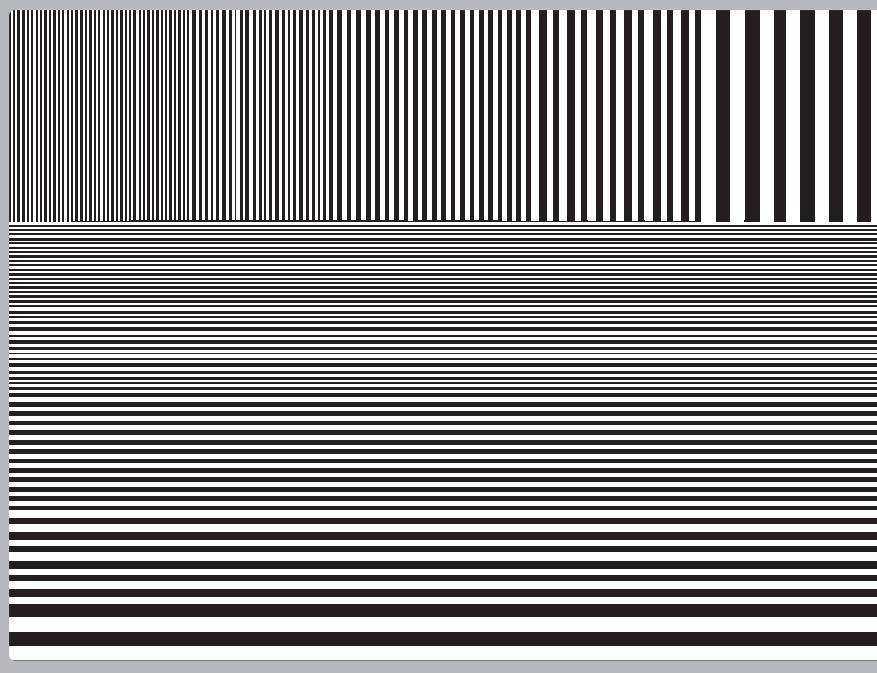


Рис. 3.1. Изображение вертикальных и горизонтальных черно-белых полос

Этот процесс называется префильтрацией, т. к. изображение фильтруется до дискретизации. Часто в качестве взвешивающей функции используют функцию Гаусса:

$$H_C(u, v) = K \cdot \exp(-au^2 - bv^2), \quad (3.2)$$

где $|u|, |v| < \delta$; K, a, b — константы.

Два варианта откликов для разных значений констант a и b приведены на псевдотрехмерной диаграмме рис. 3.2.

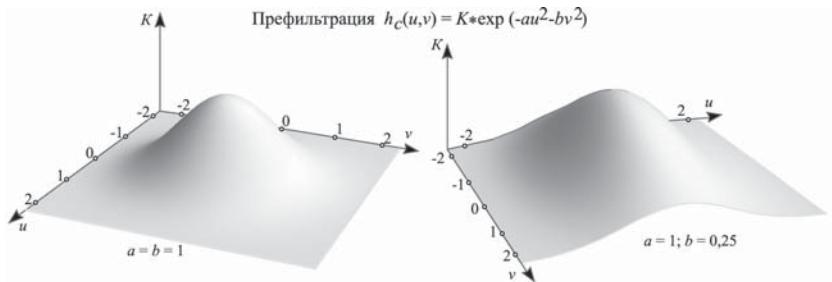


Рис. 3.2. Варианты импульсных откликов префильтра Гаусса

При восстановлении изображения (т. е. воспроизведении точек между дискретными пикселями) используют различные методы интерполяции:

$$\hat{B}(x, y) = \sum_k \hat{B}(x_k, y_k) \cdot H_d(x - x_k, y - y_k), \quad (3.3)$$

где $H_d(u, v)$ — интерполяционная функция [1.37, 1.63]. Обычно используют билинейную функцию:

$$H_d(u, v) = S(u, \Delta x) \cdot S(v, \Delta y), \quad (3.4)$$

где Δx и Δy — горизонтальные и вертикальные расстояния между пикселями. Для прямоугольных массивов пикселов используют линейную интерполяцию:

$$S(z, \Delta) = 1 - \left| \frac{z}{\Delta} \right|, \quad \left| \frac{z}{\Delta} \right| \leq 1. \quad (3.5)$$

Часто применяются более гладкие интерполяции.

Квадратичная интерполяция:

$$S(z, \Delta) = \begin{cases} \frac{3}{4} - \left| \frac{z}{\Delta} \right|^2, & \left| \frac{z}{\Delta} \right| \leq \frac{1}{2}; \\ \frac{9}{8} - \frac{3}{2} \left| \frac{z}{\Delta} \right| + \frac{1}{2} \left| \frac{z}{\Delta} \right|^2, & \frac{1}{2} < \left| \frac{z}{\Delta} \right| \leq \frac{3}{2}. \end{cases} \quad (3.6)$$

Кубическая интерполяция (непрерывная первая производная):

$$S(z, \Delta) = \begin{cases} 1 - \frac{5}{2} \left| \frac{z}{\Delta} \right|^2 + \frac{3}{2} \left| \frac{z}{\Delta} \right|^3, & \left| \frac{z}{\Delta} \right| \leq 1; \\ 2 - 4 \left| \frac{z}{\Delta} \right| + \frac{5}{2} \left| \frac{z}{\Delta} \right|^2 - \frac{1}{2} \left| \frac{z}{\Delta} \right|^3, & 1 < \left| \frac{z}{\Delta} \right| \leq 2. \end{cases} \quad (3.7)$$

Интерполяция кубическим В-сплайном (непрерывные первая и вторая производные):

$$S(z, \Delta) = \begin{cases} \frac{2}{3} - \left| \frac{z}{\Delta} \right|^2 + \frac{1}{2} \left| \frac{z}{\Delta} \right|^3, & \left| \frac{z}{\Delta} \right| \leq 1; \\ \frac{4}{3} - 2 \left| \frac{z}{\Delta} \right| + \left| \frac{z}{\Delta} \right|^2 - \frac{1}{6} \left| \frac{z}{\Delta} \right|^3, & 1 < \left| \frac{z}{\Delta} \right| \leq 2. \end{cases} \quad (3.8)$$

Псевдотрехмерное представление интерполяционных многочленов приведено на рис. 3.3. Этот процесс интерполяции называется постфильтрацией. Целиком цепочка цифровой обработки изображений выглядит так, как показано на рис. 3.4.

При этом перед цифровой обработкой изображения осуществляется его префильтрация, затем следуют дискретизация, квантование и кодирование пикселов. При восстановлении изображения после декодирования пикселов производится постфильтрация.

Наиболее часто встречающаяся форма пространственной дискретизации — сканирование раstra, преобразующее двумерное изображение в одномерную временную функцию. В частности, в телевидении используется чересстрочная развертка. Из функции $B(x, y)$ при этом создается одномерная функция [1.37, 1.47] $B(t) = B[t(x, y)]$, которая подвергается префильтрации:

$$\hat{B}(t) = \int B(\tau) \cdot h_C(t - \tau) d\tau, \quad (3.9)$$

где $h_C(\tau)$ — импульсная характеристика префильтра. Затем $\hat{B}(t)$ дискретизируется так, чтобы в строке умещалось целое количество точек $\{\hat{B}(t_k)\}$ ($k = 1 \dots N_S$), а в кадре — целое количество строк. Полученный продискретизированный сигнал

$$B_S(t) = \sum_{k=1}^{N_S} \hat{B}(t) \cdot \delta(t - t_k) \quad (3.10)$$

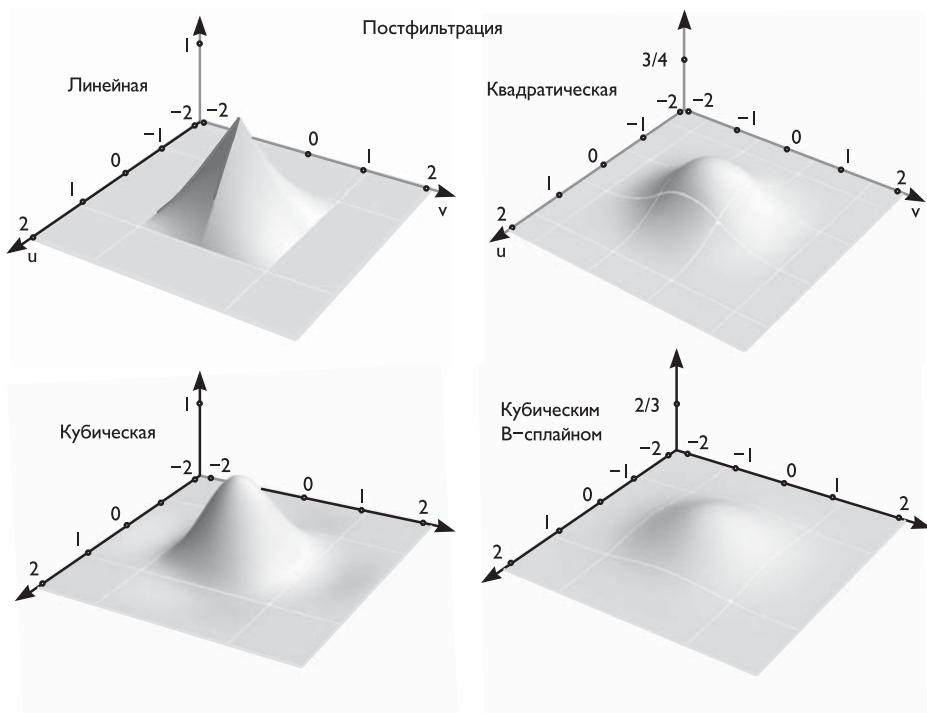


Рис. 3.3. Псевдотрехмерное представление интерполяционных многочленов, используемых при восстановлении изображений

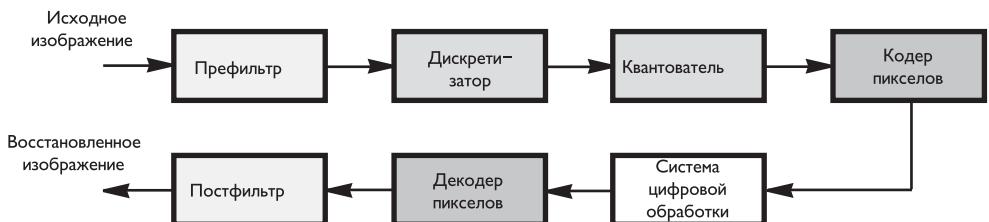


Рис. 3.4. Последовательность операций цифровой обработки изображений

квантуется и превращается в цифровой код. Здесь $\delta(\tau)$ — дельта-функция.

После декодирования и временной постфильтрации производится восстановление непрерывного сигнала и раstra:

$$\begin{aligned}\tilde{B}(t) &= \int B_S(\tau) \cdot h_d(t - \tau) d\tau = \tilde{B}(x_t, y_t) \text{ или} \\ \tilde{B}(x, y) &= \int \tilde{B}(t) \cdot h_d(x - x_t, y - y_t) dt.\end{aligned}\tag{3.11}$$

3.2. Фурье-преобразование, теорема о дискретизации

Пространственная дискретизация и квантование уровней пикселов изображения — не единственный метод представления визуальной информации конечным количеством данных.

Если исходное изображение $B(x, y)$ преобразуется в некоторую другую форму до дискретизации и квантования, то иногда можно получить более удобное представление и кодирование информации.

Одним из таких представлений является Фурье-преобразование [1.64–1.69], при котором прямоугольное изображение $B(x, y)$ в координатах $0 \leq x \leq X$; $0 \leq y \leq Y$, с помощью комплексных функций

$$\Phi_{mn}(x, y) = \frac{1}{\sqrt{XY}} \cdot \exp \left(2\pi i \left(\frac{nx}{X} + \frac{my}{Y} \right) \right) \quad (3.12)$$

может однозначно быть записано в виде ряда:

$$B(x, y) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} c_{mn} \cdot \Phi_{mn}(x, y), \quad (3.13)$$

где

$$c_{mn} = \int_0^Y \int_0^X B(x, y) \cdot \Phi'_{mn}(x, y) dx dy,$$

$\Phi'_{mn}(x, y)$ комплексно сопряжена с $\Phi_{mn}(x, y)$, а базисные функции $\Phi_{mn}(x, y)$ — ортогональны, поскольку

$$\int_0^Y \int_0^X \Phi_{mn}(x, y) \cdot \Phi'_{kl}(x, y) dx dy = \delta_{mk} \cdot \delta_{nl}. \quad (3.14)$$

Как известно, преобразование Фурье, примененное к одномерной функции $y = F(x)$, позволяет получить разложение этой функции на частотные компоненты так, что каждая точка кривой $F(x)$ может быть представлена суммой синусоид и косинусоид различных частот, взятых каждая со своим коэффициентом [1.69, 1.70]. Аналогично, представление $B(x, y)$ в сочетаниях экспоненциальных базисных Фурье-изображений есть разложение изображений на компоненты пространственных частот:

$$f_x = \frac{n}{X} \text{ периодов на единицу длины по оси } x \text{ и}$$

$$f_y = \frac{m}{Y} \text{ периодов на единицу длины по оси } y.$$

Комплексные коэффициенты c_{mn} характеризуют интенсивность и фазу этих пространственных частот относительно выбранной системы координат.

Общая пространственная частота двумерного периодического изображения определяется соотношением:

$$f_s = \sqrt{f_x^2 + f_y^2}.$$

Компоненты высших пространственных частот определяют быстрые изменения интенсивности в изображении, характеризующие мелкие детали, резкие переходы

яркости и пр. Компоненты низких пространственных частот связаны с медленными и плавными изменениями яркости.

Очевидно, можно выбрать некоторые граничные пространственные частоты F_x и F_y , исходя из того критерия, что пространственные частоты выше граничных уже ничего не добавляют к качеству изображения, поскольку игнорируются зрительным анализатором человека. Наличие в сигнале пространственных частот выше граничной уже не приводит к повышению визуальной четкости воспроизводимого изображения, а следовательно, эти составляющие сигнала можно отбросить.

Тогда изображение $\hat{B}(x, y)$, в общем случае отличающееся от $B(x, y)$, представимо конечным числом пространственных частот:

$$\hat{B}(x, y) = \sum_{m=-M}^M \sum_{n=-N}^N c_{mn} \cdot \Phi_{mn}(x, y), \quad (3.15)$$

и спектральные компоненты ограничены частотами:

$$\begin{aligned} |f_x| &< \frac{2N+1}{2X} = F_x, \\ |f_y| &< \frac{2M+1}{2Y} = F_y, \end{aligned}$$

а количество ненулевых коэффициентов c_{mn} равно $N_S = (2M+1)(2N+1)$.

При этом ограниченное число коэффициентов Фурье c_{mn} можно рассчитать, используя N_S дискретных значений ограниченного по спектру изображения $B(x, y)$, поскольку для каждого пикселя с координатами (x_k, y_k) можно записать:

$$\hat{B}(x_k, y_k) = \sum_{m=-M}^M \sum_{n=-N}^N c_{mn} \cdot \Phi_{mn}(x_k, y_k). \quad (3.16)$$

Это и есть разложение элемента изображения на пространственные частоты с помощью двумерного преобразования Фурье.

Если пиксели выбраны таким образом, что N_S этих уравнений линейно независимы, то их можно решить относительно N_S неизвестных коэффициентов, не прибегая к вычислению интегралов.

Такое множество дискретных точек состоит из прямоугольного массива с $N_x = 2N+1$ столбцов и $N_y = 2M+1$ строк.

В этом случае $N_S = N_x \times N_y$ и для $|m|, |k| \leq M; |n|, |l| \leq N$ можно показать, что

$$\sum_{i=0}^{N_x-1} \sum_{j=0}^{N_y-1} \Phi_{mn}(x_i, y_j) \cdot \Phi'_{kl}(x_i, y_j) = \frac{N_s}{XY} \cdot \delta_{mk} \cdot \delta_{nl}, \quad (3.17)$$

где $x_i = \frac{X}{N_x} \cdot i$, $y_j = \frac{Y}{N_y} \cdot j$. Отсюда вытекает, что

$$c_{mn} = \frac{XY}{N_s} \cdot \sum_{i=0}^{N_x-1} \sum_{j=0}^{N_y-1} \hat{B}(x_i, y_j) \cdot \Phi'_{mn}(x_i, y_j). \quad (3.18)$$

В этом и состоит суть теоремы отсчетов, необходимая плотность которых должна немного превышать удвоенные граничные пространственные частоты [1.37]:

$$D_x = \frac{2N+1}{X} = 2F_x, \quad D_y = \frac{2M+1}{Y} = 2F_y.$$

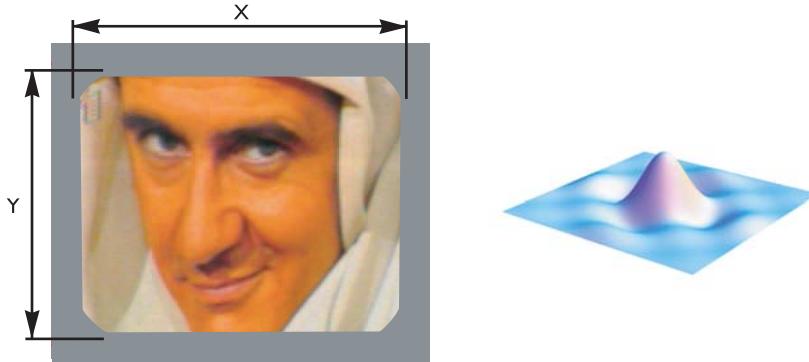


Рис. 3.5. Двумерная функция отсчетов

При этом такие дискретные отсчеты полностью определяют ограниченное по спектру изображение $\hat{B}(x, y)$ в любых его точках.

Можно показать, что значение интенсивности ограниченного по спектру изображения в любой его точке с координатами (x, y) определяется взвешенной суммой отсчетов:

$$\hat{B}(x, y) = \sum_{i=0}^{2N} \sum_{j=0}^{2M} \hat{B}(x_i, y_j) \cdot \chi(x - x_i, y - y_j), \quad (3.19)$$

где

$$\chi(u, v) = \frac{\sin \pi \frac{(2N+1)u}{X} \cdot \sin \pi \frac{(2M+1)v}{Y}}{N_s \cdot \sin \pi \frac{u}{X} \cdot \sin \pi \frac{v}{Y}} -$$

так называемая функция отсчетов, имеющая максимум в точке $u = 0, v = 0$ и быстро спадающая по мере удаления от нее.

Эта функция приведена на рис. 3.5. С ее помощью обеспечивается точная интерполяция изображений, строго ограниченных по спектру.

Приведенные выше выражения представляют собой аналог теоремы Котельникова [1.70] для двумерного сигнала. Заметим, что функция $\chi(u, v)$ очень похожа на двумерную sinc-функцию ($\text{sinc}(z) = \frac{\sin \pi z}{\pi z}$), за исключением того, что она периодична по u и v с периодами X и Y соответственно.

Префильтрация изображения фильтром с импульсной характеристикой $\chi(u, v)$ обеспечивает удаление частот выше определенного заданного предела.

Фильтрация изображения имеет одно важное отличие от фильтрации бесконечной одномерной функции времени. Для ограниченного в пространстве изображения непрерывность резко нарушается на краях.

Это нарушение часто приводит к росту высокочастотных компонент, которые не имеют отношения к информации, содержащейся в изображении.

При попытке отфильтровать эти частоты появляются сильные искажения на границах (так называемый эффект Гиббса).

В качестве примера на рис. 3.6 приведено изображение без программного ограничения спектра (а) и с резким его ограничением (б). Там же приведена разность яркостей этих двух изображений (в). Как видно, при ограничении спектра наиболее заметны искажения колебательного характера на границах изображения.

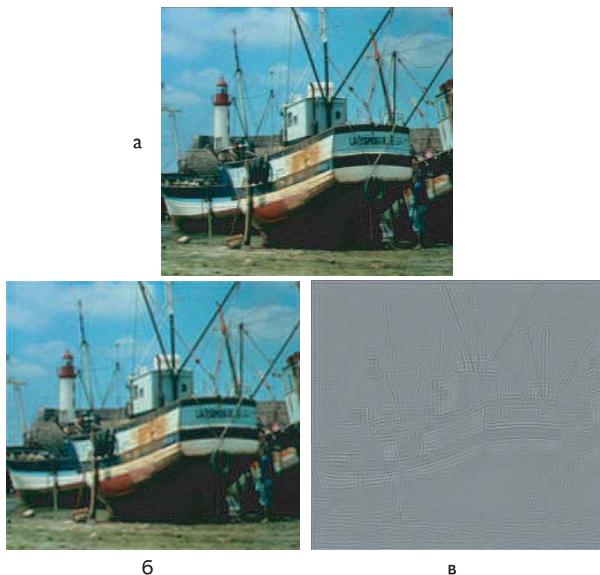


Рис. 3.6. Эффект Гиббса при резком ограничении спектра изображения

3.3. Изображение, изменяющееся во времени

Изменяющееся во времени изображение можно представить в виде трехмерной функции $B(x, y, t)$, которая для временного интервала $0 \leq t \leq T$ может быть записана следующим образом [1.37]:

$$B(x, y) = \sum_{m, n, p = -\infty}^{\infty} c_{mnp} \cdot \Phi_{mnp}(x, y, t), \quad (3.20)$$

где

$$\Phi_{mnp}(x, y, t) = \frac{1}{\sqrt{XYT}} \cdot \exp 2\pi i \left(\frac{nx}{X} + \frac{my}{Y} + \frac{pt}{T} \right),$$

$$c_{mp} = \int_0^T \int_0^Y \int_0^X B(x, y, t) \cdot \Phi'_{mnp}(x, y, t) dx dy dt,$$

p/T — временная частота.

Если после префильтрации p/T меньше некоторой величины W (Гц), то по аналогии с вышеизложенным изображение можно продискретизировать во времени с частотой $2W$ Гц. Для человеческого глаза граница этой частоты 10–15 Гц обычно вполне достаточна [1.8, 1.9].

При этом движущееся изображение в последовательности кадров представляет собой дискретные мгновенно зафиксированные статические изображения, как показано на рис. 3.7.

Практически временная префильтрация осуществляется в самом датчике, поскольку большинство датчиков до дискретизации просто накапливает световую энергию за время экспозиции T_I .



Рис. 3.7. Дискретная последовательность кадров

Это обеспечивает ослабление каждой временной частотной компоненты на $\text{sinc}(f_t \cdot T_I)$, где f_t (в герцах) — временная частота.

Следует заметить, что если время интегрирования T_I очень мало, то временная префильтрация неэффективна и возможно возникновение строб-эффекта, вызывающего, например, иллюзорное вращение колес назад и движение рывками.

Однако даже если T_I равно периоду между кадрами T_F , ослабление на половинной частоте дискретизации $f_t = 1/2 \cdot T_F$ составляет только 4 дБ, что недостаточно для устранения указанного выше эффекта согласно теореме о дискретизации.

В телевизионной камере префильтрованное указанным образом изображение ($T_I = T_F$) сканируется по раstrу, и дискретизация производится над одномерным сигналом $\hat{B}(t) = \hat{B}(x_t, y_t, t)$. Таким образом, между дискретизацией точек вверху и внизу раstra проходит некоторое время, равное длительности кадра (поля). Это приводит к определенному искажению формы больших движущихся объектов, если рассматривать один кадр. Однако форма быстров движущихся объектов также нарушается (точнее, «расплывается») при временной фильтрации. Восстановление изображения требует интерполяции или постфильтрации. Однако временная постфильтрация из-за аппаратных сложностей осуществляется не аппаратурой, а человеческим глазом.

Следует отметить, что эффективность такой интерполяции изображений между кадрами зависит от окружающего освещения: чем ярче дисплей, тем выше заметность мерцания из-за недостаточной временной постфильтрации глазом.

Как было отмечено, для большинства движущихся изображений временные частоты редко превышают 10–15 Гц. Именно по этой причине в кинокамерах смена кадров осуществляется с частотой 24 Гц. Однако для того, чтобы не было мерцаний, проектор производит смену кадров с частотой 48 или 72 Гц, что достигается путем показа каждого кадра 2 или 3 раза.

В телевидении эта технология до сих пор не использовалась, поскольку в приемнике требовалось хранить достаточно большой объем информации.

Вместо этого используется следующее свойство глаза: временная постфильтрация осуществляется глазом значительно более эффективно для мелких деталей изображения, т. е. высоких пространственных частот, чем для более крупных деталей изображения, т. е. более низких пространственных частот.

Таким образом, эффективность фильтрации может быть достигнута, если низкие пространственные частоты записываются в видеокамере и воспроизводятся ТВ-приемником с частотой 50–60 Гц, а высокие пространственные частоты — лишь с частотой 25–30 Гц.

Это как раз и достигается использованием чересстрочной развертки (2:1). Однако ее использование приводит к следующим дефектам:

- мерцанию строк при воспроизведении высоких вертикальных пространственных частот;
- искажению формы объектов при их движении в горизонтальном направлении.

3.4. Квантование. Импульсно-кодовая модуляция

Одним из основных шагов при получении представления визуальной информации потоком с конечной битовой скоростью является превращение каждого значения элемента изображения в двоичное слово конечной длины, т. е. создание кода.

В процессе квантования отсчет аналогового сигнала сравнивается с набором пороговых уровней. Если отсчет попадает в определенный амплитудный интервал, то ему приписывается значение фиксированного уровня квантования, соответствующего данному интервалу. В цифровой системе каждому квантованному отсчету ставится в соответствие двоичная кодовая комбинация.

Уровни и коды квантования могут изменяться в зависимости от обстоятельств. Например, неэффективно квантовать и кодировать все коэффициенты Фурье-изображения по одному правилу.

Квантование аналоговых величин приводит к потере части информации. Важным вопросом является рациональный выбор пороговых уровней и уровней квантования, обеспечивающий минимальную погрешность процедуры восстановления, определенную выбранным критерием.

При решении задачи о квантовании аналоговой величины u , изменяющейся в пределах от U_H до U_K , необходимо выбрать такой набор пороговых уровней d_j и уровней r_j , что если

$$d_j \leq u < d_{j+1},$$

то исходное значение u заменяется на число r_j [1.37, 1.39, 1.47].

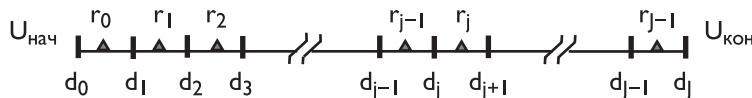


Рис. 3.8. Изображение при различном числе уровней квантования и пример размещения пороговых уровней и уровней квантования

На рис. 3.8 приведен пример размещения пороговых уровней и уровней квантования на отрезке числовой оси, содержащей J пороговых уровней.

Уровни квантования и пороговые уровни выбираются так, чтобы уменьшить до минимума некоторую заданную величину, характеризующую ошибку квантования, т. е. степень различия между u и ее квантованной величиной \hat{u} , равной соответствующему уровню квантования. В качестве меры ошибки зачастую выбирают среднеквадратичную ошибку:

$$e = \int_{U_H}^{U_K} (u - \hat{u})^2 p(u) du = \sum_{j=0}^{J-1} \int_{d_j}^{d_{j+1}} (u - r_j)^2 p(u) du, \quad (3.21)$$

где $p(u)$ — плотность распределения случайной величины u .

Если число уровней квантования велико, то положения пороговых уровней и уровней квантования достаточно точно определяются по формуле [1.37, 1.39, 1.64]:

$$d_j = \frac{(U_K - U_H) \int_{U_H}^{U_K} \frac{du}{\sqrt[3]{p(u)}}}{\int_{U_H}^{U_K} \frac{du}{\sqrt[3]{p(u)}}} + U'_H, \quad (3.22)$$

где $u_j = \frac{j(U_K - U_H)}{J} + U_H$, $j = 0, 1, \dots, J$; $r_j = \frac{d_{j+1} - d_j}{2}$, $j = 0, 1, \dots, J - 1$. Если же число уровней квантования невелико, то применяется более сложный алгоритм вычисления величины d_j и r_j , называемый алгоритмом Макса или Ллойда–Макса [1.64, 1.71]. При этом для заданной плотности вероятностей $p(u)$ оптимальные значения пороговых уровней и уровней квантования можно найти путем решения рекуррентным способом системы уравнений:

$$\begin{cases} r_j = 2d_j - r_{j-1}; \\ r_j = \frac{\int_{d_j}^{d_{j+1}} u \cdot p(u) du}{\int_{d_j}^{d_{j+1}} p(u) du}. \end{cases} \quad (3.23)$$

При этом среднеквадратическая ошибка квантования становится минимальной:

$$e_{\min} = \sum_{j=0}^{J-1} \left[\int_{d_j}^{d_{j+1}} u^2 p(u) du - r_j \int_{d_j}^{d_{j+1}} p(u) du \right]. \quad (3.24)$$

В табл. 3.1 приведены расположения только положительных пороговых уровней и уровней квантования (d_j и r_j , $j = 0, 1, \dots$) в квантователях Ллойда–Макса для плотностей равномерного распределения вероятностей в пределах $[-1, 1]$, а также распределений Гаусса и Лапласа:

$$p(u) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{u^2}{2\sigma^2}\right), \quad (3.25)$$

$$p(u) = \frac{1}{\sigma\sqrt{2}} \exp\left(-\frac{|u|\sqrt{2}}{\sigma}\right). \quad (3.26)$$

Таблица 3.1. Пороговые уровни и уровни квантования Ллойда–Макса

Число пороговых уровней	Равномерное распределение		Распределение Гаусса		Распределение Лапласа	
	d_j	r_j	d_j	r_j	d_j	r_j
3	0,0000		0,0000		0,0000	
		0,5000		0,7979		0,7071
	1,0000		∞		∞	
5	0,0000		0,0000		0,0000	
		0,2500		0,4528		0,4198
	0,5000		0,9816		1,1269	
		0,7500		1,5104		1,8340
	1,0000		∞		∞	

Таблица 3.1 (окончание)

7	0,0000	0,0000	0,0000	
	0,1250	0,5005	0,2451	0,2334
	0,2500	0,3750	0,5332	
	0,5000	1,050	0,7560	0,8330
	0,6250		1,3439	1,6725
	0,7500	1,7479		2,3796
	0,8750		2,1519	3,0867
	1,0000	∞		∞
9	0,0000	0,0000	0,0000	
	0,0625		0,1284	0,1240
	0,1250	0,2582		0,2644
	0,1875		0,3880	0,4048
	0,2500	0,5224		0,5667
	0,3125		0,6568	0,7287
	0,3750	0,7995		0,9198
	0,4375		0,9423	1,1110
	0,5000	1,0993		1,3444
	0,5625		1,2562	1,5778
	0,6250	1,4371		1,8776
	0,6875		1,6180	2,1773
	0,7500	1,8435		2,5971
	0,8125		2,0690	3,0169
	0,8750	2,4008		3,7240
	0,9375		2,7326	4,4311
	1,0000	∞		∞

Отрицательные пороговые уровни и уровни квантования соответственно равны:

$$d_{-j} = -d_j \text{ и } r_{-j} = -r_j, \quad j = 0, 1, \dots$$

В системе обработки сигналов яркости телевизионных изображений с применением импульсно-кодовой модуляции (ИКМ) каждый отсчет обычно квантуется на основе линейной шкалы, и ему ставится в соответствие двоичная кодовая комбинация.

При кодировании непосредственно яркостного сигнала применяют равномерный код, и поэтому число уровней квантования пикселов определяется соотношением [1.36, 1.37, 1.39]: $J = 2^B$, где B — число двоичных разрядов (битов), отведенных для кодирования отсчетов.

Замена исходных пикселов квантованными приближениями приводит к ошибкам восстановленного изображения, которые проявляются в виде случайного шума или «снега». В большинстве приложений этот шум практически не виден при 8-битовом квантовании, при снижении числа битов заметность шума увеличивается (см. рис. 3.8).

Следует отметить, что заметность шума ИКМ-квантования сильно зависит от того, как много шума в исходном изображении до квантования. Большинство приложений требуют, чтобы шум квантования был малым по сравнению с шумом исходного изображения.

В системах с ИКМ сокращение цифрового потока может быть осуществлено простым уменьшением числа разрядов в кодовых комбинациях.

При субъективной оценке качества воспроизводимых изображений число разрядов B понижают до тех пор, пока эффекты, вызванные квантованием, не выйдут за границы допустимого. Глаз способен определить от 10 до 15 градаций абсолютного значения светлоты, но имеет гораздо большую чувствительность к различию светлоты соседних элементов изображения [1.64, 1.45].

При уменьшении числа уровней квантования прежде всего бросается в глаза эффект появления контуров в тех областях, где светлота исходного изображения изменяется плавно (см. рис. 3.8). Появление контуров вызывается скачкообразным изменением светлоты квантованного изображения при переходе от одного уровня квантования к другому.

Наименьшее число разрядов ИКМ, позволяющее предотвратить появление контуров в областях с плавным изменением светлоты, как было указано, зависит от ряда факторов, в том числе от линейности характеристики дисплея и свойств шума до и после преобразования изображения в видеосигнал.

Литература к части I

- 1.1. Цифровая обработка телевизионных и компьютерных изображений / Под ред. Ю.Б. Зубарева и В.П. Дворковича. 2-е изд., перераб. и доп. М.: НАТ, 1997.
- 1.2. Кривошеев М.И. Международная стандартизация цифрового телевизионного вещания. М.: НИИР, 2006
- 1.3. Зубарев Ю.Б., Кривошеев М.И., Красносельский И.Н. Цифровое телевизионное вещание. Основы. Методы. Системы. М.: Мир, 2001
- 1.4. Синепол В.С., Цикин И.А. Системы компьютерной видеоконференцсвязи. М.: Связь и бизнес, 1999
- 1.5. Методы компьютерной обработки изображений / Под ред. В.А. Сойфера. М.: Физматлит, 2001
- 1.6. Гонсалес Р., Буде Р. Цифровая обработка изображений. М.: Техносфера, 2005
- 1.7. Яне Б. Цифровая обработка изображений. М.: Техносфера, 2007
- 1.8. Справочник по телевизионной технике / Пер. с англ. под ред. С.И. Катаева. Т.1–2. М.-Л.: Госэнергоиздат, 1962
- 1.9. Телевизионная техника. Справочник / Под общ. ред. Ю.Б. Зубарева и Г.Л. Глориозова. М.: Радио и связь, 1994
- 1.10. Видеоконференции и видеоконференцсвязь: словарь терминов
http://slovotolk.com/society/tvcinema/frameglossary_rus.html
- 1.11. Видеоконференции и видеоконференцсвязь: словарь терминов
http://www.stel.ru/videoconference/tech_vc/glossary/
- 1.12. NNM-Club: Глоссарий — перечень видео и кодек терминов
<http://nnm-club.ru/forum/viewtopic.php?t=28>
- 1.13. Standards — Videokonferenz|Audiodaten|Vebkonferenz
<http://www.mvc.de/contenttwocolumns.php?c=16&pn=4&pk=23#v>
- 1.14. SkyVisionDVR.Com — List of Image Resolution
http://skyvisiondvr.com/sky/index.php?option=com_content&task=view&id=11&Itemid=25
- 1.15. Ох уж эти разрешения
<http://www.mobil.ru/article.php?advice=108>
- 1.16. John McGowan's AVI Overview: Glossary
<http://www.jmcgowan.com/avigloss.html>
- 1.17. USB PC Camera
<http://www.pc-cameras.philips.com/manuals/Pdf/usermanual646en.pdf>
- 1.18. TRLabs / Video Conferencing: Promising but why did we end up traveling to this Symposium?
<http://www.win.trlabs.ca/icts/images/andrewkostiuk.pdf>

- 1.19. Приоров А.Л., Хрящев В.В., Апальков И.В. Цифровая обработка изображений: Учебное пособие. Ярославль: ЯрГУ им. П.Г. Демидова, 2007
- 1.20. Hoffman G. CIE Color Space
<http://www.fho-emde/~hoffmann/ciexyz29082000.pdf>
- 1.21. Гуревич М.М. Цвет и его измерение. М.: АН СССР, 1950
- 1.22. Кустарев А.К. Колориметрия цветного телевидения. М.: Связь, 1967
- 1.23. Кривошеев М.И., Кустарев А.К. Световые измерения в телевидении. М.: Связь, 1973
- 1.24. CIE 1931 Color Space
http://en.wikipedia.org/wiki/CIE_1931_color_space
- 1.25. MacAdam D.L. Sources of Color Science. Cambridge, MA, MIT Press, 1970
- 1.26. Джадд Д., Вышецки Г. Цвет в науке и технике. М.: Мир, 1978
- 1.27. Юстова Е.Н. Таблицы основных колориметрических величин. М.: Изд-во Комитета стандартов, 1967
- 1.28. Photometry/Colorimetry-light perception
<http://www.paladix.cz///www.wikipedia.org>
- 1.29. Chromaticity Diagrams Lab Report
<http://www.efg2.com/Lab/Graphics/Colors/Cromaticity.html>
- 1.30. Lab Color Spase
http://en.wikipedia.org/wiki/Lab_color_space
- 1.31. CIPR Still Images
<http://www.cipr.rpi.edu/resource/stills/index.html>
- 1.32. CIPR Sequences
<http://www.cipr.rpi.edu/resource/sequences/index.html>
- 1.33. The University of South California free database
<http://sipi.usc.edu/database/index/html>
- 1.34. Wikipedia materials plas
http://en.wikipedia.org/wiki/Standard_test_image#cite_note-1
- 1.35. МККР. Характеристики монохромных систем телевидения. Отчет 308
- 1.36. МККР. Параметры кодированных сигналов цифрового телевидения для студий. Рекомендация 601
- 1.37. Netravali A.N., Haskell B.G. Digital pictures: Representation and Compression.: Plenum Press, N.Y., 1991
- 1.38. Цифровое телевидение / Под ред. М.И.Кривошеева. М.: Связь, 1980
- 1.39. Цифровое кодирование телевизионных изображений / Под ред. И.И. Цукермана. М.: Связь, 1981
- 1.40. МККР. Характеристики телевизионных систем. Отчет 624
- 1.41. Темников Ф.Е., Афонин В.А., Дмитриев В.И. Теоретические основы информационной техники. М.: Энергия, 1971
- 1.42. Певзнер Б.М. Системы цветного телевидения. Л.: Энергия, 1969
- 1.43. МККР. Характеристики систем цветного телевидения. Отчет 407

- 1.44. *Новаковский С.В.* Стандартные системы цветного телевидения. М.: Связь, 1976
- 1.45. *Левинер Б.М.* Качество цветных телевизионных изображений. М.: Радио и связь, 1988
- 1.46. МККР. Номенклатура сигналов цветных полос. Рекомендация 471
- 1.47. *Птачек М.* Цифровое телевидение. Теория и техника. М.: Радио и связь, 1990
- 1.48. *Дворкович В.П., Крикошев М.И.* Особенности измерения и контроля в цветном телевидении. М.: Издательство ГКТР, 1981
- 1.49. ГОСТ 7845. Система вещательного телевидения. Основные параметры. Методы измерений
- 1.50. *Новаковский С.В., Котельников А.В.* Новые системы телевидения. М.: Радио и связь, 1992
- 1.51. ITU-R Recommendation BT.1200-1 Target standard for digital video systems for the studio and for international programme exchange, 1998
- 1.52. ITU-R Recommendation BT.1201-1 Extremely high resolution imagery, 2004
- 1.53. МККР. Подходы к разработке общемирового стандарта студийного ТВЧ. Отчет AW/11
- 1.54. *Дворкович В., Чобану М.* Проблемы и перспективы развития систем кодирования динамических изображений // MediaVision. 2011. № 2, 3, 4, 5, 7, 8
- 1.55. *Хлебородов В.А.* На пути к единому мировому стандарту ТВЧ // Техника кино и телевидения, 1988, № 2
- 1.56. ISO/IEC 11172-2 — Information Technology — Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1,5 Mbit/s — Part 2: Video, 1993 (MPEG-1 Video)
- 1.57. ISO/IEC 14496-2 — Information Technology — Coding of Audio-Visual Objects — Part 2: Visual, 2004 (MPEG-4 Video)
- 1.58. ITU-T Recommendation H.261 — Video Codec for Audiovisual services at p x 64 kbit/s, 1993; ITU-T Recommendation H.263 — Video Coding for Low Bit Rate Communication, 2005
- 1.59. ITU-T Recommendation H.262 / ISO/IEC 13818-2 — Information Technology — Generic Coding of Moving Pictures and Associated Audio Information: Video, 2000 (MPEG-2 Video)
- 1.60. ITU-T Recommendation H.264 — Advanced Video Coding for Generic Audio-visual Services / ISO/IEC 14496-10 — Information Technology — Coding of Audio-Visual Objects — Part 10: Advanced Video Coding, 2005
- 1.61. *Дорошенко И.В.* Системы телетекста. Техника средств связи. Сер. Техника телевидения. 1981, вып. 3
- 1.62. *Мартин Дж.* Видеотекс и информационное обслуживание общества. Перевод с английского. М.: Радио и связь, 1987

- 1.63. *Jayant N.S., Noll P.* Digital Coding of Waveforms, Chapter 4.: Prentice Hall, N.J., 1984
- 1.64. *Прэтт У.* Цифровая обработка изображений. Т. 1–2. М.: Мир, 1982
- 1.65. *Залманзон Л.А.* Преобразования Фурье, Уолша, Хаара и их применение в управлении, связи и других областях. М.: Наука, 1989
- 1.66. *Ахмед Н., Rao K.P.* Ортогональные преобразования при обработке цифровых сигналов. М.: Связь, 1980
- 1.67. *Гольдберг Л.М., Матюшин Б.Д., Поляк М.Н.* Цифровая обработка сигналов. Справочник. М.: Радио и связь, 1985
- 1.68. *Ярославский Л.П.* Введение в цифровую обработку изображений. М.: Сов. радио, 1979
- 1.69. *Рабинер Л.Р. Гоулд Б.* Теория и применение цифровой обработки сигналов. М., Мир, 1978
- 1.70. *Левин Б.Р.* Теоретические основы статистической радиотехники. т.1, т.2, т.3. М.: Сов. радио, 1974, 1975, 1976
- 1.71. *Max J.* Quantizing for minimum distortion. IEEE Trans., 1960, v. JT-6, № 3

ЧАСТЬ II

СТАТИСТИЧЕСКАЯ И ВИЗУАЛЬНАЯ ИЗБЫТОЧНОСТЬ ИЗОБРАЖЕНИЙ

Введение

Сокращение избыточности информации о статическом и динамическом изображении является способом обеспечения существенного уменьшения объема цифровой информации для реализации режимов ее хранения и передачи по каналам связи, использования в разнообразных приложениях, таких как видеотелефония, видеоконференцсвязь, широковещательное телевидение, дистанционное зондирование, телемедицина и др.

Избыточность информации является центральным понятием для ее цифрового сжатия. Термин «сжатие» означает уменьшение объема данных, используемых для описания определенного количества информации. Данные являются средством передачи информации. При этом для определения одного и того же количества информации может быть использовано различное количество данных.

Как оценить эффективность различных видов представления информации? Используется два вида оценки. Один из них, относительный, характеризуется коэффициентом сжатия [2.1–2.4]:

$$C_R = \frac{N_2}{N_1},$$

где N_1 и N_2 — данные, характеризующие объем информации до и после сокращения избыточности.

Абсолютная оценка эффективности представления информации определяется величиной, равной количеству информации данного представления, приходящемуся в среднем на один информационный символ. Так, если исходная информация об изображении характеризуется объемом 8 битов на пиксел, то при

эффективном кодировании на каждый пиксель в среднем может приходиться существенно меньше бита.

Поток данных об изображении имеет значительное количество излишней информации, которая может быть устранена практически без визуальной заметности его изменений.

Существует два типа избыточности:

- статистическая избыточность, связанная с корреляцией и предсказуемостью данных; эта избыточность может быть устранена без потери информации, исходные данные при этом могут быть полностью восстановлены;
- визуальная (субъективная) избыточность, которую можно устраниить с частичной потерей данных, мало влияющих на качество воспроизведенных изображений; это — информация, которую можно изъять из изображения, не нарушая визуально воспринимаемое качество изображений.

Соседние пиксели динамических изображений как в пространственной, так и во временной областях коррелированы. Устранение внутрикадровой статистической избыточности (т. е. пространственной корреляции пикселов) позволяет сократить объем информации об изображении в 1,5–4 раза.

В основе методов устранения статистической избыточности лежит информационная теорема Клода Шеннона, которая характеризует зависимость эффективности представления информации от свойств самой информации. Для количественной оценки свойств информации вводятся две связанные между собой характеристики: энтропия и избыточность. Теорема Шеннона определяет предельные возможности статистического кодирования информации.

При создании метода статистического кодирования решаются три проблемы: построения информационной модели, генерации кода и хранения описания способа кодирования. Эффективность кодирования различных видов информации зависит от применяемых принципов решения этих проблем.

Визуальная избыточность связана с психофизическими свойствами зрения. При восприятии изображения зритель не оценивает количественные параметры отдельных пикселов, а отыскивает определенные особенности, такие как контуры или текстурные области, подсознательно объединяет их в узнаваемые детали и изображения в целом.

Визуальная избыточность принципиально отличается от статистической избыточности, она связана с реальной и количественно измеримой зрительной информацией. Поскольку она не воспринимается при визуальном восприятии, поскольку она может быть устранена при цифровой обработке изображений.

Устранение визуальной избыточности позволяет очень существенно сократить объем информации за счет необратимого удаления той части информации, которая не воспринимается или плохо различается человеческим глазом. При использовании соответствующих методов объем информации об отдельном изображении может быть снижен в 8–12 раз без видимых искажений, а при обработке последовательности изображений объем информации может быть сокращен в несколько десятков и даже сотни раз.

ГЛАВА 4

СТАТИСТИЧЕСКАЯ ИЗБЫТОЧНОСТЬ ДИСКРЕТИЗИРОВАННЫХ ДАННЫХ

4.1. Информационная теорема

Пусть имеется дискретизированное на $IM \times N$ пикселов и квантованное с точностью K битов монохромное изображение. Следовательно, при импульсно-кодовой модуляции необходимо передать $M \times N \times K$ битов информации.

Если предположить, что квантованные значения яркости неравновероятны, то уменьшение информации возможно путем изменения количества битов информации для кодирования пикселов: более вероятные значения яркости кодируются словами с меньшим количеством битов, а менее вероятные — с большим. Этот метод называется кодированием словами переменной длины, или энтропийным кодированием.

Пусть квантованный уровень яркости b_i имеет вероятность $P(b_i)$, и ему присваивается слово — код длины $L(b_i)$ битов. Тогда средняя длина кода для всего изображения составит:

$$\hat{L}(B) = \sum_{i=1}^I L(b_i) \cdot P(b_i) \text{ бит/отсчет}, \quad (4.1)$$

B — последовательность отсчетов (источник данных), которые могут принимать одно из значений b_i с вероятностями $P(b_i)$.

Нижняя граница величины \hat{L} определяется информационной теоремой и называется энтропией случайной величины [2.1–2.6]. Если энтропия $H(S)$ измеряется в битах, то

$$H(B) = - \sum_{i=1}^I P(b_i) \cdot \log_2 P(b_i) \text{ бит/отсчет} \leq \hat{L}.$$

Таким образом, энтропия — это мера количества информации, которую несет случайная величина. $H(B) \geq 0$, поскольку $P(b_i) \in [0, 1]$. Из формулы $H(B)$ вытекает, что чем более неравномерно распределение $P(b_i)$, тем меньше энтропия и тем эффективнее энтропийное кодирование.

В случае, когда уровни яркости появляются равновероятно, энтропия максимальна и равна

$$H(B)_{\max} = \log_2 I.$$

По аналогии с определением энтропии вводится понятие избыточности как меры определенности (неслучайности) информации. Ее относительная величина определяется формулой:

$$R(B) = 1 - \frac{H(B)}{H(B)_{\max}}. \quad (4.2)$$

4.2. Виды статистического кодирования

Эффективность методов устранения статистической избыточности, называемых энтропийным или экономным кодированием, зависит от выбранной расчетной модели источника информации. На рис. 4.1 приведена классификация методов энтропийного кодирования.

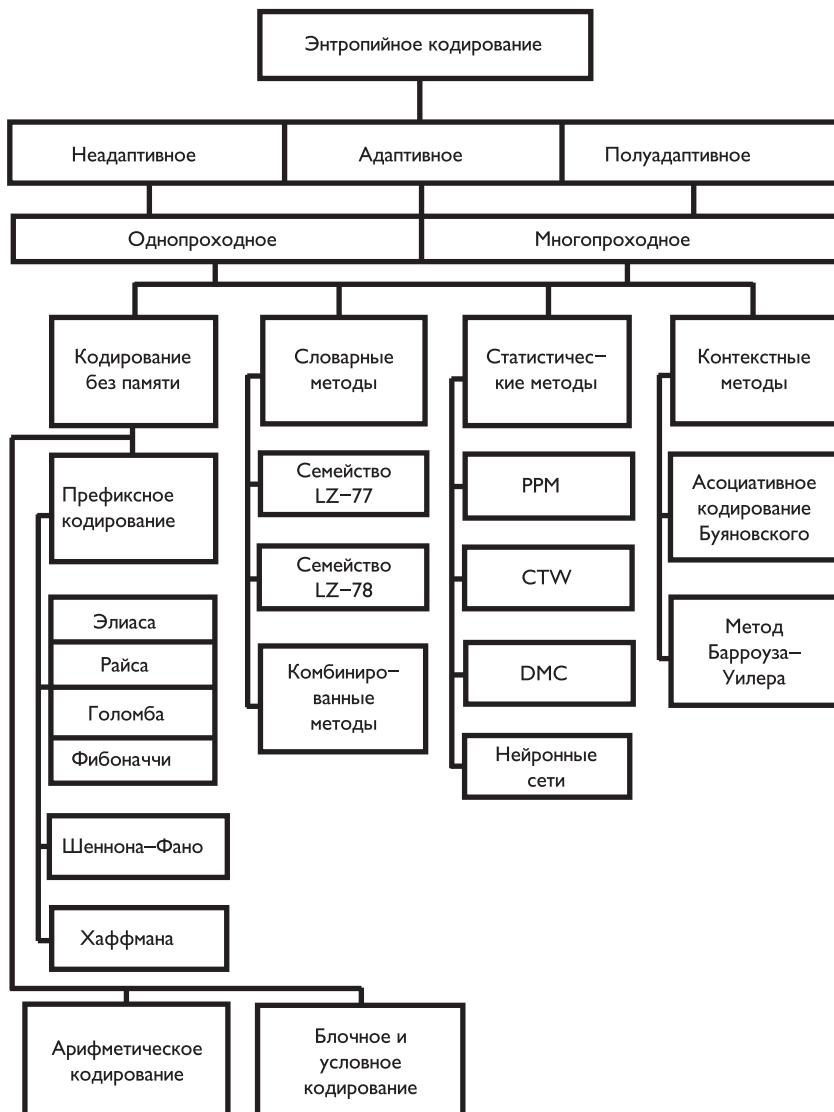


Рис. 4.1. Классификация методов устранения статистической избыточности

Методы энтропийного кодирования делят на адаптивные и неадаптивные. В неадаптивных методах способ кодирования не меняется в процессе обработки

данных, а в адаптивных методах кодирование подстраивается под изменяющиеся статистические характеристики потока входных данных.

При использовании неадаптивного варианта алгоритма необходимо передать на декодирующую сторону таблицу распределения вероятностей. При адаптивном варианте кодирования таблица распределения вероятностей пересчитывается как на передающей, так и на приемной стороне по мере поступления данных. Существуют также полуадаптивные алгоритмы, при которых применяют некоторые подобранные способы кодирования и хранят их описание вместе с закодированной информацией.

Алгоритмы кодирования могут быть однопроходными и многопроходными, т. е. требующими сбора предварительной информации об обрабатываемых данных до начала непосредственного процесса кодирования. В связи с этим процесс обработки информации зачастую состоит из двух этапов — моделирования, используемого для оценки дерева вероятностей информационного поля, и собственно кодирования источников данных без памяти.

Методы кодирования источников данных без памяти включают префиксное кодирование и арифметическое кодирование. Повышение эффективности этих методов связано с применением блочного и условного кодирования.

Наиболее известные методы префиксного кодирования — хорошо исследованные схемы кодирования целых чисел Элиаса, Райса, Голомба, Фибоначчи [2.7–2.11], алгоритмы Шеннона–Фано [2.2, 2.12], Хаффмана [2.13].

Основой арифметического кодирования послужил алгоритм П. Элайеса. Существенный вклад в его развитие в разное время внесли Р. Паско [2.14], Д. Риссанен, Г. Лэнгдон и В. Пеннебакер [2.15–2.18], Ф. Рубин [2.19], М. Гуаззо [2.20], Г. Мартин [2.21], Д. Клири, И. Уиттен, Э. Моффат и Р. Нил [2.22, 2.23], П. Говард [2.24].

Снижение битового потока возможно при применении блочного и условного кодирования [2.4], т. е. кодирование и передача информации блоками из значительного количества отсчетов.

Специфическими особенностями обладают словарные, статистические и контекстные методы кодирования информации.

Идея словарных методов кодирования, к которым относятся классические схемы Зива–Лемпела и их многочисленные модификации [2.6, 2.25–2.39], заключается в том, что входные символы и группы символов преобразуются в индексы некоторого словаря, формируемого по заданному алгоритмом закону.

В основе статистических методов моделирования (PPM, CTW, DMC, нейронные сети) лежит принцип оценки и использования условных вероятностей появления символов в сообщении, рассчитываемых путем статистического анализа информационной выборки [2.40–2.53].

Очень интересны контекстные методы ассоциативного кодирования Буяновского [2.54] и Барроуза–Уиллера [2.55], являющиеся оптимальными при кодировании информации, обладающей значительной избыточностью.

4.2.1. Методы представления целых чисел

В большинстве случаев модель источника информации неизвестна, и поэтому выбор и эффективность использования тех или иных методов кодирования входной последовательности элементов алфавита зависит от вероятности их появления.

Основная идея методов представления целых чисел состоит в том, чтобы отдельно хранить порядок значения элементов X_i («экспоненту» E_i) и отдельно —значащие цифры значения («мантиссию» M_i).

Эти методы чаще всего применяются в случаях, когда объем и свойства входных данных заранее неизвестны. Экспонента значений элементов при этом обычно представляется в виде унарного кода $a(n)$, представляющего число n в виде подряд идущих n единиц, заканчивающихся контрольным нулем, либо в виде набора n нулей, за которыми следует контрольная единица (табл. 4.1).

Таблица 4.1. Унарные коды

Код числа	Унарные коды	
0	0	1
1	10	01
2	110	001
3	1110	0001
4	11110	00001
5	111110	000001
...

Гамма- и дельта-коды Элиаса (Elias γ and δ codes) генерируются по правилам, приведенным в табл. 4.2.

Для диапазона значений $[2^k, 2^{k+1} - 1]$ числа n коды формируются следующим образом:

- гамма-код: унарное представление числа k (k нулей и единица), за которым следует двоичное представление (бинарный код) числа $(n - 2^k)$ (в таблице обозначено « x »); длина кода — $2k + 1$ бита;
- дельта-код: гамма-код Элиаса для числа $(k + 1)$ (т. е. количества битов двоичного представления чисел заданного диапазона значений), за которым следует двоичное представление числа $(n - 2^k)$ (в таблице обозначено « x »); длина кода — $2L + k + 1$ бита, где $L = \lceil \log_2(k + 1) \rceil$ — целая часть значения логарифма числа $(k + 1)$ по основанию 2.

Таблица 4.2. Гамма- и дельта-коды Элиаса

Диапазон	Гамма-коды Элиаса	Длина кода, бит	Дельта-коды Элиаса	Длина кода, бит
1	1	1	1	1
2...3	01x	3	010x	4
4...7	001xx	5	011xx	5
8...15	0001xxx	7	00100xxx	8
16...31	00001xxxx	9	00101xxxx	9
32...63	000001xxxxxx	11	00110xxxxxx	10
64...127	0000001xxxxxxx	13	00111xxxxxxx	11
128...255	00000001xxxxxxxx	15	0001000xxxxxxxx	14

Коды Голомба (Golomb codes) — семейство двоичных префиксных кодов представления натуральных чисел (с нулем). Коды различаются одним параметром — величиной m .

Для кодирования символа с номером n необходимо представить этот номер в виде $n = qm + r$, где q и r — целые положительные числа, $0 \leq r < m$.

Кодируемое число разбивается на две независимо кодируемые части: частное и остаток от деления на m : $q = [n/m]$ и $rn - mq$.

Частное q кодируется унарным кодом, а остаток r , представляющий собой число в диапазоне $[0, \dots, m-1]$, кодируется бинарным кодом длиной $\lceil \log_2 m \rceil$. Полученные двоичные последовательности объединяются в результирующее слово.

Например: положим, параметр кода $m = 4$ и кодируемое число $n=13$. Тогда частное от деления кодируемого числа на параметр кода $q = [n/m] = [13/4] = 3$ и унарный код $a(q)a(3) = 1110$. Поскольку остаток от деления кодируемого числа на m равен $rn - m \cdot q = 13 - 4 \cdot 3 = 1$ и представляет собой число в диапазоне $[0, \dots, m-1] = [0, \dots, 3]$, бинарный код $b(r)b(1) = 01$. Следовательно, результирующее кодовое слово равно $a(q)|b(r)a(3)|b(1) = 1110|01$.

Коды Райса (Rice codes) — частный случай кодов Голомба, когда m является степенью двойки.

Коды Райса различаются параметром k , связанным со значением m соотношением $m = 2^k$. Следовательно, если $k = 0$, то $m = 1$, и коды Голомба и Райса соответствуют стандартному унарному коду $a(n)$.

Омега-коды Элиаса и коды Ивен–Роде (Elias ω and Even–Rodeh codes) состоят из последовательности групп длинной $L_1, L_2, L_3, \dots, L_m$ битов, которые начинаются с бита 1, а в конце последовательности следует 0. Примеры генерации омега-кодов Элиаса и кодов Ивен–Роде приведены в табл. 4.3. Символы «xxx» обозначают двоичное представление числа.

Длина каждой следующей $(n+1)$ -й группы задается значением битов предыдущей n -й группы. Значение битов последней группы является итоговым значением всего кода, т. е. всей последовательности групп. В сущности, все первые $m-1$ групп служат лишь для указания длины последней группы.

Таблица 4.3. Омега-коды Элиаса и коды Ивен–Роде

Диапазон	Омега-коды Элиаса	Длина кода, бит	Коды Ивен–Роде	Длина кода, бит
0	—	—	000	3
1	0	1	001	3
2...3	xx 0	3	0xx	3
4...7	10 xxx 0	6	xxx 0	4
8...15	11 xxxx 0	7	100 xxxx 0	8
16...31	10 100 xxxxx 0	11	101 xxxxx 0	9
32...63	10 101 xxxxxx 0	12	110 xxxxxx 0	10
64...127	10 110 xxxxxxx 0	13	111 xxxxxxx 0	11
128...255	10 111 xxxxxxxx 0	14	100 1000 xxxxxxxx 0	16
256...511	11 1000 xxxxxxxxx 0	16	100 1001 xxxxxxxxx 0	17

В омега-кодах Элиаса длина первой группы — 2 бита. Длина следующей группы на единицу больше значения предыдущей. Первое значение (1) задается отдельно. В кодах Ивен–Роде длина первой группы — 3 бита, а длина каждой последующей группы равна значению предыдущей. Первые четыре значения (0–3) заданы особым образом.

Из таблицы видно, что как двоичное представление числа L_m , так и коды длин L_1, \dots, L_{m-1} всегда начинаются с единицы, как было указано выше.

Коды Фибоначчи (Fibonacci codes) основаны на использовании представления любого натурального числа n в виде суммы чисел Фибоначчи f_i ($f_1 = 1$;

$f_2 = 2; f_i = f_{i-1} + f_{i-2}$). Поэтому можно построить код числа как последовательность битов, каждый из которых указывает на факт наличия в n определенного числа Фибоначчи.

Очевидно, если в разложении числа n присутствует f_i , то в этом разложении не может быть числа f_{i+1} . Поэтому логично для конца кода использовать дополнительную единицу. Тогда две идущие подряд единицы будут означать окончание кодирования текущего числа. Примеры кодов Фибоначчи приведены в табл. 4.4. В кодах Фибоначчи все биты, кроме двух последних, являются мантиссами.

Таблица 4.4. Коды Фибоначчи

$n \setminus f_i$	1	2	3	5	8	13	21	34	55
1	1	(1)							
2	0	1	(1)						
3	0	0	1	(1)					
4	1	0	1	(1)					
5	0	0	0	1	(1)				
6	1	0	0	1	(1)				
7	0	1	0	1	(1)				
8	0	0	0	0	1	(1)			
12	1	0	1	0	1	(1)			
13	0	0	0	0	0	1	(1)		
20	0	1	0	1	0	1	(1)		
21	0	0	0	0	0	0	1	(1)	
33	1	0	1	0	1	0	1	(1)	
34	0	0	0	0	0	0	0	1	(1)
54	0	1	0	1	0	1	0	1	(1)

4.2.2. Алгоритм Шеннона–Фано

Первый известный метод эффективного кодирования символов известен как **кодирование Шеннона–Фано** [2.2, 2.12, 2.56]. Он основан на знании вероятности каждого символа, присутствующего в сообщении. Зная эти вероятности, строят таблицу кодов, обладающую следующими свойствами:

- различные коды имеют различное количество битов;
- коды символов, обладающих меньшей вероятностью, имеют больше битов, чем коды символов с большей вероятностью;
- хотя коды имеют различную битовую длину, они могут быть декодированы единственным образом.

Алгоритм основан на рекурсивной процедуре формирования системы префиксных кодов при известном распределении вероятностей появления символов на выходе источника информации.

Все символы информационного алфавита сортируются, например, по убыванию вероятностей их появления. После этого упорядоченный ряд символов разбивается на две части таким образом, чтобы суммы вероятностей появления символов, отнесенных к каждой части, были бы примерно равны. Очевидно, что

при этом количество символов в первой части будет меньше, чем во второй части, поскольку вероятности появления каждого из символов первой части больше (или только части символов) равны вероятностям появления каждого из символов во второй части. Символам первой части присваивается начальный код «0», а символам второй части — «1».

Затем каждая из двух выделенных частей, если она содержит более одного символа, вновь разбивается на две части аналогичным образом. Разделенным на две составляющие первой части присваиваются коды соответственно «00» и «01», а второй части — «10» и «11». Такие операции производятся до тех пор, пока в рядах останется лишь по одному символу. Можно убедиться в том, что длина кода символа b_i с вероятностью появления $P(b_i)$, полученного с применением алгоритма Шеннона–Фано, отличается от оптимального значения $-\log_2 P(b_i)$ менее чем на 1 бит.

Этот алгоритм незначительно уступает по эффективности оптимальному алгоритму Хаффмана префиксного кодирования.

4.2.3. Алгоритм Хаффмана

Алгоритм Хаффмана [2.4, 2.13, 2.56], основанный на элегантной и простой процедуре построения дерева вероятностей, является самым распространенным алгоритмом генерации кода переменной длины.

При реализации этого алгоритма средняя длина кодовых слов \hat{L} находится в диапазоне:

$$H(B) \leq \hat{L} \leq H(B) + 1 \text{ бит/отсчет и } \hat{L} \geq 1 \text{ бит/отсчет,}$$

т. е. средняя длина слов не более чем на 1 бит/отсчет больше энтропии, но не менее 1 бит/отсчет (в предельном случае, когда энтропия равна нулю).

Принцип построения дерева вероятностей можно достаточно просто пояснить на конкретном примере.

Пусть для передачи изображения используется 8 уровней квантования, распределение которых определяется гистограммой со следующими данными:

$$\begin{aligned} P(b_0) = P(b_5) = P(b_6) = P(b_7) &= 0,06; & P(b_1) &= 0,23; \\ P(b_2) &= 0,30; & P(b_3) &= 0,15; & P(b_4) &= 0,08. \end{aligned}$$

Дерево строится справа налево следующим образом (рис. 4.2, верхняя диаграмма):

- в секции I уровни пикселов сортируются по вероятности от наибольшей к наименьшей сверху вниз; при равенстве $P(b_i) = P(b_j)$ выше ставится уровень $b_i < b_j$;
- в секции II две самые нижние ветви объединяются в узел, их вероятности складываются, и узел образует новую ветвь; общее количество ветвей уменьшается на одну и они вновь сортируются по вероятности от наибольшей к наименьшей;
- в секциях III и IV и т. д. производятся операции, аналогичные проводимым в секции II, до тех пор пока не останется одна ветвь с вероятностью, равной 1.

Все это дерево можно перестроить (рис. 4.2, нижняя диаграмма), убрав пересечения.

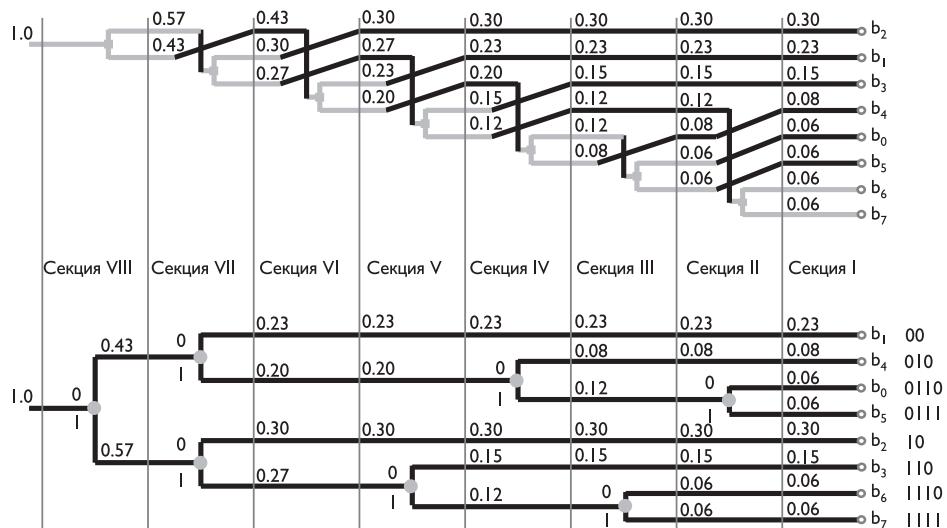


Рис. 4.2. Принцип построения алгоритма хаффмановского кодирования

Кодирование осуществляется движением слева направо по дереву к каждому кодируемому уровню b_i . При этом на каждом узле коду приписывается, например, двоичный «0», если осуществляется шаг вверх и «1», если осуществляется шаг вниз.

Таким образом, для данного случая наиболее вероятные значения b_1 и b_2 кодируются двухбитовым кодом, величины b_3 и b_4 — трехбитовым кодом, а наименее вероятные значения b_0 , b_5 , b_6 и b_7 — четырехбитовым кодом (на рис. 4.2, нижняя диаграмма, коды указаны справа).

Нетрудно понять, что эти коды легко различимы:

- если второй бит кода является двоичным нулем, то код — двухбитовый, в противном случае количество битов в коде более двух;
- если третий бит кода является двоичным нулем, то код — трехбитовый, в противном случае количество битов в коде равно четырем.

Приемник декодирует информацию, используя то же самое дерево, двигаясь вверх при получении «0» и вниз при получении «1». Средняя битовая скорость в данном случае $\hat{L} = 2,71$ бит/пиксель при энтропии $H = 2,68$ бит/пиксель (т. е. \hat{L} практически совпадает с H).

Используются неадаптивный и адаптивный варианты хаффмановского кодирования. В первом случае перед передачей сообщения передается таблица вероятностей, если она заранее неизвестна на приемной стороне. При адаптивном варианте кодирования таблица вероятностей вычисляется как на передающей, так и на приемной стороне по мере поступления данных. При этом до начала кодирования исходно предполагается, например, равновероятное распределение уровней пикселов. Рассмотренный выше пример показывает высокую эффективность хаффмановской процедуры при относительно равномерном распределении уровней пикселов.

Однако когда энтропия сообщения становится существенно меньше единицы, эффективность хаффмановского кодирования резко снижается, поскольку действует упомянутое выше ограничение снизу $\hat{L} \geq 1$ бит/отсчет.

По этой причине при использовании, например, восьмибитовой исходной информации об изображении, хаффмановское кодирование не позволит сжать информацию более восьми раз. В связи с этим применяются различные модификации неравномерного кодирования, позволяющие обеспечить более эффективное сокращение объема передаваемой информации.

Модификация алгоритма **Хаффмана с фиксированной таблицей** (CCITT, Group 3) [2.63] используется при сжатии черно-белых изображений (один бит на пиксел). Данный алгоритм был предложен третьей группой по стандартизации Международного консультационного комитета по телеграфии и телефонии (Consultative Committee International Telegraph and Telephone). Последовательности подряд идущих черных и белых точек в нем заменяются числом, равным их количеству. А этот ряд, в свою очередь, сжимается, по Хаффману, с фиксированной таблицей.

Ниже приведены два вида кодов соответственно:

- коды серий — от 0 до 63 с шагом 1 (табл. 4.5);
- дополнительные коды — от 64 до 2560 с шагом 64 (табл. 4.6).

Таблица 4.5. Таблица кодов серий длиной от 0 до 63 (CCITT, Group 3)

Длина серии	Код белой серии	Код черной серии		Длина серии	Код белой серии	Код черной серии
0	00110101	0000110111		32	00011011	000001101010
1	00111	010		33	00010010	000001101011
2	0111	11		34	00010011	000011010010
3	1000	10		35	00010100	000011010011
4	1011	011		36	00010101	000011010100
5	1100	0011		37	00010110	000011010101
6	1110	0010		38	00010111	000011010110
7	1111	00011		39	00101000	000011010111
8	10011	000101		40	00101001	000001101100
9	10100	000100		41	00101010	000001101101
10	00111	0000100		42	00101011	000011011010
11	01000	0000101		43	00101100	000011011011
12	001000	0000111		44	00101101	000001010100
13	000011	00000100		45	00000100	000001010101
14	110100	00000111		46	00000101	000001010110
15	110101	000011000		47	00001010	000001010111
16	101010	0000010111		48	00001011	000001100100
17	101011	0000011000		49	01010010	000001100101
18	0100111	0000001000		50	01010011	000001010010
19	0001100	00001100111		51	01010100	000001010011
20	0001000	00001101000		52	01010101	000000100100
21	0010111	00001101100		53	00100100	000000110111
22	0000011	00000110111		54	00100101	000000111000

Таблица 4.5 (окончание)

Длина серии	Код белой серии	Код черной серии		Длина серии	Код белой серии	Код черной серии
23	0000100	00000101000		55	01011000	000000100111
24	0101000	00000010111		56	01011001	000000101000
25	0101011	00000011000		57	01011010	0000001011000
26	0010011	000011001010		58	01011011	0000001011001
27	0100100	000011001011		59	01001010	000000101011
28	0011000	000011001100		60	01001011	000000101100
29	00000010	000011001101		61	00110010	0000001011010
30	00000011	000001101000		62	00110011	0000001100110
31	00011010	000001101001		63	00110100	0000001100111

Таблица 4.6. Таблица кодов серий длиной от 0 до 63 (CCITT, Group 3)

Длина серии	Код белой серии	Код черной серии		Длина серии	Код белой серии	Код черной серии
64	11011	0000001111		1344	011011010	00000001010011
128	10010	000011001000		1408	011011011	00000001010100
192	01011	000011001001		1472	010011000	00000001010101
256	0110111	000001011011		1536	010011001	00000001011010
320	00110110	000000110011		1600	010011010	00000001011011
384	00110111	000000110100		1664	011000	00000001100100
448	01100100	000000110101		1728	010011011	00000001100101
512	01100101	0000001101100		1792	000000001000	совп. с белой
576	01101000	0000001101101		1856	000000001100	— // —
640	01100111	0000001001010		1920	000000001101	— // —
704	011001100	0000001001011		1984	0000000010010	— // —
768	011001101	0000001001100		2048	0000000010011	— // —
832	011010010	0000001001101		2112	0000000010100	— // —
896	011010011	0000001110010		2176	0000000010101	— // —
960	011010100	0000001110011		2240	0000000010110	— // —
1024	011010101	0000001110100		2304	0000000010111	— // —
1088	011010110	0000001110101		2368	0000000011100	— // —
1152	011010111	0000001110110		2432	0000000011101	— // —
1216	011011000	0000001110111		2496	0000000011110	— // —
1280	011011001	0000001010010		2560	0000000011111	

Каждая строка изображения сжимается независимо. Считается, что в изображениях существенно преобладает белый цвет, и все строки изображения начинаются с белой точки. Если строка начинается с черной точки, то считается, что строка начинается белой серией с длиной 0. На практике в тех случаях, когда в изображении преобладает черный цвет, изображение инвертируется перед компрессией и записывается эта информация в заголовке файла.

Близким к Хаффмановскому является **алгоритм JBIG**, разработанный группой экспертов ISO (Joint Bi-level Experts Group) специально для сжатия однобитных черно-белых изображений, например, факсов или отсканированных

документов [2.64]. В принципе он может применяться и к 2-, и к 4-битовым картинкам. При этом алгоритм разбивает их на отдельные битовые плоскости. JBIG позволяет управлять такими параметрами, как порядок разбиения изображения на битовые плоскости, ширина полос в изображении, уровни масштабирования. Последняя возможность позволяет легко ориентироваться в базе больших по размерам изображений, просматривая сначала их уменьшенные копии. Распаковываться изображение на экране будет постепенно, как бы медленно «проявляясь».

Как и алгоритм Хаффмана, JBIG использует для чаще появляющихся символов короткие цепочки, а для реже появляющихся — длинные.

4.2.4. Блочное и условное кодирование

Дальнейшее снижение битового потока возможно, если использовать **блочное кодирование** [2.4], т. е. если кодировать и передавать информацию блоками из N отсчетов, которые можно представить как компоненты вектора:

$$\mathbf{b} = (b_1, b_2, \dots, b_N).$$

При квантовании этих компонент K уровнями имеем 2^{NK} возможных значений случайного вектора \mathbf{B} , чьи возможные значения \mathbf{b} имеют распределение вероятностей $\{P(\mathbf{b})\}$. Энтропия N -го порядка (блочная энтропия) определяется как

$$H(\mathbf{B}) = - \sum_b P(\mathbf{b}) \cdot \log_2 P(\mathbf{b}), \quad (4.3)$$

где сумма берется по всем 2^{NK} возможным значениям.

Очевидно, что энтропия вектора $H(\mathbf{B})$ связана с энтропией каждой его компоненты $H(B)$ соотношением

$$\frac{1}{N} H(\mathbf{B}) \leq H(B),$$

причем равенство достигается лишь в том случае, когда все компоненты вектора (пиксели) статистически независимы.

Построим хаффмановские коды для $\{\mathbf{b}\}$ в соответствии с распределением вероятностей $\{P(\mathbf{b})\}$ (всего 2^{NK} кодовых слов). При этом если среднюю длину кодового слова, определяющего вектор \mathbf{B} , обозначить \hat{L}_N , то средняя длина кодового слова, определяющего его компоненту, $\hat{L} = \frac{1}{N} \hat{L}_N$, может быть ограничена следующим соотношением:

$$\frac{1}{N} H(\mathbf{B}) \leq \frac{1}{N} \hat{L}_N = \hat{L} \leq \frac{1}{N} H(\mathbf{B}) + \frac{1}{N} \leq H(B) + \frac{1}{N}. \quad (4.4)$$

Таким образом, при достаточно больших N возможно (хотя и не всегда практически) из-за большого объема вычислений) кодировать поблочно и получать битовый поток, максимально отличающийся от энтропии первого порядка $H(B)$ на малую величину $\frac{1}{N}$.

Существенным недостатком блочного кодирования является необходимость создания таблицы кодирования (или дерева), содержащей 2^{NK} кодовых слов. При этом часто получаются весьма длинные коды.

Эта проблема частично преодолевается при так называемом **условном кодировании** [2.4], когда кодируется компонента b_N при условии, что компоненты b_1, b_2, \dots, b_{N-1} уже переданы и известны приемнику. В этом случае коды Хаффмана для b_N можно построить на основе условной вероятности $\{P(b_N | b_1, b_2, \dots, b_{N-1})\}$.

Очевидно, что такой код будет иметь 2^N слов, и средняя длина слов в битах будет не более чем на 1 бит отличаться от условной энтропии:

$$H(B_N | b_1, \dots, b_{N-1}) = - \sum_{b_N} P(b_N | b_1, \dots, b_{N-1}) \cdot \log_2 P(b_N | b_1, \dots, b_{N-1}). \quad (4.5)$$

Если такой код построен для всех возможных $(N - 1)$ -мерных значений, то получается $2^{(N-1)K}$ кодов, каждый из которых имеет 2^K кодовых слов.

Результирующая условная энтропия может быть получена усреднением по всем $(N - 1)$ -мерным векторам:

$$\begin{aligned} H(B_N | B_1, \dots, B_{N-1}) &= - \sum_{b_1 \dots b_{N-1}} P(b_N | b_1, \dots, b_{N-1}) \cdot H(B_N | b_1, \dots, b_{N-1}) = \\ &= - \sum_b P(b) \cdot \log_2 P(b_N | b_1, \dots, b_{N-1}). \end{aligned} \quad (4.6)$$

При этом средняя длина потока битов \hat{L}_c оценивается с точностью 1 бит/отсчет:

$$H(B_N | B_1, \dots, B_{N-1}) \leq \hat{L}_c \leq H(B_N | B_1, \dots, B_{N-1}) + 1 \text{ бит, но } \hat{L}_c \geq 1. \quad (4.7)$$

Эффективность условного кодирования связана с тем, что при значительной зависимости между отсчетами условная энтропия всегда меньше отнесенной на отсчет блоковой энтропии, т. е.

$$H(B_N | B_1, \dots, B_{N-1}) \leq \frac{1}{N} H(\mathbf{B}). \quad (4.8)$$

Однако в связи с тем, что \hat{L}_c оценена снизу битом на отсчет, условное кодирование неэффективно при значениях условной энтропии, меньшей единицы.

Поэтому более эффективным является так называемое **условное блочное кодирование**, сочетающее преимущества двух модификаций кодирования.

К методам блочного кодирования можно отнести **алгоритм группового кодирования RLE (Run Length Encoding)** [2.6, 2.9], который позволяет сжимать данные любых типов независимо от содержащейся в них информации. Сама информация влияет лишь на коэффициент сжатия. Он является самым быстрым и простым алгоритмом компрессии и при этом иногда оказывается весьма эффективным.

В основу алгоритма RLE положен **принцип выявления повторяющихся последовательностей данных и замены их простой структурой**, в которой указывается код данных и коэффициент повтора. Он оперирует группами данных — последовательностями, в которых один и тот же символ встречается несколько раз подряд. Суть его заключается в том, что при кодировании строка повторяющихся символов, составляющих группу, заменяется строкой, которая содержит сам повторяющийся символ (значение группы) и количество его повторов (счетчик группы).

Программные реализации алгоритма RLE отличаются простотой, высокой скоростью работы, но в среднем обеспечивают недостаточное сжатие. Наилучшими объектами для данного алгоритма являются графические файлы, в которых большие одноцветные участки изображения кодируются длинными последовательностями одинаковых байтов. Этот метод также может давать заметный выигрыш на некоторых типах файлов баз данных, имеющих таблицы с фик-

сированной длиной полей. Для текстовых данных методы RLE, как правило, неэффективны.

Из специфики работы алгоритма можно легко понять, для каких изображений коэффициент сжатия будет максимальным. Это изображения, содержащие минимальное количество цветов, без плавных переходов (с резкими границами) и с большими участками, заполненными одним цветом. Так как сканирование данных осуществляется построчно слева на право, высокий коэффициент сжатия будет достигаться на изображениях, строки которых содержат достаточно длинные цепочки одинаковых пикселов.

4.2.5. Арифметическое кодирование

Арифметическое кодирование во многих отношениях превосходит префиксные методы и теоретически обеспечивает среднюю битовую скорость передачи данных равной энтропии.

Арифметическое кодирование позволяет представить информацию более компактно, не требует ее разбиения на блоки для оптимизации сжатия, легко и эффективно использует адаптивную модель кодирования.

Основная концепция арифметического кодирования принадлежит П. Элайесу [2.56, 2.57] и относится к началу 60-х годов прошлого века. Дальнейшие исследования были направлены на практическую реализацию алгоритма [2.14–2.24].

Несмотря на объективную сложность нестандартного подхода, идея, лежащая в основе арифметического кодирования, весьма проста. Принцип этого метода можно пояснить следующим образом.

До начала кодирования сообщению ставится в соответствие интервал $(0, 1]$. Каждому символу из множества N символов, присутствующих в сообщении, на интервале $(0, 1]$ отводится интервал, ширина которого равна вероятности появления символа. Если распределение уровней квантования пикселов изображения соответствует гистограмме, аналогичной той, что приведена в качестве примера при изучении хаффмановского кодирования, то указанный интервал $(0, 1]$ разбивается так, как показано на рис. 4.3 слева.

При неадаптивной модели кодирования разбиение интервала $(0, 1]$ на диапазоны, соответствующие различным символам, происходит до начала кодирования, и информация передается декодеру. При адаптивном кодировании после кодирования каждого символа уточняется таблица вероятности символов и корректируются соответствующие им интервалы как на передающей, так и на приемной стороне.

Пусть по мере сканирования изображения первым из N символом, поступающим на кодер, является символ b_1 . Тогда на первом этапе кодирования интервал $(0, 1]$ преобразуется в новый интервал, границами которого являются границы интервала $P(b_1)$ на шкале $(0, 1]$.

Диапазон новой шкалы равен $r_1 P(b_1)$, его минимальное значение равно $l_1 P(b_0)$, а максимальное $h_1 P(b_0) + P(b_1)$.

Эта шкала также разбивается на части, пропорциональные вероятностям сообщений $P(b_i)$. Значения шкалы в n -й точке

$$g_1(n) = l_1 + r_1 \cdot \sum_{i=0}^{n-1} P(b_i), \quad (4.9)$$

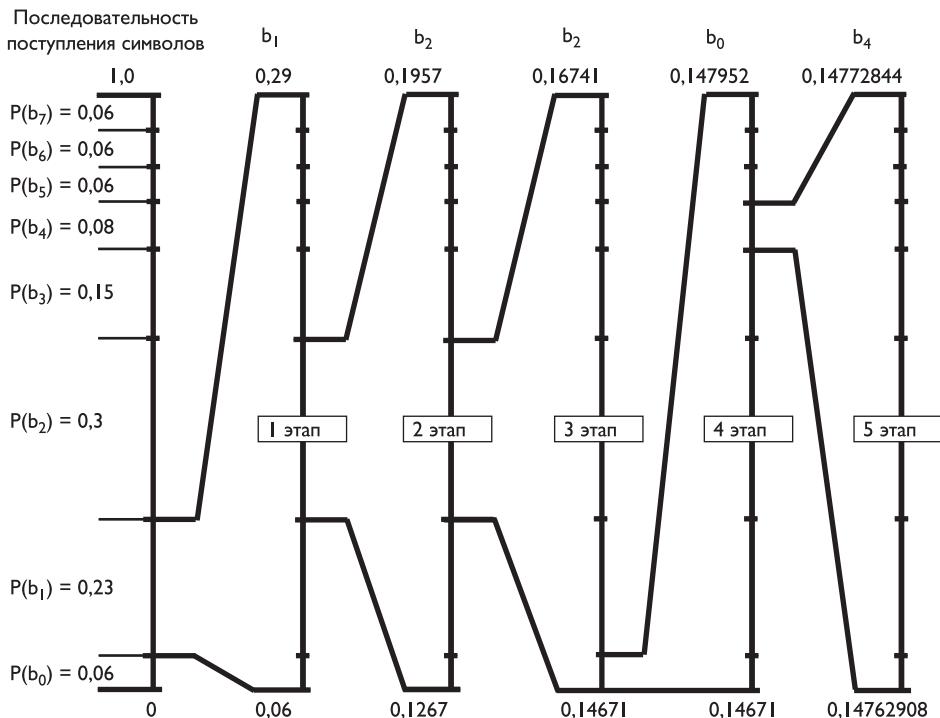


Рис. 4.3. Принцип обработки информации при арифметическом кодировании

причем $g_1(0) = l_1$, а $g_1(N)h_1$. Если на втором этапе на кодер поступает символ b_2 , то шкала вновь преобразуется, как показано на рис. 4.3. Диапазон новой шкалы равен $r_2r_1 \cdot P(b_2)$, его минимальное значение равно

$$l_2 = l_1 + r_1 \cdot \sum_{i=0}^1 P(b_i), \quad (4.10)$$

а максимальное значение —

$$h_2 = l_1 + r_1 \cdot \sum_{i=0}^2 P(b_i). \quad (4.11)$$

Значение шкалы в n -й точке

$$g_2(n) = l_2 + r_2 \cdot \sum_{i=0}^{n-1} P(b_i). \quad (4.12)$$

Таким образом, последовательно шкала по мере поступления данных уменьшается. Поступающие символы сообщения уменьшают размер интервала в соответствии с моделью вероятности символов. Более часто встречающиеся символы меньше суживают интервал, чем редкие символы, и, следовательно, добавляют меньше битов в код интервала.

На k -м этапе при поступлении на кодер n -го символа шкала итеративно изменяется таким образом, что ее диапазон равен $r_k r_{k-1} \cdot P(b_n)$. Его минимальное

значение равно $l_k = l_{k-1} + r_{k-1} \cdot \sum_{i=0}^{n-1} P(b_i)$, максимальное значение — $h_k = l_{k-1} + r_{k-1} \cdot \sum_{i=0}^n P(b_i)$, а значение шкалы p -й точки — $g_k(p) = l_k + r_k \cdot \sum_{i=0}^{p-1} P(b_i)$.

По окончании цикла кодирования формируется такое число g , что выполняется неравенство $l_k < g \leq h_k$. Это число g характеризует передаваемую последовательность символов, которая может быть однозначно восстановлена в приемнике.

Как показано на рис. 4.3, по окончании 5-го этапа число g должно быть $0,14762908 < g \leq 0,14772844$.

Если на приемной стороне сформирована исходная шкала вероятностей на интервале $(0, 1]$, то это число g попадает на интервал, характеризующий символ b_1 , поступивший в кодер на первом этапе.

Зная, что на первом этапе на кодер поступил символ b_1 , можно рассчитать новую шкалу, соответствующую шкале кодера на 1-м этапе. В этой шкале число g попадает на интервал, характеризующий символ b_2 , поступивший в кодер на втором этапе. Таким образом, последовательно могут быть определены все символы, поступающие на кодер (в данном случае — последовательность b_1, b_2, b_3, b_0, b_4).

Трудности в реализации приведенного алгоритма арифметического кодирования связаны с хранением чисел с большим количеством знаков после запятой и операциями над такими числами. Поэтому результатом дальнейших исследований стало использование более эффективных реализаций арифметического кодирования и специальных приемов повышения производительности вычислений, что позволило избежать сложных операций с плавающей точкой и существенно упростить хранение данных. Алгоритм работает быстрее, если удается оперировать двоичными дробями, задаваемыми последовательностью $a_1/2^1 + a_2/2^2 + \dots + a_3/2^3 + a_4/2^4 + \dots + a_k/2^k$. При этом на каждом этапе вычислений необходимо дописывать дополнительные последовательности дробей, пока получившееся число не попадет в интервал, соответствующий закодированной цепочке. Чтобы избежать ограничений, связанных с используемой разрядностью переменных, реальный алгоритм работает с целыми числами и оперирует с дробями, числитель и знаменатель которых являются целыми числами (например, знаменатель равен $2^{16} = 65\,536$). При этом с потерей точности можно бороться, отслеживая сближение величин l_k и h_k и умножая числитель и знаменатель дроби на целое число, обычно на 2^n .

4.2.6. Словарные методы кодирования дискретной информации

Словарные методы энтропийного кодирования вместо вероятностного используют так называемый комбинаторный подход, кодовые схемы которого используют коды только тех информационных последовательностей, которые реально порождаются информационным источником. Словарные модели опираются на информационную структуру, реализуемую словарем. Словарь включает в себя части уже обработанной информации, на основе которых осуществляется кодирование. Составляющие последовательности символов информационного источника кодируются посредством ссылок на идентичные им элементы словаря (совпадения).

Словарные методы отличаются друг от друга способом организации словаря, схемой поиска совпадений и видом ссылки на найденное совпадение [2.6, 2.9]. Впервые эти методы были описаны в работах А. Лемпела (Lempel A.) и Я. Зива (Ziv J.) [2.25, 2.26].

Первый вариант словарного алгоритма был описан в 1977 году [2.25] и был назван LZ77 по первым буквам фамилий авторов.

Классический алгоритм LZ-77 является родоначальником целого семейства схем, использующих алгоритмы со скользящим словарем или со скользящим окном. В скользящем окне помещается N символов, причем часть из них $MN - n$ — уже закодированные символы, являющиеся словарем. Оставшаяся часть символов длины n существенно меньше количества символов, записанных в словаре, используется в буфере предварительного просмотра. Предположим, к текущему моменту закодировано m символов: $s_0, s_1, s_2, \dots, s_{m-1}$, часть из которых $s_{m-M}, \dots, s_{m-2}, s_{m-1}$ записаны в словаре, а в буфере записаны ожидающие кодирования символы $s_m, s_{m+1}, \dots, s_{m-1+n}$.

Принцип кодирования заключается в поиске самого длинного совпадения последовательности символов в строке буфера, начинающегося с символа s_m , и закодированных фраз, размещенных в словаре. Эти фразы могут начинаться с любого символа $s_{m-M}, \dots, s_{m-2}, s_{m-1}$ и выходить за пределы словаря, вторгаясь в область буфера.

Полученная в результате поиска фраза $s_{m-i}, s_{m-i+1}, \dots, s_{m-i+j-1}$ кодируется с помощью двух чисел: i — смещения от начала буфера и j — длины совпадения последовательности символов. Дополнительно в выходной поток записывается символ s , непосредственно следующий за совпавшей строкой буфера.

Пример обработки информации в соответствии с алгоритмом LZ-77 приведен в табл. 4.7. Обработка подвергается фраза из 59 символов:

«на_дворе_трава_на_траве_дрова_на_дворе_трава_на_траве_дрова».

Вместо текстовой фразы могут использоваться наборы однородных элементов, пикселов, отсчетов или качественно разных объектов.

Установим длину буфера равной восьми символам, а размер словаря — большие длины кодируемой фразы.

Положим, для представления символов требуется использовать 1 байт. Следовательно, исходно фраза занимала $59 \cdot 8 = 472$ бита. Как следует из таблицы, при независимом кодировании значений коэффициентов i и j требуется использовать соответственно 5 битов и 4 бита. Тогда для представления кодированной информации требуется $(5 + 4 + 8) \cdot 20 = 340$ битов.

Если же для кодирования коэффициентов i и j использовать наиболее простой алгоритм префиксного кодирования Шеннона–Фано, то можно подсчитать, что средняя длина кода для кодирования каждого коэффициента i составит $26/20$ бита, а каждого коэффициента j — $22/20$ бита. При этом для представления кодированной информации требуется $(26/20 + 22/20 + 8) \cdot 20 = 208$ битов.

Как следует из табл. 4.7 алгоритм LZ-77 позволяет эффективно сжимать только сравнительно длинные последовательности, в которых имеет место повторение фраз.

Существует множество модификаций этого алгоритма. В частности, к алгоритмам семейства LZ-77 также относятся алгоритмы LZR [2.27], LZSS [2.28, 2.29].

Таблица 4.7. Пример работы алгоритма LZ-77

Шаг	Скользящее окно		Совпадающая фраза	Закодированные данные		
	Словарь	Буфер		<i>i</i>	<i>j</i>	<i>s</i>
1	—	на_дворе	—	1	0	«н»
2	н	а_дворе_	—	1	0	«а»
3	на	дворе_т	—	1	0	«_»
4	на_	дворе_тр	—	1	0	«д»
5	на_д	воре_тра	—	1	0	«в»
6	на_дв	оре_трав	—	1	0	«о»
7	на_дво	ре_трава	—	1	0	«р»
8	на_двор	е_трава_	—	1	0	«е»
9	на_дворе	трава_н	—	6	1	«т»
10	на_дворе_т	рava_на_	р	4	1	«а»
11	на_дворе_тра	ва_на_тр	в	8	1	«а»
12	на_дворе_трава	_на_трав	—	6	1	«н»
13	на_дворе_трава_н	а_траве_	а	15	2	«т»
14	на_дворе_трава_на_т	раве_дро	рав	9	3	«е»
15	на_дворе_трава_на_траве	_дро_ва_н	-д	21	2	«р»
16	на_дворе_трава_на_траве_др	ова_на_д	о	21	1	«в»
17	на_дворе_трава_на_траве_дро	а_на_дво	а_на_	15	5	«д»
18	на_дворе_трава_на_траве_дро_на_д	воре_тра	воре_тра	30	8	«в»
19	..._на_траве_дро_на_дворе_трав	а_на_тра	а_на_тра	30	8	«в»
20	..._дро_на_дворе_трава_на_трав	е_дро_ва	е_дро_ва	30	7	«end»

Алгоритмы LZB [2.30], LZH [2.31] представляют собой усовершенствования алгоритма LZSS с помощью использования кодов переменной длины.

Наиболее широкое распространение в компьютерных программах архивации получили в настоящее время так называемые комбинированные словарные алгоритмы семейства LZ-77, в которых статистический алгоритм устраниет недостатки способа кодирования базового словарного алгоритма.

Классический алгоритм LZ-78 [2.26], разработанный А. Лемпелом и Я. Зивом в 1978 году, положил начало другому семейству словарных алгоритмов. Алгоритм LZ-78 нередко уступает LZ-77 по эффективности кодирования, но превосходит его по скорости.

Алгоритм LZ-78 не использует скользящее окно, а в словарь помещают только «перспективные» с точки зрения вероятности последующего использования строки. При этом на каждом шаге в словарь вставляется новая фраза, пред-

ставляющая собой сцепление одной из фраз словаря (имеющей самое длинное совпадение со строкой буфера) и символа s , следующего за строкой буфера, за которой найдена совпадающая фраза. В отличие от алгоритма LZ-77 в словаре не может быть одинаковых фраз.

Кодирующая схема порождает последовательность кодов фраз. Каждый код состоит из индекса (префикса) n «родительской» фразы и символа s .

Пример обработки информации в соответствии с алгоритмом LZ-78 приведен в табл. 4.8. Обработка подвергается та же последовательность из 59 символов, которая использовалась при пояснении работы алгоритма LZ-77. В начале обработки информации словарь пуст, номером «1» задается пустая фраза словаря.

Таблица 4.8. Пример работы алгоритма LZ-78

Шаг	Добавляемая в словарь фраза		Буфер	Совпадающая фраза	Закодированные данные	
	Фраза	Ее номер			n	s
1	н	2	на_дворе	—	1	«н»
2	а	3	а_дворе_	—	1	«а»
3	—	4	дворе_т	—	1	«_»
4	д	5	дворе_тр	—	1	«д»
5	в	6	воре_тра	—	1	«в»
6	о	7	оре_трав	—	1	«о»
7	р	8	ре_трава	—	1	«р»
8	е	9	е_трава_	—	1	«е»
9	т	10	трава_н	—	4	«т»
10	ра	11	рава_на_	р	8	«а»
11	ва	12	ва_на_т	в	6	«а»
12	н	13	на_трав	—	4	«н»
13	а_	14	а_траве_	а	3	«_»
14	т	15	траве_др	—	1	«т»
15	рав	16	раве_дро	ра	11	«в»
16	е_	17	е_дро_ва_	е	9	«_»
17	др	18	дро_ва_на	д	5	«р»
18	ов	19	ова_на_д	о	7	«в»
19	а_н	20	а_на_дво	а_	14	«н»
20	а_д	21	а_дворе_	а_	14	«д»
21	во	22	воре_тра	в	6	«о»
22	ре	23	ре_трава	р	8	«е»
23	т	24	трава_н	т	10	«т»
24	ав	25	ава_на_т	а	3	«в»
25	а_на	26	а_на_тра	а_н	20	«а»
26	т	27	траве_д	т	24	«а»
27	в	28	в_дро_ва	в	6	«е»
28	д	29	дро_ва	—	4	«д»
29	ро	30	ро_ва	р	8	«о»
30	ва	31	ва	ва	12	«end»

Если для представления символов требуется использовать 1 байт, то исходно фраза занимает $59 \cdot 8 = 472$ бита. Как следует из таблицы, при независимом кодировании значений префикса n требуется использовать 5 битов. Тогда для представления кодированной информации требуется $(5 + 8) \cdot 30 = 390$ битов.

Если же для кодирования префикса n использовать алгоритм Шеннона–Фано, то можно подсчитать, что средняя длина кода для кодирования каждого коэффициента n составит $65/30$ бита. При этом для представления кодированной информации требуется $(65/30 + 8) \cdot 30 = 305$ битов.

Как следует из сравнения классических алгоритмов LZ-77 и LZ-78, последний в данном случае оказался менее эффективным. Однако алгоритм LZ-78 также имеет множество модификаций, позволяющих существенно повысить эффективность кодирования информации.

К этому семейству относятся алгоритмы LZW [2.32], LZC [2.33], LZT [2.34], LZMW [2.35], LZJ [2.36], LZFG [2.37].

Алгоритм LZW, реализованный Т. Уэлчем (Welch T.), получил широкое применение в компрессии графических компьютерных изображений (форматы GIF, TIFF) и для передачи по линиям связи (стандарт V.42bis [2.59]).

В этой модификации LZ-78 все символы алфавита входной последовательности предварительно заносятся в словарь, и результатом работы алгоритма является только выявление последовательности фраз словаря.

Пример обработки информации в соответствии с алгоритмом LZW приведен в табл. 4.9. Обработка подвергается та же последовательность из 59 символов, которая использовалась при пояснении работы алгоритмов LZ-77 и LZ-78. При этом предполагается, что исходно все символы входной последовательности, размер которых равен 8 битов (соответствующие числам от 0 до 255), занесены в словарь. Выявленные в процессе кодирования и вносимые в словарь фразы обозначим числами, начиная с 246. Ограничиваая 9 битами закодированную информацию, обозначим код очистки словаря, например, числом 511.

Таблица 4.9. Пример работы алгоритма LZW

Шаг	Входная строка	Закодированные данные	Фраза в словарь	Код
1	на_дворе_трава_на_траве_дрова_на_дворе_...	«н»	на	256
2	а_дворе_трава_на_траве_дрова_на_дворе_т...	«а»	а_	257
3	_дворе_трава_на_траве_дрова_на_дворе_т...	«_»	_д	258
4	дворе_трава_на_траве_дрова_на_дворе_т...	«д»	дв	259

Таблица 4.9 (продолжение)

Шаг	Входная строка	Закодированные данные	Фраза в словарь	Код
5	воре_трава_на_траве_ дро ¹ ва_на_дворе_трава...	«B»	во	260
6	опе_трава_на_траве_дрова_ на_дворе_трава_...	«O»	ор	261
7	ре_трава_на_траве_ дрова_на_дворе_трава_...	«P»	ре	262
8	е_трава_на_траве_дрова_ на_дворе_трава_...	«e»	е_	263
9	_трава_на_траве_дрова_ на_дворе_трава_н...	«_»	_т	264
10	трава_на_траве_дрова_на_ дворе_трава_на...	«T»	тр	265
11	рава_на_траве_дрова_на_ дворе_трава_на_...	«P»	ра	266
12	ава_на_траве_дрова_на_ дворе_трава_на_т...	«a»	ав	267
13	ва_на_траве_дрова_на_ дворе_трава_на_тра...	«B»	ва	268
14	а_на_траве_дрова_на_ дво ² ре_трава_на_трав...	«257» (а_)	а_н	269
15	на_траве_дрова_на_дворе_ трава_на_траве_дрова	«256» (на)	на_	270
16	_траве_дрова_на_дворе_ трава_на_траве_дрова	«264» (_т)	_тр	271
17	раве_дрова_на_дворе_трава_на_траве_дрова	«266» (па)	рав	272
18	ве_дрова_на_дворе_трава_на_траве_дрова	«B»	ве	273
19	е_дрова_на_дворе_трава_на_траве_дрова	«263» (е_)	е_д	274

Таблица 4.9 (окончание)

Шаг	Входная строка	Закодированные данные	Фраза в словарь	Код
20	дрова _на_дворе_трава _на_траве_дрова	«д»	др	275
21	рова _на_дворе_трава _на_траве_дрова	«р»	ро	276
22	ова _на_дворе_трава _на_траве_дрова	«о»	ов	277
23	ва _на_дворе_трава _на_траве_дрова	«268» (ва)	ва _	278
24	_на_дворе_трава _на_траве_дрова	«_»	_н	279
25	ра_дворе_трава _на_траве_дрова	«270» (на_)	на_д	280
26	дворе_трава _на_траве_дрова	«259» (дв)	дво	281
27	оре_трава _на_траве_дрова	«261» (ор)	оре	282
28	е_трава _на_траве_дрова	«263» (е_)	е_т	283
29	трава _на_траве_дрова	«265» (тр)	тра	284
30	ава _на_траве_дрова	«267» (ав)	ава	285
31	а _на_траве_дрова	«269» (а_н)	а_на	286
32	а_траве_дрова	«257» (а_)	а_т	287
33	траве_дрова	«284» (тра)	трав	288
34	ве_дрова	«273» (ве)	ве_	289
35	_дрова	«258» (_д)	_др	290
36	рова	«276»(ро)	ров	291
37	ва	«268» (ва)		«end»

Так как для представления символов требуется использовать 1 байт, то исходно фраза занимает $59 \times 8 = 472$ бита. Как следует из таблицы, при независимом кодировании значений закодированных данных требуется использовать 9 битов. Тогда для представления кодированной информации требуется $9 \times 37 = 333$ бита.

Если же для кодирования данных использовать алгоритм Шеннона-Фано, то можно подсчитать, что средняя длина кода для кодирования каждого коэффициента n составит $119/37$ бита и для представления кодированной информации требуется всего 119 битов.

4.2.7. Статистические методы моделирования дискретной информации

Для эффективного кодирования требуется знание характеристик источника информации, в частности, условные вероятности появления символов в зависимости от предшествующих символьных последовательностей. Если не имеется никакой дополнительной информации об источнике данных, то эти условные вероятности определяются с помощью статистического анализа информационных выборок. Такой принцип лежит в основе статистических методов, обладающих высокой эффективностью, но предъявляющих повышенные требования к вычислительным ресурсам.

Статистические методы позволяют оценить вероятность появления в текущем контексте (последовательность символов, непосредственно предшествующих данному символу в информационном сообщении) символа или последовательности символов. Оценка вероятности определяет длину кода, который генерируется, как правило, с помощью префиксного или арифметического кодирования. Чаще

всего используются адаптивные статистические методы. Они различаются способом получения оценок условных вероятностей появления символов в различных контекстах. В настоящее время наиболее распространены четыре основных способа: PPM, DMC, CTW и метод нейронных сетей.

В методе PPM (Prediction by Partial Matching — предсказание по частично-му совпадению), предложенном Дж. Клири (Cleary J.) и И. Уиттеном (Witten I.) и применяемом чаще всего при обработке текстовой информации, вероятность появления символа всегда оценивается в текущем контексте какого-то одного порядка. Порядком контекста называется длина последовательности символов в контексте.

Имеется множество реализаций метода PPM [2.6, 2.9, 2.40–2.53, 2.90–2.93], которые, как правило, работают по приведенной ниже схеме.

Сначала для оценки вероятности появления символа используется контекст достаточно высокого порядка. Если кодируемый символ ранее в выбранном контексте не встречался, то кодирующее устройство генерирует код служебного символа перехода к контекстной модели следующего по убыванию порядка. Если новая модель также не обеспечивает получение вероятностной оценки, вновь генерируется служебный символ перехода к следующему меньшему по порядку контексту. Для гарантии завершения процесса переходов вводится дополнительная модель «минус 1-го» порядка, при котором появление всех символов информационного алфавита оценивается с равной вероятностью.

Предположим, что к текущему моменту передана последовательность из n символов источника информации $\bar{x}_1^n = (x_1, \dots, x_n)$ и следует передать символ x_{n+1} .

Для всех возможных значений символа x_{n+1} вычисляются условные вероятности появления этого символа при уже известном контексте, определяемом последовательностью $\bar{x}_{n-d+1}^n = (x_{n-d+1}, \dots, x_n)$ наибольшей длины d , не превышающей заданного значения D . При этом выбираемая последовательность \bar{x}_{n-d+1}^n такова, что последовательность символов $\bar{x}_{n-d+1}^{n+1} = (x_{n-d+1}, \dots, x_n, x_{n+1})$ уже встречалась в переданной информации \bar{x}_1^n .

Значение символа x_{n+1} кодируется арифметическим кодом в соответствии с вычисленным условным распределением вероятностей и с учетом длины контекста $d \leq D$.

Оценивается условная вероятность $\hat{p}(a|\bar{c})$ появления символа $x_{n+1}a$ после контекста $\bar{x}_{n-d+1}^n = (x_{n-d+1}, \dots, x_n) = \bar{c}$. В случае, если эта условная вероятность оказывается равной нулю, $\hat{p}(a|\bar{c}) = 0$, используется символ *esc* с вычисленной определенным образом вероятностью его появления, и длина контекста d сокращается на единицу. Рекурсия продолжается до тех пор, пока символ не будет передан, либо длина контекста не окажется равной нулю. Это соответствует независимому кодированию символа x_{n+1} в соответствии с вычисленным по \bar{x}_1^n безусловным распределением вероятностей символов всего возможного алфавита X .

В том случае, когда символ x_{n+1} еще не встречался в последовательности \bar{x}_1^n , вновь передается символ *esc*, а также символ x_{n+1} в соответствии с равномерным распределением вероятностей на множестве неиспользованных символов алфавита источника.

Резюмируя вышеизложенное, можно сказать, что кодер PPM обеспечивает передачу очередного символа в соответствии с вероятностью, определяемой кон-

текстом наиболее возможной длины. Если же этот символ еще не встречался с таким контекстом, передается символ *esc*, контекст укорачивается на один символ. В конечном счете символ будет передан с учетом контекста меньшей или нулевой длины, либо как «новый символ», если он не встречался в уже закодированной последовательности.

Различные реализации алгоритма кодирования РРМ связаны с принципами оценки условных вероятностей появления символов алфавита источника и *esc*-символа.

Так, в алгоритме РРМА [2.40] используются следующие выражения:

$$\begin{aligned}\hat{p}(a|\bar{c}) &= \frac{N_n(\bar{c}, a)}{N_n(\bar{c}) + 1}, \quad N_n(\bar{c}, a) > 0, \\ \hat{p}(esc|\bar{c}) &= \frac{1}{N_n(\bar{c}) + 1}, \quad N_n(\bar{c}, a) = 0,\end{aligned}\tag{4.13}$$

где $N_n(\bar{x})$ — количество появлений последовательности \bar{x} в обработанной последовательности длины n .

При расчете величины условной вероятности $\hat{p}(a|\bar{c})$ дополнительно введено так называемое «правило исключений», принцип использования которого заключается в следующем.

Предположим, что при кодировании информации на некотором шаге контекст на интервале d имеет вид $\bar{c}_d = (x_{n-d+1}, \dots, x_n)$, а следующий за ним символ a , причем сочетание $\bar{c}_d, a = (x_{n-d+1}, \dots, x_n, a)$ ранее не встречалось. Предположим также, что предыдущие появления последовательности \bar{c}_d имели своими продолжениями символы b и c . В соответствии с алгоритмом передается символ *esc*, и контекст укорачивается на один символ. Следовательно, теперь контекстом будет $\bar{c}_{d-1} = (x_{n-d+2}, \dots, x_n)$.

Правило исключений требует принять во внимание тот факт, что не все сочетания \bar{c}_{d-1} следует использовать для подсчета условной вероятности появления символа a , поскольку благодаря передаче символа *esc* кодер и декодер уже знают, что символы b и c , например, не могут следовать за контекстом \bar{c}_{d-1} . По этой причине формулы (4.13) следует уточнить, подставив в них вместо $N_n(\bar{c})$ величину

$$N_n(\bar{c}) = N_n(\bar{c}_{d-1}) - N_n(\bar{c}_{d-1}, b) - N_n(\bar{c}_{d-1}, c).\tag{4.14}$$

После этой корректировки знаменатели в соотношениях (4.13) уменьшатся, что приводит увеличению вероятностей и уменьшению затрат на их передачу.

Применим алгоритм РРМА к кодированию используемой в предыдущем разделе фразы из 59 символов:

«на_дворе_трава_на_траве_дрова_на_дворе_трава_на_траве_дрова».

Результаты работы алгоритма РРМА с параметром $D = 8$ по шагам приведены в табл. 4.10. Знак $\#$ в графе «контекст» соответствует контексту нулевой длины, штрихом помечены вероятности, подсчитанные с учетом исключений. В графе $N_n(\bar{c})$ дано число появлений данного контекста, и если этот контекст укорачивался в процессе кодирования, то через запятую приведено число появлений укороченных контекстов. Предполагается, что возможно использование 256 символов и исходно их появление равновероятно.

Таблица 4.10. Пример работы алгоритма PPMA

Шаг	Интервал $D = 8$	Символ	Контекст \bar{c}	Количество появлений $N_n(\bar{c})$	PPMA	
					$\hat{p}(\text{esc} \bar{c})$	$\hat{p}(a \bar{c})$
1		н	#	0	1	1/256
2	н	а	#	1	1/2	1/255
3	на	—	#	2	1/3	1/254
4	на_	д	#	3	1/4	1/253
5	на_д	в	#	4	1/5	1/252
6	на_дв	о	#	5	1/6	1/251
7	на_дво	р	#	6	1/7	1/250
8	на_двор	е	#	7	1/8	1/249
9	на_дворе	—	#	8		1/9
10	а_дворе_	т	—	1, 9	1/2*1/9'	1/248
11	_дворе_т	р	#	10		1/11
12	дворе_тр	а	р	1, 11	1/2	1/11'
13	воре_тра	в	а	1, 12	1/2	1/12'
14	опе_трав	а	в	1, 13	1/2	2/13'
15	ре_трава	—	а	2		1/3
16	е_трава_	н	а_	1, 2, 15	1/2*1/2'	1/14'
17	_трава_н	а	н	1		1/2
18	трава_на	—	на	1		1/2
19	рава_на_	т	на_	1, 2, 3	1/2*1/2'	1/2'
20	ава_на_т	р	—т	1		1/2
21	ва_на_тр	а	_тр	1		1/2
22	а_на_тра	в	_тра	1		1/2
23	_на_трав	е	_трав	1, 1, 1, 1, 2, 22	1/2*1'*1'*1'*1/2'	1/21'
24	на_траве	—	е	1		1/2
25	а_траве_	д	е_	1, 4	1/2	1/4'
26	_траве_д	р	—д	1, 1, 25	1/2*1'	3/25'
27	_траве_др	о	р	3, 26	1/4	1/25'
28	раве_дро	в	о	1, 27	1/2	3/27'
29	аве_дров	а	в	3		1/4
30	ве_дрова	—	ва	1		1/2
31	е_дрова_	н	ва_	1		1/2
32	_дрова_н	а	ва_н	1		1/2
33	дрова_на	—	ва_на	1		1/2
34	рова_на_	д	ва_на_	1, 1, 1, 2	1/2*1'*1'	1/2'
35	ова_на_д	в	на_д	1		1/2
36	ва_на_дв	о	на_дв	1		1/2
37	а_на_дво	р	на_дво	1		1/2
38	_на_двор	е	на_двор	1		1/2
39	на_дворе	—	на_дворе	1		1/2
40	а_дворе_	т	а_дворе_	1		1/2
41	_дворе_т	р	_дворе_т	1		1/2
42	дворе_тр	а	дворе_тр	1		1/2
43	воре_тра	в	воре_тра	1		1/2

Таблица 4.10 (окончание)

Шаг	Интервал $D = 8$	Символ	Контекст \bar{c}	Количество появлений $N_n(\bar{c})$	PPMA	
					$\hat{p}(esc \bar{c})$	$\hat{p}(a \bar{c})$
44	опе_трав	а	опе_трав	1		1/2
45	ре_трава	—	ре_трава	1		1/2
46	е_трава_	н	е_трава_	1		1/2
47	_трава_н	а	_трава_н	1		1/2
48	трава_на	—	трава_на	1		1/2
49	рава_на_	т	рава_на_	1		1/2
50	ава_на_т	р	ава_на_т	1		1/2
51	ва_на_тр	а	ва_на_тр	1		1/2
52	а_на_тра	в	а_на_тра	1		1/2
53	_на_трав	е	_на_трав	1		1/2
54	на_траве	—	на_траве	1		1/2
55	а_траве_	д	а_траве_	1		1/2
56	_траве_д	р	_траве_д	1		1/2
57	траве_др	о	траве_др	1		1/2
58	раве_дро	в	раве_дро	1		1/2
59	аве_дров	а	аве_дров	1		1/2

Поясним характерные шаги алгоритма.

Шаг 1. Число $N_n(\bar{c}) = 0$. Следовательно, $\hat{p}(esc|\bar{c}) = 1$ и $\hat{p}(a|\bar{c}) = 1/256$.

Шаг 2. Число $N_n(\bar{c}) = 1$. Следовательно, $\hat{p}(esc|\bar{c}) = 1$. Учитывая, что символ на шаге 2 отличается от символа на шаге 1, $\hat{p}(a|\bar{c}) = 1/255$.

Шаг 9. Пробел встретился второй раз. Поскольку предшествующий ему символ «е» встречался только один раз, контекст имеет нулевую длину, и пробел имеет вероятность $1/(8+1)=1/9$.

Шаг 10. Пробел является контекстом длины 1 по отношению к символу «т». После пробела встречался только символ «д». Необходимо передать символ «esc» при его вероятности, равной $1/2$, и перейти к контексту нулевой длины. Но и при контексте нулевой длины символ «т» не встречался, и поэтому вновь необходимо передать символ «esc», вероятность которого равна $1/10$. Однако ясно, что среди возможных символов не может быть символа «д», и вероятность символа «esc» повышается до $1/9$.

Шаг 22. Контекстом по отношению к символу «в» является «_тра». Эта комбинация уже встречалась, и за ней следовал символ «в». Таким образом, следующими символами могут быть либо символ «в», либо символ «esc» с равными вероятностями. По этой причине для передачи символа «в» требуется всего 1 бит.

Шаг 23. Символу «е» предшествует контекст «_трав». При появлении этого контекста за ним следовал другой символ — «а». Поэтому передается символ «esc» при вероятности $1/2$. Контекст укорачивается — «трав»; за ним ранее следовал только символ «а». Вновь необходимо передать символ «esc» при его вероятности без учета правила исключений, равной $1/2$. Но при передаче предыдущего символа «esc» значение «а» уже было уже исключено, а значит, вероятность символа «esc» равна $1'$. Контекст вновь сокращается до «рав», и аналогично предыдущему вновь следует символ «esc» с вероятностью $1'$. Контекст еще сокращается до «ав», вновь следует символ «esc» с вероятностью $1'$. И, наконец,

к символу «е» контекст сократился до «в». Контекст «в» встречался два раза. Вероятность символа «*esc*» без учета правила исключений равна $1/3$, а с учетом этого факта — $1/2'$. Пустой контекст встречался 22 раза. Значит, вероятность $\hat{p}(a|\bar{c}) = 1/23$, а с учетом двух исключений (символы «а» и «о») — $1/21'$.

Используя значения $\hat{p}(\bar{c})$ и $\hat{p}(a|\bar{c})$, можно подсчитать количество битов, требуемых для передачи данной информации:

$$I = \left[- \sum_{\forall i} \log_2 \hat{p}_i(\text{esc}|\bar{c}) - \sum_{\forall i} \log_2 \hat{p}_i(a|\bar{c}) \right]. \quad (4.15)$$

В данном случае $I = \lceil 34,47 + 149,83 \rceil = 185$ битов (операция $\lceil \cdot \rceil$ — округление вверх).

Важно отметить, что в алгоритме PPMD [2.41] используются следующие принципы оценки условных вероятностей появления символов алфавита источника и *esc*-символов:

$$\begin{aligned} \hat{p}(a|\bar{c}) &= \frac{N_n(\bar{c}, a) - 1/2}{N_n(\bar{c})}, & N_n(\bar{c}, a) > 0, \\ \hat{p}(\text{esc}|\bar{c}) &= \frac{M_n(\bar{c})}{2N_n(\bar{c})}, & N_n(\bar{c}, a) = 0, \end{aligned} \quad (4.16)$$

где $M_n(\bar{c})$ — число различных символов, появившихся в последовательности длины n вслед за контекстом \bar{c} .

Здесь также используется правило исключений (4.14), причем число различных символов $M_n(\bar{c})$ также должно быть пересчитано с учетом правила исключений. Результаты работы алгоритма PPMD с параметром $D = 8$ по шагам приведены в табл. 4.11.

Таблица 4.11. Пример работы алгоритма PPMD

Шаг	Интервал $D = 8$	Символ	Контекст \bar{c}	Число $N_n(\bar{c})$	Число $M_n(\bar{c})$	PPMD	
						$\hat{p}(\text{esc} \bar{c})$	$\hat{p}(a \bar{c})$
1		н	#	0	0	1	1/256
2	н	а	#	1	1	1/2	1/255
3	на	—	#	2	2	2/4	1/254
4	на_д	д	#	3	3	3/6	1/253
5	на_д	в	#	4	4	4/8	1/252
6	на_дв	о	#	5	5	5/10	1/251
7	на_дво	р	#	6	6	6/12	1/250
8	на_двор	е	#	7	7	7/14	1/249
9	на_дворе	—	#	8	8		1/16
10	ф_дворе_	т	—	1, 9	1, 8	1/2*7/16'	1/248
11	дворе_т	р	#	10	9		1/20
12	дворе_тр	а	р	1, 11	1, 9	1/2	1/20'
13	воре_тра	в	а	1, 12	1, 9	1/2	1/22'
14	оре_трав	а	и	1, 13	1, 9	1/2	3/24'
15	ре_трава_	—	а	2	2		1/4
16	е_трава_	н	а_	1, 2, 15	1, 2	1/2*1/2'	1/26'
17	трава_	н	а	1	1		1/2
18	трава_на	—	на	1	1		1/2

Таблица 4.11 (окончание)

Шаг	Интервал $D = 8$	Символ	Контекст \bar{c}	Число $N_n(\bar{c})$	Число $M_n(\bar{c})$	PPMD	
						$\hat{p}(esc \bar{c})$	$\hat{p}(a \bar{c})$
19	рава_на_	т	на_	1, 2, 3	1, 2, 3	$1/2*1/2'$	$1/2'$
20	ава_на_т	р	т	1	1		$1/2$
21	ва_на_тр	а	тр	1	1		$1/2$
22	а_на_тра	в	тра	1	1		$1/2$
23	_на_трав	е	_трав	1, 1, 1, 1, 2, 22	1, 1, 1, 1, 2, 9	$1/2*1^*/$ $*1^*/1^*/$ $*1/2'$	$1/40'$
24	на_траве	—	е	1	1		$1/2$
25	а_траве_	д	е_	1, 4	1, 3	$1/2$	$1/6'$
26	_траве_д	р	д	1, 1, 25	1, 1, 9	$1/2*1'$	$5/48'$
27	_тра- ве_др	о	р	3, 26	2, 9	$2/6$	$1/48'$
28	раве_дро	в	о	1, 27	1, 9	$1/2$	$5/52'$
29	аве_дро	а	в	3	3		$1/6$
30	ве_дро	—	ва	1	1		$1/2$
31	е_дро	н	ва_	1	1		$1/2$
32	_дро	а	ва_н	1	1		$1/2$
33	дро	—	ва_на	1	1		$1/2$
34	рова_на_	д	ва_на_	1, 1, 1, 2	1, 1, 1, 2	$1/2*1^*/1'$	$1/2'$
35	ова_на_д	в	на_д	1	1		$1/2$
36	ва_на_дв	о	на_дв	1	1		$1/2$
37	а_на_дво	р	на_дво	1	1		$1/2$
38	_на_двор	е	на_двор	1	1		$1/2$
39	на_дворе	—	на_дворе	1	1		$1/2$
40	а_дворе_	т	а_дворе_	1	1		$1/2$
41	_дворе_т	р	дворе_т	1	1		$1/2$
42	дворе_тр	а	дворе_тр	1	1		$1/2$
43	воре_тра	в	воре_тра	1	1		$1/2$
44	опе_трав	а	опе_трав	1	1		$1/2$
45	ре_трава	—	ре_трава	1	1		$1/2$
46	е_трава_	н	е_трава_	1	1		$1/2$
47	_трава_н	а	трава_н	1	1		$1/2$
48	трава_на	—	трава_на	1	1		$1/2$
49	рава_на_	т	рава_на_	1	1		$1/2$
50	ава_на_т	р	ава_на_т	1	1		$1/2$
51	ва_на_тр	а	ва_на_тр	1	1		$1/2$
52	а_на_тра	в	а_на_тра	1	1		$1/2$
53	_на_трав	е	на_трав	1	1		$1/2$
54	на_траве	—	на_траве	1	1		$1/2$
55	а_траве_	д	а_траве_	1	1		$1/2$
56	_траве_д	р	траве_д	1	1		$1/2$
57	траве_др	о	траве_др	1	1		$1/2$
58	раве_дро	в	раве_дро	1	1		$1/2$
59	аве_дро	а	аве_дро	1	1		$1/2$

Рассчитанное с использованием соотношения (4.15) количество битов, требуемых для передачи данной информации, равно:

$$I = \lceil 23,78 + 158,32 \rceil = 183 \text{ бита.}$$

Как следует из рассмотренных вариантов реализации метода PPM, при их прак-

тической эффективности основным требованием является обеспечение хранения большого массива статистической информации, что вызывает необходимость использования значительного объема оперативной памяти.

Метод СТВ (Context-Tree Weighting — взвешивание с использованием контекстных деревьев) [2.44–2.49] представляет собой одну из возможных альтернатив методу РРМ, в этом методе оценка вероятности появления символа осуществляется с учетом весов контекстов, организованных в бинарное дерево.

Рассматриваются информационные источники, генерирующие двоичные последовательности длины N . Последовательность $\mathbf{x}_1^N = x_1 x_2 \dots x_N$ генерируется с вероятностью $\hat{p}_a(\mathbf{x}_1^N)$. Для источника без памяти:

$$\hat{p}_a(\mathbf{x}_1^N) = \prod_{n=1}^N \hat{p}_a(x_n), \quad \hat{p}_a(1) = 1 - \hat{p}_a(0) = \theta. \quad (4.17)$$

Параметр θ (вероятность генерации 1), лежит в диапазоне от 0 до 1. Если последовательность содержит a нулей и b единиц, то $\hat{p}_a(\mathbf{x}_1^N) = (1 - \theta)^a \cdot \theta^b$.

Кумулятивная вероятность $Q_a(\mathbf{x}_1^N)$, используемая для арифметического кодирования, определяется следующим образом:

$$Q_a(\mathbf{x}_1^N) = \sum_{n=1, x_n=1}^N \hat{p}_a(x_1 x_2 \dots x_{n-1} 0).$$

Рассмотрим оценку распределения вероятностей $\hat{p}_e(\mathbf{x}_1^N)$, сформулированную Кричевским и Трофимовым [2.60]. Распределение Кричевского–Трофимова (КТ-распределение) определяется следующей условной вероятностью:

$$\hat{p}_e(x_n = 1 | \mathbf{x}_1^{n-1}) = \frac{b + 1/2}{a + b + 1}; \quad \hat{p}_e(x_n = 0 | \mathbf{x}_1^{n-1}) = 1 - \hat{p}_e(x_n = 1 | \mathbf{x}_1^{n-1}) = \frac{a + 1/2}{a + b + 1}. \quad (4.18)$$

При этом оценка вероятности двоичной последовательности $\mathbf{x}_1^N = x_1 x_2 \dots x_N$ связана условными вероятностями так:

$$\hat{p}_e(\mathbf{x}_1^N) = \hat{p}_e(x_1) \cdot \hat{p}_e(x_2 | x_1) \cdot \hat{p}_e(x_3 | \mathbf{x}_1^2) \cdots \hat{p}_e(x_N | \mathbf{x}_1^{N-1}). \quad (4.19)$$

Из этих соотношений нетрудно вывести выражения для оценочной вероятности последовательности из a нулей и b единиц:

$$\hat{p}_e(a, b) = \frac{[\frac{1}{2} \cdot \frac{3}{2} \cdots (a - \frac{1}{2})] \cdot [\frac{1}{2} \cdot \frac{3}{2} \cdots (b - \frac{1}{2})]}{1 \cdot 2 \cdots (a + b)}. \quad (4.20)$$

При этом

$$\hat{p}_e(a + 1, b) = \frac{a + 1/2}{a + b + 1} \hat{p}_e(a, b) \text{ и } \hat{p}_e(a, b + 1) = \frac{b + 1/2}{a + b + 1} \hat{p}_e(a, b). \quad (4.21)$$

Пример оценок вероятности и кумулятивной вероятности:

$$\hat{p}_e(01110) = \frac{1}{2} \cdot \frac{1}{4} \cdot \frac{3}{6} \cdot \frac{5}{8} \cdot \frac{3}{10} = \frac{3}{256}$$

(см. рис. 4.4а);

$$Q_e(01110) = \frac{1}{2} \cdot \frac{3}{4} + \frac{1}{2} \cdot \frac{1}{4} \cdot \frac{3}{6} + \frac{1}{2} \cdot \frac{1}{4} \cdot \frac{3}{6} \cdot \frac{3}{8} = \frac{59}{128}.$$

Построение контекстного дерева

Контекстное дерево состоит из узлов, соответствующих контекстам s вплоть до заданной глубины D (память источника не более D). Корень контекстного дерева соответствует пустому контексту. Каждый узел s в контекстном дереве связывается с подпоследовательностью символов источника, которые появляются после контекста s .

Требуется оценить взвешенную вероятность символов \hat{p}_w^s в каждом узле контекстного дерева. Статистика источника неизвестна. Если источник не обладает памятью, то хорошей оценкой вероятности будет оценка по КТ-распределению. В противном случае необходимо расщепление последовательности на две подпоследовательности нулей — 0_s и единиц — 1_s , и хорошей оценкой вероятности будет служить произведение взвешенных вероятностей \hat{p}_w^{0s} и \hat{p}_w^{1s} . Вероятность в каждом узле контекстного дерева подсчитывается как взвешенная сумма этих двух вариантов:

$$\hat{p}_w^s = [\hat{p}_e(a_s, b_s) + \hat{p}_w^{0s} \cdot \hat{p}_w^{1s}] / 2,$$

где a_s, b_s — количество нулей и единиц в контексте s .

Полученные таким образом вероятности используются для арифметического кодирования данных источника.

Например, предположим, что глубина контекста $D = 3$ и пусть источник данных породил последовательность 010011011 после последовательности ... 110:

Источник данных	...	110 010011011
Последовательность данных		123456789

Пустому контексту соответствуют все символы от первого до девятого; символы 3, 6, 7, 9 имеют контекст «1»; контексту «0» соответствуют символы с номерами 1, 2, 4, 5, 8; контекст «11» имеет символ номер 7 и т. д. (см. табл. 4.12). На рис. 4.46 приведено соответствующее контекстное дерево.

Таблица 4.12. Пример расчета вероятностей для контекстного дерева в методе СТВ

Контекст	Номера символов	Количество нулей и единиц (a, b)	$\hat{p}_e(a, b)$	\hat{p}_w^s
Пустой	1,2,3,4,5,6,7,8,9	(4,5)	35/65536	133/262144
1	3,6,7,9	(2,2)	3/128	7/256
0	1,2,4,5,8	(2,3)	3/256	9/512
11	7	(1,0)	1/2	
01	3,6,9	(1,2)	1/16	1/16
10	1,4,8	(2,1)	1/16	1/16
00	2,5	(0,2)	3/8	
111	—	—	—	
011	7	(1,0)	1/2	
101	9	(0,1)	1/2	
001	3,6	(1,1)	1/8	
110	1,8	(1,1)	1/8	
010	4	(1,0)	1/2	
100	2,5	(0,2)	3/8	
000	—	—	—	

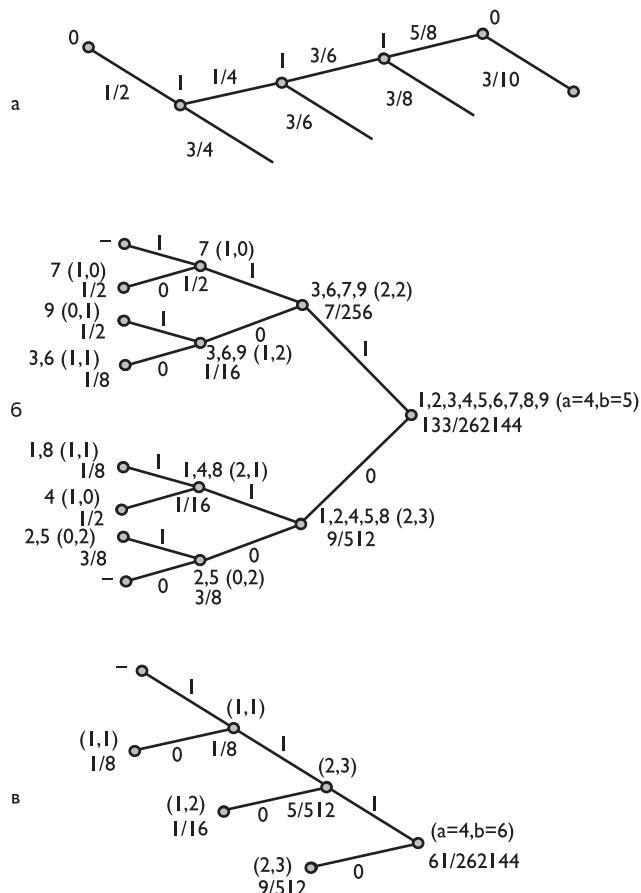


Рис. 4.4. К построению деревьев последовательностей при кодировании методом СТВ:
 а) структура дерева при расчете оценки $\hat{p}_e(01110)$; б) структура контекстного дерева для последовательности $\dots 110|010011011$; в) принцип обновления контекстного дерева при появлении следующего символа источника, равного «1»

Для каждого узла контекстного дерева подсчитывается количество нулей (a) и единиц (b), им соответствующих. Например, узлу с контекстом «1» соответствуют символы с номерами 3, 6, 7, 9, среди которых два нуля и две единицы.

Далее для каждого узла контекстного дерева рекурсивно рассчитываются вероятности. Расчет начинается с конечных веток и идет по направлению к корню. Вероятность в конечных ветках рассчитывается как $\hat{p}_e(a, b)$, а для остальных узлов используется формула взвешенной вероятности \hat{p}_w^s . В табл. 4.12 приведены соответствия символов (по номерам от 1 до 9) возможным контекстам.

Алгоритм обновления контекстного дерева при появлении следующего символа рассмотрим на том же примере (110|010011011). Пусть следующий, десятый, символ источника — «1». Тогда нужно:

- увеличить b_s ;
 - обновить значение $\hat{p}_e(a_s, b_s)$;

- обновить \hat{p}_w^s для всех контекстов вплоть до глубины $D = 3$: $\{011, 11, 1, \text{пустой}\}$.

Структура обновления контекстного дерева приведена на рис. 4.4в.

Метод DMC (Dynamic Markov Compression — динамическая марковская компрессия) значительно отличается от двух предыдущих методов. Он связан с теорией конечных автоматов и использует моделирование с конечным числом состояний [2.50, 2.51]. Благодаря простоте этот метод обладает высокой производительностью, но невысокой эффективностью при обработке последовательности достаточно сложных символов, используемых для описания большинства реальных информационных источников.

По этой причине чаще всего метод DMC используется при побитовом вводе информации, поскольку при этом нет проблем следующего состояния и существуют только два перехода от одного состояния в другое.

Марковская модель является таким источником информации, при котором в каждом состоянии она помнит только об одном предыдущем своем состоянии.

В качестве примера представим марковскую модель, при которой источник информации порождает последовательность двоичных чисел [2.61, 2.62]:

- за 0 следует 0 с вероятностью 3/4;
- за 0 следует 1 с вероятностью 1/4;
- за 1 следует 0 с вероятностью 1/3;
- за 1 следует 1 с вероятностью 2/3.

При этом первая цифра может быть «0» или «1» равновероятно (1/2). Эта модель может быть описана с помощью ориентированного графа, каждой дуге которого приписываются соответствующие вероятности перехода (см. рис. 4.5а).

По мере поступления информации марковская модель переходит из одного своего состояния в другое в соответствии с ранее прочитанными символами. Если завести счетчики переходов по каждой дуге, то можно получить оценки вероятностей переходов. Если переход из состояния A по дуге для символа «0» проходил N_0 раз, а по дуге для «1» — N_1 , то оценки условных вероятностей переходов таковы:

$$\hat{p}(0|\text{из состояния } A) = \frac{N_0}{N_0 + N_1}; \quad \hat{p}(1|\text{из состояния } A) = \frac{N_1}{N_0 + N_1}.$$

При адаптивном кодировании заранее определить значения N_0 и N_1 невозможно, и эти оценки могут быть определены так:

$$\hat{p}(0|\text{из состояния } A) = \frac{N_0 + c}{N_0 + N_1 + 2c}; \quad \hat{p}(1|\text{из состояния } A) = \frac{N_1 + c}{N_0 + N_1 + 2c},$$

где c — положительная константа.

Для файлов большого размера (намного больше значения c) выбор конкретного значения c существенной роли не играет. Для файлов малого размера при малом значении c модель быстрее оценит характеристики источника, однако при этом увеличивается вероятность неправильного прогноза, что является типичной проблемой адаптивных алгоритмов. Алгоритм динамической реализации диаграммы состояний поясняется на примере уже построенной модели, содержащей состояния A, B, C, D и E (рис. 4.5б).

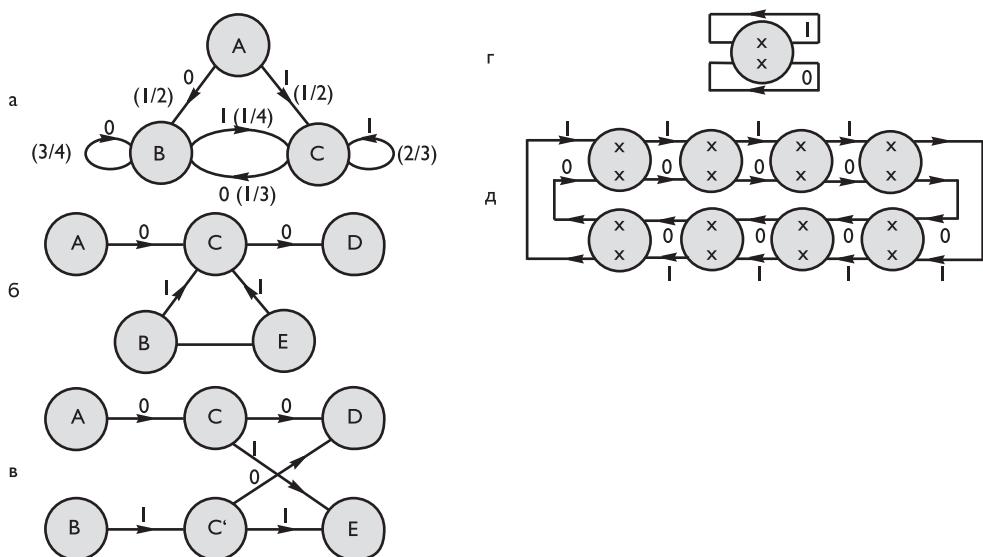


Рис. 4.5. Марковская модель источника информации: а) ориентированный граф источника заданной модели последовательности; б) исходная модель динамической реализации диаграммы состояний; в) измененная модель с разделением состояния С на две части; г) начальная модель с одним состоянием; д) начальная модель, представляющая 8-битовые последовательности

Из диаграммы следует, что в состояние С ведут две дуги из А и В; из состояния С также выходят две дуги — в D и Е.

Всякий раз, когда модель переходит в состояние С, теряется информация о контексте, т. е. неясно откуда пришли в это состояние — из А или В. Однако вполне может быть, что выбор следующего состояния (D или Е) зависит от предыдущего состояния (А или В). Чтобы учесть эту зависимость, например в случае, когда переход из состояния А в С имеет большую частоту, чем переход из В в С, состояние С обычно разделяют на две части (С и С').

Измененная модель приведена на рис. 4.5в. Значения счетчиков переходов из С в D или Е делится между счетчиками С и С' в новой модели пропорционально числу переходов из А в С и из В в С для старой модели. Следовательно, новая модель отражает степень зависимости между состояниями А, В и D, Е. При этом выбор следующего состояния (D или Е) не зависит от предыдущего состояния (А или В), но зависит от непосредственного предшественника А или В. В этом случае производится дополнительное деление состояний А, С' и В, С. Следовательно, модель теперь будет учитывать эти зависимости. Процесс деления необходимо производить только в том случае, когда существует уверенность в наличии соответствующих зависимостей, иначе информация о контексте будет потеряна.

Поэтому желательно, чтобы деление состояния С проходило тогда, когда оба счетчика переходов А-С и В-С достаточно велики. Чем больше применяется делений состояний, тем больше увеличиваются зависимости, используемые моделью для прогноза.

Для полного определения метода динамического сжатия Маркова необходимо определить начальную модель, которая затем будет изменяться в соответствии со спецификой поступающей информации. Простейшим случаем является модель с одним состоянием, приведенная на рис. 4.5г.

Несколько лучшее сжатие может быть достигнуто при побайтовом вводе информации при использовании начальной модели, представляющей 8-битовые последовательности в виде цепи, приведенной на рис. 4.5д. Впрочем, начальная модель не является особо решающей, поскольку DMC быстро приспосабливается к требованиям кодируемой информации.

Для оценки вероятностей появления символов на выходе информационного источника применяется **искусственная нейронная сеть** [2.52, 2.53].

Чаще всего используется многослойная нейронная сеть, обладающая одним нейроном выходного слоя на каждый символ информационного алфавита. На выходах этих нейронов формируются сигналы, определяющие вероятности появления различных символов в текущем контексте.

Преимуществом данного решения энтропийного кодирования является возможность адаптации системы к обрабатываемой информации с применением фиксированного объема памяти. Этот метод обладает высокой эффективностью, но очень низкой производительностью.

4.2.8. Контекстные методы энтропийного кодирования

Эта группа контекстных методов включает два алгоритма, которые рассматривают зависимости между последовательностью символов, предшествующих кодируемому (левосторонний контекст) и последовательность символов, следующих за кодируемым (правосторонний контекст). Первый из этих двух методов называется ассоциативным кодированием Буяновского (ACB) [2.54] и относится к поточным методам, т. е. он может применяться для кодирования данных, непрерывно порождаемых источником. Второй метод [2.55] основан на использовании преобразования Барроуза–Уиллера, он может применяться, если доступен весь объем кодируемой информации. Это самые современные методы, которые пока не получили заметного распространения из-за значительной сложности реализации, но имеют большие перспективы.

Ассоциативное кодирование Буяновского (ACB)

Алгоритм кодирования Буяновского поясняется на примере обработки битовой последовательности.

Предположим, что уже закодирована последовательность

...11101011100001010010001010100101110

и предстоит закодировать строку **0100010111...**, обозначенную далее как «пакуемая строка». Закодированная последовательность представляет предысторию, пакуемая строка — постисторию:

...11101011100001010010001010100101110|0100010111... (4.22)

Выпишем все подпоследовательности из предыстории, которые совпадают с первым и более начальными битами предыстории (отсчет битов ведется справа

налево, в данном случае первый бит равен «0»):

```
... 11110|1011100001010010001010100101110
... 1111010|11100001010010001010100101110
... 11110101110|0001010010001010100101110
... 111101011100|001010010001010100101110
... 1111010111000|01010010001010100101110
... 11110101110000|1010010001010100101110
... 1111010111000010|10010001010100101110
... 111101011100001010|010001010100101110
... 1111010111000010100|10001010100101110
... 111101011100001010010|001010100101110
... 1111010111000010100100|01010100101110
... 11110101110000101001000|1010100101110
... 1111010111000010100100010|10100101110
... 111101011100001010010001010|100101110
... 11110101110000101001000101010|0101110
... 111101011100001010010001010100|101110
... 11110101110000101001000101010010|1110
... 111101011100001010010001010100101110|0100010111...
```

Последней выделена строка (4.22), содержащая предысторию и постисторию, которую необходимо закодировать. Заметим, что все строки в этом списке, кроме последней доступны на декодировании, а у последней строки известна предыстория.

Максимальная длина предыстории ограничивается некоторым числом N . Полученные последовательности упорядочиваются по возрастанию лексикографического порядка предыстории. При этом анализируется последовательность символов справа налево начиная от символа «|». То есть последовательность ... 110000| лексикографически меньше последовательности ... 001000|. Слева проставлен номер строки. Пакуемой строке (с ее предысторией) присваивается номер 0, от нее номера вверх возрастают, а вниз убывают:

15	... 11110101110000	1010010001010100101110
14	... 11110101110000101001000	1010100101110
13	... 1111010111000010100100010	01010010001010100101110
12	... 111101011100001010010001010	01010100101110
11	... 1111010111000010100	10001010100101110
10	11101011100001010010001010100	101110
9	... 1111010111000	001010010001010100101110
8	... 1111010111000010	10010001010100101110
7	... 1111010111000010100100010	10100101110
6	... 111101011100001010010001010	001010100101110
5	... 111101011100001010010001010010	1110
4	... 11110101110000101001000101010	010001010100101110
3	... 111101011100001010010001010	100101110
2	... 11110101110000101001000101010	0101110
1	... 1111010	11100001010010001010100101110
0	... 111101011100001010010001010100101110	0100010111...
-1	... 11110101110	0001010010001010100101110
-2	... 11110	1011100001010010001010100101110

Этот список называют левосторонним ассоциативным списком, или «воронкой аналогий» в терминологии автора алгоритма.

Затем этот список переупорядочивается в лексикографическом порядке пост-истории (в противоположном направлении — слева направо после «|», т. е. последовательность |110000... лексикографически большие последовательности |001000...):

-1	... 11110101110	0001010010001010100101110
9	... 111101011100	001010010001010100101110
6	... 111101011100001010010	001010100101110
4	... 11110101110000101010	010001010100101110
0	... 111101011100001010010001010100101110	0100010111..
13	... 11110101110000101001011100	01010010001010100101110
12	... 1111010111000010100100	01010100101110
2	... 11110101110000101001000101010	0101110
11	... 1111010111000010100100010100	10001010100101110
8	... 1111010111000010100100010100	10010001010100101110
3	... 1111010111000010100100010101	100101110
15	... 1111010111000010100100010101	1010010001010100101110
7	... 1111010111000010100100010100	10100101110
14	... 1111010111000010100100010100	1010100101110
10	11101011100001010010001010100	101110
-2	... 11110	1011100001010010001010100101110
1	... 1111010	11100001010010001010100101110
5	... 11110101110000101001000101010010	1110

В дальнейшем используются только две строки, примыкающие к пакуемой строке:

4	... 11110101110000101010001010	010001010100101110
0	... 1111010111000010100100010100101110	0100010111...
13	... 111101011100001010010001010100101110	01010010001010100101110

Код пакуемой строки состоит из трех частей.

Номер строки, максимально совпадающей с пакуемой строкой. В данном случае — 4. Этот номер кодируется любым эффективным префиксным или арифметическим кодом. В качестве эквивалента вероятности автор алгоритма предложил использовать длину совпадения строки предыстории пакуемой строки со строками предысторий из рассмотренного списка с соответствующей нормированной. В данном случае:

Номер строки	Длина совпадения	Эквивалент вероятности
-2	4	4/(34+C)
-1	7	7/(34+C)
0	—	C/(34+C)
1	2	2/(34+C)
2	2	2/(34+C)
3	2	2/(34+C)
4	2	2/(34+C)
5	2	2/(34+C)
6	2	2/(34+C)
7	2	2/(34+C)
8	2	2/(34+C)
9	1	1/(34+C)
10	1	1/(34+C)
11	1	1/(34+C)
12	1	1/(34+C)
13	1	1/(34+C)
14	1	1/(34+C)
15	1	1/(34+C)

Величину «С» можно выбрать, например, как среднее значение длины совпадений, в данном случае сумма длин совпадений равна 34, а $C = 34/17 = 2$. Если совпадений постистории (правосторонних контекстов) нет, то записывается код номера строки «0».

Бит различия — первый бит пакуемой строки, отличающий ее от строки, максимально с ней совпадающей. В данном случае — 1 (отмечен подчеркиванием):

- в пакуемой строке — **010001011**,
- в максимально близкой к ней 4-й строке — 010001010100101110.

Этим битом определяется положение максимально близкой строки: если бит 0, то эта строка выше, если 1 — то ниже. В данном случае пакуемая строка ниже.

Длина «вытяжки» (в терминологии автора алгоритма) — количество нулей или единиц в той части пакуемой строки (ниже отмечено подчеркиванием значений **0101**), которая находится между битом различия максимально совпадающей строки (бит отмечен подчеркиванием в нижней строке) и битом различия второй строки (бит отмечен подчеркиванием в верхней строке). Если пакуемая строка лексикографически старше строки, максимально совпадающей с ней, то передается количество нулей, а если младше — то количество единиц. В данном случае подсчитывается количество нулей, которое равно 2.

```
01010010001010100101110
0100010111...
010001010100101110
```

Длина «вытяжки» также кодируется адаптивным статистическим кодером.

Таким образом, упакована строка **010001011** (до бита различия включительно). Операция восстановления строки осуществляется следующим образом:

Декодируется номер строки максимального совпадения (в данном случае 4).

По биту различия определяется положение запакованной строки (и соседней строки) — если бит «0», то она выше (лексикографически младше, длина «вытяжки» — количество единиц), в противном случае — ниже (лексикографически старше, длина «вытяжки» — количество нулей) (в данном случае номер соседней строки — 13, она лексикографически старше).

Копируется в искомую строку совпадающая часть строки максимального совпадения (4,010001010100...) и соседней (13,010100100010...), а также следующий бит из строки максимального совпадения (в данном случае — совпадающая часть — 010, а следующий бит — 0). Декодируется длина «вытяжки» (2).

Восстанавливается следующая часть искомой строки по «вытяжке» — копируются из строки максимального совпадения следующие биты, включающие соответствующее длине вытяжки количество нулей или единиц (в данном случае два нуля, так как искомая строка лексикографически старше; строка максимального совпадения — 010001010100..., из нее уже скопировано 4 бита, 0100, значит, копируется 010). Далее копируются из строки максимального совпадения все биты до следующего нуля или единицы (в данном случае — до следующего нуля, т. е. 1). Таким образом, восстановленная по «вытяжке» часть строки — 0101.

Дописывается в искомую строку бит различия (в данном случае — 1, а в строке максимального совпадения следующий бит, естественно, 0). Таким образом,

восстановлена строка **010001011**, что и требовалось. Кодирование и восстановление последующих битов информации осуществляется аналогичным образом.

Метод Барроуза–Уиллера (BWT – Burrows–Wheeler Transform)

Этот метод основан на использовании преобразования Барроуза–Уиллера, BWT, обеспечивающего преобразование всей исходной информации в более удобный для сжатия вид. Поскольку для реализации BWT требуется знать все элементы информационного потока, метод не может быть применен для сжатия при последовательном поступлении данных символ за символом.

Принцип преобразования BWT поясняется в табл. 4.13 на примере использования фразы: «на_дворе_трава_на_траве_дрова». Эта фраза обозначена в столбце 2 «Циклические перестановки» под номером «0». Последующие строки в этом столбце представляют собой циклические последовательные сдвиги фразы, каждый раз на один символ влево. Полученный набор матриц фраз пронумерован числами от 0 до 28.

Таблица 4.13. Пример преобразования Барроуза–Уиллера

№	Циклические перестановки	Сортированные перестановки	№	КС	
1	2	3	4	5	6
0	на_дворе_трава_на_траве_дрова	ава_на_траве_дрована_дворе_тр	11	р	
1	а_дворе_трава_на_траве_дрован	аве_дрована_дворе_трава_на_тр	20	р	
2	_дворе_трава_на_траве_дрована	ана_дворе_трава_на_траве_дров	28	в	
3	дворе_трава_на_траве_дрована_	а_дворе_трава_на_траве_дрован	1	н	
4	воре_трава_на_траве_дрована_д	а_на_траве_дрова_на_дворе_трав	13	в	
5	оре_трава_на_тра_ве_дрована_дв	а_траве_дрована_дворе_трава_н	16	н	
6	ре_трава_на_траве_дрована_дво	вана_дворе_трава_на_траве_дро	27	о	
7	е_трава_на_тра_ве_дрована_двор	ва_на_траве_дрова_на_дворе_тра	12	а	
8	_трава_на_траве_дрована_дворе	ве_дрована_дворе_трава_на_тра	21	а	
9	трава_на_траве_дрована_дворе_	воре_трава_на_тра_ве_дрована_д	4	д	
10	рава_на_траве_дрована_дворе_т	дворе_трава_на_траве_дрована_	3	–	
11	ава_на_траве_дрована_дворе_тр	дрована_дворе_тра_ва_на_траве	24	–	
12	ва_на_траве_дрова_на_дворе_тра	е_дрована_дворе_трава_на_трав	22	в	
13	а_на_траве_дрова_на_дворе_трав	е_трава_на_траве_дрована_двор	7	р	
14	_на_траве_дрова_на_дворе_трава	на_дворе_тра_ва_на_траве_дрова	0	а	14
15	на_траве_дрована_дворе_трава_	на_траве_дрова_на_дворе_трава_	15	–	

Таблица 4.13 (окончание)

1	2	3	4	5	6
16	а_траве_дрована_дворе_трава_н	ована_дворе_трава_на_траве_др	26	р	
17	_траве_дрована_дворе_трава_на	оре_трава_на_траве_дрована_дв	5	в	
18	траве_дрована_дворе_трава_на_	рава_на_траве_дрована_дворе_т	10	т	
19	раве_дрована_дворе_трава_на_т	раве_дрована_дворе_трава_на_т	19	т	
20	аве_дрована_дворе_трава_на_тр	ре_трава_на_траве_дрована_дво	6	о	
21	ве_дрована_дворе_трава_на_тра	рована_дворе_трава_на_траве_д	25	д	
22	е_дрована_дворе_трава_на_трав	трава_на_траве_дрована_дворе_	9	—	
23	_дрована_дворе_трава_на_траве	траве_дрована_дворе_трава_на_	18	—	
24	дрована_дворе_трава_на_траве	_дворе_трава_на_траве_дрована	2	а	
25	рована_дворе_трава_на_траве_д	_дрована_дворе_трава_на_траве	23	е	
26	ована_дворе_трава_на_траве_др	_на_траве_дрована_дворе_трава	14	а	
27	вана_дворе_трава_на_траве_дро	_трава_на_траве_дрована_дворе	8	е	
28	ана_дворе_трава_на_траве_дров	_траве_дрована_дворе_трава_на	17	а	

В столбце 3 «Сортированные перестановки» фразы столбца 2 отсортированы в соответствии с лексикографическим порядком символов, в данном случае — в порядке «а», «в», «д», «е», «н», «о», «р», «т», «_». При этом сначала следует сортировать строки по первому символу, затем строки, у которых первые символы одинаковы, — по второму символу и т. д. Номера исходных строк приведены в столбце 4.

После этой сортировки необходимо выписать из каждой фразы последние символы, приведенные в столбце 5, и запомнить номер строки, в которой записана исходная фраза (указан в столбце 6).

Приведенное преобразование Барроуза–Уиллера позволяет восстановить исходную информацию по данным столбцов 5 и 6. Последовательность операций восстановления информации приведена в табл. 4.14. Исходной информацией являются символы, приведенные в правой части столбца 2 этой таблицы и обозначенные жирным шрифтом.

Символ «а» первым встречается 6 раз; символы «в» и «р» — по 4 раза; символы «д», «е», «н», «о» и «т» — по 2 раза; символ «_» — 5 раз. Ввиду того что сортировка фраз осуществлялась в соответствии с лексикографическим порядком первых символов, их последовательность соответствует правой части столбца 2, обозначенного как «1-е символы».

Символы, идущие вслед за первыми символами, приведены в столбце 3. Принцип их формирования поясним на примерах символов, следующих за первыми шестью символами «а» в столбце 2. Поскольку каждая из строк матрицы бы-

ла получена путем циклического сдвига предыдущей строки, символы первого и последнего столбцов образуют пары.

Таблица 4.14. Восстановление данных после преобразования Барроуза–Уиллера

№	1-е символы	2-е символы	3-и символы	28-е символы
1	2	3	4	29
0	а. р	ав р	ава р	ава_на_траве_ дрована_дворе_тр
1	а. р	ав. р	аве р	аве_дрована_ дворе_трава_на_тр
2	а. в	ан в	ана в	ана_дворе_трава_ на_траве_дро
3	а. н	а_ н	а_д н	а_дворе_трава_ на_траве_дро
4	а. в	а_ в	а_н в	а_на_траве_ дрована_дворе_трав
5	а. н	а_ н	а_р н	а_траве_дрована_ дворе_трава_н
6	в. о	ва о	ван о	вана_дворе_трава_ на_траве_дро
7	в. а	ва а	ва_ а	ва_на_траве_ дрована_дворе_трав
8	в. а	ве а	ве_ а	ве_дрована_дворе_ трава_на_трава
9	в. д	во д	вор д	воре_трава_на_ траве_дрована_д
10	д. _	дв _	дво _	дворе_трава_на_ траве_дрована_
11	д. _	др _	дро _	дрована_дворе_ трава_на_траве_
12	е. в	е_ в	е_д в	е_дрована_дворе_ трава_на_трава
13	е. р	е_ р	е_н р	е_трава_на_траве_ дрована_двор
14	н. а	на а	на_ а	на_дворе_ трава_на_траве_ дро
15	н. _	на _	на_ _	на_траве_дрована_ дворе_трава_
16	о. р	ов р	ова р	ована_дворе_ трава_на_траве_др
17	о. в	оп в	опе в	опе_трава_на_ траве_дрована_д
18	р. т	ра т	рав т	рава_на_траве_ дрована_дворе_т
19	р. т	ра т	рав т	раве_дрована_ дворе_трава_на_т
20	р. о	ре о	ре_ о	ре_трава_на_ траве_дрована_д
21	р. д	ро д	ров д	рована_дворе_ трава_на_траве_д
22	т. _	тр _	тра _	трава_на_траве_ дрована_дворе_

Таблица 4.14 (окончание)

1	2	3	4	29
23	т.....	тр.....	тра.....	траве_дрована_ дворе_трава_на_
24	—.....а	—д.....а	—дв.....а	—дворе_трава_на_ траве_дрована_
25	—.....е	—д.....е	—др.....е	—дрована_дворе_ трава_на_траве_
26	—.....а	—н.....а	—на.....а	—на_траве_ дрована_дворе_ трава_
27	—.....е	—т.....е	—тр.....е	—трава_на_траве_ дрована_дворе_
28	—.....а	—т.....а	—тр.....а	—траве_дрована_ дворе_трава_на_

Строки, в которых последними символами являются символы «а» справа можно дополнить так:

Строка 7 — в.....а|в;

Строка 6 — в.....а|в;

Строка 14 — н.....а|н;

Строка 24 — _.....а|_;

Строка 26 — _.....а|_;

Строка 28 — _.....а|_.

Это означает, что первые два символа в 6 строках столбца 3 (2-е символы) будут соответственно «ав», «ав», «ан», «а_», «а_» и «а_».

Подобным же образом, используя столбец 3, формируются третьи символы, как приведено в столбце 4, поскольку строки, в которых последними символами являются символы «а» справа, можно дополнить так:

Строка 7 — ва.....а|ва;

Строка 8 — ве.....а|ве;

Строка 14 — на.....а|на;

Строка 24 — _д.....а|_д;

Строка 26 — _н.....а|_н;

Строка 28 — _т.....а|_т.

Это означает, что первые три символа в 6 первых строках столбца 3 (2-е символы) будут соответственно «ава», «аве», «ана», «а_д», «а_н» и «а_т».

Таким образом, последовательно шаг за шагом определяются символы в каждой из строк табл. 4.14, в 14-й строке которой формируется искомая последовательность.

Описанный метод восстановления исходной информации весьма сложен и требует значительного объема оперативной памяти.

На самом деле получение исходной информации значительно упрощается при применении вектора обратного преобразования, принцип нахождения которого поясняется с использованием табл. 4.15.

Таблица 4.15. Получение Т-вектора обратного преобразования Барроуза–Уиллера

№	Первые и последние символы	№ новых строк	Новые строки	№ старых строк — Т-вектор	Дан-ные	№ п/п
1	2	3	4	5	6	7
0	а.....р	18	в.....а	7	р	0
1	а.....р	19	в.....а	8	р	1
2	а.....в	6	н.....а	14	в	2
3	а.....н	14	—.....а	24	н	3
4	а.....в	7	—.....а	26	в	4
5	а.....н	15	—.....а	28	н	5
6	в.....о	16	а.....в	2	о	6
7	в.....а	0	а.....в	4	а	7
8	в.....а	1	е.....в	12	а	8
9	в.....д	10	о.....в	17	д	9
10	д.....—	24	в.....д	9	—	10
11	д.....—	25	р.....д	21	—	11
12	е.....в	8	—.....е	25	в	12
13	е.....р	20	—.....е	27	р	13
14	н.....а	2	а.....н	3	а	14
15	н.....—	26	а.....н	5	—	15
16	о.....р	21	в.....о	6	р	16
17	о.....в	9	р.....о	20	в	17
18	р.....т	22	а.....р	0	т	18
19	р.....т	23	а.....р	1	т	19
20	р.....о	17	е.....р	13	о	20
21	р.....д	11	о.....р	16	д	21
22	т.....—	27	р.....т	18	—	22
23	т.....—	28	р.....т	19	—	23
24	—.....а	3	д.....—	10	а	24
25	—.....е	12	д.....—	11	е	25
26	—.....а	4	н.....—	15	а	26
27	—.....е	13	т.....—	22	е	27
28	—.....а	5	т.....—	23	а	28

Исследуя алгоритм проведения операций восстановления информации, можно сделать вывод о том, что при формировании строк путем объединения переданных последних символов (отмеченные во всех столбцах табл. 4.14 жирным шрифтом) с выявленными столбцами 1-го, 2-го и т. д. символов формируются новые строки, расположенные на других позициях матрицы. Например, из строки с номером «0», содержащей два первых и последний символы (см. табл. 4.14, столбец 3):

ав.....р,

формируется новая строка, первые три символа которой таковы (см. табл. 4.14, столбец 4):

рав.....

В данном случае эта новая строка располагается под номером 18.

Исследуя столбец 4 табл. 4.15, можно увидеть, что строки, перемещенные в соответствии с номерами столбца 3, имеют последние символы, отсортированные в соответствии с лексикографическим порядком. Первые же символы в этом столбце определяют вторые символы строк, приведенных в столбце 2.

Полученная строка номеров строк (см. столбец 5) — $T = \{7, 8, 14, 24, 26, 28, 2, 4, 12, 17, 9, 21, 25, 27, 3, 5, 6, 20, 0, 1, 13, 16, 18, 19, 10, 11, 15, 22, 23\}$ и является искомым вектором, содержащим номера позиций символов в строке, которую необходимо декодировать (см. столбцы 5 и 6 данной таблицы).

При использовании имеющегося номера искомой строки (в данном случае равный 14) вся строка определяется весьма просто.

$T[14] = 3$, и первым символом декодируемой строки является символ «н» (см. последний символ в строке 3 столбца 2). Затем определяется номер строки в векторе T под номером 3.

$T[3] = 24$, и вторым символом декодируемой строки является символ «а» (см. последний символ в строке 24 столбца 2). Сумма двух символов — «на».

Далее по аналогии:

$$\begin{aligned} T[24] &= 10 - \langle _ \rangle; T[10] = 9 - \langle \text{д} \rangle; T[9] = 17 - \langle \text{в} \rangle; T[17] = 20 - \langle \text{o} \rangle; \\ T[20] &= 13 - \langle \text{p} \rangle; T[13] = 27 - \langle \text{e} \rangle; T[27] = 22 - \langle _ \rangle; T[22] = 18 - \langle \text{т} \rangle; \\ T[18] &= 0 - \langle \text{р} \rangle; T[0] = 7 - \langle \text{а} \rangle; T[7] = 4 - \langle \text{в} \rangle; T[4] = 26 - \langle \text{а} \rangle; \\ T[26] &= 15 - \langle _ \rangle; T[15] = 5 - \langle \text{н} \rangle; T[5] = 28 - \langle \text{а} \rangle; T[28] = 23 - \langle _ \rangle; \\ T[23] &= 19 - \langle \text{т} \rangle; T[19] = 1 - \langle \text{р} \rangle; T[1] = 8 - \langle \text{а} \rangle; T[8] = 12 - \langle \text{в} \rangle; \\ T[12] &= 25 - \langle \text{е} \rangle; T[25] = 11 - \langle _ \rangle; T[11] = 21 - \langle \text{д} \rangle; T[21] = 16 - \langle \text{р} \rangle; \\ T[16] &= 6 - \langle \text{o} \rangle; T[6] = 2 - \langle \text{в} \rangle; T[2] = 14 - \langle \text{а} \rangle. \end{aligned}$$

4.3. Кодирование с преобразованием

Весьма часто перед кодированием изображения подвергаются дискретным преобразованиям. Обобщенно можно считать, что используется два вида преобразований:

- непосредственная обработка дискретных отсчетов, к которой относится кодирование с предсказанием;
- предварительное алгоритмическое преобразование с формированием различного вида спектров сигналов.

4.3.1. Кодирование с предсказанием

Широкое применение имеют методы **кодирования с предсказанием** (так называемая дифференциальная импульсно-кодовая модуляция **ДИКМ**).

ДИКМ очень близко связана с условным кодированием. Однако в данном случае вместо кодирования и передачи значения N -го пикселя b_N , передается величина $b_N - \hat{b}_N$, где \hat{b}_N — величина, являющаяся фиксированной функцией ранее переданных пикселов b_1, b_2, \dots, b_{N-1} (рис. 4.6).

Это означает, что предсказываемое значение N -го пикселя вычисляется на передаче с использованием формулы, включающей линейную комбинацию значений нескольких предыдущих пикселов [2.4, 2.56, 2.65–2.70].

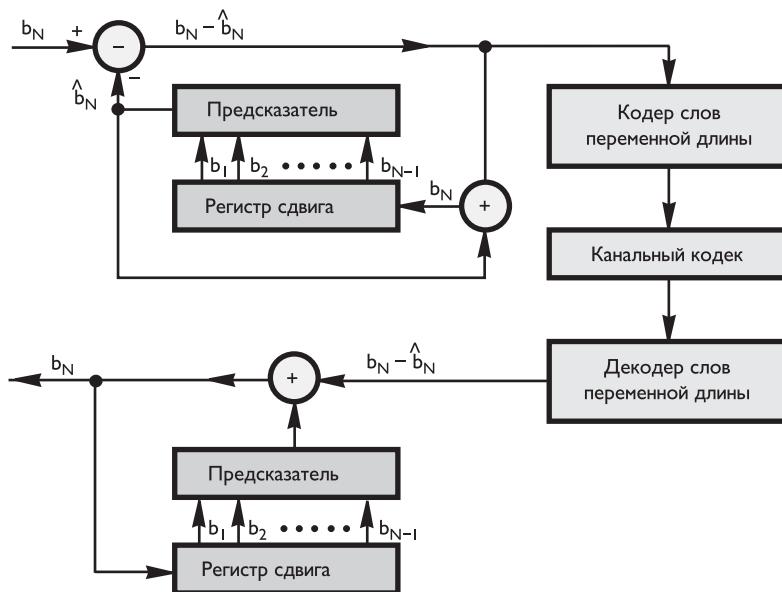


Рис. 4.6. Структура системы кодирования с предсказанием

Результатом вычисления является некая оценка («предсказание») значения N -го пикселя, которое, естественно, отличается от его истинного значения.

Таким образом, на передаче известно и истинное значение пикселя, и рассчитанное по формуле предсказание его значения. Остается вычесть одно из другого и передать по каналу связи.

На приемной стороне известно, по какой формуле рассчитывалось на передаче предсказание значения N -го пикселя. Поэтому приемнику также рассчитывает предсказанное значение N -го пикселя \hat{b}_N , используя значения нескольких предыдущих пикселов. Когда предсказание получено, приемнику остается лишь прибавить к нему значение разности между предсказанным и истинным значением N -го пикселя $b_N - \hat{b}_N$, полученное им из канала от передатчика, и сформировать таким образом истинное значение N -го пикселя.

Понятно, что такая система в принципе может работать при абсолютно любой формуле предсказания, даже при такой, которая дает результаты очень далекие от истинного значения предсказываемого пикселя, т. е. не обеспечивает вообще никакой точности — важно лишь, чтобы формула предсказания была одинаковой на передаче и на приеме. Однако чем точнее предсказание, тем меньше объем информационного потока, передаваемого по каналу, и таким образом, главная задача кодирования с предсказанием — получение дифференциального сигнала, который в среднем обладает малыми значениями и только изредка имеет большую величину.

Энтропия такого дифференциального сигнала мала, и, следовательно, мал битовый поток, необходимый для кодирования информации.

Например, эффективным является предсказатель, вычисляющий \hat{b}_N таким образом, чтобы предсказываемая величина была средневероятным значением

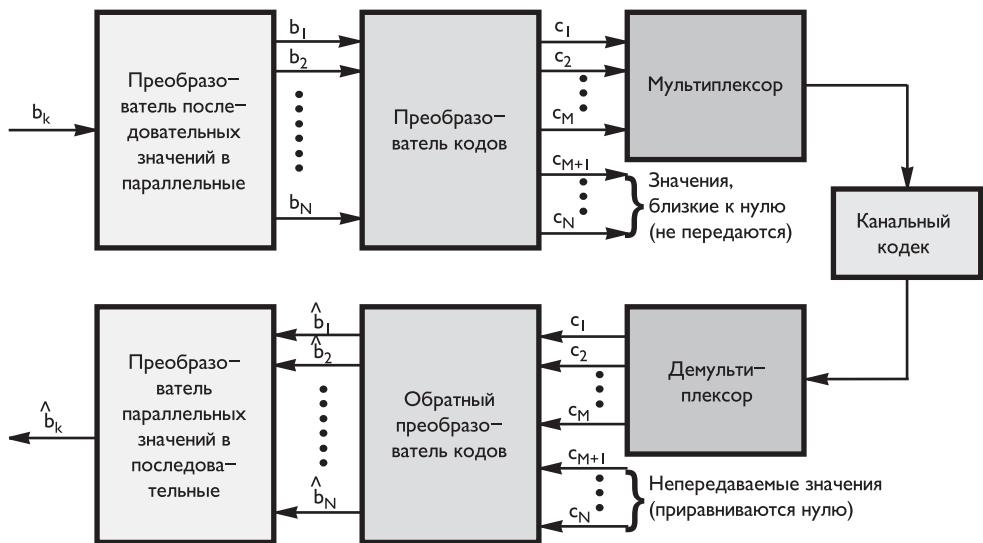


Рис. 4.7. Структурная схема кодирования с преобразованием

$$\hat{b}_N = \sum_{b_N} b_N \cdot P(b_N | b_1, \dots, b_{N-1}), \quad (4.23)$$

минимизирующими среднеквадратичную ошибку предсказания. Обычно используют линейный предсказатель

$$\hat{b}_N = \sum_{i=1}^{N-1} \alpha_i \cdot b_i, \quad (4.24)$$

в котором коэффициенты α_i часто выбирают путем минимизации среднего значения ошибки предсказания:

$$E(b_N - \hat{b}_N)^2 \Rightarrow \min. \quad (4.25)$$

При этом коэффициенты α_i могут быть получены путем решения системы уравнений:

$$\sum_{i=1}^{N-1} \alpha_i \cdot r_{ij} = d_j, \quad (4.26)$$

где $d_j = E(b_N, b_j)$, $r_{ij} = E(b_i, b_j)$, $j = 1, \dots, N-1$; $\|r_{ij}\|$ — корреляционная матрица, определяющая статистическую взаимосвязь между пикселями b_i и b_j .

4.3.2. Кодирование с дискретным преобразованием

Наиболее эффективным является **кодирование с дискретным преобразованием** N -мерных блоков \mathbf{b} ИКМ закодированных пикселов. Структурная схема такого кодирования приведена на рис. 4.7.

Обычно применяется линейное преобразование [4, 56, 65, 70–72]:

$$c_j = \sum_{m=1}^N t_{mj} \cdot b_m, \quad 1 \leq j \leq N, \quad (4.27)$$

где b_m — коды, характеризующие исходный блок пикселов изображения, c_j — коды, характеризующие преобразованные величины, $\|t_{mj}\| = \mathbf{T}$ — матрица преобразования.

В матричной форме это преобразование и обратное преобразование можно записать в виде:

$$\mathbf{c} = \mathbf{T} \times \mathbf{b}, \quad \mathbf{b} = \mathbf{T}^{-1} \times \mathbf{c}.$$

Главная цель преобразования — сделать как можно больше преобразованных величин c_j настолько малыми, чтобы они стали несущественными и не требовали бы кодирования и передачи.

Другой целью такого преобразования является минимизация статистических зависимостей между c_i и c_j ($i \neq j$), что обеспечивает само по себе устранение избыточности информации.

Если обратная матрица \mathbf{T}^{-1} равна транспонированной \mathbf{T}' ($\mathbf{T}^{-1} = \mathbf{T}'$), то преобразование является **ортонормированным**, или **унитарным**.

Примером такого преобразования является дискретное преобразование Фурье (ДПФ) [2.3, 2.4, 2.56, 2.65, 2.66, 2.73], при котором

$$t_{mk} = \frac{1}{\sqrt{N}} \cdot \exp\left[-\frac{2\pi i}{N} \cdot (k-1) \cdot (m-1)\right], \quad k, \quad m = 1, \dots, N. \quad (4.28)$$

Соотношение $\mathbf{T}^{-1} = \mathbf{T}'$ в данном случае эквивалентно

$$\sum_{k=1}^N t_{mk} \cdot t'_{lk} = \delta_{ml} = \begin{cases} 1, & m = l, \\ 0, & m \neq l. \end{cases} \quad (4.29)$$

Унитарное преобразование сохраняет энергию сигнала, т. е.

$$\sum_{l=1}^N |b_l|^2 = \sum_{m=1}^N |c_m|^2. \quad (4.30)$$

Это соотношение известно как равенство Парсеваля [2.4].

Используется большое количество унитарных преобразований. Так как обратное преобразование полностью восстанавливает исходные коды блока пикселов b_l , энтропия при преобразовании не меняется, т. е.

$$H(\mathbf{b}) = H(\mathbf{c}) \text{ бит/блок.}$$

К тому же, если преобразование устраняет статистическую зависимость между преобразованными величинами $\{c_1, \dots, c_N\}$, то

$$H(\mathbf{c}) = \sum_{m=1}^N H(c_m) \text{бит/блок.}$$

В этом случае оказывается весьма эффективным блочное кодирование (исключая, возможно, коэффициенты, энтропия которых $H(c_m) \ll 1$). До сих пор не найдено преобразование пиксельной информации, которое полностью устраивает статистическую зависимость между преобразованными величинами.

Однако парная статистическая корреляция может быть устранина использованием ортогонального линейного преобразования Кархунена–Лоэва (ПКЛ) или Хотеллинга [2.72–2.76]. Это преобразование позволяет оценить предельные степени сжатия информации, достижимые за счет снижения корреляции компонент.

Непосредственное применение ПКЛ в прикладных программах сжатия видеинформации не осуществляется по причине отсутствия быстрого алгоритма его вычисления и связанных с этим проблем. Однако исследование статистических свойств сигналов и цифровых изображений, в частности с помощью ПКЛ, облегчает поиск преобразования, наиболее близкого к оптимальному среди допускающих построение быстрого вычислительного алгоритма процессов.

Вычисление матрицы преобразования ПКЛ осуществляется следующим образом.

Предположим, в изображении имеется M блоков по N пикселов, значения которых равны b_{nm} ($1 \leq n \leq N; 1 \leq m \leq M$).

Первоначально рассчитывается ковариационная матрица $R = \|R_{ij}\|$, элементы которой равны

$$R_{ij} = \sum_{m=1}^M (b_{im} - \hat{b}_{im}) \cdot (b_{jm} - \hat{b}_{jm}), \quad 1 \leq i, j \leq N. \quad (4.31)$$

Заметим, что если пиксельные значения имеют среднее $\hat{b}_{im} = 0$, то отношение

$$r_{ij} = \frac{R_{ij}}{\sum_{m=1}^M (b_{im} - \hat{b}_{im})^2} \quad (4.32)$$

является коэффициентом корреляции между i -м и j -м пикселями.

Затем путем решения уравнения N -й степени

$$\begin{vmatrix} R_{11} - \lambda & R_{12} & \dots & \dots & R_{1N} \\ R_{21} & R_{22} - \lambda & \dots & \dots & R_{2N} \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ R_{N1} & R_{N2} & \dots & \dots & R_{NN} - \lambda \end{vmatrix} = 0 \quad (4.33)$$

определяются собственные значения ковариационной матрицы

$$\lambda_1, \lambda_2, \dots, \lambda_N \quad (\lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_N).$$

Вычисление собственных значений обычно осуществляется итерационными методами.

Ввиду того, что определитель системы уравнений

$$\begin{aligned} R_{11}t_{1k} + \dots + R_{1N}t_{Nk} &= \lambda_k t_{1k}, \\ R_{21}t_{1k} + \dots + R_{2N}t_{Nk} &= \lambda_k t_{2k}, \\ \dots & \dots \dots \dots \\ R_{N1}t_{1k} + \dots + R_{NN}t_{Nk} &= \lambda_k t_{Nk}, \end{aligned} \quad (4.34)$$

для $1 \leq k \leq N$ равен нулю, вычисление коэффициентов преобразования осуществляется следующим образом.

Первоначально один из коэффициентов, например t_{1k} , фиксируется, скажем, $t'_{1k} = 1$, количество уравнений при каждом значении k сокращается на одно, и они

вычисляются путем решения системы $N - 1$ линейных уравнений величины t'_{2k} , t'_{3k}, \dots, t'_{Nk} .

Затем вычисляются координаты коэффициентов преобразования путем нормировки:

$$t_{ik} = t'_{ik} \sqrt{\sum_{j=1}^N t'_{jk}^2}.$$

Заметим, что ковариационная матрица преобразованного изображения является диагональной, т. е. имеет значения не равные нулю только по диагонали, поскольку ПКЛ устраниет парные корреляционные связи.

Кроме декорреляции коэффициентов преобразования, ПКЛ имеет и другое полезное свойство: оно максимизирует количество коэффициентов, которые настолько малы, что не являются существенными.

Например, предположим, что коэффициенты ПКЛ (c_1, \dots, c_N) упорядочены по убыванию собственных значений $\lambda_1 \leq \dots \leq \lambda_N$. Также предположим, что из соображений сокращения объема передаваемой информации передаче подлежат только первые pN коэффициентов ($0 \leq p \leq 1$).

В этом случае в приемнике используется усеченное количество коэффициентов $(c_1, \dots, c_{pN}, 0, \dots, 0)$ и в результате преобразования вычисляются значения пикселов (b_1, \dots, b_N) .

Можно показать, что при этом среднеквадратичная ошибка преобразования определяется соотношением:

$$E \left[\frac{1}{N} \sum_{i=1}^N (b_i - \hat{b}_i)^2 \right] = E \left[\frac{1}{N} \sum_{m=pN+1}^N |c_m|^2 \right] = \frac{1}{N} \sum_{m=pN+1}^N \lambda_m.$$

Таким образом, ПКЛ минимизирует среднеквадратичную ошибку преобразования [2.4].

При кодировании реальных цифровых изображений часто используется так называемое дискретное косинусное преобразование (ДКП), реализующее парную декорреляцию пикселов, весьма близкую к преобразованию ПКЛ.

На базе применения ДКП построен **алгоритм JPEG**, являющийся стандартом кодирования полноцветных изображений [2.77–2.80].

Алгоритм разработан группой экспертов в области фотографии специально для сжатия 24-битных изображений. JPEG — Joint Photographic Expert Group — подразделение в рамках ISO — Международной организации по стандартизации.

ДКП преобразовывает изображение так, что полученные при преобразовании многие коэффициенты либо близки, либо равны нулю. Кроме того, благодаря несовершенству человеческого зрения, можно аппроксимировать коэффициенты более грубо без заметной потери качества изображения. Для этого используется квантование коэффициентов. При этом преобразовании теряется часть информации, но могут достигаться большие коэффициенты сжатия.

Существует специальная реализация JPEG без потерь — **Lossless JPEG**.

Стандарт Lossless JPEG ориентирован на полноцветные 24-битные или 8-битные в градациях серого изображения без палитры. Lossless JPEG рекомендуется применять в тех приложениях, где необходимо побитовое соответствие исходного и декомпрессированного изображений.

4.4. Статистика монохромных и цветных изображений

4.4.1. Статистика монохромных изображений без преобразования

Для оценки возможностей сжатия изображений их удобно представить как реализацию случайного процесса.

Статистика этих реализаций нестационарна и сильно изменяется в различных изображениях [2.4, 2.56, 2.81]. Так, для конкретного изображения статистика распределения уровней яркости может быть существенно неравномерной, и ее использование для кодирования, например, кодами переменной длины может обеспечить существенное сжатие изображений.

Однако плотность распределения вероятностей, усредненная по множеству изображений, является практически равномерной, и энтропия при квантовании пикселов на 256 уровней практически равна 8 битам на пиксел.

Таким образом, в общем случае использование статистик первого порядка для обеспечения сжатия изображений не является эффективным.

Более эффективно использование блочных и условных статистик отдельных изображений. Исследования различных изображений показывают:

- при увеличении размера блока энтропия на один пиксел уменьшается;
- условные энтропии с увеличением количества условий уменьшаются;
- для малых размеров блоков условная энтропия существенно меньше блочной энтропии на пиксел.

Следует заметить, что при 8-битовой информации о значении уровней пикселов для конкретного изображения трудно построить достоверную статистику высокого порядка, поскольку ограничено число пикселов. Так, например, при оценке вероятности блока, состоящего только из трех пикселов, число возможных уровней составляет $(2^8)^3 = 2^{24}$, а количество пикселов в изображении с числом пикселов в строке, равным 720, и с числом строк, равным 540, определяется неравенством $720 \times 540 < 2^{19}$.

Широко используемая мера статистической зависимости между элементами изображения — автокорреляционная функция, определяемая как среднее произведение смешенных в пространстве и времени пикселов блоков динамического (статического) изображения [2.4, 2.56, 2.82]:

$$R(x_1, y_1, t_1; x_2, y_2, t_2) = E\{b(x_1, y_1, t_1) \cdot b(x_2, y_2, t_2)\}, \quad (4.35)$$

где $\tau_x = x_2 - x_1$ — сдвиг вправо, $\tau_y = y_2 - y_1$ — сдвиг вниз, $\tau_t = t_2 - t_1$ — сдвиг во времени от пикселя (x_1, y_1, t_1) к пикселу (x_2, y_2, t_2) динамического изображения.

Автоковариационная функция изображения определяется как

$$\begin{aligned} K(x_1, y_1, t_1; x_2, y_2, t_2) &= \\ &= E\{[b(x_1, y_1, t_1) - E(b(x_1, y_1, t_1))] \cdot [b(x_2, y_2, t_2) - E(b(x_2, y_2, t_2))]\}, \end{aligned} \quad (4.36)$$

где $E(b(x, y, t))$ — средние значения яркостей пикселов изображения.

Если бы случайный процесс, порождающий изображения, был стационарным, то автокорреляционная функция зависела только от разностей, но не зависела

от того, в каких блоках изображения она вычисляется:

$$\begin{aligned} R(x_1, y_1, t_1; x_2, y_2, t_2) &= R(\tau_x, \tau_y, \tau_t); \\ K(x_1, y_1, t_1; x_2, y_2, t_2) &= K(\tau_x, \tau_y, \tau_t). \end{aligned} \quad (4.37)$$

Величина $K(0, 0, 0)$ определяет дисперсию процесса. Нормализованная автоковариационная функция

$$\rho(\tau_x, \tau_y, \tau_t) = K(\tau_x, \tau_y, \tau_t) / K(0, 0, 0)$$

дает полезную меру статистической зависимости между пикселами.

Энергетический спектр стационарного изображения по определению есть результат преобразования Фурье его автокорреляционной функции [2.83–2.88]:

$$F(\omega_x, \omega_y, \omega_t) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} R(\tau_x, \tau_y, \tau_t) \cdot \exp[-i(\omega_x \tau_x + \omega_y \tau_y + \omega_t \tau_t)] d\tau_x d\tau_y d\tau_t. \quad (4.38)$$

В телевидении пространственные и временные процессы формирования изображений разделяются. В данном случае:

$$R(\tau_x, \tau_y, \tau_t) = R_{xy}(\tau_x, \tau_y) \cdot R_t(\tau_t). \quad (4.39)$$

Для упрощения вычислений пространственную автокорреляционную функцию представляют в виде произведения автокорреляционных функций для каждой пространственной переменной:

$$R_{xy}(\tau_x, \tau_y) = R_x(\tau_x) \cdot R_y(\tau_y). \quad (4.40)$$

В изображениях часто встречаются горизонтальные и вертикальные структуры, поэтому такая аппроксимация оказывается вполне приемлемой.

Весьма часто нормализованную автоковариационную функцию аппроксимируют экспонентой:

$$\rho(\tau_x, \tau_y, \tau_t) = \exp\{-\alpha_x |\tau_x| - \alpha_y |\tau_y| - \alpha_t |\tau_t|\}, \quad (4.41)$$

где коэффициенты $\alpha_x, \alpha_y, \alpha_t$ характеризуют корреляционные свойства изображения соответственно между пикселами в одной строке, между строками и между кадрами (полями).

Если расстояние между пикселами в строке обозначить $\tau_x = nT_0$, между строками $\tau_y = mT_{\text{стр}}$, а между кадрами (полями) $\tau_t = pT_{\text{К(П)}}$, то это соотношение может быть записано в виде:

$$\rho(n, m, p) = \exp\{-\alpha_x T_0 |n| - \alpha_y T_{\text{стр}} |m| - \alpha_t T_{\text{К(П)}} |p|\}. \quad (4.42)$$

Полученные экспериментально коэффициенты корреляции в телевизионных изображениях между элементами строк, строками, кадрами и полями соответственно равны [2.82, 2.83]

$$\begin{aligned} \rho_0 &= 0,97; & \rho_{\text{стр}} &= 0,90; \\ \rho_{\text{К}} &= 0,92; & \rho_{\text{П}} &= 0,95. \end{aligned}$$

При этом соответствующие коэффициенты

$$\alpha_x T_0 = 0,0385; \quad \alpha_y T_{\text{стр}} = 0,138; \quad \alpha_t T_{\text{К}} = 0,108; \quad \alpha_t T_{\text{П}} = 0,0654.$$

Относительный энергетический спектр рассматриваемого процесса можно пред-



Рис. 4.8. Смазывание объектов в кадре

ставить в виде:

$$F(\omega_x, \omega_y, \omega_t) = \frac{8 \cdot \alpha_x \cdot \alpha_y \cdot \alpha_t}{(\alpha_x^2 + \omega_x^2) \cdot (\alpha_y^2 + \omega_y^2) \cdot (\alpha_t^2 + \omega_t^2)}. \quad (4.43)$$

Еще раз заметим, что автокорреляционная и связанные с ней функции могут существенно отличаться для различных изображений, а также внутри каждого изображения, что подтверждает нестационарность статистик изображения.

В телевизионных изображениях статистики в кадрах и их последовательностях могут существенно изменяться в зависимости от движения объектов и движения камеры. Даже в отдельном кадре имеются важные специфические особенности, характеризующие движущиеся детали, вследствие того, что в видеокамерах нет затворов, подобных используемым в фотоаппаратах, и время экспозиции практически равно кадровому интервалу.

Это приводит к тому, что быстро движущиеся объекты в каждом отдельном кадре оказываются сильно размытыми, как показано на рис. 4.8.

За счет использования чересстрочной развертки соседние строки кадра разделены во времени на интервал поля. За это время возможно существенное движение объекта, что приводит к снижению статистических зависимостей между строками ТВ-кадра. Смещение изображения за интервал поля за счет движения камеры иллюстрируется рис. 4.9.

Фактически же большая часть изображений в целом от кадра к кадру не меняется или меняется очень мало, что характеризуется близкими к единице значениями коэффициента корреляции.



Рис. 4.9. Смещение изображений в смежных полях при чересстрочной развертке

4.4.2. Статистика монохромных изображений с преобразованием

Статистики (особенно второго или более высоких порядков) сигналов в **коде-ратах с предсказанием** имеют существенно меньшие зависимости от изменения изображения [2.4, 2.56].

Они почти всегда могут быть охарактеризованы Лапласовым распределением

$$p(x) = \frac{\lambda}{2} \cdot \exp(-\lambda |x|), \quad (4.44)$$

вариация λ которого зависит в основном от вида предсказания. Среднее значение дифференциального сигнала всегда равно нулю, и его энтропия в основном определяется формой характеристики плотности вероятности в области нуля.

Следовательно, для одного и того же предсказателя один и тот же алгоритм кодирования может быть эффективным для различных изображений.

Следует заметить, что с ростом скорости движения объектов энтропия области движения растет при использовании межкадрового предсказателя, но падает при использовании внутрикадрового предсказателя за счет размазывания кадра. Однако в данном случае для улучшения межкадровых предсказаний зачастую используют методику предсказания движения, т. е. смещения соответствующих пикселов за время между кадрами.

Использование такого вида предсказания очень затрудняет чересстрочная развертка. В связи с этим при больших скоростях движения в системах с использованием межкадрового предсказания резко увеличивается загрузка канала

в результате увеличения объема передаваемой информации. Следует заметить, что использование нескольких потоков данных в одном канале передачи, как это применяется, например, при видеотелефонной связи, позволяет сгладить неравномерность загрузки канала.

При использовании в кодерах **унитарных преобразований**, как уже было сказано, достигаются две цели:

- сокращается количество передаваемых коэффициентов;
- коэффициенты, которые необходимо передавать, могут быть достаточно дескоррелированными, что обеспечивает уменьшение объема информации при блочном кодировании.

Исследования показали, что при широко используемых унитарных преобразованиях (Фурье, косинусного, Уолша–Адамара и др.) все коэффициенты преобразований, кроме первого, имеют Лапласово распределение плотностей вероятностей, что может быть использовано для выбора алгоритма квантования.

4.4.3. Статистика цветных изображений

Как и в случае монохромного изображения, статистики распределения уровней в сигналах основных цветов RGB могут быть существенно неравномерными. Однако плотности распределения вероятностей, усредненные по множеству изображений, также оказываются практически равномерными.

Для естественных изображений между сигналами основных цветов имеется существенная корреляция, поскольку они не содержат больших областей с насыщенными цветами. Таким образом, приблизительно одинаковые амплитуды сигналов основных цветов встречаются достаточно часто, и, следовательно, наибольшую вероятность появления имеют различные оттенки серого цвета.

По мере уменьшения окраски корреляции между сигналами R, G и B возрастают.

Эти корреляции практически могут быть оценены с использованием, например, преобразования Кархунена–Лоэва. При этом большая часть энергии будет сосредоточена в одной из компонент.

Хотя такое преобразование и изменяется от изображения к изображению, но компонента с максимальной энергией всегда состоит из примерно равных количеств основных цветов, т. е. практически соответствует сигналу яркости.

Результаты применения преобразования Кархунена–Лоэва к ряду тестовых изображений подтверждают, что используемые на практике комбинации сигналов основных цветов — YIQ (NTSC) и YUV (PAL, SECAM) имеют близкое к оптимальному распределение мощности, при том что большая часть этой мощности приходится на сигнал яркости [2.4, 2.81]. Этот факт иллюстрируется табл. 4.16.

Вследствие пониженной разрешающей способности глаза для передачи сигналов цветности в сравнении с сигналами яркости изображения используются более узкие частотные полосы. Считается, что доля цифрового потока, несущего информацию о цветности, может быть сведена к величине, не превышающей квадрат отношения ширины соответствующих полос [2.4, 2.66].

Так, в системе NTSC ширина полос сигналов I и Q составляет соответственно 36% и 12% от ширины полосы сигнала яркости, и цифровой поток может

Таблица 4.16. Распределение мощности по компонентам

Изображение	Система координат	Удельная мощность компонент, %		
		1	2	3
Лицо женщины	RGB	45,14	35,41	19,45
	YIQ	78,32	17,54	4,14
	Кархунена-Лоэва	85,84	12,1	2,06
Человеческие фигуры	RGB	51,55	31,09	17,36
	YIQ	84,84	13,81	1,35
	Кархунена-Лоэва	92,75	6,46	0,79

быть уменьшен по сравнению с потоком данных сигнала яркости Y до $[(0,36)^2 + (0,12)^2] \times 100 = 14\%$ просто за счет прореживания и передискретизации массивов величин I и Q в 0,36 и 0,12 раз соответственно сначала по горизонтали, а затем по вертикали.

Есть и дополнительные резервы сокращения относительного объема информации о цветности [2.69, 2.89]:

- во-первых, возможность более глубокого квантования благодаря меньшей чувствительности глаза к ошибкам уровней цветности по сравнению с ошибками уровня яркости;
- во-вторых, известно, что большие перепады цветности почти всегда совпадают с большими перепадами яркости (резкими контурами в изображении), причем обратное свойство не имеет места.

Последний факт может быть использован для дополнительного сокращения цифрового потока при раздельном кодировании контурно-объемной и текстурно-полутоновой составляющих изображения.

Статистики цветоразностных сигналов в кодерах с предсказанием или с унитарным преобразованием похожи на соответствующие статистики сигнала яркости, т. е. имеют Лапласово распределение. Однако эти статистики обладают большей вариацией λ и, следовательно, быстрее спадают. Это — результат меньшего динамического диапазона изменений цветоразностных сигналов для большинства изображений.

Измерения показывают [2.4, 2.89], что корреляция между компонентами Y , U , V в кодере с преобразованием практически отсутствует (за исключением специфической пространственной корреляции при резких переходах цвета).

4.5. Кодирование с сокращением объема передаваемой информации

До сих пор предполагалось, что визуальная информация дискретизируется и квантуется с достаточной точностью и полученные цифровые элементы изображения кодируются, передаются и декодируются без существенного изменения их величин. При этом обсуждалось лишь устранение статистической избыточности визуальной информации.

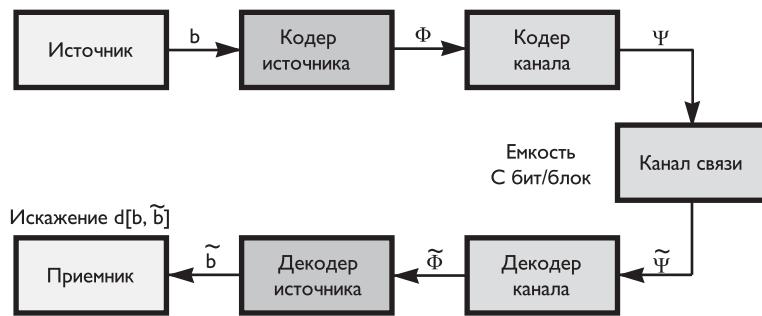


Рис. 4.10. Структура классической системы связи

Однако возможно дополнительное уменьшение объема передаваемой информации за счет устранения субъективной избыточности, т. е. той части информации, которую человек не может различить.

Устранение субъективной избыточности влечет за собой использование кодирования с уменьшением точности или информационного содержания цифровых данных до приемлемого уровня перед передачей, чтобы удовлетворить конкретным практическим требованиям.

Теоретический подход к кодированию с заданным критерием точности был развит Шенноном в 1948 г. в его знаменитом трактате об информационной теореме [2.2].

На рис. 4.10 приведена классическая система связи, содержащая источник информации, кодеры источника и канала, канал связи, декодеры канала и источника, приемник.

Источник формирует N -мерные блоки квантованных пикселов \mathbf{b} , которые кодер источника преобразует в символы Φ , а кодер канала — в символы Ψ . При этом на вход декодера канала поступают символы $\Psĩ$, преобразуемые в символы $\hat{\Phi}$, которые декодер приемника использует для получения пикселов $\hat{\mathbf{b}}$.

Пусть канал характеризуется емкостью C битов на пиксель или $N \times C$ битов на символ канала.

По определению, емкость канала — это верхняя граница потока битов, который может пропустить канал при пренебрежимо малых ошибках передачи. Более того, для любой скорости потока, меньшей C , поток ошибок может быть сделан сколь угодно малым при соответствующем кодировании в канале.

Пусть условная вероятность $P(\hat{\Psi} | \Psi)$ канала не зависит от ранее переданного символа, т. е. канал не обладает памятью, что практически всегда справедливо.

Тогда при плотности вероятности символов на входе канала $P(\Psi)$ **взаимная информация** определяется как

$$I(\Psi, \hat{\Psi}) = \sum_{\Psi, \hat{\Psi}} P(\Psi) \cdot P(\hat{\Psi} | \Psi) \cdot \log_2 \frac{P(\hat{\Psi} | \Psi)}{P(\Psi)}. \quad (4.45)$$

$I(\Psi, \hat{\Psi})$ — мера количества информации о символе Ψ бит/символ, которая возни-

кает при приеме символа $\hat{\Psi}$. Шенон показал, что емкость канала равна

$$C = \frac{1}{N} \cdot \max_{P(\Psi)} [I(\Psi, \hat{\Psi})] \text{ бит/пиксел,} \quad (4.46)$$

где максимум берется по всем возможным распределениям вероятности символа Ψ . Например, если в канале не возникает помех, то

$$P(\Psi, \hat{\Psi}) = \begin{cases} 1, & \hat{\Psi} = \Psi; \\ 0, & \hat{\Psi} \neq \Psi; \end{cases}$$

и взаимная информация равна энтропии

$$I(\Psi, \hat{\Psi}) = H(\Psi) \text{ бит/пиксел.} \quad (4.47)$$

В этом случае $I(\Psi, \hat{\Psi})$ максимальна, если все возможные символы Ψ равновероятны. Следовательно, если множество символов Ψ имеет J возможных членов, то

$$C = \frac{1}{N} \cdot \max_{P(\Psi)} [H(\Psi)] = \frac{1}{N} \cdot \log_2 J \text{ бит/пиксел,} \quad (4.48)$$

или $J = 2^{N \cdot C}$. Таким образом, множество символов Ψ может быть использовано для передачи двоичных слов длины $N \cdot C$ битов, и передача происходит со скоростью C бит/пиксел. Если канал шумит, передача информации со скоростью, близкой к C , еще возможна, но в кодер канала необходимо включать коды исправления ошибок.

Вопросы канального кодирования выходят за рамки настоящего материала, но, тем не менее, следует отметить, что чем мощнее и сложнее коды исправления ошибок, тем больше задержка передачи между входом и выходом кодека канала.

Для неискаженной передачи информации энтропия источника $H(\mathbf{b})$ должна быть меньше энтропии потока на выходе кодера $H(\Phi)$, которая, в свою очередь, должна быть меньше емкости канала:

$$N \cdot C \geq H(\Phi) \geq H(\mathbf{b}) \text{ бит/блок.} \quad (4.49)$$

Однако во многих ситуациях требование $\hat{\mathbf{b}} = \mathbf{b}$ не является ни необходимым, ни экономичным. В этом случае кодер источника должен производить оценку информации \mathbf{b} и создавать символ Φ с энтропией, меньшей, чем $H(\mathbf{b})$.

При $H(\Phi) < H(\mathbf{b})$ кодирование источника необратимо и получаемая информация $\hat{\mathbf{b}}$ не всегда равна информации источника \mathbf{b} . Таким образом, можно прийти к ошибке передачи $d(\mathbf{b}, \hat{\mathbf{b}})$, параметры которой (например, среднее значение, среднеквадратичное отклонение и др.) могут быть мерой общей работы системы связи.

Естественным представляется требование ограничения усредненного параметра искажения \hat{d} некоторой заданной величиной $\hat{d} \leq D$. При этом для нижней границы емкости канала $R(D)$ очевидно выполнение неравенства $C \geq R(D)$, где C — потенциальная емкость канала без помех.

Шенон показал [2.2], что при соответствующем кодировании источника и канала величину C можно сделать сколь угодно близкой к $R(D)$.

ГЛАВА 5

ВИЗУАЛЬНАЯ ИЗБЫТОЧНОСТЬ ИЗОБРАЖЕНИЙ

Устранение визуальной избыточности изображений является основным резервом сокращения передаваемой информации.

Для оптимизации процесса кодирования с точки зрения обеспечения передачи наименьшего объема информации необходимо, с одной стороны, не передавать избыточную информацию, а с другой — не допустить чрезмерной потери качества изображения.

Изменение визуального восприятия изображения связано как со специфическими особенностями физиологии зрения, так и с системными ограничениями, заложенными в используемых ТВ-стандартах. Особо отметим, что до сих пор не существует простой и адекватной модели визуального восприятия изображений, пригодной для оптимизации их кодирования, и по этой причине визуальные тесты остаются важными для оценки качества изображений.

Ошибки в изображении заметны глазу, если они превышают «порог заметности». Существуют различные подходы к определению «порога заметности» в различных условиях наблюдения изображений.

Следует отметить, что восприятие искажений, заведомо превышающих порог заметности, мало изучено. Различают пространственную и временную заметность искажений изображений.

5.1. Заметность пространственных изменений яркости изображений

Заметность искажений изображений зависит от ряда факторов объективного и субъективного характера [2.4, 2.94–2.98]:

- оптической неидеальности зрения;
- непроизвольных движений глаза (смещение точки внимания каждые 0,3–0,7 с и даже чаще) при восприятии неподвижных изображений;
- средней яркости фона;
- временных и пространственных вариаций яркости в окрестности рассматриваемого элемента;
- пространственной и временной формы сигнала изображения и др.

Важнейшим параметром, определяющим диапазон изменений сигнала, является максимальная визуально воспринимаемая яркость изображения.

Общая яркость цветного экрана определяется выбранной цветностью белого. Исследования показывают, что едва заметным искажениям белого соответствуют довольно большие, порядка 6–8% изменения сигналов основных цветов [2.4, 2.99].

Контраст изображения характеризуется отношением величины максимальной яркости изображения и яркости наиболее темных его участков.

Телевизионное изображение оценивается как хорошее при контрасте 30, качество его значительно улучшается при увеличении контраста до 100. При дальнейшем повышении контраста качество телевизионного изображения улучшается незначительно.

Качество воспроизводимых градаций яркости в диапазоне между максимальной и минимальной яркостями изображения определяют требования к шкале квантования сигнала яркости.

При исследовании порога заметности пространственных изменений яркости изображения обычно используют детали изображения, имеющие угловой размер порядка $1,5^\circ$ — при отношении расстояния наблюдения к высоте изображения на экране, равном 6, размер детали составляет порядка $1/8$ ширины экрана.

Предположим, имеется изображение с тремя одинаковыми круглыми деталями, расположенными в левом верхнем углу, в центре и в правом нижнем углу. На сером фоне окружающей среды ($1/2$ от максимальной яркости — код 128 при 8-битовой шкале) эти детали четко различимы при их яркости, например, равной $1/4$ от максимальной яркости изображения (код 64). Изменяя яркость расположенных в углах деталей в разные стороны, отметим, что **едва заметные изменения яркости** возникают при изменении кода, характеризующего пиксели деталей, на 4–6 единиц.

Следовательно, при изменении кодов на 6 единиц соотношения яркости элементов изображения в левом верхнем углу, в центре и в правом нижнем углу (по отношению к максимальному значению яркости экрана) соответственно равны $58/255 : 64/255 : 70/255$. При этом, как следует из рис. 5.1, изменения яркости едва заметны.

Аналогичный результат получается при яркости деталей большей, чем яркость окружающей среды, как показано на рис. 5.2 (соотношения — $186/255 : 192/255 : 198/255$).

Заметность изменений яркости резко возрастает, если указанные выше три детали на изображении разместить рядом (рис. 5.3). В таком случае изменения яркости оказываются заметными, если уровни сигналов отличаются лишь на две единицы. Еще большая заметность изменений яркости имеет место, если яркость фона (ближайшего участка) близка к яркости деталей (рис. 5.4).

Итак, порог заметности пространственных изменений яркости зависит от многих факторов — яркости деталей изображения, яркости фона, относительного положения деталей различной яркости, условий внешнего освещения [2.94, 2.95, 2.99, 2.100].

В целом высокие пороги заметности имеют место в очень темных или очень ярких участках изображения. На среднеярких, серых участках заметность изменений яркости наибольшая.

Неоднородность фона, на котором передается рассматриваемый элемент изображения, создает маскирование пространственных изменений яркости.



Рис. 5.1. Пример визуальной заметности изменений яркости удаленных друг от друга темных деталей на сером фоне

На рис. 5.5 показано изображение вертикальных белых линий, расположенных на различных расстояниях от вертикальной белой широкой полосы.

Слева от первой левой белой широкой полосы линия располагается на расстоянии одного пикселя, а справа — на расстоянии двух пикселов. Слева от второй белой широкой полосы линия располагается на расстоянии трех пикселов, а справа — на расстоянии четырех пикселов. У последующих двух широких белых полос линии располагаются соответственно на расстоянии пяти, шести, семи и восьми пикселов. Нетрудно заметить, что визуальная яркость этих линий возрастает при увеличении расстояния от маскирующей поверхности.

На некотором расстоянии от указанных выше вертикальных белых линий располагаются линии, визуальная яркость которых подобрана равной яркости близлежащих белых линий. При этом чем ближе белая линия располагается к белой полосе, тем меньше ее визуальная яркость.

Этот эффект, получивший название эффекта пространственного маскирования был использован в некоторых экспериментах для повышения эффективности ДИКМ [2.4].

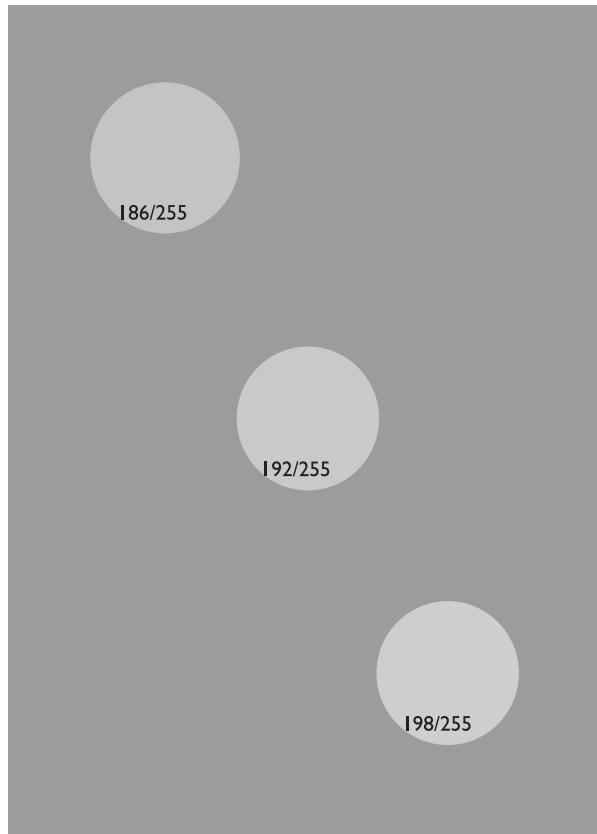


Рис. 5.2. Пример визуальной заметности изменений яркости удаленных друг от друга светлых деталей на сером фоне

Пороговая заметность изменений яркости существенно зависит от формы сигнала, от относительного положения его искажений [2.4, 2.94, 2.99].

На рис. 5.6 приведено изображение нескольких вертикальных полос, форма сигнала которых имеет вид продискретизированной синусквадратичной функции.

Сигнал располагается на темном фоне (64 дискрета) и имеет размах в 128 дискретов. Справа и слева от линий располагаются эхо-сигналы двух полярностей (справа — отрицательный, а слева — положительный), расположенных на различных расстояниях от основного сигнала (2 пикселя, 4 пикселя, 8 пикселов и 16 пикселов). Их уровни установлены таким образом, что соответствуют однаковой визуальной пороговой заметности.

При этом эхо-сигналы, расположенные на расстоянии 8 и 16 пикселов, едва заметны при размахе примерно в 2% относительно основного сигнала, на расстоянии 4 пикселов — 4%, а на расстоянии 2 пикселов — более 8%.

Потенциально достижимое качество телевизионного изображения определяется параметрами ТВ стандарта. Стандарт практически не ограничивает точности воспроизведения приведенной яркости на деталях крупных и средних раз-

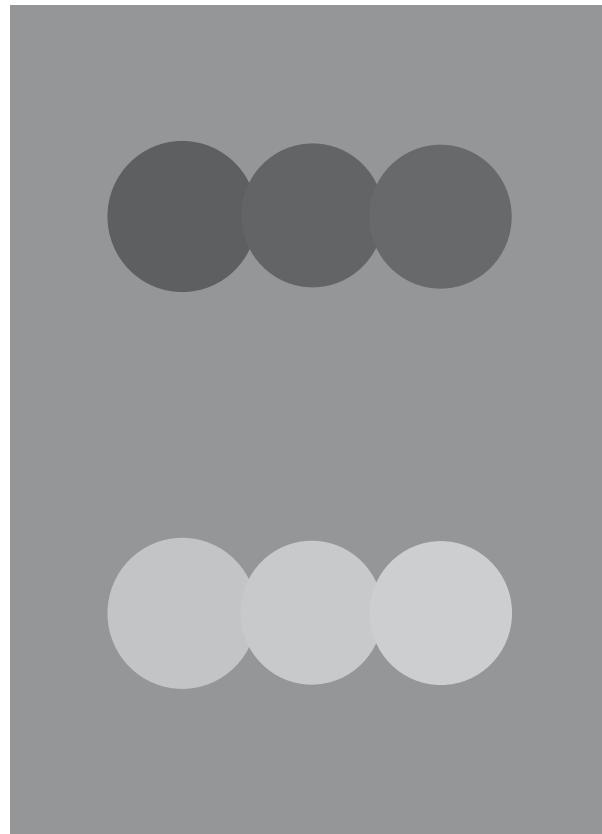


Рис. 5.3. Пример визуальной заметности изменений яркости совмещенных друг с другом темных (как на рис. 5.1) и светлых (как на рис. 5.2) деталей на сером фоне

меров. Он ограничивает главным образом точность воспроизведения мелких деталей и резких переходов, т. е. четкостные параметры изображения [2.94, 2.95, 2.99].

Четкость по горизонтали (вдоль строк) пропорциональна отношению полосы частот сигнала яркости к частоте строк, т. е. числу строчных гармоник в видеоспектре.

Верхнюю граничную частоту видеоспектра $f_{\text{гр}}$ определяют из условия, что один период сигнала этой частоты воспроизводит два элемента изображения (например, белый и черный штрихи).

Отсюда очевидно, что [2.99]

$$f_{\text{гр}} = \frac{L_{\text{стр}}}{2\Delta_{\text{гор}}} \cdot \frac{1}{T_{\text{стр}}(1 - \alpha)}, \quad (5.1)$$

где $L_{\text{стр}}$ — размер активной части строки, $\Delta_{\text{гор}}$ — размер одного горизонтального элемента, $T_{\text{стр}}$ — длительность строчного интервала, α — относительная величина пассивной части строки. Если учесть, что

$$L_{\text{стр}} = k \cdot h, \quad T_{\text{стр}} = 1/(Z \cdot n),$$

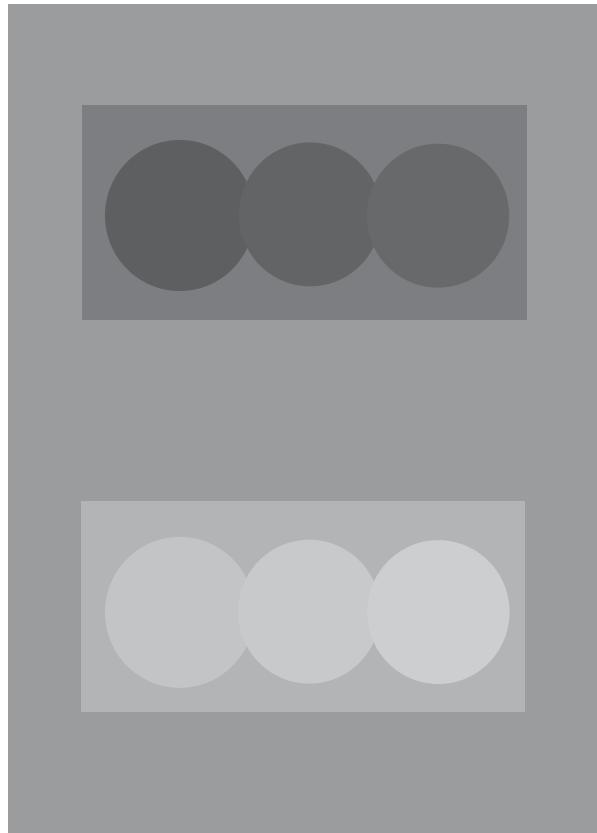


Рис. 5.4. Пример визуальной заметности изменений яркости совмещенных друг с другом темных и светлых деталей (как на рис. 5.3) на фоне, близком к яркости деталей

где $k = 4/3$ — формат кадра, h — его высота, Z и n — число строк в кадре и число кадров в секунду, причем

$$h = Z(1 - \Theta)\delta_{\text{верт}},$$

где Θ — относительная величина пассивной части кадра, $\delta_{\text{верт}}$ — шаг развертки по вертикали, то

$$f_{\text{гр}} = \frac{\delta_{\text{верт}}}{\Delta_{\text{гор}}} \cdot \frac{knZ^2}{2} \cdot \frac{1 - \Theta}{1 - \alpha}. \quad (5.2)$$

Член $K = \frac{\delta_{\text{верт}}}{\Delta_{\text{гор}}}$ — так называемый коэффициент Келла [2.99, 2.101] учитывает, что из-за дискретности развертки по вертикали реальный размер вертикального элемента $\Delta_{\text{верт}}$ превосходит шаг развертки $\delta_{\text{верт}}$. Строго говоря, согласно теореме отсчетов должно выдерживаться соотношение $\Delta_{\text{верт}} \leq \delta_{\text{верт}}$.

Но поскольку в стандартном ТВ-вещании фильтрация оптического изображения по вертикали ни до развертки в камере, ни после развертки в приемнике не осуществляется, то указанное соотношение не выполняется.

По этой причине, с одной стороны, возникают искажения — муары, связанные с перекрытием исходного и преобразованного спектров из-за отсутствия

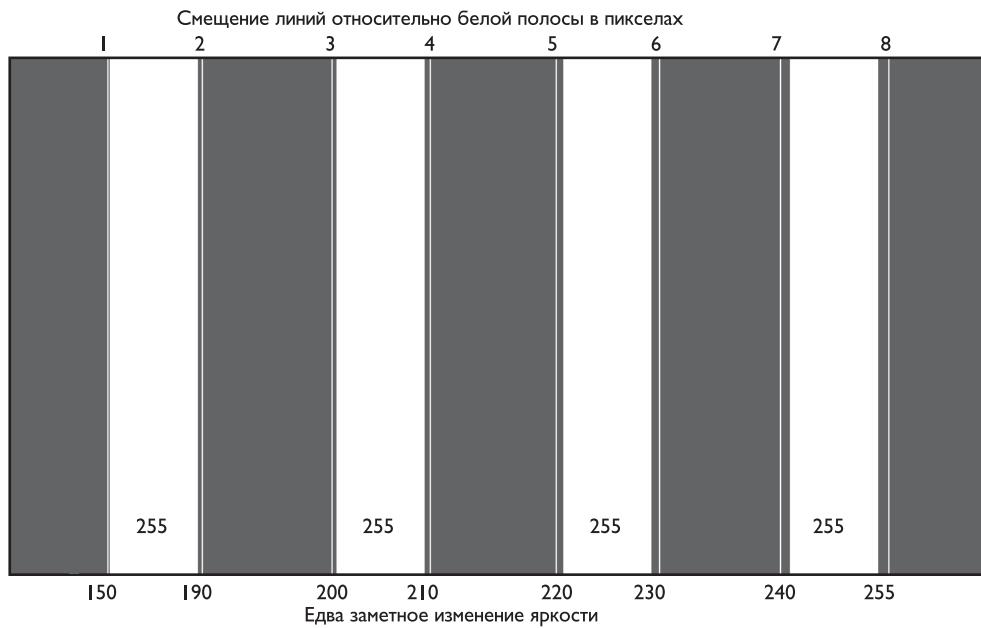


Рис. 5.5. Изменение визуальной яркости линий в зависимости от их относительного смещения от маскирующей полосы

префильтрации, и заметная строчная структура, связанная с сохранением преобразованного спектра из-за отсутствия постфильтрации.

Но с другой стороны, отсутствие фильтрации делает возможным воспроизведение более мелких деталей до $\delta_{верт}$ при благоприятном фазовом положении этих деталей относительно строчного раstra.

Очевидно коэффициент Келла может быть рассчитан из формулы, определяющей $f_{гр}$ при заданной ее величине. Так, при стандарте развертки 625/25 и $f_{гр} = 6$ МГц — $K = 0,81$, при $f_{гр} = 5$ МГц — $K = 0,67$. При стандарте развертки 525/30 и $f_{гр} = 4,2$ МГц — $K = 0,68$.

Существенную роль играет и чересстрочность разложения: так как четное и нечетное поля возникают в разные моменты времени, то визуально наблюдаемый межстрочный интервал оказывается большим, чем указанная величина $\delta_{верт}$; возникают также межстрочные мерцания, ухудшающие четкость по вертикали.

В цифровом телевидении при использовании построчного разложения и эффективной постфильтрации четкость по горизонтали и вертикали будет **строго одинакова**.

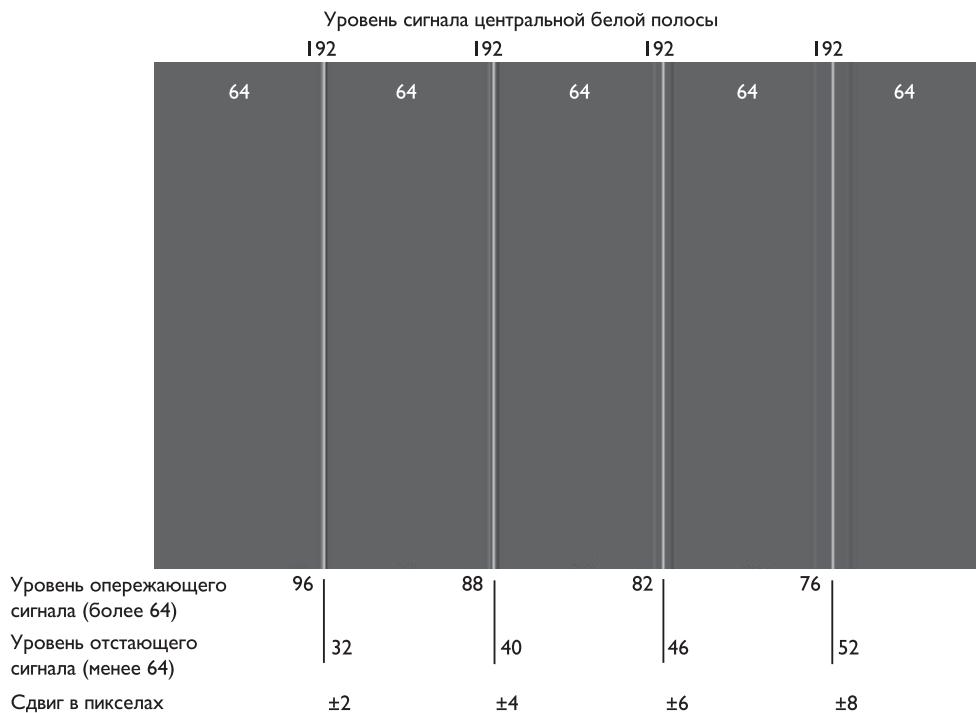


Рис. 5.6. Зависимость заметности эхо-сигналов от расстояния до основного сигнала

5.2. Заметность временных изменений яркости изображений

Как уже было отмечено выше, на качестве воспроизведения движущихся объектов в телевидении особенно сказываются два фактора:

- использование чересстрочной развертки;
- размытие объектов в последовательности кадров из-за нарушения потенциального рельефа в ТВ-камере при быстром движении.

На рис. 4.9 приведен фрагмент изображения и его увеличенный участок, поясняющие смещение движущегося изображения за интервал поля: контур движущейся детали искажен за счет смещения в строках 2-го поля относительно строк 1-го поля.

На рис. 4.8 изображены фрагменты движущегося изображения в трех соседних кадрах, поясняющие размытие в кадре за счет движения его деталей.

Помимо ухудшения качества изображения за счет сканирования считывающего луча в ТВ-камере восприятие движущегося объекта зависит от того, отслеживает ли глаз его движение.

Пространственное разрешение также заметно падает при резкой смене сюжета — приблизительно в 10 раз на длительность 0,5 сек [2.4, 2.102]. Этот факт

может быть использован для обеспечения эффективного сжатия телевизионных изображений.

Следует заметить, что параметры системы телевизионного вещания являются доминирующим фактором ухудшения пространственного восприятия движущихся изображений, если глаз следует за объектом. В противном случае изменения качества изображений связаны с особенностями зрения. Однако не существует количественных критериев, позволяющих снижать разрешающую способность воспроизведения «неотслеживаемых» объектов изображения с целью повышения эффективности кодирования.

5.3. Влияние помех

Мешающее действие помех на изображении при одном и том же их эффективном значении и одинаковых условиях наблюдения в основном зависит от распределения мощности помехи по спектру, разрешающей способности глаза, яркости и цветности участков воспроизводящего устройства, на котором наблюдаются помехи [2.4, 2.103].

Визуальное восприятие флюктуационной помехи связано с используемыми методами обработки и передачи изображений, поскольку эти процессы в значительной мере определяют форму спектра помех и распределение их амплитуд.

Субъективная оценка воздействия флюктуационных помех на изображение зависит от многих факторов — формы, цвета и размера деталей изображения, внешней освещенности экрана, дистанции просмотра и т. д.

Относительная степень мешающего действия различных составляющих флюктуационных помех характеризуется так называемыми весовыми функциями помех $\Phi(f)$.

При этом визуально воспринимаемая мощность помехи P_B характеризует ее заметность. Этот термин употребляется в смысле выходной мощности некоторого светочувствительного прибора, аналогичного человеческому глазу:

$$P_B = \int_0^{f_{\text{рп}}} P(f) \cdot \Phi(f) df, \quad (5.3)$$

где $P(f)$ — спектральная плотность мощности помехи.

Фильтры, реализующие весовую функцию помех, называются взвешивающими. В телевидении они нашли широкое распространение.

На рис. 5.7 приведены схемы, параметры и характеристики затухания взвешивающих фильтров, используемых в монохромном (кривая 1, фильтр — справа сверху) и цветном (кривая 2, фильтр — справа снизу) телевидении при передаче сигналов стандартных систем NTSC, PAL, SECAM [2.104–2.106].

Влияние помех на качество воспроизводимых изображений обычно характеризуют выраженным в дБ отношении Ψ_{Π} размаха сигнала яркости U_Y к эффективному напряжению помехи $U_{\text{эфф}}$, измеренному на выходе взвешива-

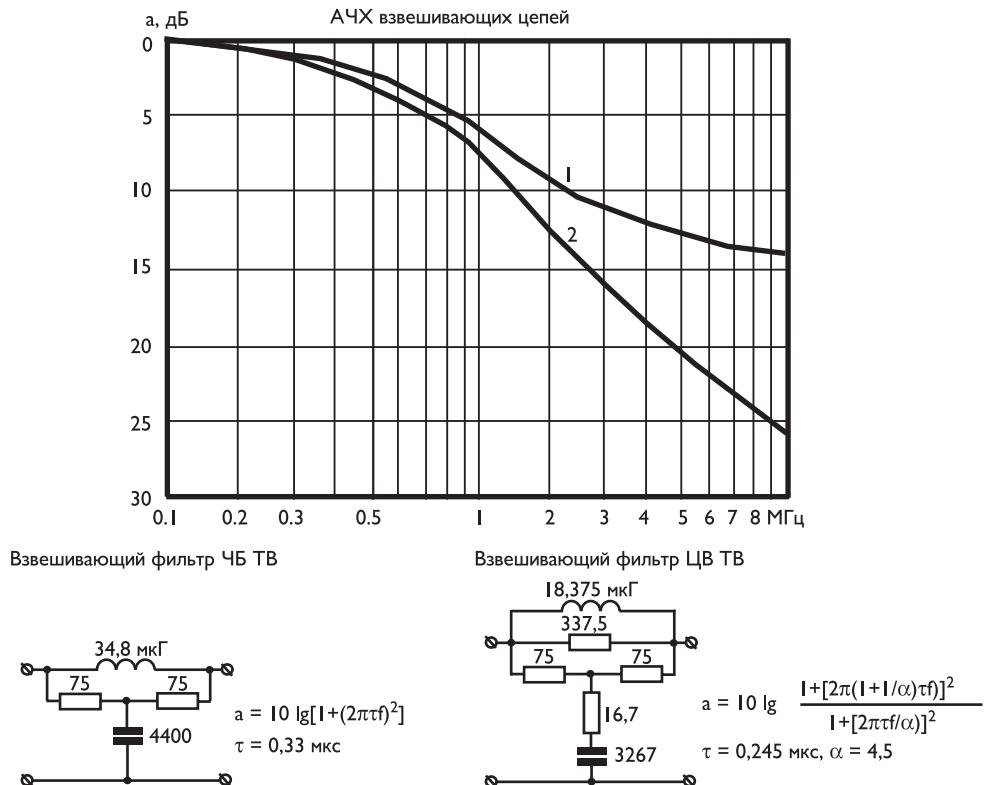


Рис. 5.7. Взвешивающие фильтры помех и их характеристики

ящего фильтра [2.103]:

$$\Psi_{\Pi} = 20 \lg \frac{U_Y}{U_{\text{эфф}}} \text{ (дБ).} \quad (5.4)$$

Общепринято, что при отношении сигнала к помехе более 52 дБ случайные помехи незаметны, а при менее чем 26 дБ они сильно мешают.

Еще раз отметим, что квантование исходных пикселов изображения приводит к ошибкам восстановленного изображения в виде случайного шума, который практически невидим при 8-битовом квантовании.

При уменьшении числа битов заметность шума увеличивается. Отметим также, что заметность шума квантования в сильной степени зависит от наличия флюктуационных помех в исходном изображении. Это следует учитывать при цифровом кодировании изображений.

5.4. Специфические особенности восприятия цветов

В отличие от объективного описания цветов в рамках колориметрической теории, субъективное восприятие цвета является сложной проблемой, и пока не предложены алгоритмы оценки визуально воспринимаемого цвета.

Экспериментальные исследования показывают, что на визуально воспринимаемый цвет влияют следующие факторы [2.94, 2.95, 2.98, 2.99, 2.107]:

- спектральный состав света, который излучается источником;
- спектральное распределение излучения, попадающего в глаз наблюдателя от всех объектов, находящихся в поле зрения;
- пространственное распределение и форма объектов;
- наличие цветовой памяти глаза (хроматическая адаптация) и др.

Количественная оценка всех этих факторов является невыполнимой задачей. Экспериментально показано, что восприятие цвета обычно можно характеризовать тремя факторами [2.99, 2.107–2.109]: для источников света — светимостью, цветовым тоном и насыщенностью, для объектов — освещенностью, цветовым тоном и насыщенностью.

Светимость характеризуется интенсивностью излучаемого источником светового потока, а освещенность определяет долю отраженного или пропущенного объектом падающего света. Эти два понятия отражают одну и ту же характеристику цветового восприятия объекта — яркость.

Цветовой тон — субъективно воспринимаемый признак цвета, который позволяет отнести его по сходству к тому или иному спектральному или пурпурному цвету. Численно его определяют доминирующую длиной волны, т. е. длиной волны монохромного излучения, которое в смеси с белым дает данный цвет (для пурпурных цветов задают длину волны дополнительного цвета, дающего белый цвет в смеси с данным).

Насыщенность — признак цвета, выражающий степень его отличия от белого цвета той же яркости. Численно она характеризуется чистотой цвета P , определяемой отношением яркости доминирующего спектрального цвета к полной яркости смеси. По определению насыщенность максимальна для спектральных и пурпурных цветов ($P = 1$) и минимальна — для белого цвета ($P = 0$).

Цветовой тон и насыщенность изображения достаточно просто можно оценить с использованием диаграммы цветности [2.110].

Если цвет объекта на этой диаграмме характеризуется некоторой точкой (x_0, y_0) , то пересечение линии, соединяющей эту точку с точкой, характеризующей цвет стандартного источника белого (например, D), с границей локуса определяет доминирующий спектральный цвет, т. е. цветовой тон. Длина отрезка между точкой (x_0, y_0) и точкой белого, отнесенными к длине линии, соединяющей точку белого с пересечением границы локуса, характеризует насыщенность цвета объекта.

Для практического использования диаграммы цветности важно знать **пороги цветоразличения**, т. е. минимальные изменения цветности, различаемые глазом человека.

Как показали измерения Мак-Адама и Райта [2.98, 2.111], связь между порогом цветоразличения и изображающим его расстоянием на диаграмме x, y существенно нелинейна — она изменяется до 30 раз от одного края диаграммы к другому.

В 1960 г. МКО приняла диаграмму uv , на которой различие цветовых порогов существенно меньшее.

Переход от координат x, y к координатам u, v осуществляется линейным преобразованием:

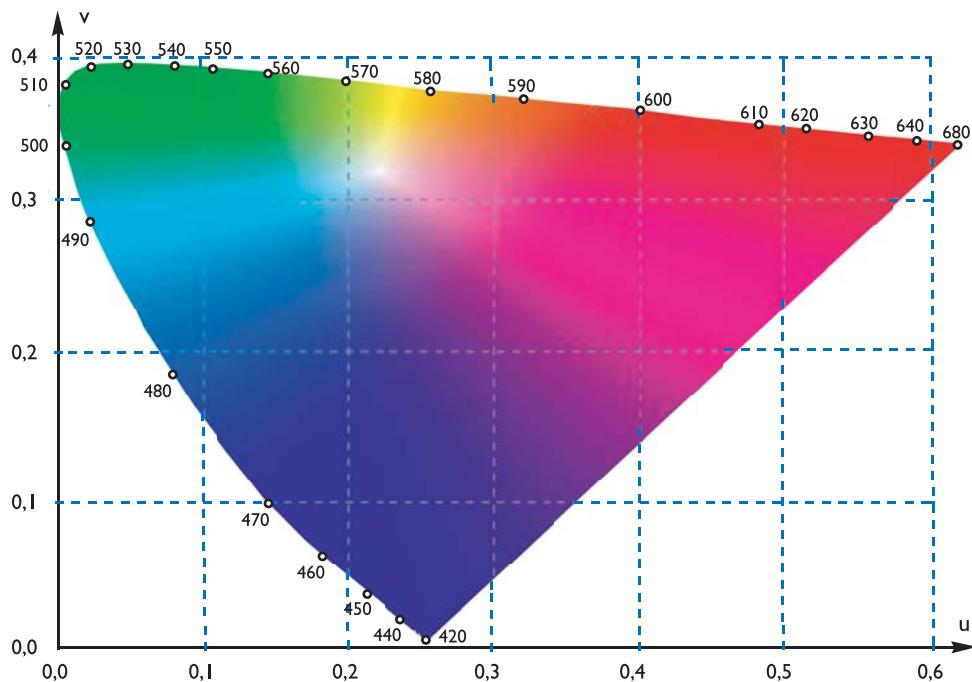


Рис. 5.8. Диаграмма цветности МКО-1960

$$u = \frac{4x}{-2x + 12y + 3}; \quad v = \frac{6y}{-2x + 12y + 3}. \quad (5.5)$$

На рис. 5.8 изображена диаграмма цветности МКО-1960, использование которой позволяет получить более надежные данные для установления параметров воспроизводимых на экране приемного устройства цветных изображений.

Усредненный порог цветоразличения установлен равным 0,0038 единиц шкал u , v (порог Мак-Адама). При этом участком неразличимых на глаз цветностей на диаграмме uv считается кружок с диаметром, равным одному порогу.

Пользуясь этим определением порога цветоразличения, можно подсчитать, что на площади цветового треугольника может быть размещено порядка 3900 кругов диаметром равным порогу Мак-Адама [2.99, 2.112].

Таким образом, стандартным кинескопом может быть воспроизведено менее четырех тысяч цветностей (человек способен различить примерно 12 тысяч цветностей).

Из приведенных данных вытекает, что в стандартных ТВ-системах достаточно передавать до четырех тысяч цветностей, для чего можно использовать два шестиразрядных кода, определяющих цветоразностные сигналы.

Вдоль линии спектральных цветов укладывается около 300 порогов Мак-Адама, а вдоль линии пурпурных цветов — 130. Вдоль периметра же цветового треугольника размещается лишь 256 цветовых тонов.

Все приведенные числа в силу зависимости цветовых порогов от цветности, яркости, содержания изображения и условий наблюдения могут рассматривать-

ся лишь как ориентировочные, но, тем не менее, представляют большой практический интерес и могут быть использованы для повышения эффективности кодирования цветных изображений.

Поскольку не существует никакой теории, поясняющей условия восприятия цветов, изменяющихся в пространстве и во времени, для оценки свойств зрения проводился ряд экспериментов.

На рис. 5.9–5.11 приведены едва заметные изменения насыщенности зеленого, красного и синего цветов соответственно. В центре экрана круг имеет 75%-ю насыщенность, насыщенность круга в левом верхнем углу уменьшена примерно на 5%, а в правом нижнем — увеличена примерно на 5%.

Заметность изменений насыщенности существенно увеличивается, если круги расположены рядом и образуется поле сравнения (рис. 5.12–5.14).

На рис. 5.15–5.17 приведены едва заметные изменения цветового тона для дополнительных цветов — желтого, голубого и пурпурного соответственно. В центре расположен круг соответствующего цвета 100%-й насыщенности. В левом верхнем углу изменение желтого цвета — происходит в сторону зеленого, голубого — в сторону синего, пурпурного — в сторону красного. В правом нижнем углу изменение желтого цвета в сторону красного, голубого — в сторону зеленого и пурпурного — в сторону синего. При этом едва заметные искажения вызываются изменениями размаха цветоразностных сигналов не менее чем на пять процентов.

На рис. 5.18–5.20 приведены эти же круги дополнительных цветов, расположенные рядом. Заметность изменений цветового тона увеличивается.

Отмечается маскирование цветных относительно мелких деталей как яркостной составляющей фона изображения, так и цветовой компонентой. Что касается временного восприятия цвета, то известно, что вариации цветности менее заметны, чем вариации яркости. Наиболее заметны изменения зеленого цвета, затем — красного, и наименее всего — синего.

5.5. Методы субъективной оценки качества изображений

Отдельные характеристики изображения — яркость, цветность, контраст, форма переходов и другие — могут быть измерены объективно с помощью специальных измерительных приборов. Однако общее качество изображения можно оценить исключительно посредством субъективных экспертиз или на их основе [2.113–2.115].

При проведении экспертиз особое внимание уделяется выбору испытательных изображений, числу и квалификации экспертов, условиям наблюдения, порядку проведения экспериментов, принципам обработки их результатов и т. д.

В телевидении в качестве испытательных используют изображения, близкие к наиболее встречающимся (изображение лица, группы людей, пейзажи, архитектурные композиции и т. д.) [2.103, 2.116, 2.117].

Особые требования к испытательным изображениям предъявляются в системах с цифровой обработкой сигналов. При этом испытательные таблицы, используемые для оценки качества передачи сигналов аналогового телевидения,

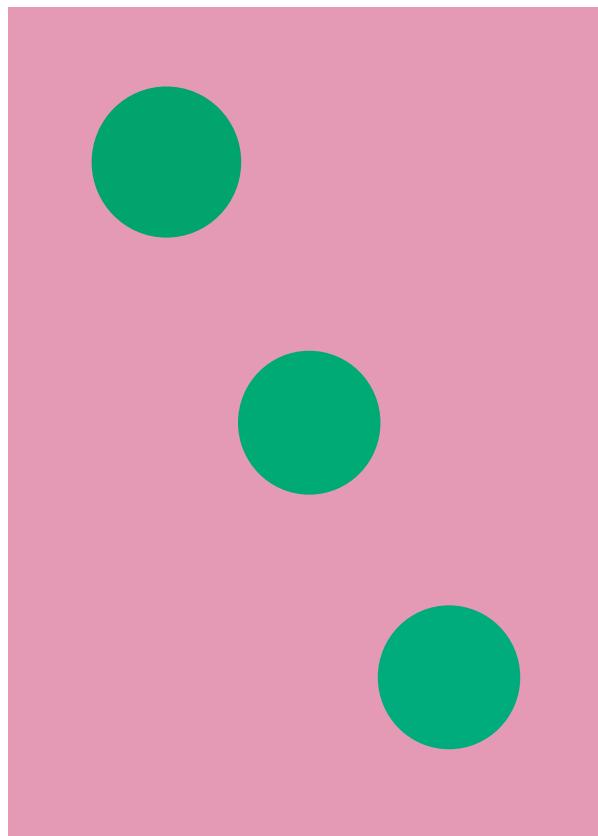


Рис. 5.9. Изменение насыщенности далеко расположенных друг от друга деталей зеленого цвета (сверху вниз — 70%, 75% и 80%)

оказываются малопригодными для оценки передачи изображений в цифровых системах связи.

Зачастую применяются движущиеся эталонные изображения. В этом случае они должны содержать объекты, движущиеся в различных направлениях и с различными скоростями, в частности движущиеся титры.

Должны также использоваться изображения, получаемые в процессах масштабирования и панорамирования.

При выборе наблюдателей обычно отдают предпочтение неспециалистам [2.99, 2.103, 2.117–2.120].

В основном используются следующие методы субъективной оценки изображений:

- **метод эквивалентности**, при котором наблюдатель самостоятельно уравнивает качество двух по-разному искаженных изображений;
- **метод различения**, при котором наблюдатель определяет абсолютный или разностный порог заметности искажений;
- **метод оценок**, когда наблюдатель оценивает качество изображений по

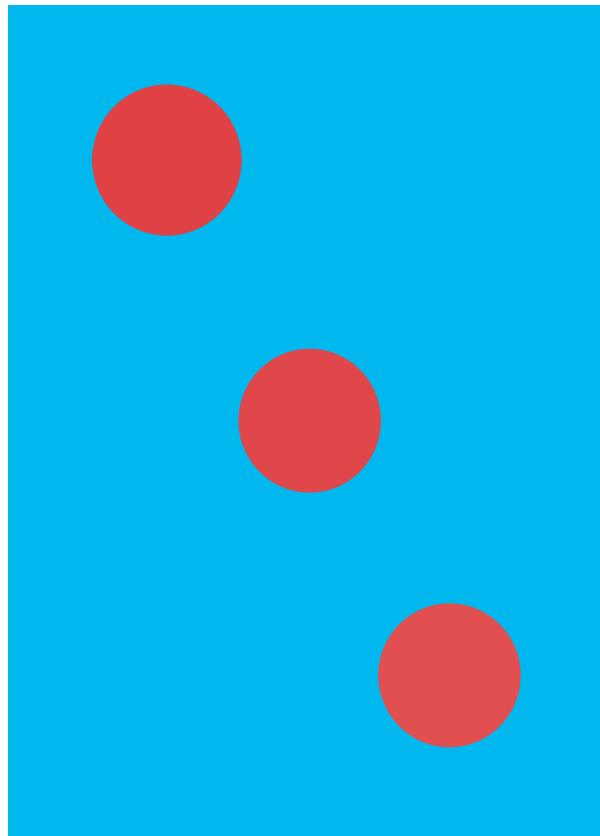


Рис. 5.10. Изменение насыщенности далеко расположенных друг от друга деталей красного цвета (сверху вниз — 70%, 75% и 80%)

определенной шкале, при этом в начале эксперимента предъявляется «опорное» изображение, качество которого принимается соответствующим высшей оценке шкалы;

- **метод сравнения**, когда наблюдатель сопоставляет качество двух изображений, давая оценки по шкале сравнения.

Первые два метода применяются для определения заметности отдельных видов искажений и построения их характеристик, например таких, как кривая заметности помех, функция восприятия яркости и др. Непосредственно для оценки качества изображения они не используются.

Метод оценок и метод сравнения используются в телевидении, их шкалы регламентированы в документах МККР [2.114, 2.115] и сведены в табл. 5.1.

Иногда вместо пятибалльных шкал оценок применяют девятибалльную шкалу со ступенями оценок через 0,5:

$$5; 4,5; 4; 3,5; 3; 2,5; 2; 1,5; 1.$$

Хотя оценки с промежуточными значениями (4,5; 3,5; 2,5; 1,5) и не имеют соб-

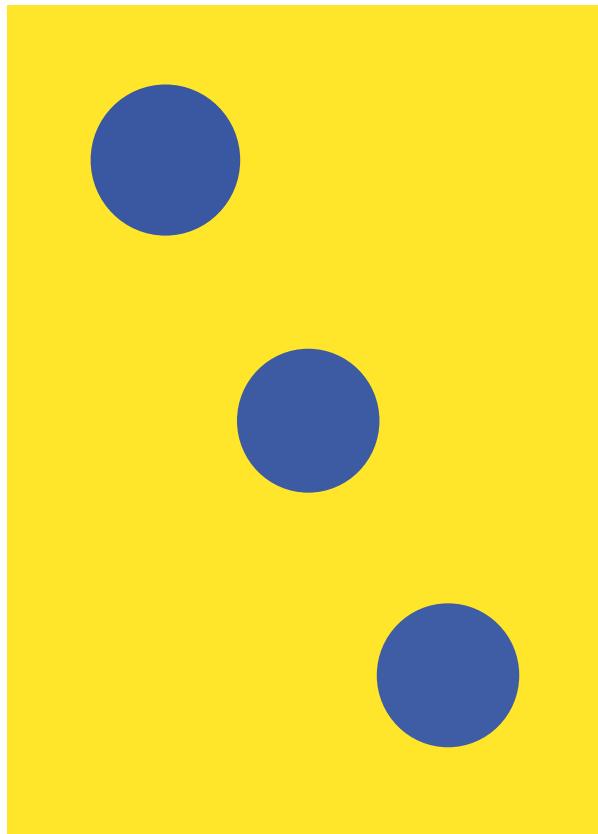


Рис. 5.11. Изменение насыщенности далеко расположенных друг от друга деталей синего цвета (сверху вниз — 70%, 75% и 80%)

ственных определений, но используются достаточно просто. Например, оценка 4,5 может быть использована как оценка «между отлично и хорошо».

Таблица 5.1. Основные шкалы оценок и сравнения

Пятибалльные шкалы оценки		Семибалльная шкала сравнения
Шкала качества	Шкала ухудшений	
5 — отлично	5 — незаметно	+3 — намного лучше
4 — хорошо	4 — заметно, но не мешает	+2 — лучше
3 — удовлетворительно	3 — заметно, немного мешает	+1 — немного лучше
2 — плохо	2 — мешает	0 — одинаково
1 — очень плохо	1 — сильно мешает	-1 — немного хуже
		-2 — хуже
		-3 — намного хуже

В ходе субъективных экспертиз обычно получают большое количество оценок, и вопрос их рационального представления и интерпретации результатов имеет большое значение.

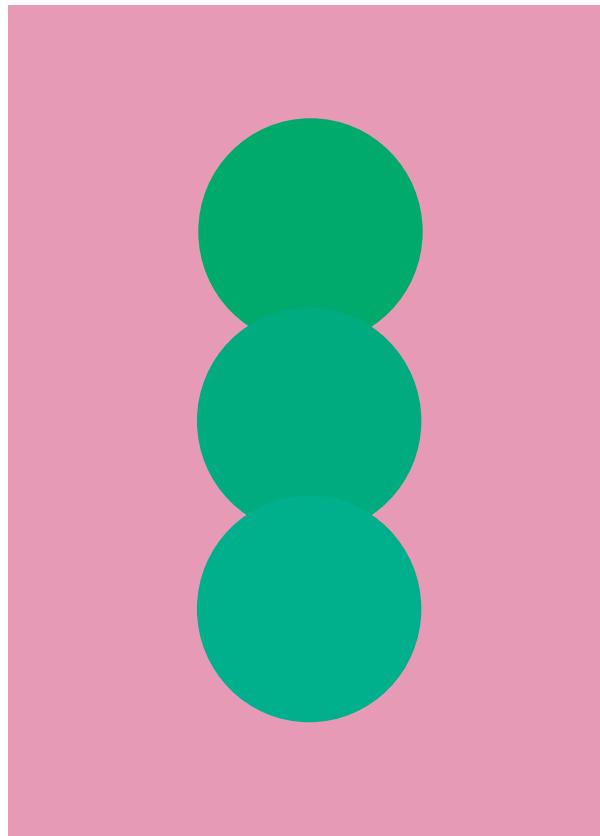


Рис. 5.12. Изменение насыщенности наложенных друг на друга деталей зеленого цвета (сверху вниз — 70%, 75% и 80%)

Чаще всего используют среднюю оценку в баллах, вычисляемую по формуле [2.99, 2.103]:

$$Q = \frac{1}{N} \cdot \sum_{i=1}^r a_i n_i,$$

где N — общее число оценок, n_i — число оценок в a_i баллах, r — общее число баллов.

Результаты оценки качества изображений, полученные для ряда значений искажений одного и того же вида, обычно представляются в виде зависимости средней оценки качества изображения от величины искажения.

5.6. Компьютерные методы хранения изображений

Компьютерные технологии хранения, обмена и воспроизведения черно-белых и цветных изображений различного разрешения реализуются в виде файлов гра-

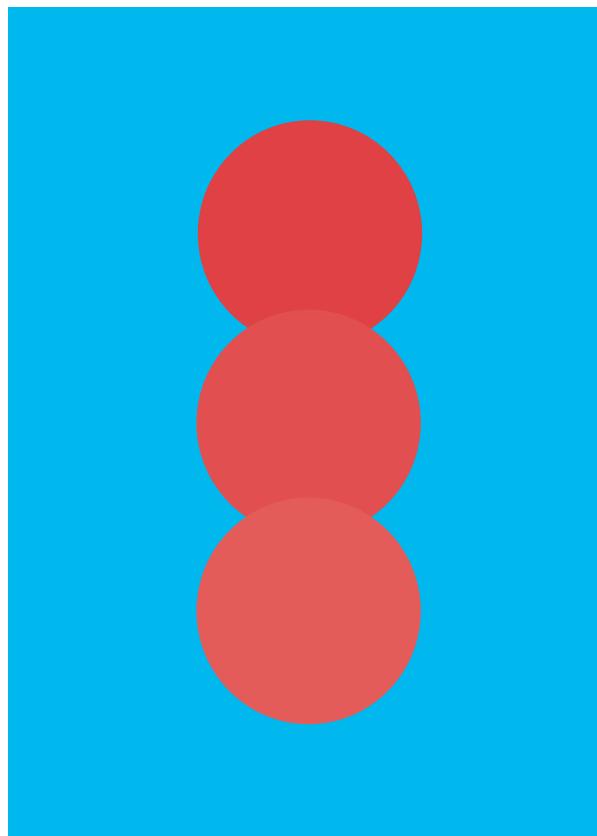


Рис. 5.13. Изменение насыщенности наложенных друг на друга деталей красного цвета (сверху вниз — 70%, 75% и 80%)

фических данных, способы организации которых обычно называют графическими форматами [2.121].

После записи в файл изображение превращается в некоторую последовательность данных, причем формат этих данных может существенно измениться, например, в результате преобразования файла.

Графические данные традиционно подразделяют на *векторные* и *растровые*. Первые из них обычно применяют для представления прямых и кривых линий, многоугольников с заданием их цветности и толщины линий. Растровые данные представляют собой набор числовых значений, определяющих уровни и цвет отдельных пикселов. Воспроизведение в определенной последовательности массива числовых значений пикселов обеспечивает реализацию изображений на компьютерном или телевизионном дисплее. Векторные и растровые графические данные могут содержаться в едином формате, называемом *метафайловым* форматом.

В настоящее время известно несколько десятков графических форматов. Как правило, графические форматы включают в себя какие-либо методы сжатия изображений без потерь (RLE, LZW, Lossless JPEG и др.) или с потерями (JPEG,

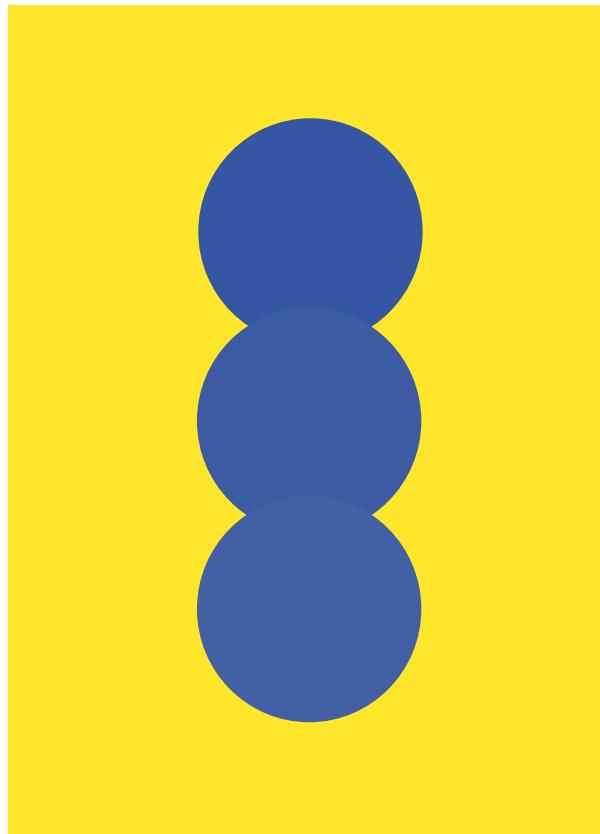


Рис. 5.14. Изменение насыщенности наложенных друг на друга деталей синего цвета (сверху вниз — 70%, 75% и 80%)

фрактальное кодирование и др.). Помимо статистического и визуального сокращения избыточности изображений алгоритмы их сжатия включают различные методы эффективной упаковки данных с тем, чтобы они занимали меньший объем дискового пространства.

В табл. 5.2 приведены параметры различных файлов хранения цветного изображения «Гавань» (рис. 5.21), имеющего размер 640×512 пикселов ($640 \times 512 \times 3 = 983\,040$ байтов).

Метафайловый формат AI (Adobe Illustrator) [2.121] — формат, используемый известной и широко распространенной программой создания изображений, поддерживающей приложения большинства настольных издательских систем. В качестве формата чисел используется американский стандартный код для обмена информацией ASCII (American Standard Code for Information Interchange), компрессия информации отсутствует. Файлы AI состоят из последовательности записей, которые могут быть комментариями, данными, командами или комбинациями команд и данных. Воспроизводимые цвета — без ограничения.

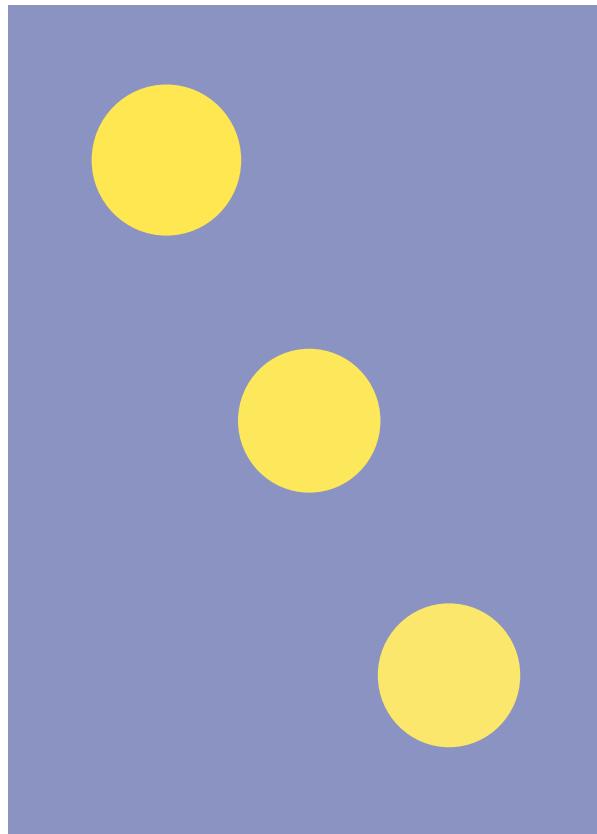


Рис. 5.15. Изменение цветового тона далеко расположенных друг от друга деталей желтого цвета

Масштабируемый векторный формат SVG (Scalable Vector Graphics) [2.122] предназначен для описания двумерной векторной и смешанной векторно-растровой графики. SVG-файлы хорошо поддаются сжатию. Существует возможность увеличить любую часть изображения SVG без потери качества. Дополнительно, к элементам SVG-файла возможно применять фильтры — специальные модификаторы для создания эффектов, подобных применяемым при обработке растровых изображений (размытие, выдавливание, сложные системы трансформации и др.). SVG — открытый стандарт. Воспроизводимые цвета — без ограничения. В SVG-документах доступно использование растровой графики. Имеется возможность вставлять элементы с изображениями в форматах PNG, GIF или JPG. В SVG реализована анимация с помощью языка SMIL (Synchronized Multimedia Integration Language).

Формат SVGZ является расширением формата SVG, обеспечивая некоторое сжатие файла с использованием метода GZIP [2.123], относящегося только к текстовой части информации. Графика, видео, архивы и прочие нетекстовые данные не сжимаются. Тем не менее экономия может быть весьма существенной.

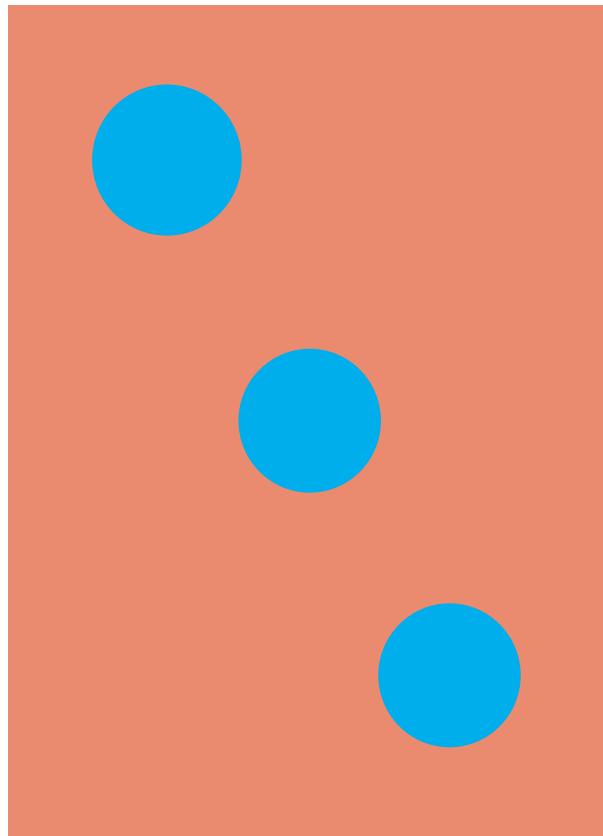


Рис. 5.16. Изменение цветового тона далеко расположенных друг от друга деталей голубого цвета

Растровый формат TIFF (Tag Image File Format) [2.121] — стандартный метод хранения монохромных, индексированных, полутоночных и полноцветных изображений в моделях RGB и CMYK с восьми- и шестнадцатибитовыми каналами. Его модификации 4.0–6.0 поддерживают обработку цветных RGB-изображений с цветовым форматом до 24 битов/пикселя. TIFF — один из многоцелевых и разносторонних растровых форматов, обладающих возможностью расширения и поддержки различных схем сжатия цифровых данных — RLE, LZW, CCITT, JPEG. TIFF-файлы содержат заголовки файла изображений IFH (Image File Header), директории файла изображений IFD (Image File Directory) и растровые данные. Допускается возможность существования файла, не содержащего растровых данных. Директория файла изображений содержит информацию о ширине, высоте и пиксельном формате изображения, количестве цветовых плоскостей и типе сжатия данных.

Формат PXR (Pixar) [2.124] предназначен для высокоразвитых современных графических станций и программ трехмерной графики, используется для обмена со специализированными графическими станциями Pixar, ориентированными на трехмерное моделирование и анимацию. Поддерживаются только полутоночные и полноцветные RGB-изображения. Сокращение избыточности не применяется.

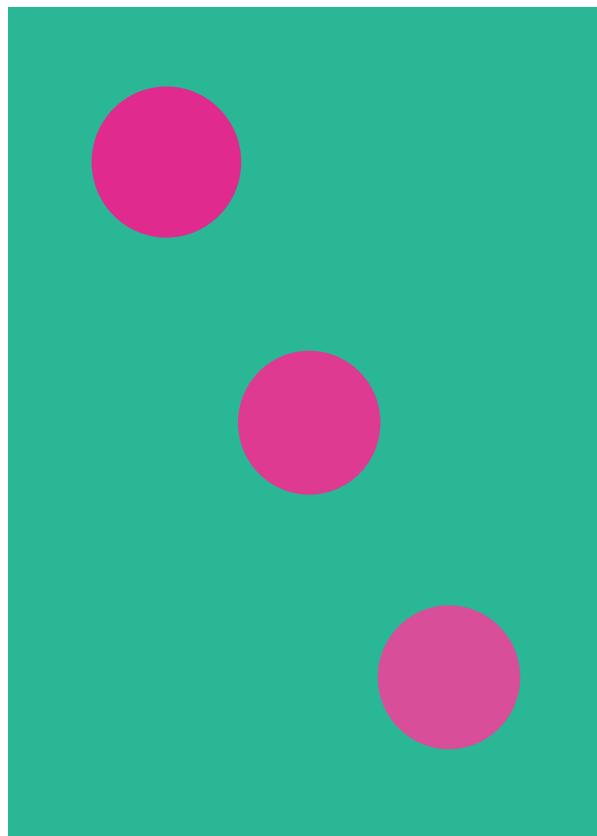


Рис. 5.17. Изменение цветового тона далеко расположенных друг от друга деталей пурпурного цвета

Метафайловый формат EMF (Microsoft Enhanced Metafile) [2.125] — относительно новый формат. При сохранении в формате EMF растровые изображения переводятся в векторную форму. Если простые растровые образы передаются после векторизации достаточно точно, то аппроксимация картинок с плавными цветовыми переходами, градиентными заливками и полутонами иногда требует очень большого количества векторных объектов. Это замедляет перерисовку экрана и выполнение всех вычислительных операций. Файлы EMF экономно расходуют дисковое пространство. Максимальная глубина цвета — 32 бита/пиксел, сжатие не предусматривается.

Растровый формат TGA (Truevision Graphics Adapter, Targa Graphics Adapter Image) [2.121] широко применяется для обработки и хранения изображений, данные которых имеют глубину до 32 бита/пиксел. Он известен также под сокращенными названиями VST, VDA, ICB, TPIC. Хорошо документированный формат широкого применения, схема сжатия недостаточно эффективна — используется алгоритм сжатия RLE. Формат TGA использует три метода упорядочения данных изображения: псевдоцвет, direct-color и truecolor. Цветовые таблицы псевдоцветных изображений определяют каждый цвет в виде отдельного

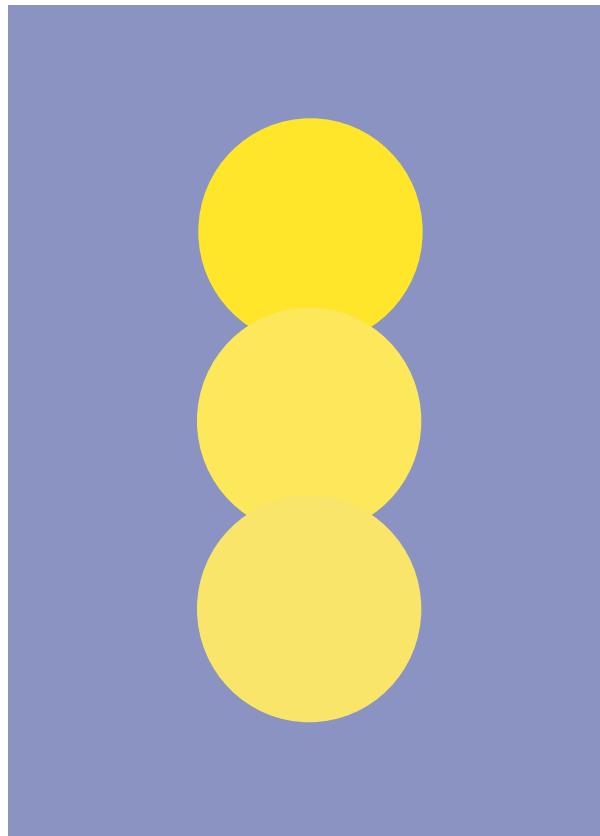


Рис. 5.18. Изменение цветового тона наложенных друг на друга деталей желтого цвета

элемента, причем значения отдельного цветового канала индивидуально недоступны. Изображения direct-color подобны псевдоцветным изображениям, однако в них каждый цветной канал RGB записывается как отдельный элемент и может быть модифицирован индивидуально — значения каждого пикселя содержат три индекса, по одному на каждый канал цветовой таблицы. Изображения truecolor (полноцветные) хранят цветовую информацию пикселя непосредственно в данных изображения, для их отображения цветовая таблица не применяется.

Растровый формат BMP (Microsoft Windows Bitmap) [2.121] — это собственный формат Windows, использующийся для хранения практически всех растровых данных. Формат известен также как DIB, Windows BMP, Windows DIB, Compatible Bitmap. В настоящее время используется множество версий формата BMP, которые разрабатывались для компьютеров, базирующихся на процессорах Intel, и по этой причине общей их чертой является хранение данных в порядке байтов «little endian». Данные об изображениях имеют цветовую глубину до 32 бита/пиксель при максимальных размерах до $2^G \times 2^G$ пикселов. Файлы BMP содержат заголовок файла, информационный заголовок, цветовую палитру и растровые данные. Несжатые данные изображения представляют собой либо

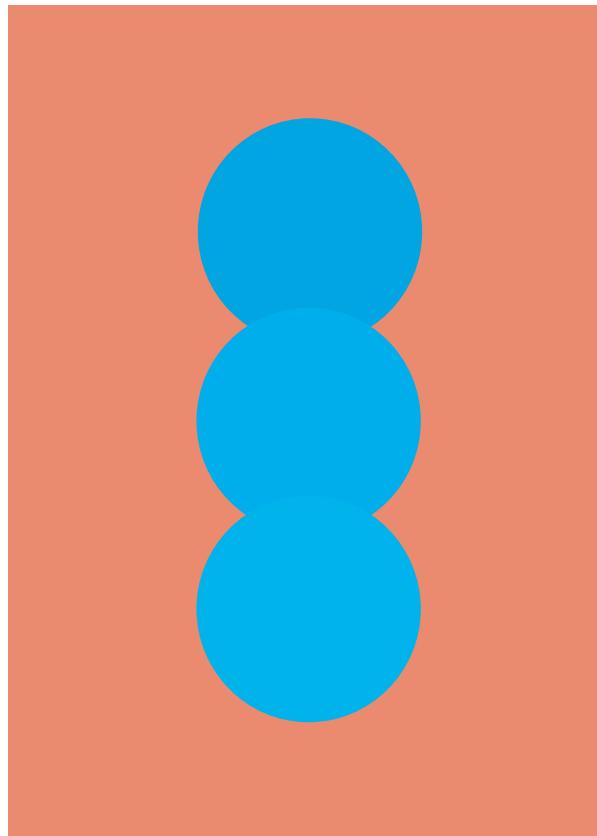


Рис. 5.19. Изменение цветового тона наложенных друг на друга деталей голубого цвета

индексы в цветовой палитре, либо реальные RGB-значения. Пиксели упаковываются в байты и упорядочиваются по строкам развертки. Формат Windows BMP поддерживает простую схему группового кодирования RLE, обеспечивающую сжатие только 4-битовых и 8-битовых растровых данных.

Метафайловый формат PCT (Macintosh PICT, PICT) [2.126] — собственный формат Macintosh, применим и для растровых, и для векторных изображений. Несмотря на то, что он используется преимущественно для обмена графическими данными между приложениями Mac, его поддерживают и многие программы для PC. Современная версия поддерживает до 32 битов на пиксель. Файлы PICT-формата могут поддерживать объектно-ориентированную векторную графику и растровые изображения. Изображения в PICT могут содержать графические данные в разнообразных подформатах, включая двухцветные растровые изображения, полноцветные изображения JPEG или список команд рисования. В формате PICT используется схема сжатия RLE.

Растровый формат PCX (PC Paintbrush File Format) [2.121] — один из широко используемых форматов, применяется для хранения иллюстраций в настольных издательских системах. Он известен также под сокращенными назва-

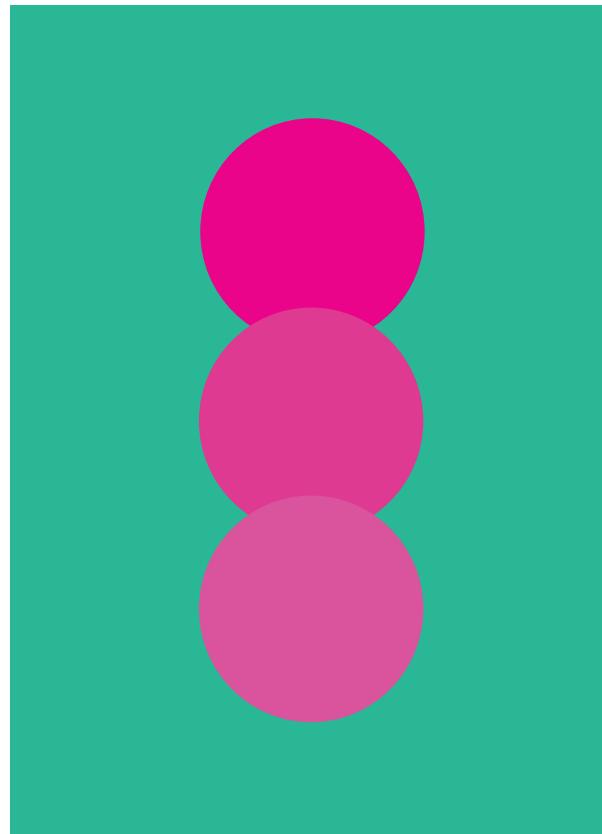


Рис. 5.20. Изменение цветового тона наложенных друг на друга деталей пурпурного цвета

ниями DCX и PCX. Данные об изображениях имеют цветовую глубину до 24 битов/пиксель при максимальных размерах до $64K \times 64K$ пикселов. Данные изображения могут кодироваться с использованием алгоритма RLE, который нельзя назвать эффективным. Файлы PCX состоят из заголовка, данных изображения и цветовой палитры. Пиксельные данные строк развертки хранятся либо в пиксельно-ориентированном, либо в плоскостно-ориентированном виде. Пиксельно-ориентированные данные хранятся в виде непрерывной строки последовательно записанной пиксельной информации, независимо от того, это реальные данные или индексы цветовой палитры. При хранении по плоскостям данные разбиваются на RGB-составляющие и группируются по цвету в строки развертки.

Векторный формат SWF (Adobe Flash, Shockwave Flash) [2.127] может включать в себя текст, векторную и растровую графику; проигрывается в веб-браузерах, которые имеют установленный модуль проигрывания flash мультимедийного контента для платформы «Adobe Engagement Platform» (такого, как веб-приложения, игры и мультфильмы). SWF-формат является в настоящее время единственным векторным форматом, файлы которого могут использоваться

Таблица 5.2. Размеры файлов различных форматов с изображением «Гавань»

№ п/п	Наименование файла	Компрессия	Объем файла, байт
1	Adobe Illustrator (AI)	без компрессии	1 579 144
2	Scalable Vector Graphics (SVG)	- <-	1 339 006
3	Scalable Vector Graphics Zip (SVGZ)	gzip	1 009 416
4	Tag Image File Format (TIFF)	без компрессии	1 000 408
5	Pixar (PXR)	- <-	984 064
6	Microsoft Enhanced Metafile (EMF)	- <-	983 452
7	Truevision Graphics Adapter, Targa Graphics Adapter Image (TGA)	- <-	983 084
8	Microsoft Windows Bitmap (BMP, DIB)	- <-	983 064
9	Macintosh PICT (PCT, PICT)	- <-	945 708
10	PC Paintbrush File Format (PCX, DCX, PCC)	- <-	928 217
11	Truevision Graphics Adapter, Targa Graphics Adapter Image (TGA)	RLE	892 254
12	Adobe Flash, Shockwave Flash (SWF)	Lossless JPEG	615 082
13	Tag Image File Format (TIFF)	LZW	559 396
14	Tag Image File Format (TIFF)	JPEG	529 364
15	Portable Network Graphic Format (PNG)	LZ77	490 257
16	JPEG File Interchange Format (JPEG, JFIF, JFI, JPG)	максимальное качество	283 279
17	Graphics Interchange Format (GIF)	LZW	223 632
18	JPEG File Interchange Format (JPEG)	высокое качество	165 642
19	JPEG File Interchange Format (JPEG)	среднее качество	76 671
20	JPEG File Interchange Format (JPEG)	низкое качество	61 428
21	JPEG File Interchange Format (JPEG)	очень низкое качество	41 267

при создании Web-страниц. На Web-странице можно как разместить отдельные элементы, так и разработать практически всю страницу целиком на основе технологии Flash. Предусматривается сжатие информации по стандарту Lossless JPEG.

Растровый формат PNG (Portable Network Graphic Format) [2.121] позволяет хранить изображения, битовая глубина которых до 16 битов в полутонах или до 48 битов в полноцветных изображениях при максимальном размере изображения $2^G \times 2^G$ пикселов. Этот формат обеспечивает поэтапное отображение данных изображения, хранение информации о показателе гамма, о прозрачности, о текстовой информации. В нем используется эффективный метод сжатия без потерь LZW. Данные изображения в формате PNG представлены в виде растра, строки которого записаны слева направо, сверху вниз. В начале каждой строки развертки записывается дополнительный байт «типа фильтра», который используется при сжатии и распаковке изображения и указывает тип алгоритма филь-



Рис. 5.21. Статическое изображение «Гавань» («Harbour»)

трации, примененный при обработке строки. Формат чисел — «big endian». Формат PNG разработан как альтернатива стандарту Graphics Interchange Format (GIF) и существенно его превосходит по функциональным возможностям.

Растровый формат GIF (Graphics Interchange Format) [2.121] является форматом, обеспечивающим передачу и хранение данных об изображениях, однако он не поддерживает хранение полноцветных изображений. Используемая палитра цветов — до 256, максимальный размер изображения — $64K \times 64K$ пикселов. GIF в отличие от многих растровых форматов использует потоковое хранение данных. GIF-файл состоит из последовательности пакетов данных и дополнительной протокольной информации, при этом файл читается так, как будто он является непрерывным потоком данных. Формат чисел — «little endian». Для кодирования данных изображения, записанного в GIF-файле, применяется метод LZW, позволяющий сжимать строки байтов с одинаковыми значениями в отдельное кодовое слово и способный уменьшить размер 8-битовых пиксельных данных практически до двух раз. Формат позволяет хранить в одном файле несколько изображений, но эта возможность редко используется. Формат поддерживает возможность хранения и воспроизведения изображений с чересстрочной разверткой.

Растровый формат JPEG (JPEG File Interchange Format) [2.121] изначально предназначался для хранения графики и изображений. Он известен также под сокращенными названиями JFIF, JFI, JPG. JPEG File Interchange Format ис-

пользуется для хранения данных, закодированных с использованием стандарта JPEG, разработанного организацией Joint Photographic Experts Group (Объединенная группа экспертов по фотографии). Формат позволяет хранить цветные изображения, битовая глубина которых достигает максимум 24 бита/пиксель при размере изображения до $64K \times 64K$ пикселов. Данные представляют собой потоки байтов, причем 16-битовые слова хранятся в формате чисел «big endian». Формат обеспечивает сжатие информации с потерями. При сохранении JPEG-файла можно указать степень сжатия, которую обычно задают в некоторых условных единицах. При этом побитно восстановленное изображение всегда отличается от оригинала. Распространенным заблуждением является мнение о том, что качество JPEG тождественно доле сохраняемой информации. При сжатии изображение переводится в цветовую систему YCbCr. Далее каналы изображения Cr и Cb, отвечающие за цвет, уменьшаются в 4 раза (в два раза по горизонтали и по вертикали). Уже на этом этапе необходимо хранить только четверть информации о цвете изображения. Далее каналы изображения, включая черно-белый канал Y, разбиваются на блоки 8×8 пикселов. Каждый блок подвергается дискретно-косинусному преобразованию. Полученные коэффициенты подвергаются квантованию и упаковываются с помощью кодов переменной длины. Матрица, используемая для квантования коэффициентов, хранится вместе с изображением. Обычно она строится так, что высокочастотные коэффициенты подвергаются более сильному квантованию, чем низкочастотные. Это приводит к огрублению мелких деталей на изображении. Чем выше степень сжатия, тем более сильному квантованию подвергаются все коэффициенты. Применяются различные способы записи сжатого изображения. Progressive JPEG — такой способ записи сжатого изображения в файл, при котором старшие (низкочастотные) коэффициенты находятся в начале файла. Это позволяет получить уменьшенное изображение при загрузке лишь небольшой части файла и повышать детализацию изображения по мере загрузки оставшейся части. Поэтому Progressive JPEG получил широкое распространение в Интернете. К недостаткам формата следует отнести то, что при сильных степенях сжатия дает знать о себе блочная структура данных, изображение «дробится на квадратики» (каждый размером 8×8 пикселов). Этот эффект особенно заметен на областях с низкой пространственной частотой (плавные переходы изображения, например, чистое небо). В областях с высокой пространственной частотой (например, контрастные границы изображения) возникают характерные «артефакты» — иррегулярная структура пикселов искаженного цвета и/или яркости. Кроме того, из изображения пропадают мелкие цветные детали. Однако несмотря на недостатки JPEG получил очень широкое распространение из-за высокой степени сжатия по отношению к существовавшим во время его появления форматам.

Литература к части II

- 2.1. Цифровая обработка телевизионных и компьютерных изображений. 2-е изд., перераб. и доп. / Под ред. Ю.Б. Зубарева и В.П. Дворковича. М.: НАТ, 1997.
- 2.2. Шенон К. Работы по теории информации и кибернетике/ Пер. с англ под ред. Р.Л. Добрушина и О.Б. Лупанова. М.: Иностранная литература, 1963.
- 2.3. Gallager R.G. Information Theory and Reliable Communication/ John Wiley and Sons, Inc. N.Y. 1968.
- 2.4. Netravali A.N., Haskell B.G. Digital pictures: Representation and Compression: Plenum Press. N.Y. 1991.
- 2.5. Гонсалес Р., Будс Р. Цифровая обработка изображений. М.: Техносфера, 2005.
- 2.6. Семенюк В.В. Экономное кодирование дискретной информации. СПб.: ИТМО, 2001.
- 2.7. Golomb S. W. Run-Length Encoding// IEEE Trans. Inform. Theory. 1966. Vol. 12. № 3.
- 2.8. Rice R.F. Some Practical Universal Noiseless Coding Techniques: JPL Publications 79-22. Jet Propulsion Lab. Pasadena, California, USA, Mar., 1979.
- 2.9. Ватолин Д., Ратушняк А., Смирнов М., Юкин В. Методы сжатия данных. Устройство архиваторов, сжатие изображений и видео. М.: ДИАЛОГ-МИФИ, 2003.
- 2.10. Дискретная математика: Алгоритмы. Кодирование целых чисел. СПб.: СПбГУ информационных технологий, механики и оптики
<http://rain.ifmo.ru/cat/view.php/theory/coding/integer-2005>
- 2.11. Fenwick P. Punctured Elias Codes for variable-length coding of the integers// Technical Report 137. 1996. Dec., 5.
- 2.12. Fano F.M. Technical. N 65. The Research Laboratory of Electronics// Technical. № 65., MIT Mar., 17, 1949.
- 2.13. Хаффмен Д.А. Метод построения кодов с минимальной избыточностью// Кибернетический сборник. М., ИЛ, 1961. Вып. 3.
- 2.14. Pasco R. Source Coding Algorithms For Fast Data Compression, Ph.D. dissertation, Dept of Electrical Engineering. Stanford University, California, USA, 1976.
- 2.15. Rissanen J.J. Generalized Kraft Inequality and Arithmetic Coding// IBM J. Res. Develop., 1976. Vol. 20, 6. № 3.
- 2.16. Rissanen J.J., Langdon G.G. Arithmetic Coding// IBM J. Res. Develop., 1979. Vol. 23. № 2.
- 2.17. Pennebaker W.B., Mitchell J.L., Langdon G.G., Arps R.B. An Overview of the Basic Principles of the Q-Coder Adaptive Binary Arithmetic Coder// IBM J. Res. Develop., 1988. Vol. 32. № 6.



- 2.18. *Rissanen J.J., Langdon G.G.* Arithmetic Coding// IBM J. Res. Develop., 1979. Vol. 23. № 2.
- 2.19. *Rubin F.* Arithmetic Stream Coding Using Fixed Precision Registers// IEEE Trans. Inform. Theory, 1979. Vol. 25. № 6.
- 2.20. *Guass M.* A General Minimum-Redundancy Source-Coding Algorithm// IEEE Trans. Inform. Theory, 1980. Vol. 26. № 1.
- 2.21. *Martin G.N.N.* Range encoding: an algorithm for removing redundancy from a digitized message// Video & Data Recording Conference. IBM UK Scientific Center, Southampton. July, 1979.
- 2.22. *Cleary J., Witten I.H.* Data Compression Using Adaptive Coding and Partial String Matching// IEEE Trans. Commun., 1984. Vol. 32. № 4.
- 2.23. *Moffat A., Neal R., Witten I.H.* Arithmetic Coding Revisited// Proc IEEE Data Compression Conf. Snowbird, Utah, USA. Mar., 1995.
- 2.24. *Howard P.G., Vitter J.S.* Practical Implementations of Arithmetic Coding// Storer A. Image and Text Compression, Kluwer Academic Publishers. Massachusetts, USA, 1992.
- 2.25. *Ziv J., Lempel A.* A Universal Algorithms for Sequential Data Compression// IEEE Trans. Inform. Theory. 1977. Vol. 23 / № 3.
- 2.26. *Ziv J., Lempel A.* Compression of Individual Sequences via Variable-Rate Coding// IEEE Trans. Inform. Theory/ 1978. Vol. 24. № 5.
- 2.27. *Rodeh M., Pratt V.R., Even S.* Linear Algorithm for Data Compression via String Matching // J. ACM. 1981. Vol. 28., № 1.
- 2.28. *Storer J.A., Szymanski T.G.* Data Compression via Textual Substitution// J. ACM. 1982. Vol. 29. № 4.
- 2.29. *Bell T.C.* Better OPM/L Text Compression// IEEE Trans. Commun. 1986. Vol. 34. № 12.
- 2.30. *Bell T.C.* A Unifying Theory and Improvements for Existing Approaches to Text Compression, Ph.D. Dissertation. Dept. of Computer Sciences, University of Canterbury. New Zealand, 1987.
- 2.31. *Brent R.P.* A Linear Algorithm for Data Compression// Aust. Comput. J. 1987. Vol. 19. № 2.
- 2.32. *Welch T.A.* A Technique for High-Performance Data Compression// IEEE Computer. 1984. Vol. 17. № 6.
- 2.33. *Thomas S.W., McKie J., Davies S., Turkowski K., Woods J.A., Orost J.W.* Compress (Version 4.0): Program and Documentation, 1985.
- 2.34. *Tischer P.* A Modified Lempel-Ziv-Welch Data Compression Scheme// Aust. Comput. Sci. Commun., 1987. Vol. 9. № 1.
- 2.35. *Miller V.S., Wegman M.N.* Variations on a Theme by Ziv and Lempel// Apostolico A., Galil Z. Combinatorial Algorithms on Words. Springer-Verlag, New York, USA, 1985.
- 2.36. *Jacobson M.* Compression of Character String by an Adaptive Dictionary// BIT. 1985. Vol. 25. № 4.

- 2.37. *Fiala E.R., Greene D.H.* Data Compression with Finite Windows// CACM, 1989. Vol. 32. № 4.
- 2.38. *Bellman R.* Introduction to Matrix Analysis. McGraw-Hill, N.J., 1960.
- 2.39. *Bhushan A.K.* Efficient Transmission and Coding of Colour Components. M.S. Thesis/ Massachusetts Institute of Technology. Cambridge, MA. June, 1977.
- 2.40. *Cleary J., Witten I.H.* Data Compression Using Adaptive Coding and Partial String Matching// IEEE Trans. Commun. 1984. Vol. 32. № 4.
- 2.41. *Howard P.G.* The design and analysis of efficient lossless data compression systems, Report CS-93-28, Dept. of Comp. Sci. Brown University. Providence, Rhode Island, 1993.
- 2.42. *Moffat A.M.* Implementing the PPM Data Compression Scheme// IEEE Trans. Commun. 1990. Vol. 38. № 11.
- 2.43. *Cleary J., Teahan W., Witten I.* Unbounded Length Context for PPM// Proc IEEE Data Compression Conf. Snowbird. Utah, USA, Mar., 1995.
- 2.44. *Tjalkens T.J., Volf P.A.J., Willems F.M.J.* A Context-Tree Weighting Method for Text-Generating Sources// Proc. IEEE Data Compression Conf., Snowbird. Utah, USA, Mar., 1997.
- 2.45. *Volf P.A.J.* Text Compression Methods Based on Context Weighting: Tech. rept., Stan Ackermans Institute, Eindhoven University of Technology. Netherlands, June, 1996.
- 2.46. *Willems F.M.J.* The Context-Tree Weighting Method: Extensions// IEEE Trans. Inform. Theory, 1998. Vol. 44. № 2.
- 2.47. *Willems F.M.J., Shtarkov Y.M., Tjalkens Tj.J.* Context Tree Weighting: A Sequential Universal Source Coding Procedure for FSMX Sources// Proc. IEEE Int. Symp. Inform. Theory. San Antonio, Texas, USA, 1993.
- 2.48. *Willems F.M.J., Shtarkov Y.M., Tjalkens Tj.J.* Context Tree Weighting: Basic Properties// IEEE Trans. Inform. Theory. 1995. Vol. 41. № 3.
- 2.49. *Willems F.M.J., Shtarkov Y.M., Tjalkens Tj.J.* Context Weighting for General Finite Context Sources// IEEE Trans. Inform. Theory. 1996. Vol. 42. № 5.
- 2.50. *Cormack G.V., Horspoll R.N.* Data Compression Using Dynamic Markov Modeling// Comput. J. 1987. Vol. 30. № 6.
- 2.51. *Horspoll R.N., Cormack G.V.* Dynamic Markov Modeling. A Prediction Technique// Proc. Int. Conf. on the System Sciences. Honolulu, Hawaii, USA, 1986.
- 2.52. *Long P.M., Natsev A.I., Vitter J.S.* Text Compression via Alphabet Representation// Proc. IEEE Data Compression Conf. Snowbird, Utah, USA, Mar., 1997.
- 2.53. *Schmidhuber J., Heil S.* Sequential Neural Text Compression// IEEE Trans. Neural Networks, 1996. Vol. 7. № 1.
- 2.54. *Буяновский Г.* Ассоциативное кодирование// Монитор. 1994. № 8.
- 2.55. *Burrows M., Wheeler D.J.* A Block-Sorting Lossless Data Compression Algorithm: Res. Rept. 124, DIGITAL Systems Research Center, 1994.

- 2.56. Прэйтт У. Цифровая обработка изображений. Т.1–2. М.: Мир, 1982.
- 2.57. Abramson N. Information Theory and Coding. McGraw-Hill, N.J., 1963.
- 2.58. Langdon G. An Introduction to Arithmetic Coding/ IBM J. Res. Develop. 28. Mar., 1984.
- 2.59. ITU-T, CCITT Recommendation V.42 bis: Data Compression Procedures for Data Circuit Terminating Equipment (DCE) Using Error Correction Procedures, 1990.
- 2.60. Krichevsky R.E., Trofimov V.K. The Performance of Universal Coding// IEEE Trans. Inform. Theory. 1981. Vol. 27. № 1. Pp. 199–207.
- 2.61. Bell T., Witten I, Cleary J. Modeling for Text Compression// ACM Computing Surveys. Vol.21. № 4, pp. 557–591. Dec., 1989.
- 2.62. Обратимое сжатие или сжатие без наличия помех
<http://www.citforum.ru/programming/theory/szhatie.shtml>
- 2.63. CCITT. Standardization of Group 3 Facsimile apparatus for document transmission. Rec. Fascicle VII.2/ 1980. Т. 4.
- 2.64. ISO/IEC JNC1/SC2/WG9, CD 11544/ Progressive Di-level Image Compression, Revision 4.1. 1991. September, 16.
- 2.65. Ярославский Л.П. Введение в цифровую обработку изображений. М.: Сов. радио, 1979.
- 2.66. Цифровое кодирование телевизионных изображений/ Под ред. И.И. Цукермана. М.: Связь, 1981.
- 2.67. Cutler C.C. Differential Quantization of Communication Signals// U.S. Patent 2 605 361. July, 1952.
- 2.68. O’Neal I.B.Jr. Predictive Quantizing System (Differential Pulse Code Modulation) for the Transmission of Television Signal// BSTJ, 15,1966.
- 2.69. Limb J. Source-Receiver Encoding of Television Signals// Proc. IEEE, № 3. 1967.
- 2.70. Харатишвили Н.Г. Цифровое кодирование с предсказанием непрерывных сигналов. М.: Радио и связь, 1986.
- 2.71. Залманзон Л.А. Преобразования Фурье, Уолша, Хаара и их применение в управлении, связи и других областях. М.: Наука, 1989.
- 2.72. Ахмед Н., Rao K.P. Ортогональные преобразования при обработке цифровых сигналов. М.: Связь, 1980.
- 2.73. Оппенгейм A.B., Шафер P.B. Цифровая обработка сигналов. М.: Связь, 1979.
- 2.74. Wintz P.A. Transform Picture Coding// Proc. IEEE. V. 60. July, 1972.
- 2.75. Лоули Д., Максвелл А. Факторный анализ как статистический метод. М.: Мир, 1967.
- 2.76. Обработка изображений при помощи ЦВМ/ Пер. с англ. под ред. Д.С. Лебедева. М.: Мир, 1973.

- 2.77. ISO/IEC DIS 10918-1. Information Technology. Digital Compression and Coding of Continuous-tone Still Images: Requirements and Guidelines// Ed. 1, JTS 1/ SC 29, 1994.
- 2.78. ISO/IEC DIS 10918-2. Information Technology. Digital Compression and Coding of Continuous-tone Still Images: Compliance Testing// Ed. 1, JTS 1/ SC 29, 1994.
- 2.79. ISO/IEC DIS 10918-3. Information Technology. Digital Compression and Coding of Continuous-tone Still Images: Extensions// Ed. 1, JTS 1/ SC 29, 1994.
- 2.80. The JPEG Still Picture Compression Standard// Communications of the ACM. V.34. № 4. April, 1991.
- 2.81. *Frei W. and Jaeger P.A.* Some Basic Considerations for the Source Coding of Color Pictures: Proc. of International Conference on Communications. Seattle Wash., 1973.
- 2.82. *Левин Б.Р.* Теоретические основы статистической радиотехники. Т.1–3. М.: Сов. радио, 1974, 1975, 1976.
- 2.83. *Красильников Н.Н.* Статистическая теория передачи изображений. М.: Связь, 1976.
- 2.84. *Лебедев Д.С., Цукерман И.И.* Телевидение и теория информации. Л.: Энергия, 1965.
- 2.85. *Игнатьев Н.К.* Применение δ -функций для анализа систем с дискретизацией: Научные доклады высшей школы// Радиотехника и электроника, ¹₂, 1959.
- 2.86. *Kojiro Kinoshita.* Image Transmission Carrier and Image Spectrum in Generalized Photographic System// NHK Technical Monograph. 1964. № 3.
- 2.87. *Дерюгин Н.Г.* Спектр мощности и функция корреляции телевизионного сигнала// Электросвязь. 1957. № 7.
- 2.88. *Игнатьев Н.К.* Энергетический спектр телевизионного сигнала// Электросвязь. 1959. № 1.
- 2.89. *Netravali A.N. and Rubinstein C.B.* Luminance Adaptive Coding of Chrominance Signals // IEEE Trans. on Communication. V. COM-27. April, 1979.
- 2.90. *Witten J.S., Bell T.C.* The Zero Frequency Problem: Estimating the Probabilities of Novel Events in Adaptive Text Compression// IEEE Trans. Inform. Theory. 1987. Vol. 37. № 7.
- 2.91. Teahan W.J. Probability Estimation for PPM
<http://www.cs.waikato.ac.nz/wjt/papers/NZCSRSC.ps.gz>
- 2.92. *Bloom C.* Algorithm PPMZ2
<http://www.cbloom.com/src/ppmz.html>
- 2.93. *Шкарин Д.* Практическая реализация алгоритма PPM
<http://sochi.net.ru/maxime/doc/PracticalPPM.doc.gz>
- 2.94. Справочник по телевизионной технике. Т.1–2/ Пер. с англ. под ред. С.И. Кацаева. М.-Л.: Госэнергоиздат, 1962.



- 2.95. Телевизионная техника: Справочник/ Под общ. Ред. Ю.Б. Зубарева и Г.Л. Глориозова. М.: Радио и связь, 1994.
- 2.96. Роз А. Зрение человека и электронное зрение/ Пер. с англ. под ред. В.С. Вавилова. М.: Мир, 1977.
- 2.97. Кустарев А.К. Колориметрия цветного телевидения. М.: Связь, 1967.
- 2.98. Джадд Д., Вышецки Г. Цвет в науке и технике. М.: Мир, 1978.
- 2.99. Певзнер Б.М. Качество цветных телевизионных изображений. М.: Радио и связь, 1988.
- 2.100. Крибошев М.И., Кустарев А.К. Световые измерения в телевидении. М.: Связь, 1973.
- 2.101. Mitsuhashi J. High- Quality Television and Visual Characteristics// J. of the Institute of TV Engineers of Japan. , 1979. V.33. ¹².
- 2.102. Красильников Н.Н. Теория передачи и восприятия изображения. М.: Радио и связь, 1986.
- 2.103. Крибошев М.И. Основы телевизионных измерений. М.: Радио и связь, 1989.
- 2.104. МККР. Требования к передаче телевидения на большие расстояния (за исключением системы I): Рекомендация 421.
- 2.105. МККР. Характеристики телевизионных каналов связи, предназначенных для международных передач: Рекомендация 567.
- 2.106. МККР. Единое значение отношения сигнал/шум для всех телевизионных систем: Рекомендация 658.
- 2.107. Аббуд И. О допустимых цветовых искажениях ТВ изображения// Техника кино и телевидения. 1974. № 5.
- 2.108. Pazderak J., Kepr M. Objektivne meritelne Kriterium Barevného podany televizního obrazu// Slaboproudý obzor. 1980. Sv.41. ¹³.
- 2.109. Pazderak J., Kepr M. Experimentalní stanovení rozlišitelných diferenci barev a jejich aplikace// Slaboproudý obzor. 1984. Sv.45. ¹¹.
- 2.110. CIE 1931 Color Space http://en.wikipedia.org/wiki/CIE_1931_color_space
- 2.111. McAdam D.L. Sources of Color Science. Cambridge, MA, MIT Press, 1970.
- 2.112. Буксбаум Г., Бедросян С.Д. Зависимость количества цветов от числа уровней серого: количественное соотношение// ТИИР. Т.72. 1984. № 10.
- 2.113. МККР. Метод субъективной оценки качества телевизионных изображений: Рекомендация 500.
- 2.114. МККР. Субъективная оценка качества телевизионных изображений: Отчет 405.
- 2.115. Гофайзен О.В., Певзнер Б.М. Характеристики зависимости качества изображения от основных параметров ТВ-тракта. Ч. I.
- 2.116. Sallio P., Kretz F. Qualité subjective en télévision numérique. Premiere partie: Methodologie de son evalution // Revue de radiodiffusion-télévision. 1978. ¹⁵².

- 2.117. ГОСТ 26320: Оборудование телевизионное студийное и внестудийное. Методы субъективной оценки качества цветных телевизионных изображений.
- 2.118. *Антипин М.В.* Интегральная оценка качества телевизионного изображения. М.: Наука, 1970.
- 2.119. *Тилькин В.И.* О системе оценок качества телевизионного изображения// Техника кино и телевидения. 1975. № 9.
- 2.120. *White T.A., Allnatt Y.W., Lewis N.W.* Double-stimulus Quality Rating method for Television Digital Codecs// Electron. Letters. 1980. V. 16.
- 2.121. *Джеймс Д. Мюррей, Уильям ван Райпер.* Энциклопедия форматов графических файлов/ Пер. с англ. ВНВ, Киев, 1997.
- 2.122. SVG файлы <http://ru.wikipedia.org/wiki/SVG>
- 2.123. GZIP сжатие <http://sjwebway.info/service.htm>
- 2.124. Жвалевский А., Гурский Ю., Корабельникова Г. Adobe Illustrator CS в теории и на практике. Изд-во «Новое знание», 2004.
- 2.125. Описание формата EMF
[http://msdn.microsoft.com/en-us/library/cc230514\(v=prot.20\).aspx](http://msdn.microsoft.com/en-us/library/cc230514(v=prot.20).aspx)
- 2.126. PICT (Macintosh QuickDraw Picture Format)
http://arttower.ru/tutorial/Svetilkin/spravochnicki/Flash5/ch14_8.html#25
- 2.127. Формат ShockWave Flash <http://www.filetypes.ru/swf>

ЧАСТЬ III

ОСНОВНЫЕ МЕТОДЫ СЖАТИЯ ИЗОБРАЖЕНИЙ

Введение

Цифровые методы обработки изображений намного превосходят аналоговые по гибкости и эффективности. Основным преимуществом цифровых методов обработки является возможность сокращения объема передаваемой информации и обеспечения сжатия и передачи нескольких телевизионных программ в одном стандартном радиоканале. Добиться этого аналоговыми средствами невозможно.

В табл. III.1 приведена классификация основных используемых методов кодирования изображений.

Наиболее простой алгоритм кодирования изображений — импульсно-кодовая модуляция (ИКМ), при которой после квантования значений уровней пикселов производится их преобразование в цифровой код. Преобразование может осуществляться неадаптивно, когда каждый пиксель кодируется кодом с заранее заданным числом битов (например, 8), или адаптивно. В последнем случае кодирование производится кодами переменной длины в зависимости от текущей статистики пикселов.

Различные варианты дифференциальной импульсно-кодовой модуляции (ДИКМ) — кодирования с предсказанием — обеспечивают более эффективное сжатие изображений. Кодирование с предсказанием также может осуществляться неадаптивным (фиксированным) и адаптивным образом. При этом возможно повышение коэффициента сжатия изображения за счет использования условного замещения кодов, при котором если ошибка предсказания меньше некоторой заданной величины, то она не передается. Более точное предсказание осуществляется с использованием задержанного кодирования, когда предсказанные значения определяются не только предыдущими, но и последующими пикселями.

Таблица III.1. Классификация основных методов кодирования изображений

Методы и алгоритмы кодирования изображений

```

graph TD
    A[Импульсно-кодовая модуляция] --- B[Кодирование с предсказанием]
    A --- C[Кодирование с преобразованием]
    A --- D[Кодирование с интерполяцией и экстраполяцией]
    A --- E[Статистическое кодирование]
    A --- F[Другие методы кодирования]
    
    B --- B1[Алантиния]
    B --- B2[Неалантиния]
    
    C --- C1[Алантиниес]
    C --- C2[Неалантиниес]
    
    D --- D1[Алантиное]
    D --- D2[Неалантиное]
    
    E --- E1[Алантиниес]
    E --- E2[Неалантиниес]
    
    F --- F1[Квантование]
    F --- F2[Предсказание]
    F --- F3[Преобразования]
    F --- F4[Подбор коэффициентов]
    F --- F5[Квантование]
    F --- F6[Дискретное преобразование Фурье (ДПФ)]
    F --- F7[Преобразование Кархунена-Лоэва]
    F --- F8[Дискретное Кошиуское Преобразование]
    F --- F9[Преобразования Хаара, Уолша-Адамара и др.]
    F --- F10[Шеннона-Фано]
    F --- F11[Хаффмана]
    F --- F12[Арифметическое]
    F --- F13[Блочное и условное]
    F --- F14[Словарное, статистическое, контекстное]
    F --- F15[Векторное квантование]
    F --- F16[Выделение контуров]
    F --- F17[Последовательность одинаковых символов]
    F --- F18[Битовая плоскость]
    F --- F19[Фрактальное кодирование]
    F --- F20[Вейвлет-кодирование]
  
```

Кодирование с преобразованием в настоящее время наиболее распространено. При этом используются различные унитарные преобразования — Фурье, Уолша—Адамара, Хаара, косинусное, синусное и др. Для оценки возможностей использования ортонормированных преобразований при кодировании конкретных изображений применяется преобразование Кархунена—Лоэва.

Кодирование с интерполяцией и экстраполяцией также используется в ряде случаев. В частности, его использование позволяет снизить частоту дискретизации, обеспечив при этом достаточно высокое качество воспроизведенного изображения.

При обработке статических изображений используется только пространственная субдискретизация, а при обработке подвижных изображений весьма эффективно применение временной субдискретизации.

Статистическое кодирование обеспечивает сжатие изображений без потери информации. Различные варианты статистического кодирования рассмотрены в Части II, посвященной вопросам статистической и визуальной избыточности изображений.

И наконец, в классификационной таблице имеется колонка «другие методы кодирования», в которой обозначены некоторые используемые методы обработки изображений, рассматриваемые в последующих главах. Особое внимание уделено принципам расчета и организации систем вейвлет-кодирования, при которых изображение разбивается на несколько различных по спектру составляющих с неизменным суммарным количеством пикселов. Такое преобразование изображения позволяет более эффективно использовать цифровую его обработку со значительным сокращением статистической и визуальной избыточности.

ГЛАВА 6

НЕПОСРЕДСТВЕННОЕ КОДИРОВАНИЕ ИЗОБРАЖЕНИЙ И КОДИРОВАНИЕ С ПРЕДСКАЗАНИЕМ

6.1. Импульсно-кодовая модуляция

В процессе преобразования изображения с использованием импульсно-кодовой модуляции производятся следующие операции [3.1–3.6]:

- ограничение спектра;
- дискретизация во времени (или пространстве);
- квантование дискретных отсчетов;
- представление отсчетов в виде бинарных слов.

Эти операции иллюстрируются структурной схемой и диаграммой рис. 6.1.

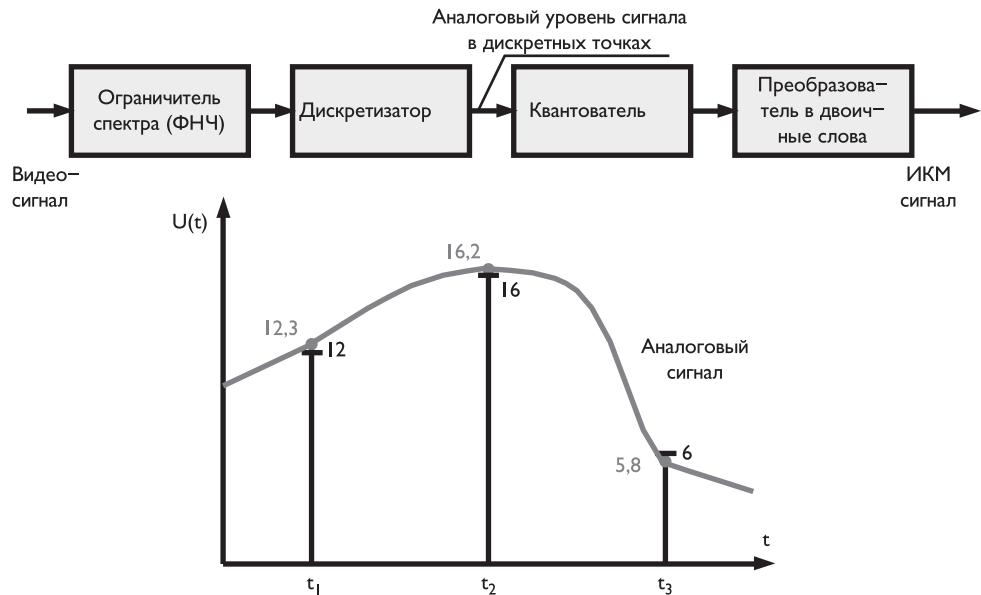


Рис. 6.1. Последовательность видов обработки сигналов при ИКМ

При преобразовании сигналов яркости оцифровка его дискретных значений обычно производится после гамма-коррекции, обеспечивающей подстройку шкалы квантования в соответствии с законом Вебера–Фехнера [3.7–3.8].



Рис. 6.2. Изменение качества изображения при восстановлении чересточечной структуры

На практике часто используются линейные квантователи с 256 уровнями, что обеспечивает достаточно высокое качество воспроизводимых изображений. Уменьшение числа уровней квантования приводит к появлению визуально заметных шумов квантования, проявляющихся в виде ложных контуров на участках изображения, не содержащих мелких деталей. Уменьшение визуальной заметности этих искажений возможно путем добавления к восстановленному сигналу специального высокочастотного шума. При этом качество изображения может быть дополнительно повышенено путем оптимизации характеристик такого шума.

В ряде случаев, например в прикладном телевидении, для сокращения объема информации используют чересточечное разложение изображений, при котором в одном из двух последовательных кадров в нечетных строках преобразуются нечетные пиксели, а в четных строках — четные; в другом же кадре преобразуются четные пиксели в нечетных строках, а нечетные пиксели — в четных строках. Это позволяет вдвое сократить объем передаваемой информации. При восстановлении изображения применяется интерполяция пропущенных пикселов по четырем их окружающим [3.5, 3.6].

На рис. 6.2 приведены исходное и восстановленное указанным образом изображения при чересточечном кодировании пикселов.

При кодировании цветных изображений оцифровке обычно подвергаются не сигналы основных цветов R, G, B, а их комбинации Y, I, Q или Y, U, V.

Это связано со свойствами зрительного анализатора человека. Шумы квантования менее заметны на красном и синем цветах деталей изображения, что позволяет снизить разрядность оцифровки цветоразностных сигналов до 6 битов/пиксел. Кроме того, при использовании комбинаций Y, I, Q или Y, U, V допустимо значительно меньшее пространственное разрешение для цветоразностных сигналов [3.5, 3.10–3.13].

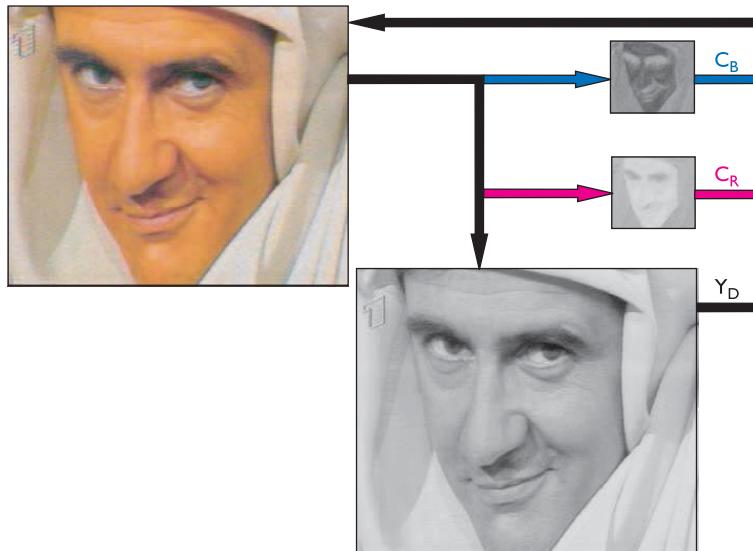


Рис. 6.3. Разложение цветного изображения на три компонента

В частности, в стандартных телевизионных системах PAL и SECAM разрешение и по горизонтали и вертикали для сигналов U и V может быть уменьшено в четыре раза, с учетом того, что их граничная частота в четыре раза меньше граничной частоты сигнала яркости Y . На рис. 6.3 приведено исходное цветное изображение, а также изображения сигналов Y_D , C_R и C_B .

Напомним,

$$\begin{aligned} Y_D &= \text{round} (219E'_Y + 16); \\ C_R &= \text{round} (160E'_{R-Y} + 128); \\ C_B &= \text{round} (126E'_{B-Y} + 128). \end{aligned}$$

При обратном преобразовании этих изображений получаемое цветное изображение практически не отличается от исходного. Объем информации в каждом из сигналов C_R и C_B в 16 раз меньше объема информации монохромного изображения Y_D .

Небольшое сокращение объема информации при использовании ИКМ возможно за счет использования адаптивного кодирования.

Хотя плотность распределения вероятностей уровней пикселов, усредненная по множеству изображений, подчиняется практически равномерному закону, для конкретных изображений она может быть существенно неравномерной, что позволяет использовать вероятностные методы кодирования, снижающие среднее количество используемых битов на пиксель.

На рис. 6.4 приведено монохромное изображение «Залив», а также плотность распределения вероятностей уровней его пикселов. При этом если при неадаптивном кодировании требуется использование $\hat{L}_n = 8$ битов/пиксель, то адаптивное кодирование снижает ее величину до $\hat{L}_a = 7,3$ бита/пиксель.

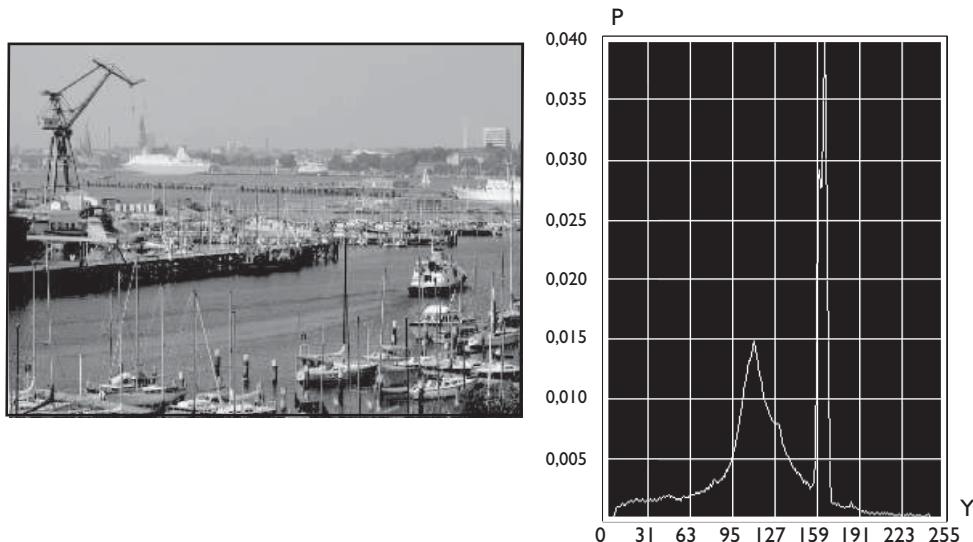


Рис. 6.4. Распределение плотности вероятностей яркости пикселов изображения

6.2. Кодирование изображений с предсказанием, дифференциальная импульсно-кодовая модуляция

Кодирование изображений с использованием ИКМ не позволяет существенно сократить объем исходной информации, поскольку не учитывает корреляционные связи между пикселями. Более эффективными являются системы кодирования с предсказанием, принцип работы которых заключается в том, что в соответствии с некоторыми алгоритмами по значениям предшествующих пикселов производится предсказание значения текущего пикселя. При этом кодируются и передаются лишь ошибки предсказания, по которым на приемной стороне восстанавливается исходное изображение.

Методы такого кодирования принято называть дифференциальной импульсно-кодовой модуляцией (ДИКМ).

Подробнее процедура ДИКМ была уже описана в Части II.

Очевидно, что эффективность предсказания связана с согласованием алгоритма предсказания со статистическими свойствами передаваемого изображения [3.14-3.17].

На рис. 6.5 приведены изображение «Залив» и зависимости, характеризующие корреляционные связи между соседними элементами вдоль и поперек строк.

Концентрация точек вокруг прямой линии, исходящей из начала координат, говорит о высокой корреляции между соседними пикселями и, следовательно, о возможности достаточно точного предсказания последующих отсчетов по известным предыдущим.

Аналогичная картина наблюдается и при оценке корреляционных связей в цветоразностных сигналах. На рис. 6.6 приведены цветное изображение «Залив»

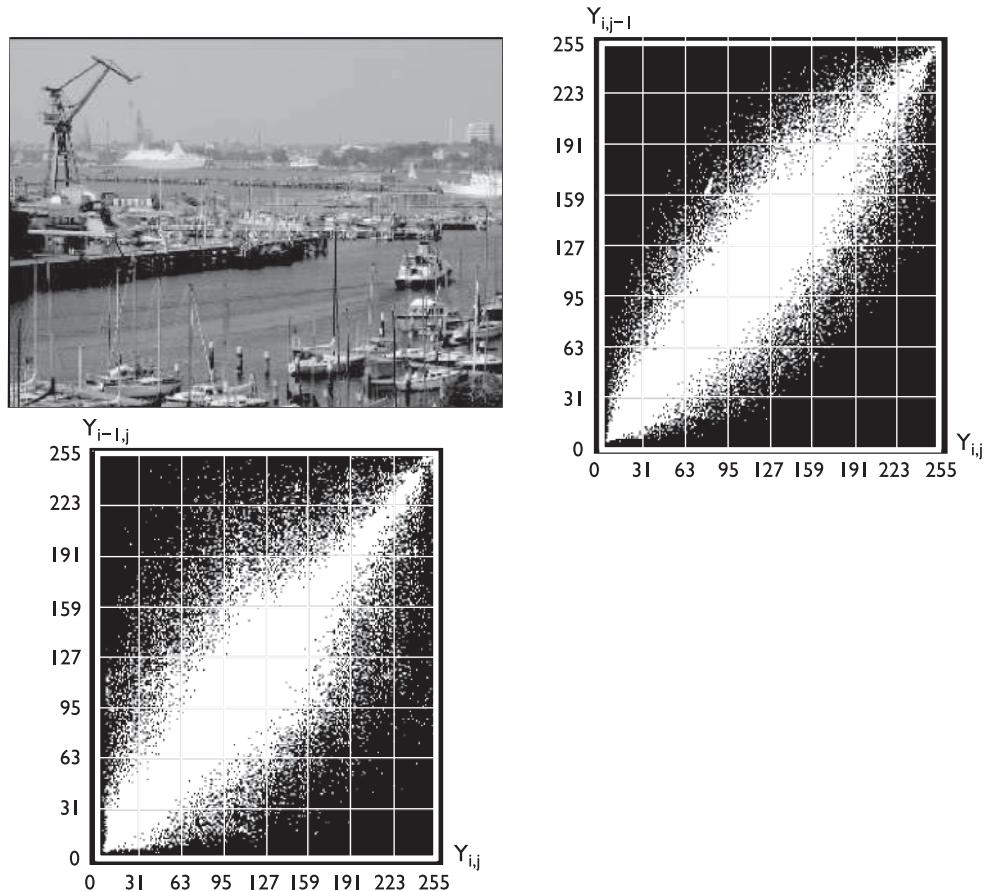


Рис. 6.5. Корреляционные связи между яркостями соседних пикселов вдоль и поперек строк

и зависимости, характеризующие корреляционные связи между значениями цветоразностных сигналов U и V , соответствующими смежным пикселям в строке и поперек строк.

Таким образом, изучение статистики изображений показывает, что в предшествующих пикселях содержится значительная часть информации об очередном пикселе.

Наибольшее распространение получили методы линейного предсказания, в которых значение текущего пикселя предсказывается в виде линейной комбинации $(n - 1)$ известных пикселов [3.1, 3.5, 3.17, 3.18]:

$$\hat{x}_k = \sum_{i=1}^{n-1} \alpha_i x_{k-i}. \quad (6.1)$$

Заметим, что x_{k-i} — не обязательно отсчеты сигнала, предшествующие x_k в строчном интервале или в предшествующих строках. Разность между истинным значением пикселя x_k и предсказанной величиной \hat{x}_k

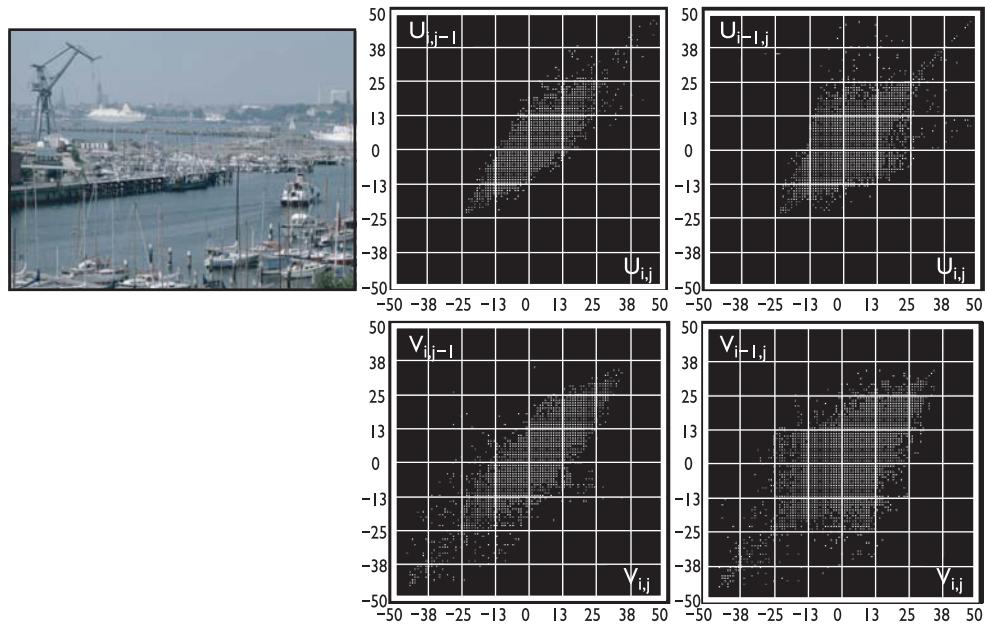


Рис. 6.6. Корреляционные связи между цветностями соседних пикселов

$$e_k = x_k - \hat{x}_k \quad (6.2)$$

называется ошибкой предсказания, а соответствующий ей сигнал — разностным сигналом.

Разностный сигнал имеет больший динамический диапазон по сравнению с исходным и может принимать как положительные, так и отрицательные значения.

Однако распределение плотности вероятностей разностного сигнала выгодно отличается от распределения плотности вероятностей исходного, а именно:

- оно значительно меньше зависит от вида изображения;
- статистики разностных сигналов почти всегда могут быть охарактеризованы Лапласовым распределением, вариация которого в основном зависит только от вида предсказания.

Отметим еще раз, что при этом для одного и того же вида предсказателя алгоритм кодирования оказывается эффективным для различных изображений.

Плотность распределения вероятностей разностного сигнала имеет ярко выраженный неравномерный характер с максимумом в области нулевых значений. Это означает, что ошибка предсказания благодаря сильным связям между соседними элементами в большинстве случаев мала. При этом энтропия разностного сигнала оказывается существенно меньше энтропии исходного сигнала.

Корреляционные связи между элементами разностного сигнала существенно ослабевают, хотя они все-таки остаются, что проявляется, например, в сохранении контуров объектов в разностном сигнале.

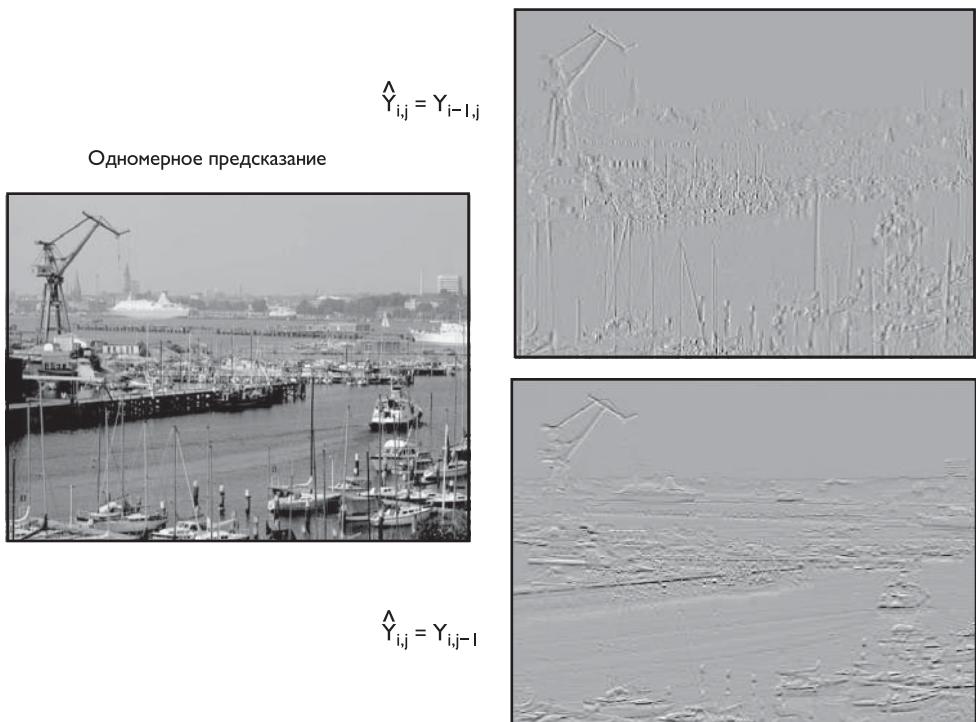


Рис. 6.7. Исходное изображение и изображения разностных сигналов (справа сверху — между смежными пикселями в строке, справа снизу — между смежными пикселями соседних строк)

На рис. 6.7 приведено исходное изображение и изображения двух разностных сигналов — между смежными пикселями в строке и смежными пикселями в соседних строках.

В качестве критерия, обеспечивающего расчет весовых коэффициентов линейного предсказания, зачастую используют минимум среднеквадратичного отклонения ошибки предсказания:

$$\min_N E(x_N - \hat{x}_N). \quad (6.3)$$

Это условие приводит к определению коэффициентов α_i из системы уравнений [3.17, 3.18]:

$$\sum_{i=1}^{N-1} \alpha_i r_{ij} = r_{Nj}, \quad 1 \leq j \leq N-1, \quad (6.4)$$

где r_{ij} — коэффициенты автокорреляции между пикселями, смещеными один относительно другого на интервал $|i - j|$.

Наиболее простой способ линейного предсказания реализуется в системах кодирования с дельта-модуляцией [3.19–3.21], при которой обычно предусматривается квантование разностного сигнала лишь на два уровня, что позволяет передать лишь знак ошибки предсказания. Если знак ошибки положителен, то

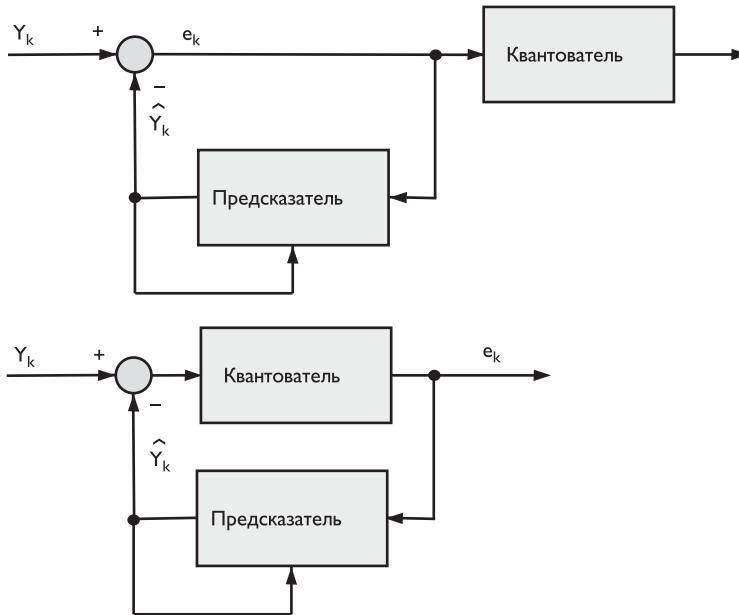


Рис. 6.8. Варианты включения квантователя в кодирующих устройствах с предсказанием

сигнал в декодере увеличивается на некоторую заданную величину Δ , равную минимальному дискрету шкалы квантования; в противном случае сигнал в декодере уменьшается на эту же величину Δ .

Характерной особенностью такого алгоритма работы является возникновение значительных искажений сигнала при резких его изменениях (так называемые перегрузки по крутизне). Эта особенность приводит к снижению четкости восстанавливаемых изображений.

Повышение качества передачи изображений при использовании дельта-модуляции связано с применением алгоритмов адаптивного изменения шкалы квантования (или компандирования).

Принцип компандирования заключается в том, что если, например, знак приращения сигнала не изменяется в течение, скажем, трех-четырех интервалов дискретизации, то шкала изменения сигнала удваивается.

При отсутствии изменения знака приращения процесс удвоения шкалы может осуществляться несколько раз.

Недостатки дельта-модуляции устраняются использованием многоуровневого квантования разностных величин. При этом кодирующее устройство может быть выполнено двумя способами, зависящими от места включения блока квантования [3.5, 3.18, 3.22–3.24], как показано на рис. 6.8.

В первом варианте квантователь включен вне цепи обратной связи, формирующей сигнал предсказания. В этом случае требуется использование большого числа уровней квантования, поскольку шумы квантования при восстановлении изображений укрупняются предсказателем на приемной стороне.

Поясним это следующим образом. Пусть при квантовании k -го отсчета разностного сигнала возникла некоторая ошибка квантования Δ_k . Так как на пере-

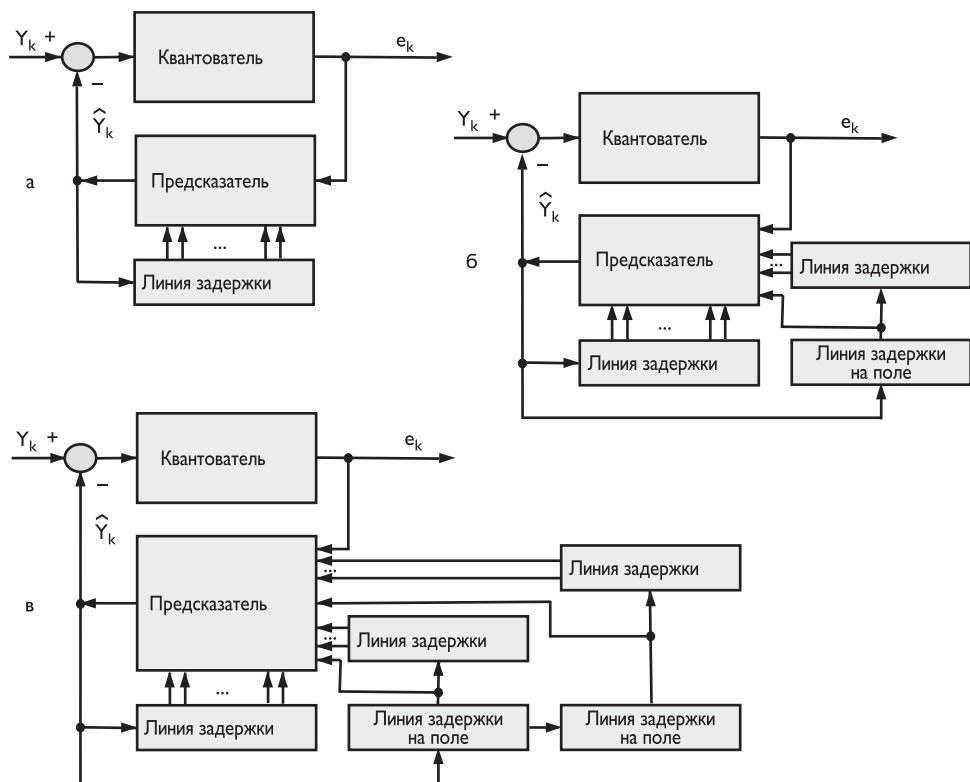


Рис. 6.9. Структурные схемы предсказателей

дающей стороне имеется истинное значение k -го отсчета x_k , то при определении $(k+1)$ -го отсчета разностного сигнала ошибка Δ_k учтена не будет, в то время как на приемной стороне ошибка квантования k -го отсчета будет присутствовать при вычислении всех последующих отсчетов.

Это приводит к накоплению ошибок и к увеличению их заметности на переданных и восстановленных изображениях.

Устранение такого эффекта достигается включением квантователя в цепь обратной связи.

В зависимости от того, каким образом выбираются элементы изображения, используемые для предсказания, системы ДИКМ могут быть разделены на три основные группы — одномерного, двумерного и трехмерного предсказания, структурные схемы которых приведены на рис. 6.9а, б, в, соответственно.

Одномерные предсказатели используют предсказание по одному измерению, например межэлементное, межстрочное, межкадровое предсказание.

При межэлементном предсказании (рис. 6.9) требуется наименьший объем памяти.

Использование в предсказателях элементов телевизионного изображения, взятых по вертикали, осложняется принятым в вещательном телевидении принципом чересстрочного разложения, поскольку корреляция между смежными эле-

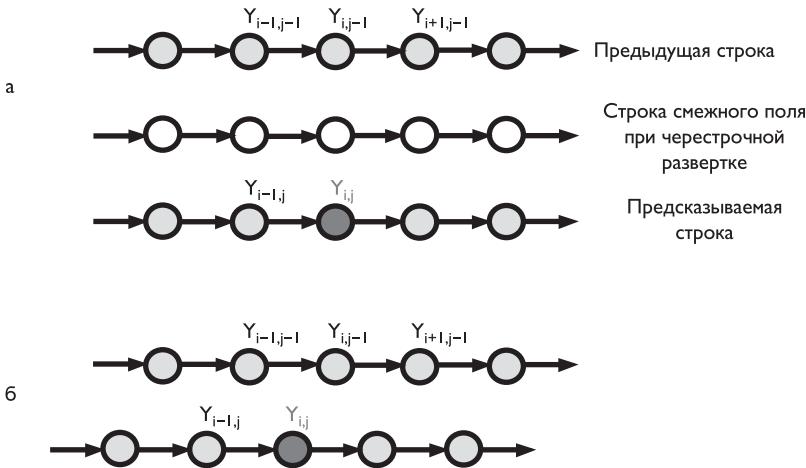


Рис. 6.10. Позиции и наименования элементов поля изображения с ортогональной и шахматной структурами

ментами соседних строк одного поля значительно ниже, чем между смежными элементами соседних строк двух последовательно передаваемых полей. Существенным препятствием в использовании в предсказателях элементов смежных полей является влияние изменений изображения в результате движения.

Двумерные предсказатели (рис. 6.9, справа сверху) используют для оценки уровней текущих пикселов элементы, смещенные как по горизонтали, так и по вертикали.

Менее всего до сих пор использовались трехмерные предсказатели (рис. 6.9, снизу), однако следует полагать, что их эффективность более высока, чем у двумерных и тем более одномерных.

Одним из важнейших вопросов реализации ДИКМ является создание рациональной шкалы квантования, использование которой позволяет уменьшить число уровней квантования.

6.3. Основные виды предсказания

6.3.1. Внутрикадровое предсказание

Алгоритм внутрикадрового предсказания зависит от структуры дискретизации изображений.

На рис. 6.10 в качестве примера приведены позиции и наименования элементов поля телевизионного изображения, используемых для предсказания элемента $X(i, j)$ при двумерной дискретизации с одинаковой начальной фазой для каждой строки.

Чаще всего для предсказания используются значения пикселов $X(i - 1, j)$, $X(i - 1, j - 1)$ и $X(i, j - 1)$.

В табл. 6.1 приведены варианты используемых коэффициентов предсказания для сигнала яркости и цветоразностных сигналов [3.17, 3.24]. В правой крайней

Таблица 6.1. Оценка различных методов предсказания

Видеосигнал	Коэффициенты предсказания для пикселов			Дисперсия разностного сигнала, σ_e^2
	$X(i-1, j)$	$X(i-1, j-1)$	$X(i, j-1)$	
E'_Y	1	—	—	53,1
	1	-1/2	1/2	29,8
	3/4	-1/2	3/4	27,9
	7/8	-5/8	3/4	26,3
E'_{R-Y}	1	—	—	22,6
	—	—	1	6,8
	1/2	-1/2	1	4,9
	5/8	-1/2	7/8	4,7
E'_{B-Y}	1	—	—	13,3
	—	—	1	3,2
	1/2	-1/2	1	2,5
	3/8	-1/4	7/8	2,5

колонке даны усредненные по множеству цветных изображений дисперсии разностных сигналов σ_e^2 . Дискретизация сигнала яркости производилась с частотой 10 МГц, а цветоразностных сигналов — с частотой 2 МГц, квантование — восьмибитовое.

Из этой таблицы следует, что предсказание цветоразностных сигналов особенно эффективно. Эти варианты приведены в табл. 6.2.

Мощность сигнала ошибки предсказания — не единственный критерий, определяющий качество предсказателя. Более важными являются субъективная оценка качества изображений при их восстановлении, влияние порядка предсказания на характер помех, вызываемых появлением ошибок в каналах связи.

Переход от одномерных видов предсказания (пп. 1–3 табл. 6.2) к двумерным хотя и приводит к уменьшению мощности ошибки предсказания, но радикально не изменяет ситуацию.

Однако при этом возможно резкое повышение качества воспроизводимых изображений и снижение влияния помех канала передачи информации. Можно выделить две основные группы алгоритмов предсказания [3.5]. Одна из них предусматривает расчет предсказуемой величины как взвешенной суммы значений ближайших к нему элементов. Весовые коэффициенты при этом положительны, характер предсказания и число учитываемых при предсказании элементов не меняется в зависимости от локальной структуры изображения.

Другая группа алгоритмов использует для предсказания не только положительные, но и отрицательные весовые коэффициенты, причем зачастую имеются два весовых коэффициента, близкие по величине, но разные по знаку. Такой алгоритм позволяет адаптироваться к локальной структуре изображения: при кодировании участков изображения с плавно изменяющимся или не изменяющимся фоном происходит компенсация влияния элементов с коэффициентами, равными по величине, но разными по знаку.

На рис. 6.11 приведены варианты изображений, определяемые разностными сигналами при использовании одномерного и двумерного алгоритмов внутрикадрового предсказания.

Для данного изображения алгоритм, использующий двумерное предсказание, является более эффективным — границы контуров изображения менее выражены.

Таблица 6.2. Различные варианты коэффициентов предсказания

№ п/п	Количество элементов, применяемых для предсказания	Коэффициенты предсказания для пикселов				Структура дискретиза- ции
		$X(i-1, j)$	$X(i-1, j-1)$	$X(i, j-1)$	$X(i+1, j-1)$	
1	1	1	–	–	–	рис. 6.10а
2	–»–	–	–	1	–	–»–
3	–»–	–	1	–	–	–»–
4	2	1/2	–	1/2	–	рис. 6.10а, б
5	–»–	1/2	–	–	1/2	–»–
6	3	1/2	1/4	1/4	–	рис. 6.10а
7	–»–	1/2	–	1/4	1/4	–»–
8	–»–	1	–1	1	–	рис. 6.10а, б
9	–»–	1	–1/2	1/2	–	–»–
10	–»–	1/2	–1/2	1	–	–»–
11	–»–	3/4	–1/2	3/4	–	–»–
12	–»–	3/4	–9/16	13/16	–	–»–
13	–»–	7/8	–5/8	3/4	–	–»–
14	4	7/8	–1/2	1/2	1/8	рис. 6.10б

ны. Заметим также, что выбор лучшего алгоритма предсказания существенно зависит от используемой структуры дискретизации изображения. Изменение периода дискретизации при любой структуре расположения отсчетов приводит к изменению корреляционных связей между близко расположенными друг к другу пикселями.

6.3.2. Межкадровое предсказание

Движущееся телевизионное изображение представляет собой последовательность кадров (полей) неподвижных изображений. Свойства неподвижных изображений с точки зрения сжатия объема, необходимого для их передачи и воспроизведения информации, рассмотрены выше.

Специфическими особенностями движущихся телевизионных изображений являются наличие временной корреляции между соответствующими элементами в последовательности сменяющих друг друга кадров и связанной с этим временной статистической избыточностью информации об уровнях пикселов изображений.

Межкадровое предсказание использует комбинацию из пикселов текущего и предыдущих кадров. Этот вид предсказания особенно эффективен при малом изменении изображения от кадра к кадру. При увеличении скорости движения в предсказателях эффективно использование лишь пикселов смежных полей. Для сцен с большим движением целесообразно использование лишь внутрикадрового предсказателя.

На рис. 6.12 приведены обозначения пикселов в двух соседних кадрах, используемых в предсказателях. Смежные поля каждого кадра условно обозначены как первое (второе) и второе (первое) поля. $X(i, j, k, 1)$ – i -й пиксел j -й строки первого поля k -го кадра.

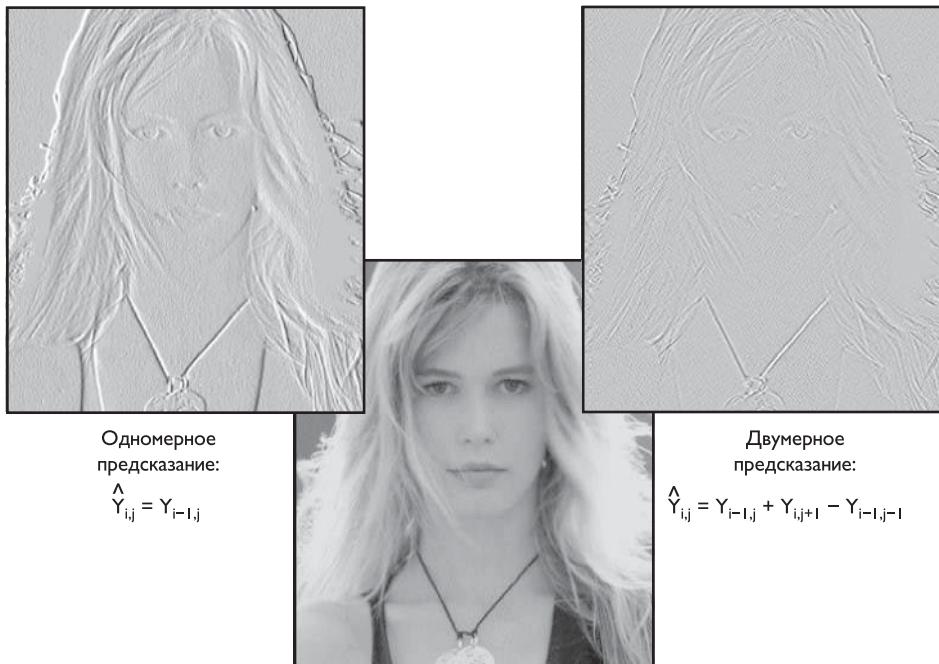


Рис. 6.11. Варианты одномерного и двумерного преобразований изображения

В табл. 6.3 приведены характеристики различных видов предсказателей и дана усредненная оценка их эффективности, характеризуемая величиной энтропии (при 35-уровневом квантовании разностного сигнала) [3.17, 3.28].

Изменение свойств предсказателей при наличии движения в передаваемом изображении связано с двумя факторами:

- при движении снижается корреляция между пикселями текущего и предыдущего кадров (полей);
- интегрирование сигнала видеокамерой приводит к возрастанию корреляции внутри поля в направлении движения.

По этой причине эффективно использование адаптивных предсказателей, изменяющих свою структуру в зависимости от скорости движения.

Например, Грэхем (Graham) [3.16] предложил адаптивный предсказатель, использующий для предсказания значение пикселя $X(i, j, k, 1)$ величины предыдущего пикселя в строке $X(i - 1, j, k, 1)$, смежного пикселя в предыдущей строке $X(i, j - 1, k, 1)$ или пикселя предыдущего кадра $X(i, j, k - 1, 1)$ в зависимости от того, какая из разностей

$$\begin{aligned}
 &|X(i - 1, j, k, 1) - X(i - 2, j, k, 1)| \text{ (межэлементная разность),} \\
 &|X(i - 1, j, k, 1) - X(i - 1, j - 1, k, 1)| \text{ (межстрочная разность),} \\
 &|X(i - 1, j, k, 1) - X(i - 1, j, k - 1, 1)| \text{ (межкадровая разность)}
 \end{aligned}$$

является минимальной.

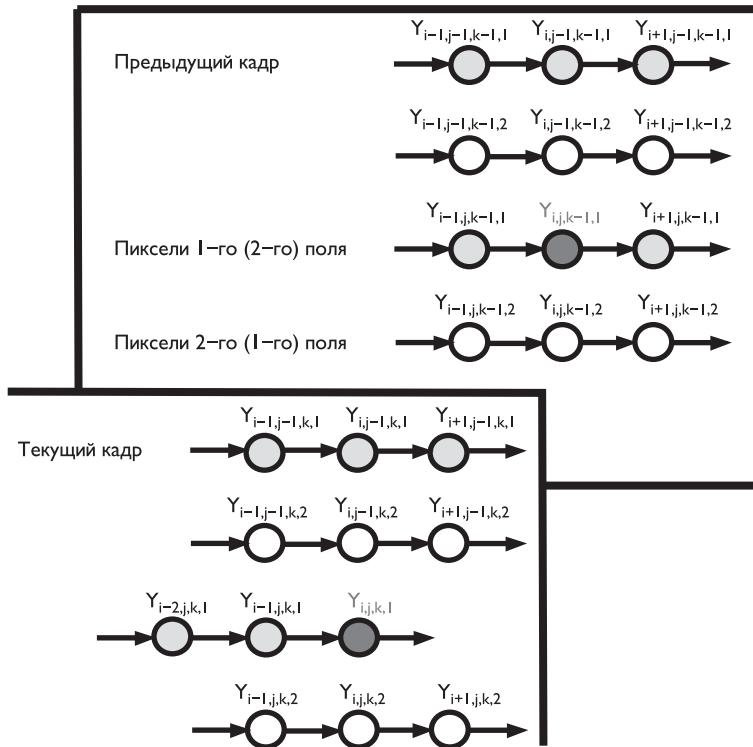


Рис. 6.12. К использованию межкадрового предсказания

Как следует из рассмотренного материала, способы обработки массивов данных статических изображений обобщены для динамических изображений.

6.4. Квантование сигналов, ошибки предсказания

Выбор типа шкалы квантования зависит от принципа формирования и метода кодирования разностного сигнала. При малом шаге квантования обычно выбирают линейную шкалу, поскольку в данном случае линейная шкала минимизирует энтропию квантованного сигнала. Однако при квантовании разностных сигналов для обеспечения достаточного качества восстановления изображений не требуется использование малого шага квантования, и разностные сигналы обычно квантуются нелинейным способом [3.1, 3.4, 3.5, 3.17, 3.18, 3.29].

Для определения шкалы квантования используют критерий минимизации среднеквадратичного отклонения и субъективную оценку качества восстанавливаемого изображения.

Применение первого критерия не вызывает особых осложнений, поскольку для этого необходимо знать лишь плотность распределения вероятностей разностного сигнала и число уровней квантования.

Таблица 6.3. Межкадровое предсказание

№ п/п	Вид предсказателя $[X(i, j, k, 1) - P_x]$	Элемент, по которому осуществляется предсказание (P_x)	Энтропия разностного сигнала, биты/пиксел
1	Межкадровая разность	$X(i, j, k - 1, 1)$	2,1–3,9
2	Межэлементная разность	$X(i - 1, j, k, 1)$	2,0–3,7
3	Межэлементная и межкадровая разность	$X(i, j, k - 1, 1) +$ $+ X(i - 1, j, k, 1) -$ $- X(i - 1, j, k - 1, 1)$	1,8–3,1
4	Межстрочная и межкадровая разность	$X(i, j, k - 1, 1) +$ $+ X(i, j - 1, k, 1) -$ $- X(i, j - 1, k - 1, 1)$	1,5–3,5
5	Межполевая разность	$[X(i, j - 1, k, 2) +$ $+ X(i, j, k, 2)]/2$	1,8–2,5
6	Межэлементная и межполевая разность	$X(i - 1, j, k, 1) +$ $+ [X(i, j - 1, k, 2) +$ $+ X(i, j, k, 2)]/2 -$ $- [X(i - 1, j - 1, k, 2) +$ $+ X(i - 1, j, k, 2)]/2$	1,5–2,5

Плотность распределения вероятностей разностного сигнала, как уже было отмечено, подчиняется закону Лапласа

$$p(x) = \frac{\lambda}{2} \cdot \exp(-\lambda|x|), \quad (6.5)$$

где λ — вариация, связанная со среднеквадратичным отклонением σ соотношением $\lambda = 2/\sigma$.

При определении уровней и порогов квантования обычно пользуются критерием Ллойда–Макса [3.17, 3.18, 3.30], описанным в Части I.

При этом плотность распределения разностного сигнала разбивают на ряд участков таким образом, чтобы вероятности попадания уровней сигнала в каждый участок были одинаковыми. Границы этих участков и являются пороговыми уровнями нелинейной шкалы квантования.

Далее вычисляется «центр тяжести» каждого интервала, являющийся соответствующим уровнем квантования для всех значений разностного сигнала, попадающий в интервал между соседними пороговыми уровнями.

Процесс определения пороговых уровней и уровней квантования поясняется на рис. 6.13.

Определение пороговых уровней d_j и уровней квантования r_j сводится к решению уравнений:

$$\frac{1}{m} = \int_{d_{j-1}}^{d_j} p(x) dx, \quad i = 1, \dots, m - 1, \quad (6.6)$$

$$r_j = \int_{d_{j-1}}^{d_j} xp(x) dx, \quad i = 1, \dots, m - 1, \quad (6.7)$$

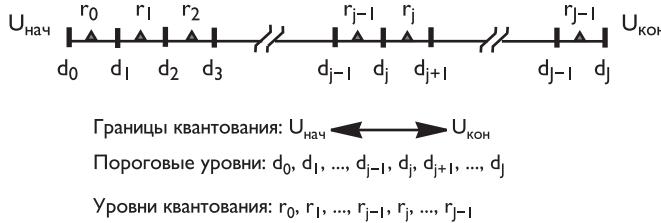


Рис. 6.13. К определению пороговых уровней и уровней квантования

Таблица 6.4. Пороги и уровни квантования, оптимальные для распределения Лапласа

1 разряд		2 разряда		3 разряда		4 разряда	
Пороговые уровни	Уровни квантования						
						$-\infty$	-4,4311
						-3,7240	-3,0169
						-2,5971	-2,1773
						-1,8776	-1,5778
						- ∞	-3,0867
						-2,3796	-1,6752
						- ∞	-0,9198
						-1,2527	-0,8330
						-0,5332	-0,2334
$-\infty$	-0,7071	-1,1269	-0,4198	-0,5332	-0,2334	-0,2664	-0,1240
0,0000	0,7071	0,0000	0,4198	0,0000	0,2334	0,0000	0,1240
∞		1,1269	1,8340	0,5332	0,8330	0,2664	0,4048
		∞		1,2527	1,6752	0,5667	0,7287
				2,3796	3,0867	0,9198	1,1110
					∞	1,3444	1,5778
						1,8776	2,1773
						2,5971	3,0169
						3,7240	4,4311
						∞	

где d_j , r_j — соответственно j -е порог и уровень квантования, m — число уровней шкалы.

В табл. 6.4 дано расположение уровней квантования и пороговых уровней квантователей Ллойда–Макса при Лапласовой плотности распределения разностного сигнала и $\sigma = 1$. В случае если преобразуется цифровой многоуровневый сигнал, значения пороговых уровней и уровней квантования могут быть определены следующим образом:

$$d_{j\text{ц}} = \text{round}(N\sigma d_j), \quad r_{j\text{ц}} = \text{round}(N\sigma r_j),$$

где d_j , r_j — значения пороговых уровней и уровней квантования, определяемые по табл. 6.4; σ — относительная величина среднеквадратичного отклонения разностного сигнала; N — число уровней квантования исходного сигнала.

Другой стратегией в определении шкалы квантования в системах с предсказанием является учет визуального восприятия при выборе пороговых уровней и уровней квантования [3.1, 3.5, 3.31–3.33].

При субъективной экспертизе оптимизация шкалы квантования может определяться различными способами.

Например, один из них предполагает, что в распоряжении исследователей имеется достаточно большой набор различных шкал квантования, из которых необходимо выбрать шкалу, обеспечивающую наилучшее субъективное качество воспроизведимых изображений. В этом случае используется абсолютная шкала оценок качества.

Другой способ основан на изменении расстановки пороговых уровней и уровней квантования с использованием результатов субъективных экспертиз, приводимых по относительной шкале оценки качества. При этом исходят из предпосылки, что отдельные участки шкалы квантования искажают в основном различные части изображения.

Так, уровни шкалы квантования, расположенные в районе нуля шкалы мгновенных значений, в основном обеспечивают качество передачи монотонных перепадов яркости кодируемого изображения и отсутствие/наличие гранулярного шума, т. е. шума квантования на участках изображения с мало изменяющейся яркостью.

Средние участки шкалы квантования в основном определяют качество воспроизведения участков изображения со средней скоростью изменения яркости. Кроме того, эти участки используются при квантовании мелких деталей.

Участки шкалы квантования, наиболее удаленные от центра шкалы, предназначены для передачи резких перепадов яркости изображений.

Обращают на себя внимание значительные отличия шкал квантования, оптимизированных по субъективному критерию, и шкал квантования, рассчитанных по критерию Ллойда–Макса.

Основные отличия этих шкал заключаются в различной расстановке уровней квантования в средней части шкалы. Это понятно, поскольку заметность ложных контуров на изображении с большим количеством переходов и мелких деталей значительно ниже.

На рис. 6.14 приведены три монохромных изображения (и их увеличенные фрагменты): исходный (слева) и восстановленные из разностных сигналов с двумя вариантами четырехбитовой шкалы квантования, определенной по минимуму квадрата ошибки (в центре) и по максимуму субъективной оценки качества (справа).

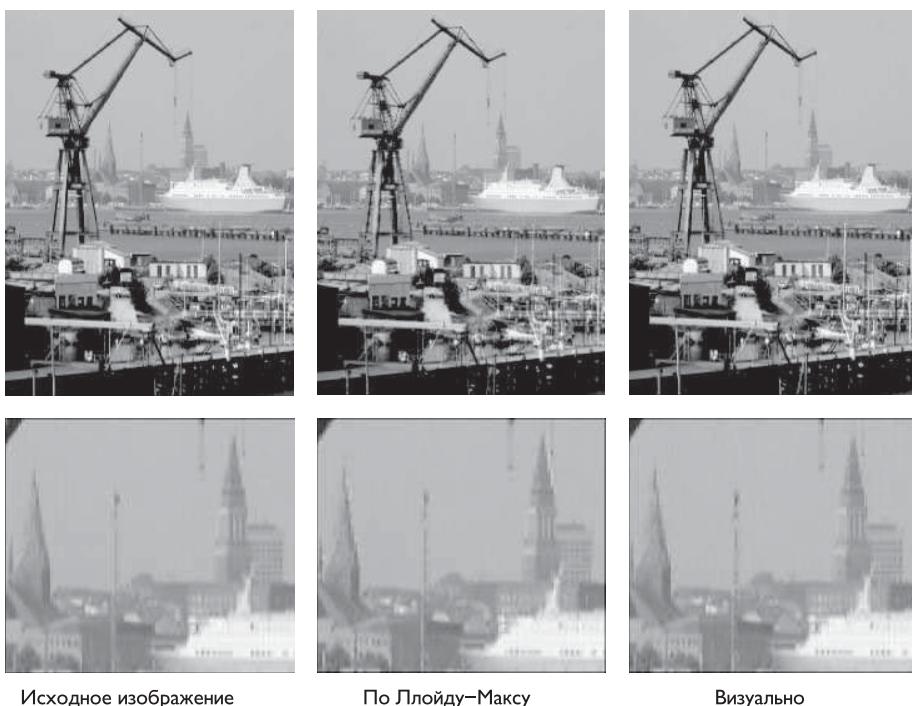
Следует указать, что эффективность использования второй шкалы невысока.

Для уменьшения ошибки квантования иногда применяют адаптивный метод, при котором производится мгновенное компандирование шкалы квантования. В зависимости от распределения яркости на локальном участке изображения изменяется расстановка уровней квантования, например, переключением со шкалой квантования с малым шагом на шкалу с большим шагом. Путем такой адаптации к локальным распределениям яркости участков изображения обеспечивается дополнительное сокращение объема передаваемой информации или повышение качества ее передачи.

Команда на переключение шкал квантования производится по результатам анализа окружающих элементов и предыдущего предсказания.

При передаче участков ровного фона или плавных перепадов яркости используется точная шкала квантования, при резких изменениях сигнала применяется более грубая шкала.

Линейное предсказание с оптимизированным квантованием



Исходное изображение

По Ллойду-Максу

Визуально

Рис. 6.14. Монокромные изображения с 4-битовой шкалой квантования

В табл. 6.5 приведены результаты субъективных экспертиз качества телевизионных изображений, кодируемых с использованием предсказателей [3.5].

Таблица 6.5. Субъективная оценка кодирования с предсказанием

№ п/п	Вид предсказания	Число бит/пиксель	Оценка качества по пятибалльной шкале
1	Одномерное неадаптивное	3	3,0
		4	4,0
		5	4,5
		6	5,0
2	Одномерное адаптивное с коммутацией шкал	4	4,5
		5	5,0
3	Двумерное неадаптивное	3	4,0
		4	4,5
4	Двумерное адаптивное с коммутацией шкал	3	4,5
		4	5,0



Рис. 6.15. Примеры влияния помех на искажения при использовании межэлементного (слева) и межстрочного (справа) предсказания

6.5. Помехоустойчивость кодирования с предсказанием

При передаче изображений с использованием предсказаний особо важным является вопрос помехоустойчивости канального кодирования информации. При импульсно-кодовой модуляции (ИКМ) видеинформации помеха искажает кодовые посылки только того пикселя, который передавался в момент возникновения помехи. Если помеха одиночная, то она поражает лишь один пиксель изображения. По этой причине заметны лишь те помехи, которые поражают старшие разряды ИКМ-кодов.

Усреднение яркостей пикселов смежных кадров в значительной мере снижает заметность даже существенно нерегулярных искажений пикселов.

В системах с предсказанием даже одиночная помеха при восстановлении вызывает искажения ряда пикселов [3.5, 3.34, 3.35]. Возникает трек ошибки, характер которого зависит от способа предсказания, а также частоты передачи опорных отсчетов.

При одномерном предсказании по одному пикселу будут искажены все последующие пиксели в направлении предсказания. На рис. 6.15 приведены примеры таких искажений для случаев предсказания в горизонтальном и вертикальном направлениях.

Помеха как бы «выбивает» следующую за ней часть изображения в направлении предсказания. Главная причина, из-за которой возникают столь заметные искажения, вызвана правилом декодирования кодовой последовательности на приемной стороне — для восстановления k -го пикселя необходимо сложить все поступающие из канала связи ошибки предсказания:

$$\hat{x}_k = \sum_{i=1}^k \hat{e}_i,$$

где i — порядковый номер элемента разностного сигнала, отсчитываемый в направлении предсказания от опорного пикселя x_0 .

Таким образом, каждая возникающая во время передачи j -го элемента предсказания ошибка $\Delta \hat{e}_i$ будет присутствовать при восстановлении всех пикселов, расположенных в направлении предсказания $k \geq j$.

Чтобы уменьшить влияние ошибок канала связи, нужно либо сократить период между опорными отсчетами, либо уменьшить вес, с которым учитывается каждое приходящее значение ошибки предсказания. Последнее эффективно реализуется при увеличении порядка предсказания (см. табл. 6.2).

Скорость убывания размаха помехи и ее заметность определяются соотношениями весовых коэффициентов и алгоритмом предсказания. При этом происходит перераспределение энергии помехи в нескольких направлениях, что вызывает резкое снижение ее заметности.

ГЛАВА 7

ГРУППОВОЕ КОДИРОВАНИЕ ИЗОБРАЖЕНИЙ

При кодировании изображений с предсказанием возможности сокращения цифрового потока ограничены. Это связано в первую очередь с тем, что предсказание производится поэлементно, и использование нелинейного квантования не позволяет существенно сократить визуальную избыточность без снижения качества восстанавливаемого изображения. При кодировании с предсказанием практически не удается обойтись меньшим объемом информации, чем 4 бита на пиксель.

Альтернативой кодированию, использующему декорреляцию видеинформации предсказанием, является групповое кодирование, основанное на разбиении изображений на участки и одновременном кодировании сразу группы элементов. При этом более экономное кодирование связано с одной стороны с использованием условного кодирования (см. часть I), а с другой стороны — с возможностью нарушения соотношений между пикселями таким образом, что это не приведет к визуально заметным искажениям.

При групповом кодировании обычно используют преобразование исходных данных. В этом случае повышение эффективности кодирования связано, в основном, с тремя факторами [3.1, 3.5, 3.17, 3.18, 3.36–3.39]:

- в процессе преобразования ряд коэффициентов становится настолько малыми по величине, что их можно отбросить без заметного изменения качества восстанавливаемых изображений;
- в процессе преобразования осуществляется декорреляция данных, обеспечивающая повышение эффективности статистического кодирования;
- различное нелинейное квантование коэффициентов преобразования позволяет существенно сократить объем передаваемой информации без заметного изменения качества изображения при его восстановлении.

7.1. Дискретные линейные ортогональные преобразования

Широкое распространение получили так называемые унитарные преобразования, являющиеся частным случаем линейных преобразований, при которых линейный оператор обратим (т. е. матрица, умножением на которую реализуется преобразование, обязательно имеет обратную ей матрицу, умножением на которую исходные данные могут быть точно восстановлены), а его ядро удовлетворяет условиям ортогональности [3.18, 3.40–3.44].

Предположим, что прямоугольный участок изображения размером $N_1 \times N_2$ пикселов $X(n_1, n_2)$, $1 \leq n_1 \leq N_1$; $1 \leq n_2 \leq N_2$, отображается линейным преобразованием в компоненты $F(m_1, m_2)$:

$$F(m_1, m_2) = \sum_{n_1=1}^{N_1} \sum_{n_2=1}^{N_2} X(n_1, n_2) \cdot A(n_1, n_2; m_1, m_2), \quad (7.1)$$

где $1 \leq m_1 \leq N_1$, $1 \leq m_2 \leq N_2$, $A(n_1, n_2; m_1, m_2)$ — ядро прямого преобразования. Исходный участок изображения можно получить с помощью обратного преобразования:

$$X(n_1, n_2) = \sum_{m_1=1}^{N_1} \sum_{m_2=1}^{N_2} F(m_1, m_2) \cdot B(n_1, n_2; m_1, m_2), \quad (7.2)$$

$B(n_1, n_2; m_1, m_2)$ — ядро обратного преобразования. Преобразование является унитарным, если выполняются условия:

$$\begin{aligned} & \sum_{m_1=1}^{N_1} \sum_{m_2=1}^{N_2} A(n_1, n_2; m_1, m_2) \cdot A^*(j_1, j_2; m_1, m_2) = \\ &= \sum_{m_1=1}^{N_1} \sum_{m_2=1}^{N_2} B(n_1, n_2; m_1, m_2) \cdot B^*(j_1, j_2; m_1, m_2) = \delta(n_1 - j_1, n_2 - j_2), \end{aligned} \quad (7.3)$$

а также

$$\begin{aligned} & \sum_{n_1=1}^{N_1} \sum_{n_2=1}^{N_2} A(n_1, n_2; m_1, m_2) \cdot A^*(n_1, n_2; k_1, k_2) = \\ &= \sum_{n_1=1}^{N_1} \sum_{n_2=1}^{N_2} B(n_1, n_2; m_1, m_2) \cdot B^*(n_1, n_2; k_1, k_2) = \delta(m_1 - k_1, m_2 - k_2). \end{aligned} \quad (7.4)$$

Здесь $*$ — знак сопряженного преобразования, получаемого комплексным сопряжением и транспонированием матрицы преобразования.

Преобразование называется разделимым, если

$$\begin{aligned} A(n_1, n_2; m_1, m_2) &= A_C(n_1, m_1) \cdot A_R(n_2, m_2), \\ B(n_1, n_2; m_1, m_2) &= B_C(n_1, m_1) \cdot B_R(n_2, m_2), \end{aligned} \quad (7.5)$$

где $A_C(B_C)$ и $A_R(B_R)$ — одномерные операторы преобразования столбцов и строк.

В этом случае результат воздействия двумерного унитарного преобразования можно получить в два этапа — сначала выполняется одномерное преобразование по всем столбцам блока изображения, а затем над полученными результатами проводится преобразование по всем строкам:

$$\begin{aligned} P(m_1, n_2) &= \sum_{n_1=1}^{N_1} X(n_1, n_2) \cdot A_C(n_1, m_1); \\ F(m_1, m_2) &= \sum_{n_2=1}^{N_2} P(m_1, n_2) \cdot A_R(n_2, m_2). \end{aligned} \quad (7.6)$$

Возможны различные интерпретации унитарных преобразований.

Преобразование изображения можно рассматривать как разложение исходного изображения в обобщенный двумерный спектр. Каждая спектральная состав-

ляющая характеризует вклад соответствующей спектральной (базисной) функции в энергию исходного изображения.

При такой трактовке понятие частоты можно обобщить так, чтобы оно было применимо не только к косинусам и синусам, но и к другим функциям, на которых основываются преобразования.

Наглядное представление об унитарных преобразованиях изображений можно получить, рассматривая такое преобразование как поворот многомерной системы координат, при котором область возможных изменений векторов максимально сжимается («компрессируется»).

Еще одна возможность интерпретации преобразований заключается в том, что преобразование $F(m_1, m_2)$ в $X(n_1, n_2)$ можно представить как способ представления изображения в виде набора двумерных функций $B(n_1, n_2; m_1, m_2)$, каждая из которых соответствует точке (m_1, m_2) плоскости обобщенных частот. В подобной интерпретации ядро $B(n_1, n_2; m_1, m_2)$ называется двумерной базисной функцией, а коэффициент $F(m_1, m_2)$ указывает «вес» этой базисной функции, необходимый для получения рассматриваемого изображения.

Обычно при анализе изображений используют квадратные блоки изображений размером $N \times N$.

7.1.1. Дискретное преобразование Фурье

Дискретное двумерное преобразование Фурье матрицы отсчетов изображения определяется в виде ряда [3.5, 3.18, 3.36, 3.45]:

$$F(u, v) = \frac{1}{N} \cdot \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} X(j, k) \cdot \exp\left[-\frac{2\pi i}{N}(uj + vk)\right], \quad (7.7)$$

а обратное преобразование имеет вид:

$$X(j, k) = \frac{1}{N} \cdot \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} F(u, v) \cdot \exp\left[\frac{2\pi i}{N}(uj + vk)\right]. \quad (7.8)$$

По аналогии с терминологией непрерывного преобразования Фурье переменные u и v называют пространственными частотами. Поскольку ядра преобразования симметричны и разделимы, двумерное преобразование можно выполнить в виде последовательных одномерных преобразований по строкам и столбцам матрицы изображения.

Базисными функциями преобразования являются экспоненты с комплексными показателями, которые можно разложить на косинусную и синусную составляющие:

$$\begin{aligned} A_R(n, m) = A_C(n, m) = t_n = \frac{1}{\sqrt{N}} \exp\left(-\frac{2\pi i}{N}nm\right) = \\ = \frac{1}{\sqrt{N}} \cos \frac{2\pi}{N}nm - \frac{i}{\sqrt{N}} \sin \frac{2\pi}{N}nm; \end{aligned} \quad (7.9)$$

$$\begin{aligned} B_R(n, m) = B_C(n, m) = t_n^* = \frac{1}{\sqrt{N}} \exp\left(\frac{2\pi i}{N}nm\right) = \\ = \frac{1}{\sqrt{N}} \cos \frac{2\pi}{N}nm + \frac{i}{\sqrt{N}} \sin \frac{2\pi}{N}nm; \end{aligned} \quad (7.10)$$

$0 \leq n, m \leq N - 1$.

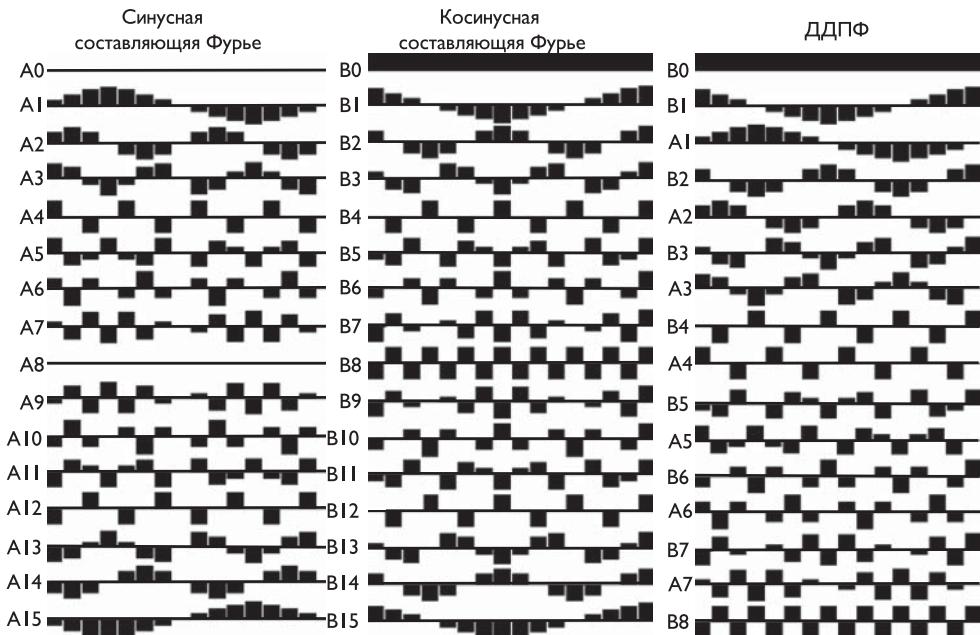


Рис. 7.1. Одномерные базисные функции преобразования Фурье при $N = 16$

На рис. 7.1 слева и в центре приведены графики соответственно синусных и косинусных составляющих одномерных базисных функций преобразования Фурье для $N = 16$.

Из этого рисунка видно, что для низких частот эти функции являются грубыми аппроксимациями непрерывных синусоид. С повышением частоты сходство базисных функций с синусоидами теряется. Для наивысшей частоты базисная функция представляет собой меандр.

Можно заметить избыточность наборов синусных и косинусных составляющих: при преобразовании N чисел X_1, \dots, X_N формируется N комплексных коэффициентов

$$\begin{aligned}
 P(u) &= \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} X_n \exp\left(-\frac{2\pi i}{N} nu\right) = \\
 &= \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} X_n \cos \frac{2\pi}{N} nu - \frac{i}{\sqrt{N}} \sum_{n=0}^{N-1} X_n \sin \frac{2\pi i}{N} nu; \quad (7.11)
 \end{aligned}$$

$0 \leq u \leq N$, состоящих из N действительных и N мнимых чисел.

При этом очевидно, что $P(1 + m) = P^*(N - m - 1)$, $0 \leq m \leq N - 2$. По этой причине обычно используется так называемое действительное дискретное Фурье преобразование (ДДПФ), базисные функции которого с учетом нормирующего множителя определяются, например, следующим образом [3.17]:

$$\begin{aligned}
 t_0^R &= t_0; \\
 &\dots \\
 t_{2n-1}^R &= \sqrt{2} R e t_n; \\
 t_{2n}^R &= \sqrt{2} I m t_n; \\
 &\dots \\
 t_N^R &= \begin{cases} \sqrt{2} I m t_{\frac{N-1}{2}}, & N = 2M-1, \\ t_{\frac{N}{2}}, & N = 2M. \end{cases}
 \end{aligned} \tag{7.12}$$

Эти базисные функции приведены на рис. 7.1. справа.

Возможен и другой вариант реализации ДДПФ. Например, для четного N рассчитываются значения новых дискретных отсчетов:

$$\begin{aligned}
 g_1(j) &= \frac{X_{\frac{N}{2}-j} + X_{\frac{N}{2}+j}}{2}, & j = 0, \dots, \frac{N}{2}-1; \\
 g_2(j) &= \frac{-X_{\frac{N}{2}-j} + X_{\frac{N}{2}+j}}{2}, & j = 0, \dots, \frac{N}{2}-1.
 \end{aligned}$$

и формируется N также действительных коэффициентов:

$$\begin{aligned}
 P_{\cos}(u) &= \sqrt{\frac{2}{N}} \sum_{j=0}^{\frac{N}{2}-1} g_1(j) \cos \frac{2\pi}{N} j u, & u = 0, \dots, \frac{N}{2}; \\
 P_{\sin}(u) &= \sqrt{\frac{2}{N}} \sum_{j=0}^{\frac{N}{2}-1} g_2(j) \sin \frac{2\pi}{N} j u, & u = 0, \dots, \frac{N}{2},
 \end{aligned}$$

характеризующих величину и фазу соответствующих спектральных компонентов сигнала

$$\begin{aligned}
 \text{mod } P(0) &= P_{\cos}(0), \\
 &\dots \\
 \text{mod } P(u) &= \sqrt{P_{\cos}^2(u) + P_{\sin}^2(u)}, \\
 \arg P(u) &= \operatorname{Arctg} \frac{P_{\sin}(u)}{P_{\cos}(u)}, \quad u = 1, \dots, \frac{N}{2}-1, \\
 &\dots \\
 \text{mod } P\left(\frac{N}{2}\right) &= P_{\cos}\left(\frac{N}{2}\right), \quad \arg P\left(\frac{N}{2}\right) = 0.
 \end{aligned} \tag{7.13}$$

Величины спектральных компонент сигнала $P(u)$, как правило, с ростом частоты u/N падают. Это приводит к уплотнению энергии в коэффициентах с меньшими частотными индексами.

Действительное дискретное преобразование Фурье удобно для применения в связи с возможностью реализации быстрых алгоритмов расчета (так называемое быстрое преобразование Фурье — БПФ). Если для реализации ДПФ требуется выполнение примерно N^2 комплексных умножений и сложений, то при БПФ необходимо обеспечить $N \cdot \log_2 N$ таких операций [3.46–3.50].



Сокращение визуальной избыточности информации при использовании унитарных преобразований, в том числе преобразования Фурье, связано с осуществлением различных видов квантования полученных спектральных компонент [3.17]. Наиболее простой вариант квантования связан с формированием кодов

$$Q'(u) = \text{round}[Q(u)/K_Q], \quad (7.14)$$

где обозначение $Q(u)$ характеризует действительные и мнимые части коэффициентов Фурье-преобразования $P_{\cos}(u)$ и $P_{\sin}(u)$, либо их модуль $\text{mod}(P(u))$ и фазу $\arg(P(u))$; K_Q — некоторые числа (обычно кратные 2), зависящие только от вида информации Q .

При восстановлении изображения определение коэффициентов, по которым производится расчет уровней соответствующих пикселов, осуществляется по формуле

$$\hat{Q}(u) = Q \cdot K_Q. \quad (7.15)$$

Увеличение коэффициента K_Q приводит к увеличению коэффициента сжатия изображения и к увеличению погрешностей восстановления пикселов.

Таким образом, степень сжатия и погрешность преобразования являются функцией коэффициента K_Q : при уменьшении этого коэффициента уменьшается погрешность преобразования, но падает степень сжатия. Для каждого унитарного преобразования существуют свои оптимальные значения K_Q .

Исследования сокращения избыточности изображений с использованием Фурье-преобразования показали его недостаточную эффективность.

В то же время известное свойство инвариантности модуля спектра Фурье относительно сдвига входного сигнала при простом преобразовании фазового спектра — смещении на некоторую величину, линейно зависящую от частоты спектральных компонент, позволяет надеяться на возможность использования ДПФ при анализе изображений, содержащих элементы, подобные друг другу с точностью до сдвига.

Эта возможность представляет собой альтернативу распространенным методам поиска величины сдвига, основанным на итерационной минимизации суммы модулей (или квадратов) разностей соответствующих величин в двух блоках.

7.1.2. Преобразование Хаара

Преобразование Хаара основывается на использовании ортогональной матрицы Хаара [3.17, 3.18, 3.38, 3.39, 3.51]. В табл. 7.1 приведены примеры ортогональных матриц Хаара четвертого, восьмого и шестнадцатого порядков. Преобразование Хаара можно рассматривать как процесс дискретизации исходного сигнала, при котором с переходом к следующей строке вдвое уменьшается шаг дискретизации.

Хааровский спектр описывает распределение энергии компонент, соответствующих разностям яркостей соседних пикселов, разностям средних значений яркостей соседних пар пикселов и вообще разностям средних значений яркостей соседних групп из 2^m элементов.

На рис. 7.2 приведены псевдотрехмерные матрицы блока изображения с числом пикселов 8×8 (слева), хааровского спектра (с округлением коэффициентов) и ошибок восстановления значений пикселов исходного блока изображения (справа внизу).

Отметим, что в спектре Хаара наблюдается концентрация энергии в областях низких хааровских частот.

Таблица 7.1. Ортогональные матрицы Хаара

Размерность матрицы	Матрица Хаара
4×4	$H_4 = \frac{1}{\sqrt{4}} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & \sqrt{2} & -\sqrt{2} & 0 \\ \sqrt{2} & 0 & 0 & \sqrt{2} \\ 0 & 0 & \sqrt{2} & -\sqrt{2} \end{bmatrix}$
8×8	$H_8 = \frac{1}{\sqrt{8}} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 \\ \sqrt{2} & \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sqrt{2} & \sqrt{2} & -\sqrt{2} \\ -2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -2 & -2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -2 & -2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 \end{bmatrix}$
16×16	$H_8 = \frac{1}{\sqrt{16}} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ \sqrt{2} & \sqrt{2} & \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} & -\sqrt{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \sqrt{2} & \sqrt{2} & \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} & -\sqrt{2} & -\sqrt{2} \\ 2 & 2 & -2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 2 & -2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 2 & -2 & -2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 & -2 & 2 & 2 & -2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$

7.1.3. Преобразование Уолша–Адамара

Преобразование Уолша–Адамара можно рассматривать как дискретный аналог непрерывного преобразования по базису, составленному из функций Уолша [3.5, 3.17, 3.18, 3.36, 3.37, 3.52–3.54]. Оно основано на квадратной матрице, элементы которой равны плюс или минус единице, а строки и столбцы образуют ортогональные векторы.

Среди ортогональных матриц Уолша–Адамара наименьшей является матрица второго порядка $N = 2$:

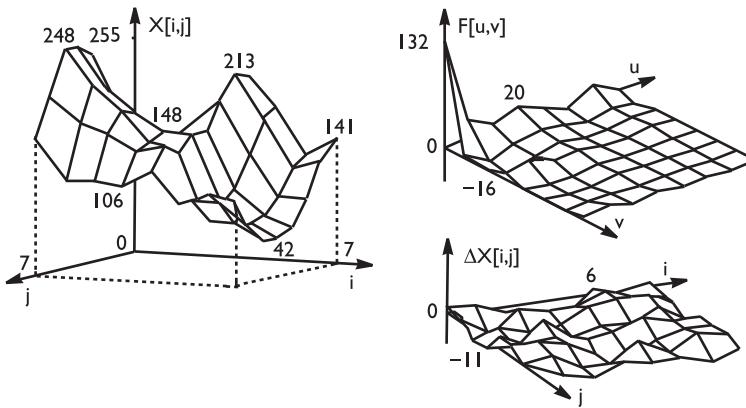


Рис. 7.2. Блок изображения 8×8 пикселов и его преобразование Хаара

$$H_2 = \frac{1}{\sqrt{2}} \begin{vmatrix} 1 & 1 \\ 1 & -1 \end{vmatrix}.$$

Наиболее просто удается построить матрицы Уолша–Адамара, если они имеют порядок $N = 2^n$, где n — целое число.

Если H_N — матрица Уолша–Адамара N -го порядка, то

$$H_{2N} = \frac{1}{\sqrt{2}} \begin{vmatrix} H_N & H_N \\ H_N & -H_N \end{vmatrix} \dots$$

Например, число изменений знака

$$H_4 = \frac{1}{\sqrt{2}} \begin{vmatrix} H_2 & H_2 \\ H_2 & -H_2 \end{vmatrix} = \frac{1}{2} \begin{vmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{vmatrix} = \begin{matrix} \dots & \dots & \dots & -0 \\ \dots & \dots & \dots & -3 \\ \dots & \dots & \dots & -1 \\ \dots & \dots & \dots & -2 \end{matrix}$$

Строки матрицы H_{2N} можно рассматривать как последовательность отсчетов прямоугольных периодических колебаний (сигналов), период которых кратен $1/N$. Следовательно, матрица Уолша–Адамара описывает преобразование, связанное с разложением блоков изображений по семейству прямоугольных базисных функций, а не по синусам и косинусам, характерным для преобразований Фурье.

Обычно матрицу Уолша–Адамара упорядочивают путем перестановок строк в порядке возрастания числа изменений знака.

Например, упорядоченная матрица 4-го порядка имеет вид:

$$\text{число изменений знака } H_{4y} = \frac{1}{2} \begin{vmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{vmatrix} = \begin{matrix} \dots & \dots & \dots & -0 \\ \dots & \dots & \dots & -1 \\ \dots & \dots & \dots & -2 \\ \dots & \dots & \dots & -3 \end{matrix}$$

Для упорядоченной матрицы Адамара порядка $N = 2^n$ двумерное преобразование можно представить в виде

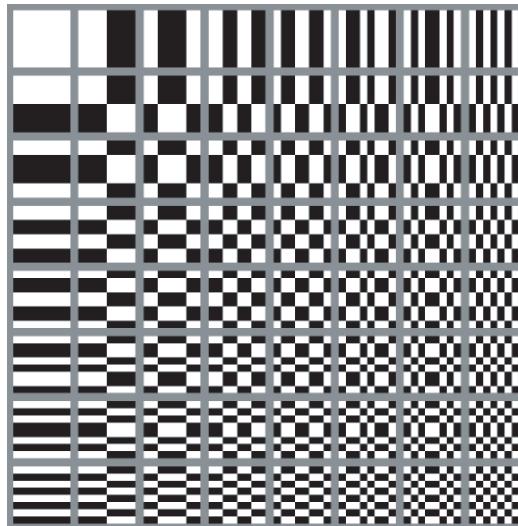


Рис. 7.3. Базисные функции упорядоченного преобразования Уолша–Адамара при $N = 8$

$$F(u, v) = \frac{1}{N} \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} X(j, k) \cdot (-1)^{q(j, k, u, v)}, \quad (7.16)$$

где $q(j, k, u, v) = \sum_{i=0}^{n-1} [g_i(u) \cdot j_i + g_i(v) \cdot k_i]$, причем

$$\begin{aligned} g_0(u) &= u_{n-1}; \\ g_1(u) &= u_{n-1} + u_{n-2}; \\ g_2(u) &= u_{n-2} + u_{n-3}; \\ \dots & \\ g_{n-1}(u) &= u_1 + u_0; \end{aligned}$$

u_i, v_i, j_i, k_i — цифры в двоичном представлении чисел u, v, j, k соответственно. Например, если

$$u = 13 = 1 \cdot 2^0 + 0 \cdot 2^1 + 1 \cdot 2^2 + 1 \cdot 2^3,$$

то

$$u_0 = 1; u_1 = 0; u_2 = 1; u_3 = 1.$$

На рис. 7.3 приведено изображение базисных функций упорядоченного преобразования Уолша–Адамара при $N = 8$.

На рис. 7.4 приведены псевдотрехмерные матрицы блока изображения, аналогичного рис. 7.2, спектра Уолша–Адамара (с округлением коэффициентов) и ошибок восстановления значений пикселов исходного блока изображения.

Отметим, что, как и в преобразовании Хаара, здесь наблюдается концентрация энергии в областях низких частот. Эффективность использования преобразования Уолша–Адамара для сжатия изображений практически такая же, как эффективность применения преобразования Хаара.

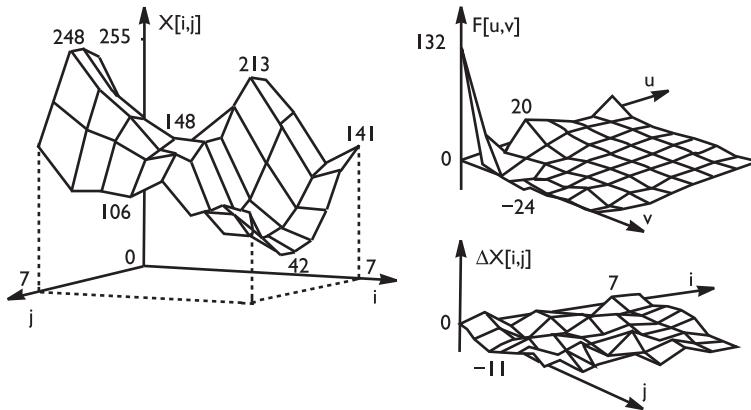


Рис. 7.4. Блок изображения 8×8 пикселов и его преобразование Уолша–Адамара

7.1.4. Дискретное синусное преобразование

Двумерное синусное преобразование определяется соотношением [3.18, 3.55]:

$$F(u, v) = \frac{2}{N+1} \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} X(j, k) \cdot \sin \frac{\pi(j+1)(u+1)}{N+1} \times \sin \frac{\pi(k+1)(v+1)}{N+1}, \quad u, v = 0, \dots, N-1. \quad (7.17)$$

Обратное преобразование имеет такой же вид. Графики базисных функций синусного преобразования для $N = 8$ приведены на рис. 7.5.

На рис. 7.6. показаны три псевдотрехмерные матрицы: блока изображения, аналогичного рис. 7.4, спектра синусного преобразования и ошибок восстановления значений пикселов исходного блока.

Концентрация энергии в областях низких частот синусного преобразования также очевидна. Ошибка восстановления в данном случае меньше, чем при использовании преобразований Уолша–Адамара и Хаара, что говорит о большей эффективности синусного преобразования.

7.1.5. Дискретные косинусные преобразования

Известно, что ряд Фурье для любой непрерывной действительной и симметричной (четной) функции содержит только действительные коэффициенты, соответствующие косинусным членам ряда. В соответствующей интерпретации этот результат можно распространить и на дискретное преобразование Фурье изображений.

Имеется несколько способов преобразования блоков изображений с целью создания симметричных дискретных функциональных соотношений [3.18, 3.56].

Один из них заключается в том, что к изображению вплотную пристраивают его зеркальное отражение, как показано на рис. 7.7 слева.

Второй способ заключается в том, что оригинал и отражения пристраиваются, посредством наложения крайних элементов (рис. 7.7, справа).

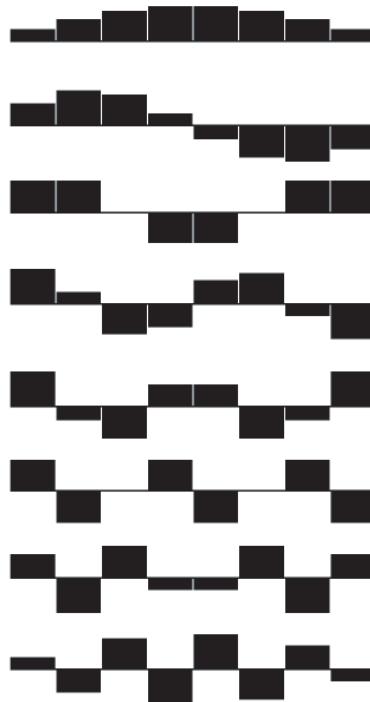


Рис. 7.5. Базисные функции синусного преобразования при $N = 8$

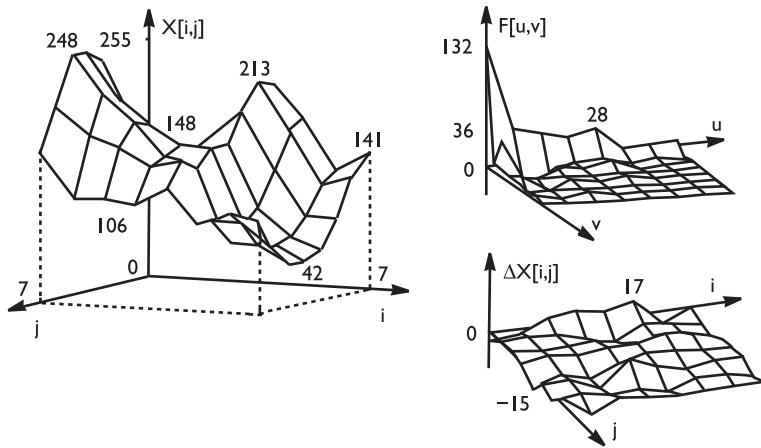


Рис. 7.6. Блок изображения 8×8 пикселов и его синусное преобразование

Таким образом, из первоначального массива, содержащего $N \times N$ пикселов, в первом случае, называемом четным косинусным преобразованием, получают массив $2N \times 2N$ элементов, а во втором случае — нечетном косинусном преобразовании — массив $(2N - 1) \times (2N - 1)$ элементов.

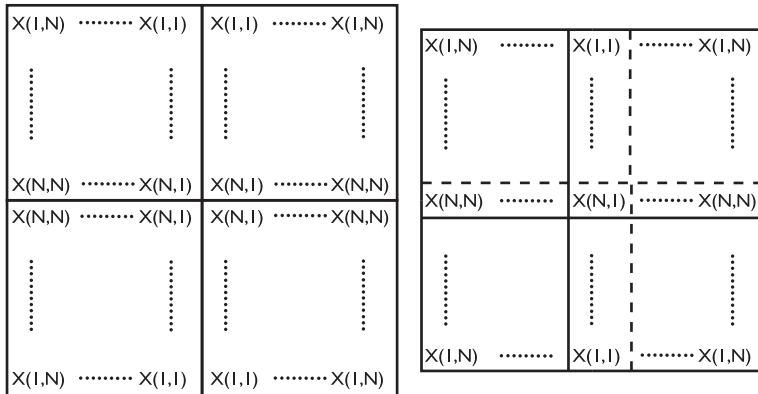


Рис. 7.7. Способы обработки блоков изображения при создании косинусных преобразований

Вычисление двумерного преобразования Фурье при использовании наложения крайних элементов блока приводит к **нечетному косинусному преобразованию**, определяемому соотношениями:

$$F(u, v) = \frac{1}{N} \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} \tilde{X}(j, k), u, v = 0; \quad (7.18)$$

$$F(u, v) = \frac{2}{N} \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} \tilde{X}(j, k) \cdot \cos \frac{2\pi ju}{2N-1} \cdot \cos \frac{2\pi kv}{2N-1}, u, v \neq 0, \quad (7.19)$$

где значения $\tilde{X}(j, k)$ связаны со значениями пикселов исходного изображения так:

$$\tilde{X}(j, k) = \begin{cases} \frac{1}{4}X(j, k) & \text{при } j = 0, k = 0; \\ \frac{1}{2}X(j, k) & \text{при } j = 0, k \neq 0 \text{ или } j \neq 0, k = 0; \\ X(j, k) & \text{при } j \neq 0, k \neq 0. \end{cases} \quad (7.20)$$

Обратное преобразование вычисляется аналогичным образом:

$$\tilde{X}(j, k) = \frac{1}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} F(u, v), j, k = 0; \quad (7.21)$$

$$\tilde{X}(j, k) = \frac{2}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} F(u, v) \cdot \cos \frac{2\pi ju}{2N-1} \cdot \cos \frac{2\pi kv}{2N-1}, j, k \neq 0; \quad (7.22)$$

$$X(j, k) = \begin{cases} 4\tilde{X}(j, k), & \text{при } j = 0, k = 0; \\ 2\tilde{X}(j, k), & \text{при } j = 0, k \neq 0 \text{ или } j \neq 0, k = 0; \\ \tilde{X}(j, k), & \text{при } j \neq 0, k \neq 0. \end{cases} \quad (7.23)$$

Базисные функции нечетного косинусного преобразования являются, как следует из приведенных выше соотношений, разделимыми, так что двумерное нечетное косинусное преобразование можно осуществлять с использованием одномерных операций, выполняемых последовательно.

Следует заметить, что это преобразование уступает по эффективности **четному косинусному преобразованию**, которое имеет весьма широкое применение.

Выполнение двумерного преобразования Фурье при построении четных функциональных соотношений пристраиванием к блоку изображения его зеркальных отражений приводит к четному косинусному преобразованию [3.18, 3.38, 3.57]:

$$F(u, v) = \frac{2}{N} \cdot C(u) \cdot C(v) \cdot \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} X(j, k) \cdot \cos \frac{\pi u(2j+1)}{2N} \cdot \cos \frac{\pi v(2k+1)}{2N}, \quad (7.24)$$

где $C(n) = \begin{cases} \frac{1}{\sqrt{2}}, & n = 0, \\ 1, & n \neq 0. \end{cases}$ Обратное преобразование определяется соотношением

$$X(j, k) = \frac{2}{N} \cdot \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} C(u) \cdot C(v) \cdot F(u, v) \cdot \cos \frac{\pi u(2j+1)}{2N-1} \cdot \cos \frac{\pi v(2k+1)}{2N-1}. \quad (7.25)$$

Следует заметить, что базисные функции четного косинусного преобразования принадлежат к классу дискретных полиномов Чебышева.

На рис. 7.8–7.12 приведены псевдотрехмерные изображения структур, образуемых базисными функциями двумерного косинусного преобразования и определяемыми соотношениями:

$$\tilde{F}[i, j] = C(i) \cdot C(j) \cdot \cos \frac{\pi i(2u+1)}{2N} \cdot \cos \frac{\pi j(2v+1)}{2N},$$

где $(i, j, u, v) \in [0, N-1]; N = 8$.

Очевидно, структуры базисных функций $\tilde{F}[i, j]$ и $\tilde{F}[j, i]$ кососимметричны и могут быть построены путем замены коэффициентов u на коэффициенты v и наоборот.

Компонента $\tilde{F}[0, 0]$ характеризует усредненную постоянную составляющую яркости пикселов преобразуемого блока изображения. Компоненты $\tilde{F}[0, 1], \tilde{F}[1, 0]$ характеризуют соответственно плавное изменение яркости в блоке вдоль строки и плавное изменение яркости в блоке поперек строк. Разночастотные изменения яркостей пикселов с диагональными структурами характеризуются диагональными спектральными компонентами $\tilde{F}[1, 1], \tilde{F}[2, 2], \tilde{F}[3, 3], \tilde{F}[4, 4], \tilde{F}[5, 5], \tilde{F}[6, 6]$ и $\tilde{F}[7, 7]$.

На рис. 7.13 показаны псевдотрехмерные матрицы блока изображения, аналогичного рис. 7.6, спектра его косинусного преобразования и ошибок восстановления пикселов исходного блока.

В табл. 7.2 приведены результаты статистических исследований использования рассмотренных выше унитарных преобразований для сжатия изображений.

Как следует из этой таблицы, косинусное преобразование имеет несомненные преимущества перед преобразованиями Хаара, Уолша–Адамара и синусным.

Таблица 7.2. Результаты статистических исследований ортогональных преобразований

Степень сжатия	Среднеквадратичные (максимальные) ошибки преобразований			
	Хаара	Уолша–Адамара	Синусное	Косинусное
4,0	2,38 (7)	2,42 (7)	2,04 (5)	1,61 (4)
4,5	2,56 (7)	2,82 (8)	2,22 (7)	1,61 (5)
5,0	2,72 (7)	2,77 (9)	2,34 (6)	1,61 (6)

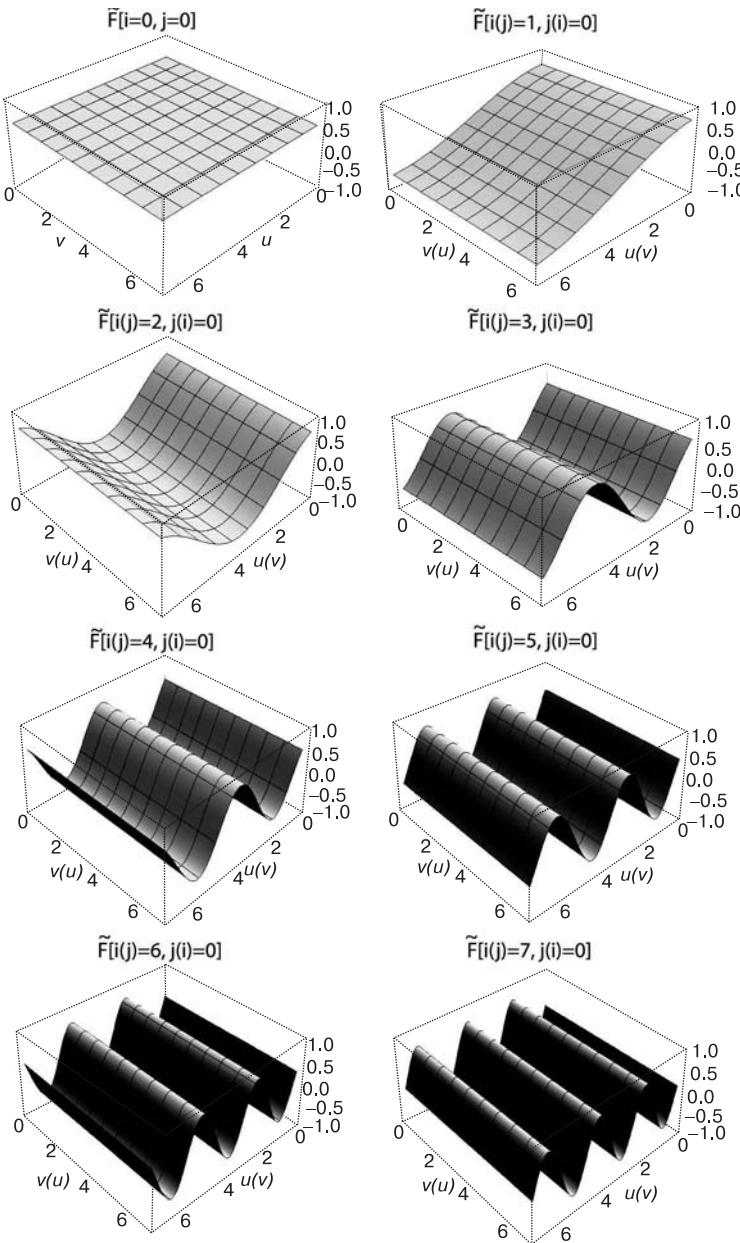


Рис. 7.8. Двумерные базисные функции $\tilde{F}[0, 0]$, $\tilde{F}[1, 0](\tilde{F}[0, 1])$, $\tilde{F}[2, 0](\tilde{F}[0, 2])$, $\tilde{F}[3, 0](\tilde{F}[0, 3])$, $\tilde{F}[4, 0](\tilde{F}[0, 4])$, $\tilde{F}[5, 0](\tilde{F}[0, 5])$, $\tilde{F}[6, 0](\tilde{F}[0, 6])$ и $\tilde{F}[7, 0](\tilde{F}[0, 7])$ косинусного преобразования

7.1.6. Преобразование Кархунена–Лоэва

Принципиальную возможность преобразовать блоки изображения в образы с попарно полностью декоррелированными элементами предоставляет так называемое

мое преобразование Кархунена–Лоэва или Хотеллинга [3.17, 3.18, 3.36, 3.38, 3.39, 3.59, 3.60].

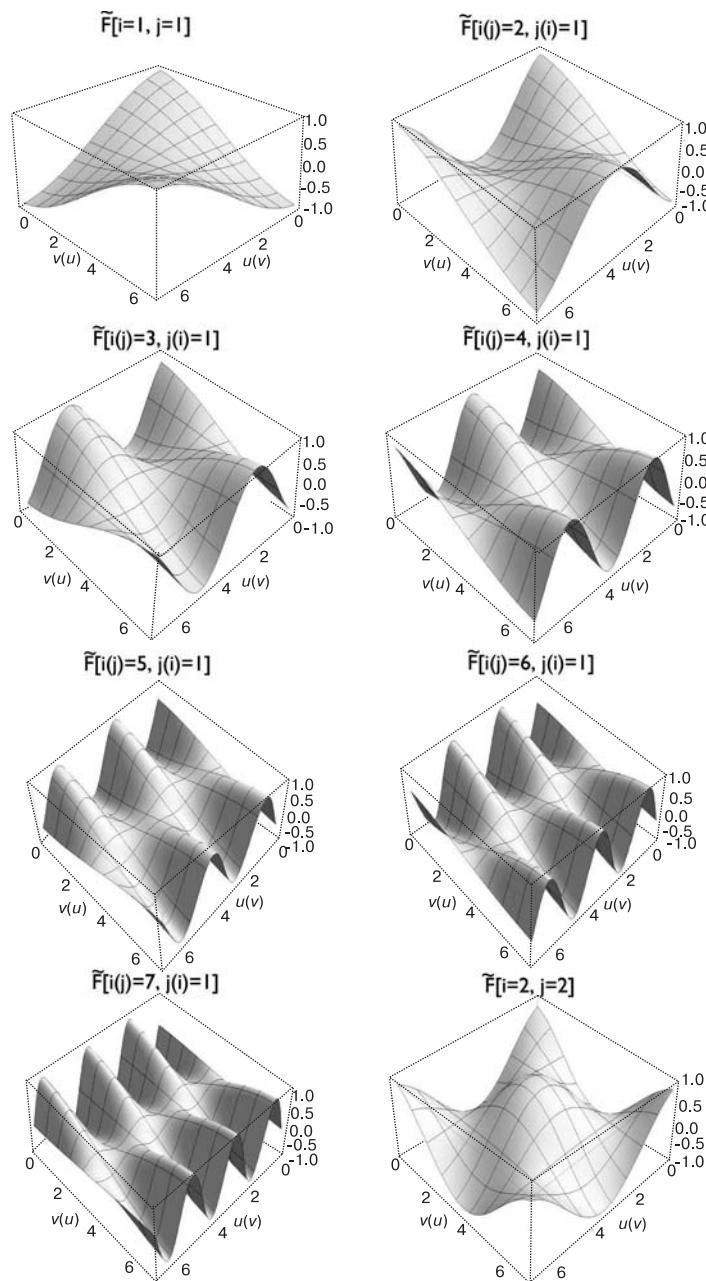


Рис. 7.9. Двумерные базисные функции $\tilde{F}[1,1]$, $\tilde{F}[2,1]$ ($\tilde{F}[1,2]$), $\tilde{F}[3,1]$ ($\tilde{F}[1,3]$), $\tilde{F}[4,1]$ ($\tilde{F}[1,4]$), $\tilde{F}[5,1]$ ($\tilde{F}[1,5]$), $\tilde{F}[6,1]$ ($\tilde{F}[1,6]$), $\tilde{F}[7,1]$ ($\tilde{F}[1,7]$) и $\tilde{F}[2,2]$ косинусного преобразования

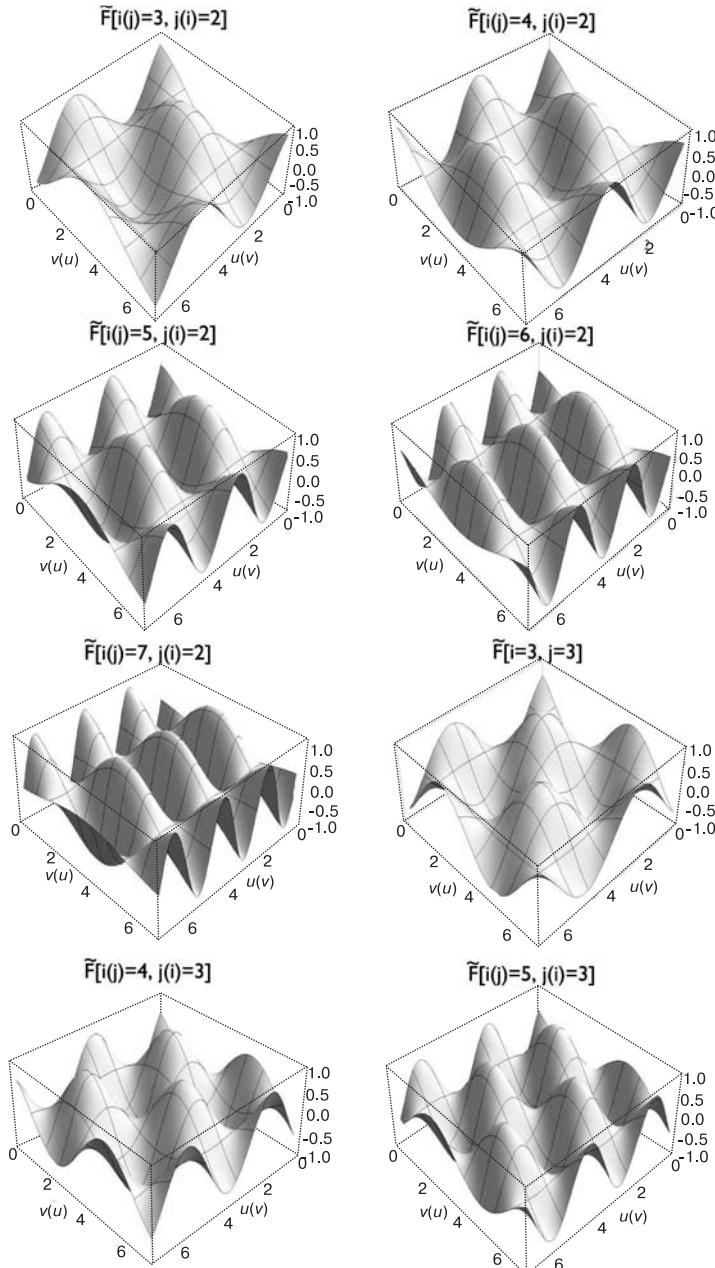


Рис. 7.10. Двумерные базисные функции $\tilde{F}[3, 2](\tilde{F}[2, 3])$, $\tilde{F}[4, 2](\tilde{F}[2, 4])$, $\tilde{F}[5, 2]$ ($\tilde{F}[2, 5]$), $\tilde{F}[6, 2](\tilde{F}[2, 6])$, $\tilde{F}[7, 2](\tilde{F}[2, 7])$, $\tilde{F}[3, 3]$, $\tilde{F}[4, 3](\tilde{F}[3, 4])$ и $\tilde{F}[5, 3]$ ($\tilde{F}[3, 5]$) косинусного преобразования

Это преобразование позволяет оценить предельные степени сжатия, достижимые за счет указанной декорреляции компонент, а также за счет допущения заданной ошибки, вносимой при усечении или квантовании компонент образа.

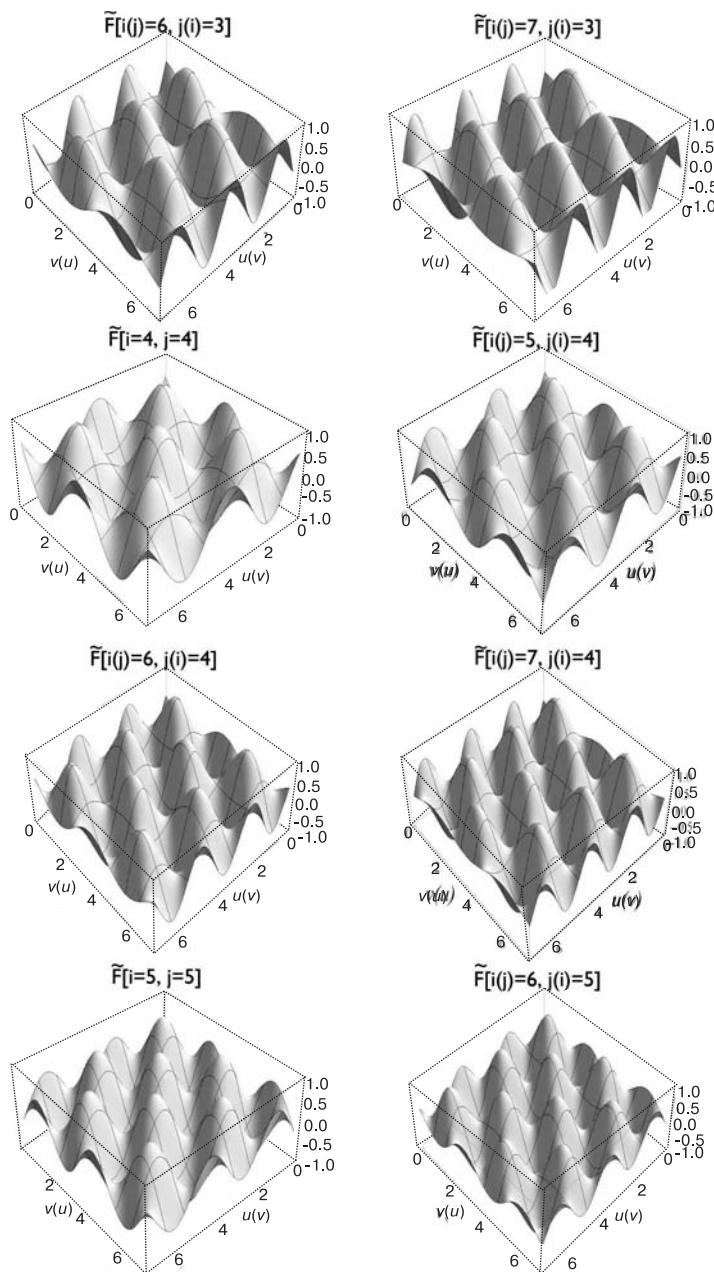


Рис. 7.11. Двумерные базисные функции $\tilde{F}[6, 3](\tilde{F}[3, 6])$, $\tilde{F}[7, 3](\tilde{F}[3, 7])$, $\tilde{F}[4, 4]$, $\tilde{F}[5, 4](\tilde{F}[4, 5])$, $\tilde{F}[6, 4](\tilde{F}[4, 6])$, $\tilde{F}[7, 4](\tilde{F}[4, 7])$, $\tilde{F}[5, 5]$ и $F[6, 5](\tilde{F}[5, 6])$ косинусного преобразования

Рассмотрим несколько подробнее принцип формирования базисных функций преобразования Кархунена–Лоэва.

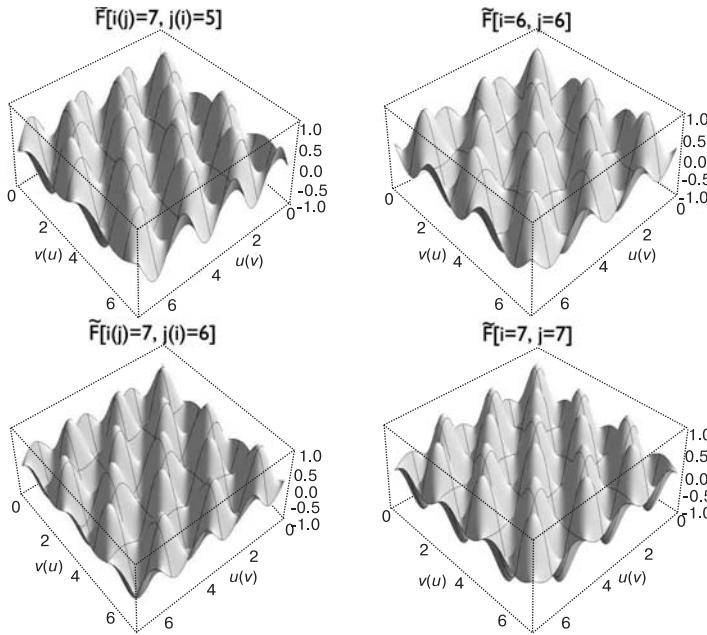


Рис. 7.12. Двумерные базисные функции $\tilde{F}[7, 5](\tilde{F}[5, 7])$, $\tilde{F}[6, 6]$, $\tilde{F}[7, 6](\tilde{F}[6, 7])$ и $\tilde{F}[7, 7]$ косинусного преобразования

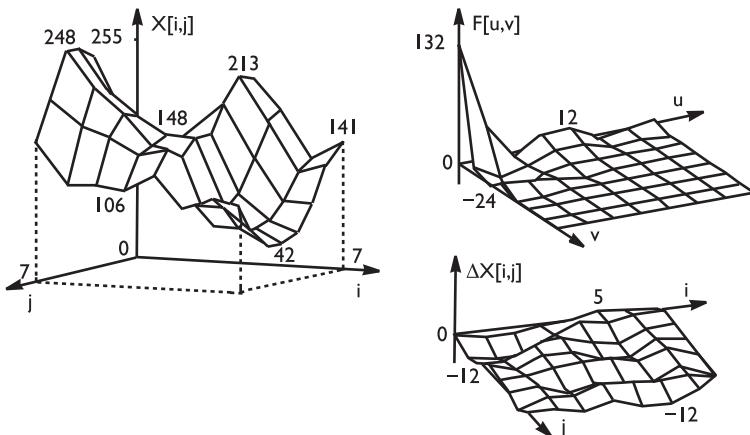


Рис. 7.13. Блок изображения 8×8 пикселов и его косинусное преобразование

Пусть имеется некоторый класс векторов $\{\mathbf{X}\}$, компоненты которых коррелируют между собой. Необходимо построить некоторое ортогональное преобразование T , переводящее данный класс векторов в класс векторов $\{\mathbf{f}\}$, компоненты которых имеют диагональную ковариационную матрицу, т. е. некоррелированы.

Пусть ковариационная матрица $\|K\{\mathbf{X}\}\|$ класса векторов $\{\mathbf{X}\}$ в терминах ее элементов определяется соотношением

$$k(i, j) = E\{(x_i - E\{x_i\}) \cdot (x_j - E\{x_j\})\},$$

где E — оператор математического ожидания.

Тогда базисом преобразования Кархунена–Лоэва является система собственных векторов $\{\mathbf{t}_m\}$, таких, что

$$\mathbf{t}_m \|K\{\mathbf{X}\}\| \mathbf{t}_m^T = \lambda_m, \mathbf{t}_m \mathbf{t}_n^T = \delta_{mn},$$

где λ_m — собственные значения диагональной ковариационной матрицы $\|K\{\mathbf{X}\}\|$ класса некоррелированных векторов $\{\mathbf{f}\}$, δ_{mn} — символ Кронекера, $\|K\{\mathbf{f}\}\| = \|\delta_{im}\delta_{jm}\lambda_m\|$.

Собственные значения λ_m имеют смысл квадратов величин соответствующих коэффициентов разложения векторов \mathbf{x} по базису \mathbf{t} , усредненному по всему классу $\{\mathbf{X}\}$.

Несколько более подробно преобразование Кархунена–Лоэва было рассмотрено в материале части II.

Непосредственное применение преобразования Кархунена–Лоэва (ПКЛ) в прикладных программах сжатия изображений не осуществляется по причине невозможности разработки быстрого алгоритма его вычисления и связанных с этим проблем.

Однако исследование статистических свойств сигналов и цифровых изображений с помощью ПКЛ облегчает поиск преобразования, наиболее близкого к оптимальному, среди преобразований, допускающих построение быстрого алгоритма вычислений.

Именно по этой причине широкое распространение получило применение дискретного косинусного преобразования, поскольку оно дает результаты, близкие к результатам преобразования Кархунена–Лоэва. Это имеет теоретическое обоснование: косинусные функции хорошо аппроксимируют собственные векторы матрицы Тейлора [3.46]:

$$\|K\{i, j\}\| = \sigma^2 \left\| \rho^{|i-j|} \right\|, \quad \rho < 1.$$

В свою очередь, эта матрица близка к ковариационным матрицам характерных телевизионных изображений.

В качестве подтверждения этого факта на рис. 7.14 приведены базисные функции Кархунена–Лоэва для изображения «Залив» (см. рис. 6.4) при $N = 8$ и базисные функции дискретного косинусного преобразования.

Использование преобразования Кархунена–Лоэва при исследованиях методов сжатия изображений привело к выводу о перспективности поиска методов расщепления ТВ изображений на отдельные составляющие, раздельная обработка которых обеспечивает более эффективное суммарное сжатие за счет преодоления нестационарности статистики.

7.1.7. Другие виды преобразований

Возможно построение других видов унитарных преобразований. Так, например, известно ортогональное наклонное или слэнт-преобразование, которое обладает базисными функциями, имеющими вид кусочно-линейных функций [3.18, 3.61–3.63]. При $N = 2$ наклонное преобразование совпадает с преобразованием Ада-

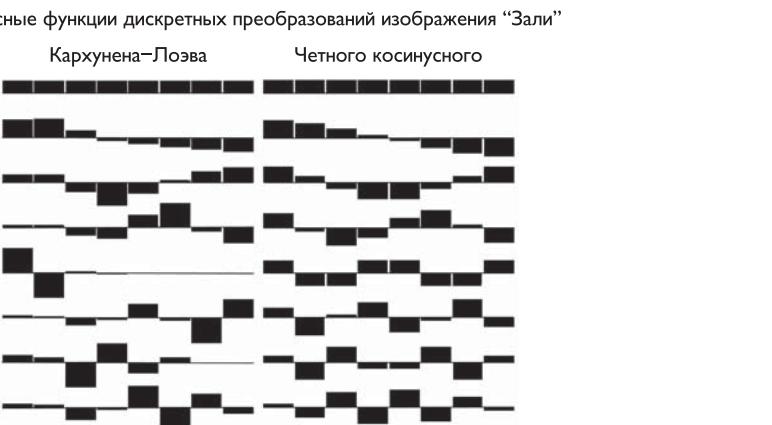


Рис. 7.14. Сравнение базисных функций преобразования Кархунена–Лоэва для конкретного изображения и базисных функций косинусного преобразования

матрица второго порядка, т. е. $S_2 = \frac{1}{\sqrt{2}} \begin{vmatrix} 1 & 1 \\ 1 & -1 \end{vmatrix}$. Матрицы наклонного преобразования могут быть записаны следующим образом:

$$S_N = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 \\ a_N & b_N & 0 & -a_N & b_N & 0 \\ 0 & 0 & I_{N/2-1} & 0 & 0 & I_{N/2-1} \\ 0 & 1 & 0 & 0 & -1 & 0 \\ -b_N & a_N & b_N & a_N & 0 & I_{N/2-1} \\ 0 & I_{N/2-1} & 0 & 0 & 0 & I_{N/2-1} \end{bmatrix} \quad (7.26)$$

где I_K — единичная матрица K -го порядка; a_N, b_N определяются рекуррентно:

$$a_2 = 1, \quad b_N = \frac{1}{\sqrt{1 + 4a_{N/2}^2}}, \quad a_N = 2b_N a_{N/2}.$$

Это преобразование при обработке изображений имеет несколько меньшую эффективность, чем дискретное косинусное преобразование. Существуют быстрые алгоритмы вычисления этого преобразования.

Унитарное преобразование может быть построено также на основе использования полиномов Лежандра

$$P_n(x) = \sum_{i=0}^n a(i, n)x^i, \quad n = 0, \dots, N-1, \quad (7.27)$$

где коэффициенты $a(i, n)$ выбираются таким образом, что

$$\sum_{i=0}^{N-1} p_n \left(\frac{2i+1}{2N} \right) \cdot P_m \left(\frac{2i+1}{2N} \right) = \delta_{mn}. \quad (7.28)$$

Например, для $N = 8$ матрица преобразования имеет вид, представленный в табл. 7.3.

Применение этого преобразования для обработки изображений показало, что оно обладает такой же эффективностью, как и дискретное косинусное преобразование. Однако быстрый алгоритм вычисления этого преобразования не был найден.

Таблица 7.3. Матрица преобразования Лежандра для $N = 8$

0,3536	0,3536	0,3536	0,3536	0,3536	0,3536	0,3536	0,3536
0,5401	0,3858	0,2315	0,0772	-0,0772	-0,02315	-0,3858	-0,5401
0,5401	0,0771	-0,2315	-0,3858	-0,3858	-0,2315	0,0771	0,5401
0,4308	-0,3077	-0,4309	-0,1646	0,1846	0,4308	0,3077	-0,4308
0,2821	-0,5238	-0,1209	0,3626	0,3626	-0,1209	-0,5238	0,2821
0,1439	-0,4922	0,3638	0,3210	-0,3210	-0,3638	0,4922	-0,1498
0,0615	-0,3077	0,5539	-0,3077	-0,3077	0,5539	-0,3077	0,0615
0,0171	-0,1195	0,3585	-0,5974	0,5974	-0,3585	0,1195	-0,0171

Рассмотрев наиболее распространенные унитарные преобразования, применяемые в цифровой обработке сигналов, следует сделать некоторые замечания, связанные с двумерностью преобразований.

При разбиении изображения на прямоугольные блоки и применении, скажем, к строкам этих блоков линейных преобразований, компоненты образов оказываются коррелированными по вертикали, что позволяет применить к столбцам из этих компонент свое преобразование.

Оно, вообще говоря, необязательно должно быть тем же, что и для обработки горизонтальных отрезков исходного изображения.

В ряде исследований использовался метод гибридного преобразования, при котором сначала производилось одномерное косинусное преобразование отрезков строк, а затем над полученными коэффициентами осуществлялась вертикальная ДИКМ. Любопытно, что при достаточно глубоком квантовании (объем передаваемой информации менее 0,5 бита/пиксель) результат получился лучше, чем при применении двумерного косинусного преобразования (ДКП).

Однако с учетом того, что в большинстве случаев алгоритмы вычислений для $N \times N$ двумерных преобразований работают быстрее, чем для линейных порядка N^2 , практическое применение находят именно двумерные преобразования.

7.2. Квантование коэффициентов преобразования

Сами по себе унитарные дискретные преобразования определены над функциями, имеющими дискретную область определения, но непрерывную область значений, т. е. значения, характеризующие уровни пикселов и величины компонент преобразования, могут быть и не целочисленными.

В то же самое время на вход кодера нужно подавать одномерную последовательность дискретных символов. Она получается при квантовании величин компонент преобразования и расстановке их в линейный ряд.

В самом общем случае процесс квантования — это осуществление нелинейного преобразования, характеристика которого имеет вид монотонной ступенчатой функции. Распределение величин шагов квантования можно оптимизировать, например, по среднеквадратичной ошибке для заданного распределения плотности вероятностей исходной функции и количества дискретных уровней (процедура Ллойда–Макса) [3.17, 3.18, 3.64, 3.65].

Исследования показывают, что компоненты унитарных преобразований телевизионных изображений имеют распределение, подчиняющееся либо нормальному, либо Лапласовому законам.

Для этих случаев при большом числе уровней преобразования близким к оптимальному является квантование с равномерной шкалой в соответствии с формулой:

$$F_q = \text{round}(F/k),$$

где F и F_q — соответственно квантуемая и квантованная величины, k — коэффициент преобразования. При восстановлении изображения компоненты унитарных преобразований умножаются на коэффициент k . Этот алгоритм имеет наибольшее распространение в цифровой обработке сигналов.

Следует отметить, что при этом минимизируется среднеквадратичное отклонение восстановленного после обратного ортогонального преобразования T -изображения от исходного, поскольку [3.17]

$$\begin{aligned} \sum_{i,j} [X_{\text{вых}}(i,j) - X_{\text{вх}}(i,j)]^2 &= \sum_{u,v} \{T[k \cdot F_q(u,v) - F(u,v)]\}^2 = \\ &= \sum_{u,v} [k \cdot F_q(u,v) - F(u,v)]^2 = \sum_{u,v} \{k \cdot \text{round}[F(u,v)/k] - F(u,v)\}^2. \end{aligned}$$

Последняя строка вытекает из равенства Парсеваля. Заметим, что минимизация среднеквадратичного отклонения значений пикселов при восстановлении изображений, строго говоря, не обеспечивает минимизации максимальной ошибки в преобразуемом блоке:

$$\max_{i,j} \{\text{abs}[X_{\text{вых}}(i,j) - X_{\text{вх}}(i,j)]\} \Rightarrow \min.$$

Эта задача может быть решена лишь методами целочисленного программирования. На рис. 7.15 приведено три варианта изображений, восстановленных после дискретного косинусного преобразования с использованием различных значений коэффициента квантования k .

Как следует из этого рисунка, с ростом k увеличивается коэффициент сжатия и существенно возрастает заметность искажений.

Возможен другой подход к выбору алгоритма квантования коэффициентов унитарных преобразований, связанный с учетом субъективной чувствительности глаза к различным пространственным частотам, учитывая, что при передаче более высокочастотных компонент допустима большая погрешность.

При этом результаты унитарного преобразования подвергаются квантованию [3.17, 3.66–3.68] в соответствии с формулой:

$$F_q(u,v) = \text{round}\{F(u,v)/[k \cdot Q(u,v)]\}, u,v = 0, \dots, N-1. \quad (7.29)$$

Для различных массивов данных, например для сигналов яркости и цветоразностных сигналов, могут применяться различные таблицы коэффициентов преобразования. Такие таблицы могут передаваться вместе с кодируемыми изображениями.

Применение общего множителя k позволяет линейно изменять таблицу квантования. Выбор и применение конкретных таблиц могут быть осуществлены по усмотрению пользователей, т. к. таблицы могут быть оптимизированы для конкретных приложений.

Так, известна функция VHS (video home system) [3.69]:

$$Q(u,v) = |a(\omega)| \cdot h(\omega), \quad (7.30)$$



Рис. 7.15. Влияние изменений коэффициента квантования на качество восстанавливаемых изображений

где

$$h(\omega) = (0,31 + 0,69\omega) \cdot \exp(-0,29\omega),$$

$$a(\omega) = \sqrt{\frac{1}{4} + \frac{1}{\pi^2} \cdot \left\{ \ln \left[\frac{2\pi\omega}{\alpha} + \sqrt{1 + \left(\frac{2\pi\omega}{\alpha} \right)^2} \right] \right\}^2},$$

$$\alpha = 11,636 \text{ deg}^{-1}, \quad \omega = \sqrt{u^2 + v^2}.$$

Группой JPEG, занимающейся стандартизацией алгоритмов цифровой обработки статических изображений, эмпирически получено несколько таблиц квантования и, в частности, данные, приведенные в табл. 7.4 для обработки сигналов яркости и цветности [3.68, 3.70]. На рис. 7.16 приведены варианты использования указанных таблиц при дискретном косинусном преобразовании изображений для двух значений коэффициента k .

Сравнивая изображения, приведенные на рис. 7.15 и рис. 7.16 при одинаковых коэффициентах сжатия, следует сказать, что использование таблиц квантования позволяет получить более высокое качество восстанавливаемых изображений при одной и той же степени сокращения избыточности информации.

Таблица 7.4. Таблицы квантования сигналов яркости и цветности

Таблица квантования сигнала яркости							
16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Таблица квантования сигнала цветности							
17	18	24	47	99	99	99	99
18	21	26	66	99	99	99	99
24	26	56	99	99	99	99	99
47	66	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99



Исходное изображение



Степень сжатия – 15



Степень сжатия – 30

Рис. 7.16. Сжатие с применением неравномерной матрицы квантования компонентов дискретного косинусного преобразования

7.3. Кодирование коэффициентов преобразования

После квантования компоненты унитарных преобразований, в частности ДКП, блоков изображения группируются в определенной последовательности, а затем кодируются хаффмановским или арифметическим кодером.

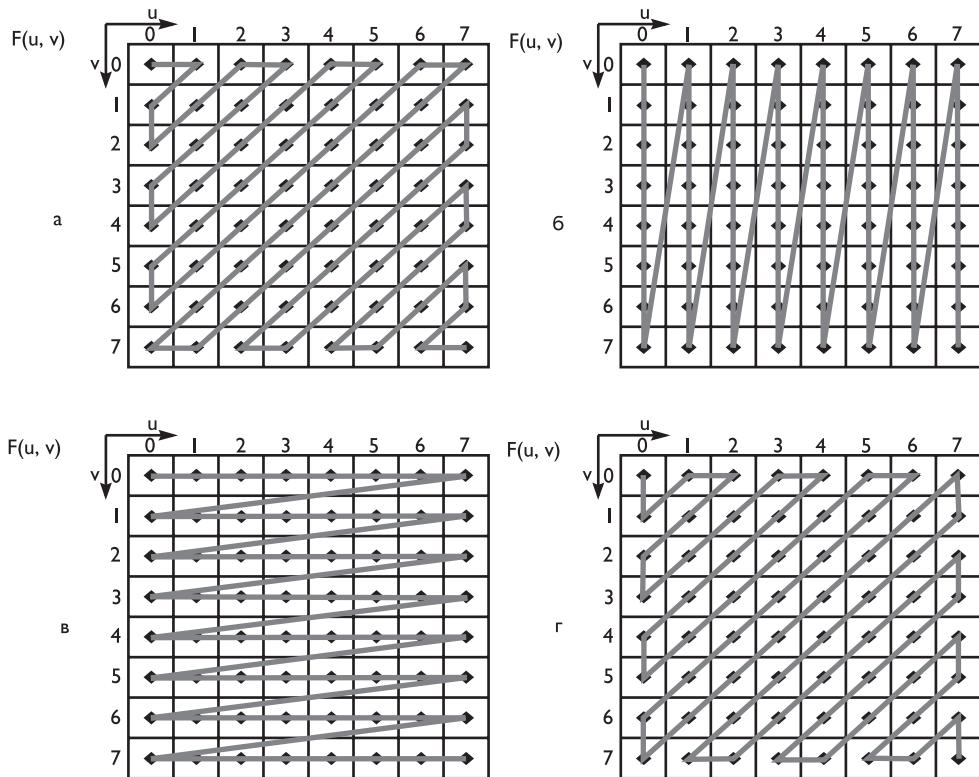


Рис. 7.17. Варианты упорядочивания коэффициентов дискретного косинусного преобразования

Метод группировки коэффициентов преобразования имеет большое значение для уменьшения объема передаваемой информации.

Одним из вариантов группировки коэффициентов ДКП является Z -упорядочивание в соответствии со схемой рис. 7.17, вверху слева [3.69].

В этом случае коэффициенты выстраиваются в последовательности возрастания пространственных частот, причем если пространственные частоты одинаковы, то предпочтение отдается коэффициентам с меньшими вертикальными частотами.

Возможны и другие варианты группирования коэффициентов:

- если в блоке изображения в основном имеется горизонтальная структура изменения значений пикселов, то более эффективно вертикальное упорядочивание (рис. 7.17, вверху справа);
- при вертикальной структуре изменения значений пикселов более удобно горизонтальное упорядочивание (рис. 7.17, внизу слева);
- иногда используется структура (рис. 7.17, внизу справа), подобная Z -упорядочиванию, но с предпочтением к коэффициентам с меньшими горизонтальными частотами.

Часто для повышения эффективности кодирования коэффициенты $F_{kl}(0,0)$ разных блоков (k, l – соответственно номера блоков по горизонтали и вертикали)

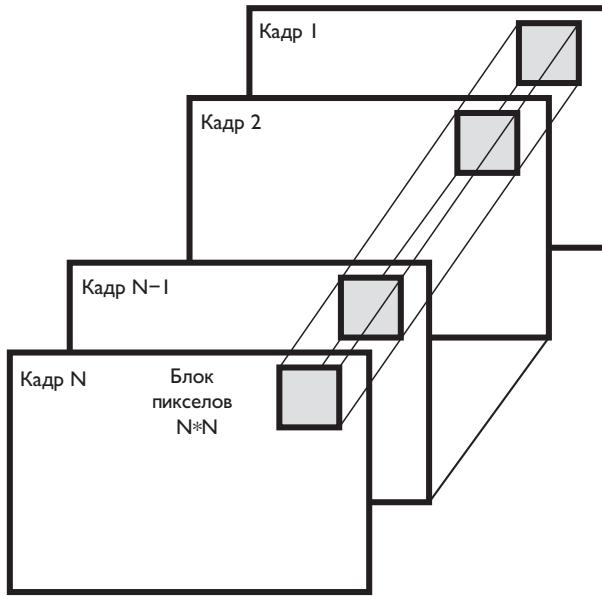


Рис. 7.18. К использованию трехмерного группового кодирования

изображения, характеризующие средние их яркости, группируют и передают отдельно, предварительно осуществляя их обработку с использованием алгоритмов предсказания.

7.4. Межкадровое или трехмерное кодирование

Исследования статистики временных изменений яркости соответствующих элементов телевизионных изображений дают результаты, в целом аналогичные тем, что получаются при изучении пространственных свойств в неподвижных изображениях, причем коэффициенты корреляции получаются зачастую даже большими, чем для соседних пикселов или соседних строк в одном кадре.

Учитывая, что все рассмотренные в данной главе способы обработки массивов данных об изображениях легко обобщаются на трехмерные объекты (рис. 7.18), можно с уверенностью предположить, что результаты применения таких трехмерных преобразований подобны соответствующим результатам для плоского случая [3.17, 3.71, 3.72].

Например, дискретное прямое и обратное косинусное преобразование для трехмерного случая может быть записано в виде:

$$\begin{aligned}
 F(u, v, w) = & \sqrt{\frac{8}{N^3}} C(u) C(v) C(w) \times \\
 & \times \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} X(i, j, k) \cos \frac{\pi u(2i+1)}{2N} \cos \frac{\pi v(2j+1)}{2N} \cos \frac{\pi w(2k+1)}{2N}, \quad (7.31)
 \end{aligned}$$

$$X(i, j, k) = \sqrt{\frac{8}{N^3}} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} \sum_{w=0}^{N-1} C(u)C(v)C(w)F(u, v, w) \times \cos \frac{\pi u(2i+1)}{2N} \cos \frac{\pi v(2j+1)}{2N} \cos \frac{\pi w(2k+1)}{2N}, \quad (7.32)$$

$$\text{где } C(n) = \begin{cases} \frac{1}{\sqrt{2}}, & n = 0, \\ 1, & n \neq 0. \end{cases}$$

Одним из важных аргументов использования трехмерных унитарных преобразований является возможность применения быстрых алгоритмов вычислений. Однако серьезным препятствием при этом является происходящая время от времени резкая смена сюжета, разрушающая межкадровые корреляционные связи.

Учитывая инерционные свойства зрения, влияние этого дефекта практически может быть существенно уменьшено путем замены некоррелированных кадров созданными с использованием экстраполяции или интерполяции. Важной проблемой при этом является разработка алгоритмов оценки резких изменений структуры кадра.

ГЛАВА 8

ДРУГИЕ МЕТОДЫ КОДИРОВАНИЯ ИЗОБРАЖЕНИЙ

Хотя групповое кодирование практически обеспечивает декорреляцию информации в преобразуемых блоках изображения, однако и после преобразований сохраняется избыточность информации в связи с тем, что имеется корреляция между блоками.

8.1. Кодирование с использованием гибридных преобразований

Указанные выше недостатки, как уже было рассмотрено выше, уменьшаются при использовании гибридных преобразований [3.17, 3.73–3.76].

Например, при внутрикадровом гибридном преобразовании изображение разбивается на вертикальные полосы, по N пикселов вдоль строк внутри каждой полосы. Отрезки из N горизонтально расположенных пикселов подвергаются линейному дискретному косинусному преобразованию, после чего коэффициенты преобразования кодируются с предсказанием по вертикали.

Эта процедура дает выигрыш при кодировании коэффициентов низких пространственных частот, но более высокочастотные коэффициенты плохо коррелируют между собой, и их лучше передавать независимо.

Оценки таких преобразований показывают, что гибридное преобразование дает приблизительно такие же среднеквадратичные ошибки, что и двумерное дискретное косинусное преобразование.

Гибридное преобразование широко применяется при кодировании движущихся изображений, когда используется двумерное косинусное преобразование в кадре и предсказание между кадрами.

В квазистационарных изображениях вектора движения малы независимо от числа пространственных деталей. Однако часто шум камеры и ее нестабильность вносят различия между кадрами даже в отсутствие движения.

В зависимости от используемого алгоритма и детальности изображения адаптивное гибридное кодирование позволяет уменьшить требуемую скорость передачи в 2–10 раз.

Дополнительно она может быть уменьшена при использовании **компенсации движения**.

8.2. Другие методы блочного и интерполяционного кодирования

Одним из методов блочного кодирования является **векторное кодирование**, при котором весь набор блоков (векторов) изображения (как одного кадра, так и последовательности кадров) приводится к некоторому конечному набору [3.17, 3.76–3.81].

При этом вместо непосредственной передачи кодовых последовательностей, характеризующих значения пикселов в каждом блоке, передается лишь номер блока. Количество возможных векторов (блоков изображения) определяется при этом пропускной способностью канала.

Основным преимуществом векторного квантования является простая структура приемника, который имеет лишь кодовую таблицу векторов, а воспроизведение изображения осуществляется путем последовательной расстановки этих векторов согласно поступающей из канала связи информации.

При адаптивном векторном квантовании, кроме того, передается также информация об эталонных векторах.

Существенными недостатками векторного кодирования является, с одной стороны, сложность кодера, а с другой — плохая аппроксимация изображений, отличающихся от тех эталонных, по которым составлялась кодовая таблица.

Следует отметить, что существуют специальные алгоритмы формирования кодовых таблиц с использованием эталонных изображений. Иногда для ускорения процесса векторного кодирования и увеличения скорости передачи блоки изображения разбиваются на классы, для каждого из которых составляется своя кодовая таблица.

Алгоритмы векторного кодирования обеспечивают формирование специального набора блоков изображения (кодовой книги), составленного по принципу наилучшего приближения к блокам исходного изображения (например, минимизацией среднеквадратичного отклонения).

Содержание кодовой книги и номера (кодовые слова) ее элементов, на которые были заменены блоки исходного изображения, упорядоченные в порядке сканирования этих блоков (например, слева направо, сверху вниз), передаются декодеру, при этом количество элементов этой книги должно быть меньше количества блоков исходного изображения.

Если рассматривать различные блоки размера $n \times m$ пикселов как точки-вектора в $(n \times m)$ -мерном пространстве изображений, то процесс замены их на элементы кодовой книги выглядит как квантование (вообще говоря, неравномерное) такого пространства на области сложной формы, окружающие точки, соответствующие изображениям — элементам кодовой книги (отчего и происходит название метода).

Пусть исходное изображение имеет размерность $N \times M$ пикселов. При прямоугольном разбиении получается $\frac{N}{n} \cdot \frac{M}{m}$ блоков размерности $n \times m$. Если кодовая книга будет содержать k элементов, то на ее передачу потребуется $k \times n \times m \times l$ битов, а последовательность кодовых слов займет еще $\frac{N}{n} \cdot \frac{M}{m} \cdot \log_2 k$ битов, где l — разрядность оцифровки пикселов исходного изображения, в большинстве случаев равная 8.

При этом достигается степень сжатия, равная

$$\text{comp} = \frac{N \cdot M \cdot l}{k \cdot n \cdot m \cdot l / c_1 + \frac{N}{n} \cdot \frac{M}{m} \cdot \log_2 k} = \frac{1}{k / \left(\frac{N}{n} \cdot \frac{M}{m} \cdot c_1 \right) + \frac{\log_2 k}{l} \cdot \frac{1}{n \cdot m}}, \quad (8.1)$$

где c_1 — степень сжатия кодовой книги, которая может быть упакована с помощью каких-либо методов компрессии без потерь информации, причем реально достижимое c_1 составляет около 2.

Очевидно, что сжатие с помощью векторного квантования предполагает обязательную потерю части информации, т. е. внесение искажений в исходное изображение. В процессе оптимизации процесса векторного квантования эти искажения минимизируются (при заданной степени сжатия, или, наоборот, достигается максимальное сжатие при заданном уровне допустимых искажений) путем:

- выбора оптимального размера блока $n \times m$;
- выбора оптимального размера кодовой книги k ;
- составления оптимального набора изображений — элементов кодовой книги.

Вообще говоря, существует и может быть найден глобальный оптимум одновременно по всем трем перечисленным параметрам. Положение этого оптимума определяется распределением $\frac{N}{n} \cdot \frac{M}{m}$ точек-блоков из исходного изображения в $(n \times m)$ -мерных пространствах изображения при различных $n \times m$. При равномерном распределении, что соответствует некоррелированному шуму в исходном изображении, оптимальным оказывается равномерное квантование каждого пикселя исходного изображения, т. е.

$$n \cdot m = 1; \quad k = 2^l / (2d); \quad K_i = 2 \cdot i \cdot d;$$

где d — максимальная ошибка квантования; 2^l — количество дискретных уровней в исходном изображении, K_i — i -й уровень воспроизведимых дискретных значений яркости пикселов.

В изображении с коррелированными значениями яркости пикселов точки-блоки распределены в пространстве изображений неравномерно, и центром «сгустков» таких точек, т. е. точек, в которых плотность вероятности нахождения блока из исходного изображения имеет локальный максимум, могут составить набор элементов кодовой книги, так что усредненная по исходному изображению ошибка, вносимая векторным квантованием, окажется меньшей, чем, например, при равномерном квантовании пространства изображений.

Следует заметить, что использованная выше в целях изложения сути метода векторного квантования модель представления блоков изображения в виде точек в многомерном пространстве изображений с вычислительной точки зрения слишком сложна, т. к. при этом придется работать с массивами данных объемом в $(2^l)^{n \cdot m}$ битов, что составляет, например, для блоков 4×4 пикселя и $l = 8$: $2^{8 \cdot 4 \cdot 4} = 2^{128}$ — неприемлемую величину.

В описываемых в литературе реальных исследовательских схемах [3.76, 3.82, 3.83], как правило, объем кодовой книги и размерность блоков выбирают, исходя из ожидаемой степени сжатия (для $\text{comp}=10$ обычно выбирают $n \times m=16$ (блоки 4×4); для $\text{comp} = 30$ — $n \times m = 64$ (блоки 8×8); количество элементов кодовой книги k обычно достигает нескольких тысяч). Основными задачами оптимизации кодирования при этом оказываются:

- составление оптимального набора изображений элементов кодовой книги;
- поиск элемента кодовой книги, наилучшим образом соответствующего очередному кодируемому блоку из исходного изображения.

Алгоритм LBG

В 1980 году Linde, Buzo и Gray предложили алгоритм составления кодовой книги, дающий наилучшие до сего дня результаты [3.17, 3.78]. Этот алгоритм, имеющийся LBG, состоит из двух процедур, последовательно-итерационно применяемых к системе — исходное изображение плюс кодовая книга, полученная в результате применения какого-либо иного предварительного алгоритма или составленная наугад из случайных изображений (от качества этой предварительной кодовой книги зависит лишь число итераций, которые необходимо будет произвести, прежде чем кодовая книга перестанет изменяться при каждой следующей итерации).

Алгоритм предусматривает реализацию последовательности итерационных процедур.

1. Процедура присваивания блокам исходного изображения соответствия тому или иному элементу текущей кодовой книги. При этом множество блоков, составляющих исходное изображение, разбивается на непересекающиеся подмножества, состоящие из блоков исходного изображения (необязательно в одном и том же количестве), близких друг к другу и к некоторому элементу кодовой книги, в соответствии с некоторой мерой (например, среднеквадратичным или максимальным отклонением яркости по блоку). При этом число непересекающихся подмножеств должно быть равно количеству элементов текущей кодовой книги (оно может быть меньше конечной величины k).

2. Процедура «улучшения» текущей кодовой книги — замена ее элементов путем построения для каждого из полученных в предыдущей процедуре подмножеств блоков такого элемента кодовой книги, который бы минимизировал некоторое среднее отклонение от него всех блоков подмножества в смысле выбранной меры. При этом блоков — новых элементов кодовой книги, соответствующих каждому подмножеству, может быть и больше одного. Тогда происходит «расщепление» элемента текущей кодовой книги, так что ее объем может быть постепенно доведен до заранее выбранного значения k .

Если «расщепления» не происходит, то новый блок — элемент кодовой книги находится, например, путем попиксельного арифметического усреднения яркостей всех блоков подмножеств при использовании в качестве меры среднеквадратичного отклонения или же путем определения медианы значений яркостей соответствующих пикселов по всему подмножеству блоков при использовании в качестве меры максимального отклонения.

Расщепление представляет собой задачу нахождения оптимальной кодовой книги малого объема, где в качестве исходного изображения выступает рассматриваемое подмножество блоков. Однако при таком решении этой задачи сложность общего алгоритма оказывается неприемлемо высокой.

Поэтому обычно нахождение оптимальной кодовой книги осуществляется упрощенными способами — от случайного разбиения подмножеств на ряд более мелких, до выбора в качестве новых элементов кодовой книги векторов собственной



системы ковариационной матрицы, рассчитанной для подмножества блоков, имеющих наибольшие соответствующие собственные значения (метод Кархунена–Лоева) и т. п. [3.38, 3.82, 3.83].

Решение вопроса о том, осуществлять ли «расщепление» в данном подмножестве или нет, может быть принято из различных соображений, например исходя из анализа величины дисперсии блоков в подмножестве.

Алгоритм LBG весьма гибок, и предполагает дальнейшую оптимизацию в конкретном применении.

Недостатком алгоритма LBG является то, что большой объем необходимых для его исполнения вычислительных операций не поддается ни сокращению, ни распараллеливанию: на каждой итерации в первой процедуре приходится перебирать все блоки исходного изображения, сравнивая каждый из них со всеми элементами текущей кодовой книги, что не позволяет непосредственно применять LBG в системах реального времени.

Данный недостаток в значительной мере компенсируется тем, что в качестве предварительной кодовой книги можно использовать результат применения иных алгоритмов (например, алгоритм Эквитца), используя LBG лишь для окончательной оптимизации кодовой книги на малом количестве итераций (возможно) без «расщеплений».

Алгоритм Эквитца

Данный алгоритм составления кодовой книги [3.79, 3.82] основан на последовательном сокращении (в противоположность LBG) числа элементов исходной кодовой книги.

При этом в качестве исходной кодовой книги используется само исходное изображение. Сокращение осуществляется путем поиска пары элементов текущей кодовой книги, менее всего отличающихся друг от друга в смысле некоторой меры, с последующей заменой такой пары блоком, полученным попиксельным арифметическим усреднением значений яркости в этой паре.

Эту операцию осуществляют до тех пор, пока размер кодовой книги не станет равен требуемому значению.

Алгоритм Эквитца более благоприятен с вычислительной точки зрения, чем LBG, но уступает ему по качеству получаемой кодовой книги.

Намного меньших объемов вычислительных операций требуют алгоритмы, основанные на принятии в качестве элемента кодовой книги собственных векторов ковариационных матриц, рассчитанных для последовательностей блоков исходного изображения. При этом получаются кодовые книги, как правило, нуждающиеся в уточнении методом LBG [3.83].

Среднеквадратичные ошибки в приводимых в литературе экспериментальных данных составляют -24 дБ при степенях сжатия порядка 10 [3.76, 3.81].

В связи с проблемой поиска элемента кодовой книги, наилучшим образом сходного с блоком исходного изображения, следует заметить, что в алгоритмах, использующих метод Кархунена–Лоева, параллельно с созданием самой кодовой книги происходит построение системы многомерных плоскостей (гиперплоскостей), делящих пространство изображений на ячейки, так что поиск соответствия блока исходного изображения элементу кодовой книги из полного перебора всей кодовой книги превращается в бинарный поиск соответствующей ячейки.

Поиск осуществляется путем вычисления знаков скалярных произведений векторов-нормалей, опущенных из точки-блока на упомянутые выше гиперплоскости, и характеристических векторов этих гиперплоскостей [3.83]. При этом количество операций на блок снижается со значения порядка k до $\log_2 k$.

Еще раз заметим, что даже самые эффективные по скорости обработки варианты векторного квантования намного сложнее алгоритмов сжатия видеинформации, использующих ортогональные преобразования или тем более предсказания, что затрудняет их применение в кодерах, работающих в реальном времени. Восстановление же закодированного с помощью векторного квантования изображения, наоборот, осуществляется намного проще, чем в любых других методах группового кодирования, и требует (в простейшем варианте) лишь одного косвенного обращения к памяти на пиксел или одного умножения с накоплением на пиксел, если перед подачей на кодер векторного квантования изображение было отнормировано по яркости и контрасту (с передачей соответствующих коэффициентов).

Очевидным фактором, лимитирующим общую достижимую степень сжатия при векторном квантовании, является необходимость передавать изображения — элементы кодовой книги, причем без потери информации при возможной упаковке. Таким образом, большей эффективности векторного квантования можно ожидать в тех случаях, когда одна и та же кодовая книга может считаться оптимальной для аппроксимации как можно большего количества блоков.

Вместе с тем существует теоретическая и, как оказалось, практическая возможность обойти проблему передачи кодовых книг при использовании так называемого фрактального сжатия (см. раздел 8.3).

Еще одним специфическим вариантом блочного кодирования является использование **неунитарного интерполяционного преобразования** [3.17].

При этом изображение разбивается на блоки, скажем 4×4 пикселов $X(i, j)$, $i, j = 0, \dots, 3$. Алгоритм передачи для каждого блока определяется, например, следующим образом:

- значение $X(0, 0)$ передается непосредственно,
- величина $X(3, 3)$ передается путем ее предсказания по переданному значению $\hat{X}(0, 0)$,
- величины $X(0, 3)$ и $X(3, 0)$ предсказываются по значению $[\hat{X}(0, 0) + \hat{X}(3, 3)]/2$.

Значения остальных пикселов передаются с предсказанием по четырем пикселям. Такой алгоритм существенно более прост в вычислениях, чем рассмотренные ранее унитарные преобразования.

Несколько слов следует сказать об **интерполяционном и экстраполяционном кодировании** [3.17]. При этом в изображении выбирается набор базовых пикселов, для передачи которых используются различные варианты рассмотренных методов кодирования. Остальные пиксели восстанавливаются при помощи интерполяции/экстраполяции. В методах с фиксированными позициями базовых пикселов передаются всегда одни и те же значения пикселов.

В адаптивных методах порядок передачи меняется: учитывается тип локальной корреляции с тем, чтобы интерполяция проводилась либо по вертикали, либо по горизонтали, либо по времени и т. д.

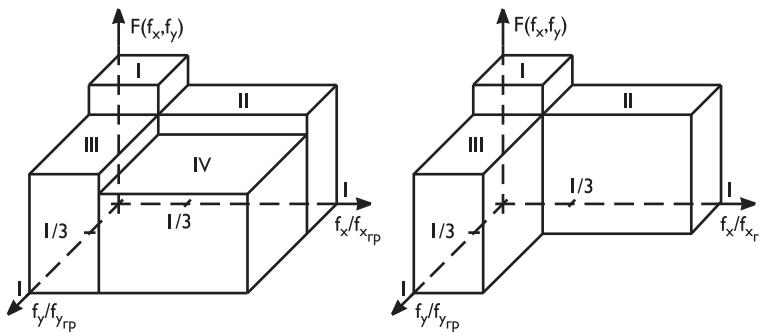


Рис. 8.1. Разбиение пространственно-частотной области изображений

В системах передачи малоподвижных изображений интерполяция обеспечивает устранение передачи промежуточных кадров. Это, естественно, ухудшает временное разрешение, но весьма эффективно при переполнении буфера передачи.

Особенно эффективной следует считать интерполяцию, адаптивную к движению объектов в изображении [3.17, 3.84]: интерполяции подвергается пиксель $(k-1)$ -го кадра с координатами $(x + \Delta x/2, y + \Delta y/2)$ по значениям пикселов $(k-2)$ -го кадра с координатами (x, y) и k -го кадра с координатами $(x + \Delta x, y + \Delta y)$.

При обзоре различных методов кодирования следует упомянуть также о методе, имеющем название «пирамида Лапласа» [3.17, 3.85–3.87].

В этом случае изображение расщепляется при помощи полосовых фильтров пространственных частот так, что границы (одни и те же по вертикали и по горизонтали) составляют гармонический ряд:

$$f_{\text{гр}}; \quad f_{\text{гр}}/k; \quad f_{\text{гр}}/k^2; \dots$$

Обработка таких изображений-диапазонов состоит в передискретизации с прореживанием в k раз по вертикали и по горизонтали, а суммарное сжатие получается благодаря тому, что спектральная мощность сигнала изображения падает с ростом частоты, а также из-за меньшей субъективной заметности ошибок квантования более высокочастотных составляющих изображения.

Возможно использование различных модификаций этого метода. В частности, один из вариантов предполагает разбиение пространственно-частотной области на четыре части, как показано на рис. 8.1, слева.

Исследования показывают, что, если выбрать $f_{x1} = \frac{1}{3}f_x \text{гр}$ и $f_{y1} = \frac{1}{3}f_y \text{гр}$ и исключить область IV, как показано на рис. 8.1, справа, искажения телевизионных изображений малозаметны, в то время как эффективное напряжение помехи в изображении уменьшается почти в два раза.

Это свидетельствует о том, что несущие смысловую нагрузку детали изображения практически всегда имеют форму, протяженную хотя бы в одном направлении раstra.

Поскольку изображение, характеризуемое областью I, имеет спектр, ограниченный пространственными частотами, в три раза меньшими граничных частот исходного изображения, оно определяется прореженными в три раза по горизонтали и вертикали отсчетами, следовательно, объем информации в 9 раз ниже объема информации об исходном изображении. На рис. 8.2 приведено исходное



Рис. 8.2. Прореживание спектральной области I по горизонтали и вертикали изображение «Залив» и прореженное изображение области I.

Информация об областях II и III может обрабатываться отдельно, что имеет смысл, так как в них концентрируются контуры деталей, имеющих по преимуществу вертикальную (низкочастотные по вертикали и высокочастотные по горизонтали составляющие спектра, область II) и горизонтальную (низкочастотные по горизонтали и высокочастотные по вертикали составляющие спектра, область III) протяженности.

Первая стадия обработки заключается в возможности передискретизации с прореживанием по низкочастотному направлению, как показано на рис. 8.3. При этом объем информации каждого из этих изображений сокращается в три раза по сравнению с исходными.

Дальнейшее уменьшение объема информации об областях II и III без устранения избыточности возможно путем свертки их спектров $g_2(x, y)$ и $g_3(x, y)$, приведенных на рис. 8.3, по высокочастотным направлениям с комплексными величинами:

$$g'_2(x, y) = g_2(x, y) \cdot \exp \left(-2\pi i \frac{f_{x\text{grp}} - f_{x1}}{2} x \right);$$

$$g'_3(x, y) = g_3(x, y) \cdot \exp \left(-2\pi i \frac{f_{y\text{grp}} - f_{y1}}{2} y \right).$$

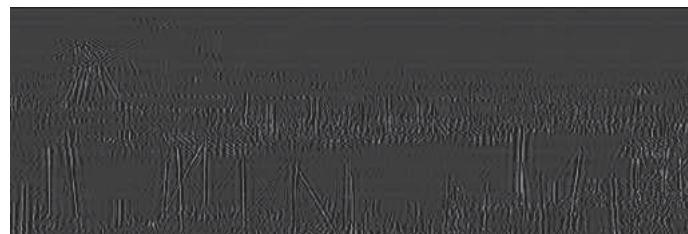
При этом выделяются четыре изображения:

$$\text{Re}[g'_2(x, y)]; \quad \text{Im}[g'_2(x, y)]; \quad \text{Re}[g'_3(x, y)] \quad \text{и} \quad \text{Im}[g'_3(x, y)]$$

и прореживается каждое из них в три раза, как показано на рис. 8.4.

Каждое из этих изображений имеет объем информации в 9 раз меньший, чем исходное изображение.

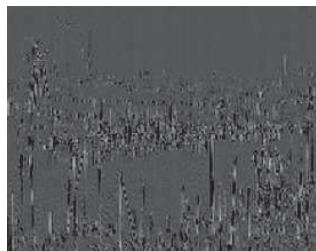
Последующая обработка каждого из пяти выделенных элементов изображений — рис. 8.2 — область I и рис. 8.4 — может осуществляться различными методами.



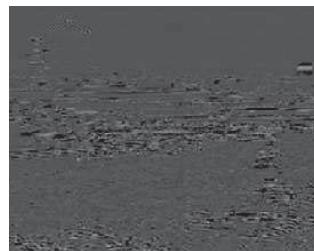
Спектральная область II,
прореженная по вертикали

Спектральная область III,
прореженная по горизонтали

Рис. 8.3. Прореживание спектральных областей II и III



Область II



Область III

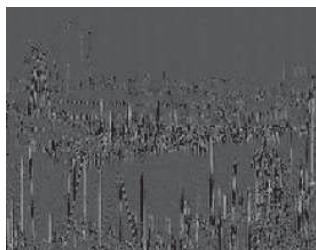


Рис. 8.4. Преобразованные составляющие изображения, приведенные на рис. 8.3

8.3. Фрактальные методы кодирования изображений

Фрактальные методы сжатия можно рассматривать как модификацию векторного квантования, при которой в качестве элементов кодовой книги используют блоки, вырезанные всевозможными способами из самого исходного изображения.

Фрактальные методы сжатия считаются одними из наиболее эффективных методов по степени сжатия изображения [3.88–3.97].

Рассмотрим, например, следующее преобразование изображения: уменьшение линейных размеров исходного изображения (рис. 8.5, слева) в два раза (рис. 8.5, в центре) и тройное его копирование (рис. 8.5, справа).

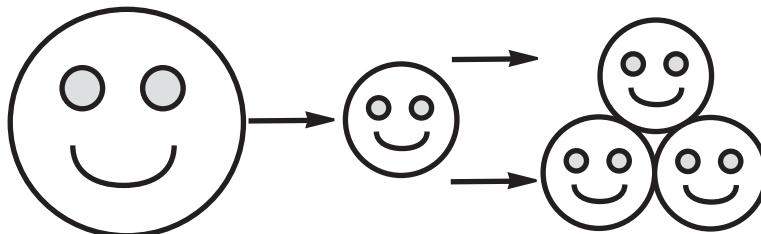


Рис. 8.5. Способ фрактального преобразования изображения

Если этот процесс повторять итеративно несколько раз, то возникающие из различных исходных изображений рисунки станут похожими друг на друга (рис. 8.6).

При достаточно большом количестве итераций эти рисунки перестанут различаться. Конечное изображение обладает рядом интересных свойств:

- во-первых, оно не зависит от начального изображения, поскольку при достаточно большом числе итераций исходное изображение уменьшится до точки;
- во-вторых, оно определяется исключительно процедурой преобразования;
- в-третьих, дальнейшие преобразования будут преобразовывать его в самого себя;
- в-четвертых, оно может иметь сколь угодно мелкие детали.

Изображения такого типа называются **аттракторами**, а преобразования — **фрактальными**.

Разумеется, процедуры преобразования могут быть и другие. Единственное ограничение — это требование сходимости изображения в указанном выше смысле. В противном случае если две разные точки исходного изображения в результате последовательности преобразований не сойдутся в одну, то конечное изображение будет зависеть от исходного и не будет аттрактором.

На практике достаточно большое количество преобразований можно описать с помощью матричного уравнения, определяющего линейное преобразование координат x и y :

$$W_i \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} a_i & b_i \\ c_i & d_i \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} e_i \\ f_i \end{bmatrix}, \quad (8.2)$$

где $a_i, b_i, c_i, d_i, e_i, f_i$ — параметры преобразования.

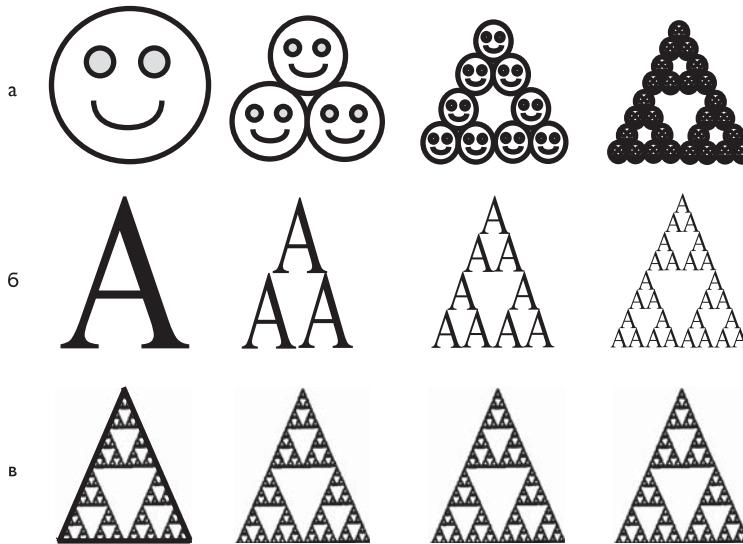


Рис. 8.6. Фрактальное преобразование различных изображений

Такие преобразования называются **аффинными преобразованиями** плоскости и могут переворачивать, вытягивать, вращать и масштабировать исходное изображение [3.89, 3.94].

На рис. 8.7 показано несколько процедур преобразования и их аттракторов:

- в верхнем ряду — описанное выше преобразование;
- в среднем ряду — аналогичное преобразование при дополнительном повороте верхней детали;
- в нижнем ряду — более сложное преобразование с поворотом деталей на разные углы и их масштабированием.

Последний случай наиболее интересен и имеет название «папоротник Барнсли» (Barnsley). Этот внешне сложный рисунок получен за счет четырех аффинных преобразований, каждое из которых имеет шесть параметров (см. приведенное матричное уравнение).

Такие самоподобные рисунки называются **фракталами**. Если перемножить число преобразований (4), число параметров (6) и число битовых под хранение каждого из параметров (например, 32) то получим $4 \times 6 \times 32 = 768$ битов — столько битов необходимо для хранения способа получения этого изображения.

В то же время приведенное штриховое изображение (1 бит/пикセル) папоротника имеет разрешение 256×256 пикселов. Для прямого хранения такого изображения необходимо 65 536 битов. То есть рассматриваемая схема позволяет «сжать» изображение примерно в 85 раз. Обратим внимание на некоторую условность такого определения коэффициента сжатия. Дело в том, что для хранения алгоритма преобразования требуется определенное, заранее известное, количество битов, но этот алгоритм позволяет создать изображение любого размера с достаточно мелкими деталями, для хранения которого требуется другое (воз-

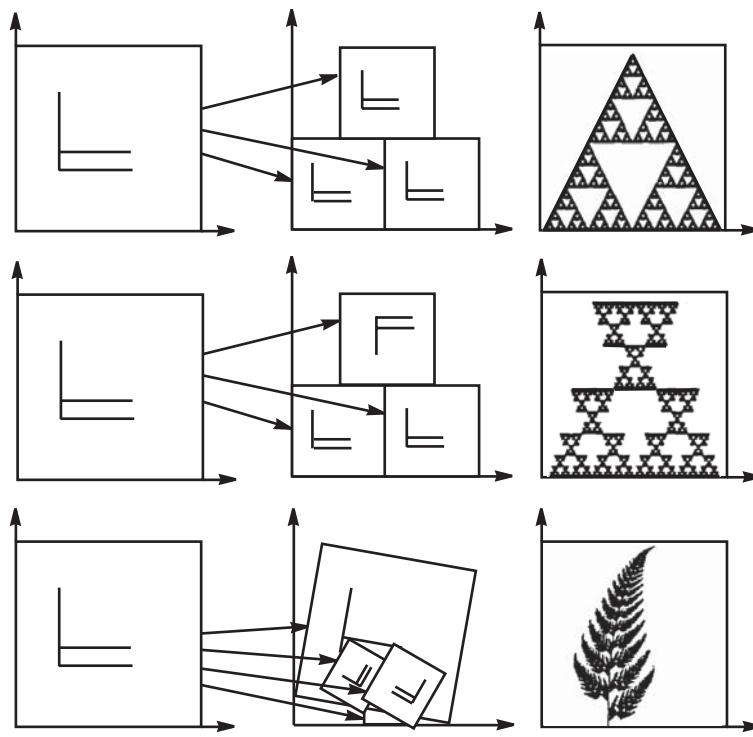


Рис. 8.7. Различные фрактальные преобразования изображений

можно, существенно большее) количество битов. Соответственно размеру изображения будет меняться и коэффициент сжатия.

Телевизионные изображения, как правило, нельзя считать аттракторами, однако они практически всегда содержат подобные детали. На рис. 8.8, слева, приведено изображение, достаточно часто встречающееся в публикациях по фрактальному сжатию, а на этом же рисунке, справа, — это же изображение, но с выделенными подробными деталями, например: участок контура плеча, контур отражения шляпы в зеркале и др.

Возникает вопрос: можно ли для любой детали изображения подобрать деталь, которая после некоторых преобразований станет достаточно похожей на исходную? Строгое математическое доказательство этого отсутствует, однако практика показывает, что это возможно почти во всех случаях.

Такие преобразования для полутоновых монохромных изображений можно формально описать следующим образом.

Пусть яркость пикселов изображения z задана некоторой функцией

$$z = f(x, y),$$

где x, y — координаты пикселов. Положим, z может иметь 256 фиксированных уровней. Само аффинное преобразование i -го блока полутонового изображения будет выглядеть следующим образом:



Рис. 8.8. Иллюстрация к подобию деталей в одном изображении

$$w \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} a_i & b_i & 0 \\ c_i & d_i & 0 \\ 0 & 0 & s_i \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} e_i \\ f_i \\ o_i \end{bmatrix}, \quad (8.3)$$

где $a_i, b_i, c_i, d_i, e_i, f_i$ — параметры аффинного преобразования, s_i, o_i — коэффициенты преобразования контраста и яркости блока.

Теперь необходимо разбить исходное изображение на блоки (назовем их доменами), для которых будут подбираться подобные блоки (ранговые области).

Весьма часто используется один из самых простых способов разбиения и преобразования исходного изображения. Он заключается в следующем. Исходное изображение разбивается на квадратные домены, например, 4×4 пикселя. Затем для каждого домена подбирается ранговая область следующим образом.

Берется квадрат, например, в два раза большего размера 8×8 , и путем усреднения по четырем соседним пикселям из него формируется квадрат 4×4 .

Этот квадрат ориентируется в одном из восьми различных положений: четыре положения определяются его поворотом на 90 градусов и еще четыре — поворотом на 90 градусов его зеркального отражения.

После преобразования контраста (путем умножения на некоторое число значений яркости каждого пикселя) и яркости (путем увеличения/уменьшения яркости каждого пикселя на некоторую величину) каждый из полученных квадратов сравнивается с доменом.

При минимальном различии в среднеквадратичном смысле квадрат считается ранговой областью рассматриваемого домена.

Вычисление оптимальных коэффициентов преобразования контраста и яркости квадратов путем минимизации среднеквадратичной разности яркостей пикселов домена и кандидата в ранговую область может быть произведено по следующим формулам:

$$s_i = \left[n \sum_{k=1}^n z_{pk} \cdot z_{dk} - \left(\sum_{k=1}^n z_{pk} \right) \cdot \left(\sum_{k=1}^n z_{dk} \right) \right] \Bigg/ \left[n \sum_{k=1}^n z_{pk}^2 - \left(\sum_{k=1}^n z_{pk} \right)^2 \right], \quad (8.4)$$

$$o_i = \left[\sum_{k=1}^n z_{dk} - s_i \sum_{k=1}^n z_{pk} \right] \Bigg/ n; \quad (8.5)$$

если $n \sum_{k=1}^n z_{pk}^2 - \left(\sum_{k=1}^n z_{pk} \right)^2 = 0$, то $s_i = 0$ и $o_i = \left(\sum_{k=1}^n z_{dk} \right) / n$; z_{pk} , z_{dk} — яркости k -х пикселов i -й ранговой области и домена, n - число пикселов в домене.

В этом случае среднеквадратичное отклонение яркостей домена и ранговой области может быть определено формулой:

$$R_i = \left[\sum_{k=1}^n z_{dk}^2 + s_i \cdot \left(s_i \sum_{k=1}^n z_{pk}^2 - 2 \left(\sum_{k=1}^n z_{pk} \cdot z_{dk} \right) + 2 o_i \sum_{k=1}^n z_{pk} \right) + o_i \left(o_i \cdot n - 2 \sum_{k=1}^n z_{dk} \right) \right] \Bigg/ n^2. \quad (8.6)$$

Такие операции проводятся по всем восьми возможным ориентациям квадрата. Для определения истинной ранговой области данного домена необходимо перебрать все возможные квадраты 8×8 изображения.

С этой целью вместо исходного квадрата используется квадрат, смешенный сначала по горизонтали на один пиксел, затем по вертикали на один пиксел, и таким образом перебираются все возможные варианты. В каждом конкретном случае при этом процесс преобразования и сравнения повторяется. Номер ориентации, коэффициенты преобразования контраста и яркости, положение верхнего левого угла кандидата в ранговую область и среднеквадратичные отклонения фиксируются, а затем ранговая область выбирается по наименьшей величине из всех указанных отклонений. Результат определяет ранговую область для данного домена.



Рис. 8.9. Исходное изображение для фрактального сжатия

На рис. 8.9 представлено исходное изображение, которое использовалось для сжатия. Рис. 8.10 иллюстрирует процесс восстановления изображения, сжатого фрактальным методом. В верхнем левом углу представлено исходное изображение (белое поле).

Пять первых итераций процесса генерации изображения (аттрактора), близкого к исходному по установленным алгоритмам преобразования координат ранговых областей, а также изменения коэффициентов их контраста и яркости представлены далее на рис. 8.10 слева направо и сверху вниз.

Исходное изображение было разбито на квадраты 4×4 пикселя, и каждый из них преобразован в соответствии с подобранными для него параметрами. В результате получилось следующее изображение (верхний правый угол). Далее таким же образом.



Рис. 8.10. Восстановление из белого поля изображения (аттрактора), сжатого фрактальным методом

После пятой итерации видимого изменения изображения в данном случае не происходит, и можно сказать, что процесс восстановления завершен.

Поскольку в процессе упаковки далеко не всегда удается подобрать ранговую область, абсолютно совпадающую (после преобразований) со своим доменом, восстановленное изображение несколько отличается от исходного.

На рис. 8.11 представлено восстановленное изображение для разбиения на домены 4×4 пиксела. Следует заметить, что такой размер домена обеспечивает очень хорошее воспроизведение исходного изображения.

На рис. 8.12 представлено восстановленное изображение для разбиения на домены 8×8 пикселов. Видно, что такой размер домена обеспечивает более низкое качество воспроизведения исходного изображения на рисунке в областях с высокочастотными деталями (например, волосы и серьги девушки) виден характерный межблочный эффект.



Рис. 8.11. Аттрактор для доменов 4×4 пиксела



Рис. 8.12. Аттрактор для доменов 8×8 пикселов

Следует обратить внимание на огромную разницу во времени при упаковке и восстановлении изображения. Процесс упаковки при рассмотренных вариантах подбора ранговых областей занимает интервал времени, на несколько порядков больший, чем процесс восстановления, реализуемый в реальном масштабе времени.

Рис. 8.13 иллюстрирует одно из основных свойств аттрактора — процесс восстановления этого же аттрактора но из другого исходного изображения (левый верхний угол). Как видно, после пятой итерации восстановления получается тот же аттрактор.

Таким образом, фрактальные методы обработки изображений уже сейчас могут найти применение в телевидении при заранее сжатых сюжетах (например, кинофильмах) и распаковке в реальном масштабе времени.

Дальнейшие исследования в этом направлении, возможно, позволят реализовать фрактальные алгоритмы и для непрерывного процесса кодирования/декодирования динамических изображений.



Рис. 8.13. Восстановление изображения (аттрактора), сжатого фрактальным методом, из другого (произвольного) изображения

ГЛАВА 9

ВЕЙВЛЕТЫ И КРАТНОМАСШТАБНАЯ ОБРАБОТКА ИЗОБРАЖЕНИЙ

Вейвлеты — это семейство математических функций определенной формы, локальных как во временной, так и в частотной областях [3.98-3.106]. Вейвлеты обычно формируются с помощью одной порождающей базовой функции путем ее сдвигов и растяжений по оси времени. Вейвлет-преобразование (WT — Wavelet Transform) сигналов является обобщением спектрального анализа (типичный представитель — классическое преобразование Фурье). Для анализа формы и спектра сигналов обычно используют так называемое непрерывное преобразование (НВП, CWT — Continuous Wavelet Transform), при преобразовании и кодировании сигналов применяют дискретное преобразование (ДВП, DWT — Discrete Wavelet Transform).

Вейвлет-преобразования в общем случае используют две непрерывные, взаимозависимые и интегрируемые по всей временной оси t функции:

- вейвлет-функции $\psi(t)$ с нулевым значением интеграла $\left(\int_{-\infty}^{\infty} \psi(t) dt = 0 \right)$ и с частотным Фурье-образом $\Psi(\omega)$; с использованием этих функций можно выделить локальные особенности сигнала;
- масштабирующие, так называемые скейлинг-функции $\varphi(t)$ с единичным значением интеграла $\left(\int_{-\infty}^{\infty} \varphi(t) dt = 1 \right)$; с их помощью реализуется грубое приближение (аппроксимация) сигнала, они необходимы при раздельном анализе низкочастотных и высокочастотных составляющих.

Вейвлет-функция $\psi(t)$ определяется той или иной базисной функцией $\psi_0(t)$ с применением к ней операций смещения во времени и масштабирования:

$$\psi(t) \equiv \psi_{a,b}(t) = a^{-1/2} \psi_0 \left(\frac{t-b}{a} \right), \quad (9.1)$$

где a и b — действительные числа ($a, b \in R$), параметр $a > 0$ определяет масштаб функции $\psi(t)$, а b — ее смещение по временной оси.

Фурье-образ $\psi(t)$ определяется выражением:

$$\Psi(\omega) = \int_{-\infty}^{\infty} \psi(t) \cdot e^{-j\omega t} dt. \quad (9.2)$$

В качестве примеров аналитически заданных базисных вейвлет-функций $\psi_0(t) \equiv \psi_{1,0}(t)$ можно привести:

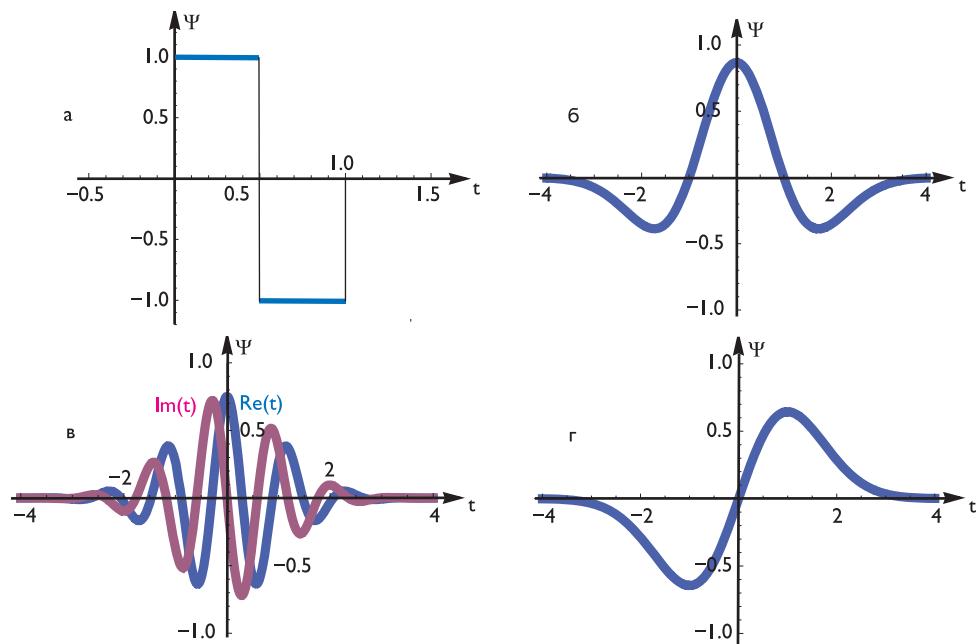


Рис. 9.1. Базисные вейвлет-функции: а) Хаара; б) мексиканская шляпа; в) Морле; г) производная гауссовой кривой

вейвлет Хаара (рис. 9.1а):

$$\psi_H(t) = \begin{cases} 1, & \text{при } 0 \leq t < 1/2, \\ -1, & \text{при } 1/2 \leq t < 1, \\ 0, & \text{иначе;} \end{cases} \quad (9.3)$$

мексиканская шляпа (рис. 9.1б):

$$\psi_{\text{Mhat}}(t) = \frac{2}{\sqrt{3}}\pi^{-1/4}(1-t^2) \cdot e^{-t^2/2}; \quad (9.4)$$

вейвлет Морле — модулированная Гауссова кривая (рис. 9.1в):

$$\psi_M(t) = \pi^{-1/4} \left(e^{-j\alpha t} - e^{-\alpha^2/2} \right) \cdot e^{-t^2/2}, \quad \text{где } \alpha = \pi \sqrt{\frac{2}{\ln 2}}; \quad (9.5)$$

производная Гауссовой кривой (рис. 9.1г):

$$\psi_{DG}(t) = \sqrt{2}\pi^{-1/4}t \cdot e^{-t^2/2}. \quad (9.6)$$

9.1. Непрерывное вейвлет-преобразование

Допустим, сигнал $u(t)$ стремится к нулю при $t \rightarrow \pm\infty$ и обладает конечной энергией: $P = \int_{-\infty}^{\infty} u^2(t)dt < \infty$. По аналогии с преобразованием Фурье **прямое**

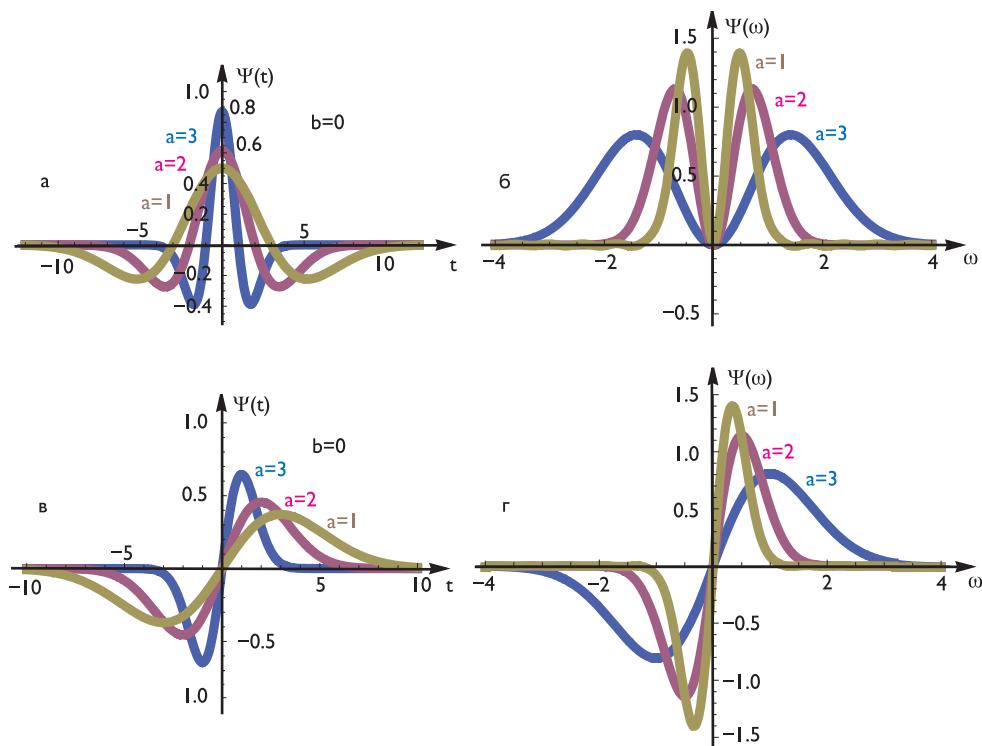


Рис. 9.2. Форма и спектр вейвлетов четного типа при разных значениях параметра «*a*» (мексиканская шляпа) — а, б; форма и спектр вейвлетов нечетного типа при разных значениях параметра «*a*» (производная Гауссовой кривой) — в, г

непрерывное вейвлет-преобразование задается вейвлет-коэффициентами:

$$C(a, b) = \int_{-\infty}^{\infty} u(t)\psi_{a,b}(t)dt, \quad (a, b \in R), \quad a > 0. \quad (9.7)$$

Прямое непрерывное вейвлет-преобразование является разложением сигнала по всем возможным сдвигам и растяжениям/сжатиям сигнала $u(t)$.

В отличие от Фурье-спектра $C(a, b)$ является функцией двух аргументов: временного масштаба «*a*» (в единицах, обратных частоте) и временного смещения «*b*» (в единицах времени).

Локальные особенности сигналов наиболее эффективно можно проанализировать путем выбора или расчета формы используемых вейвлет-функций и их частотного спектра. Так, при анализе гармонических сигналов вейвлетами четного типа (например, рис. 9.2а, б) коэффициенты $C(a, b)$ воспроизводятся в виде ярких горизонтальных полос с пиками и впадинами на доминирующих частотах вейвлетов, совпадающих с частотой гармоник сигналов. Нарушения гладкости сигналов фиксируются вертикальными полосами, пики в сигналах выделяются максимумами, а впадины — минимумами вейвлетных коэффициентов. Вейвлеты нечетного типа (например, рис. 9.2в, г) более резко реагируют на скачки

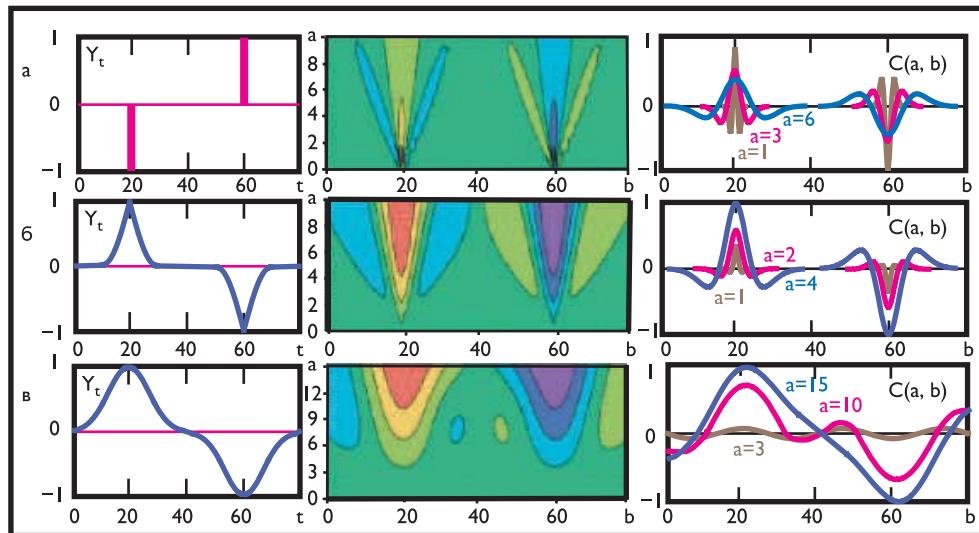


Рис. 9.3. Преобразования импульсов Кронекера (а), сигналов Лапласа (б) и сигналов Гаусса (в) с применением вейвлета «мексиканская шляпа»

и быстрые изменения в сигналах, отмечая их максимумами или минимумами в зависимости от знака производных.

На рис. 9.3 приведены результаты применения вейвлета $\psi_{Mhat}(t)$ для анализа следующих сигналов: двух разнополярных импульсов Кронекера, сигналов, форма которых подчиняется законам Лапласа и Гаусса. Цветовая гамма спектра на этих рисунках соответствует естественному цветовому ряду от красного (большие значения коэффициентов) к фиолетовому (малые значения коэффициентов). На сечениях спектра видно, что свертка единичных импульсов с разномасштабными вейвлетами повторяет форму вейвлетов, как это и положено при операции свертки. Соответственно, линии максимальных экстремумов на сечениях (в зависимости от полярности) определяют временное положение импульсов.

На рис. 9.4 приведены результаты применения вейвлета $\psi_{DG}(t)$ для анализа импульсов с крутым, плавным и линейным переходами при использовании вейвлета «производная Гауссовой кривой». Центры переходов фиксируются нулевыми значениями коэффициентов $C(a, b)$, а их крутизна отражается в основном на значениях функции $C(a, b)$ при малых величинах параметра « a ».

Формула реконструкции сигнала $u(t)$ с применением **обратного** непрерывного вейвлет-преобразования представима в виде:

$$u(t) = C_{\psi}^{-1} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} C(a, b) \cdot \psi_{a,b}(t) \cdot \frac{dadb}{a^2}, \quad (9.8)$$

где постоянная C_{ψ} зависит только от базисной функции $\psi_0(t)$ и определяется так:

$$C_{\psi} = 2\pi \int_{-\infty}^{\infty} \frac{|\Psi_0(\omega)|^2}{|\omega|} d\omega < \infty. \quad (9.9)$$

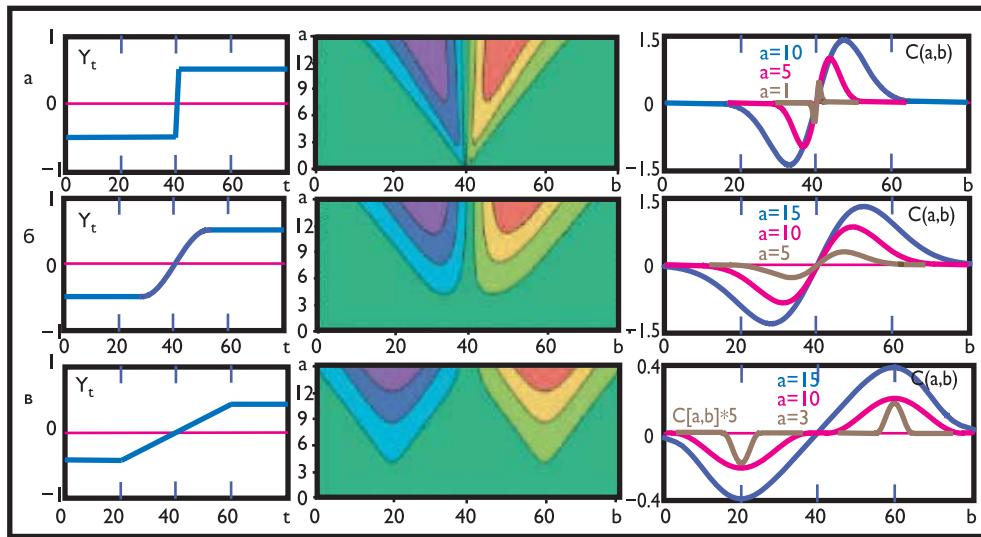


Рис. 9.4. Преобразования импульсов с крутым (а), плавным (б) и линейным (в) переходами при использовании вейвлета «производная гауссовой кривой»

Условие конечности C_ψ ограничивает класс функций, которые можно применить в качестве вейвлетов. Для обеспечения сходимости интеграла (9.8) в нуле значение $\Psi_0(0)$ должно быть тождественно равно нулю и, следовательно, функция $\psi(t)$ должна иметь нулевое среднее значение. Это означает, что не для всех сигналов возможна их точная реконструкция при использовании базисной функции $\psi_0(t)$.

9.2. Дискретное вейвлет-преобразование

Для уменьшения избыточности вейвлет-преобразований обычно задают дискретные значения коэффициентов a и b на множестве целых чисел $I \in \{-\infty, \infty\}$. Чаще всего используют так называемое *диадное* преобразование, при котором

$$a = 2^m, \quad b = k \cdot 2^m, \quad \psi_{m,k} = 2^{-m/2} \psi(2^{-m}t - k), \quad m, k \in I. \quad (9.10)$$

Вейвлет-коэффициенты прямого преобразования:

$$C_{m,k} = d_{m,k} = \int_{-\infty}^{\infty} u(t) \cdot \psi_{m,k}(t) dt, \quad (9.11)$$

где $d_{m,k}$ — дискретные детализирующие коэффициенты для вейвлет-представления сигнала уровня k . Обратное дискретное преобразование для непрерывных сигналов задается формулой:

$$u(t) = C_\psi \sum_{-\infty}^{\infty} \sum_{-\infty}^{\infty} d_{m,k} \cdot \psi_{m,k}. \quad (9.12)$$

Часто осуществляется нормировка базисных функций в частотной области таким образом, что $C_\psi = 1$.

Как и при непрерывном вейвлет-преобразовании, обратное дискретное преобразование (9.12) не может обеспечить восстановление сигналов, имеющих ненулевую постоянную составляющую. В этом случае восстановление дает близкий к исходному сигнал, причем близость воспринимается в смысле обеспечения минимума среднеквадратичной погрешности восстановления.

Поэтому при обработке таких сигналов вейвлеты используются, как правило, в паре со связанными с ними дискретными (масштабирующими) скейлинг-функциями. Если вейвлеты рассматривать как аналоги полосовых фильтров сигнала, в основном высокочастотных, при выделении локальных особенностей в сигнале, то скейлинг-функции вейвлетов представляют собой аналоги низкочастотных фильтров, которыми из сигнала выделяются в отдельный массив составляющие, не прошедшие вейвлетную фильтрацию.

Аналитика масштабирующей функций при произвольном $m = m_0 - \varphi_k(t) = \varphi_{m_0, k}(t)$ практически повторяет выражения (9.10)–(9.11):

$$\varphi_k = 2^{-m_0/2} \varphi(2^{-m_0}t - k), \quad m_0, k \in I, \quad c_k = \int_{-\infty}^{\infty} u(t) \cdot \varphi_k(t) dt,$$

и образует дополнительный базис пространства $L^2(R)$ на всей действительной оси $R(-\infty, \infty)$.

Сумма вейвлет-коэффициентов и скейлинг-коэффициентов разложения сигналов соответственно дает возможность выполнять точную реконструкцию сигналов, при этом вместо (9.12) используется следующее соотношение при обратном преобразовании:

$$u(t) = \sum_{k=-\infty}^{\infty} c_k \cdot \varphi_k(t) + \sum_{m=m_0}^{\infty} \sum_{k=-\infty}^{\infty} d_{m, k} \cdot \psi_{m, k}(t), \quad (9.13)$$

где c_k — скейлинг-коэффициенты, которые обычно называют коэффициентами аппроксимации сигнала, $d_{m, k}$ — вейвлет-коэффициенты или коэффициенты детализации.

Практическое использование вейвлет-преобразований связано в основном с дискретными вейвлетами, поскольку широко применяются цифровые методы обработки данных, а также в силу различий дискретного и непрерывного преобразований, хотя последние дают несколько более наглядное представление результатов анализа в виде поверхностей вейвлет-коэффициентов по непрерывным переменным.

На рис. 9.5 анализируемый сигнал состоит из двух модулированных гауссианов. Преобразование вейвлетом Морле четко показывает их пространственную и частотную локализацию, в то время как спектр Фурье дает только частотную локализацию.

Однако базисы на основе непрерывных вейвлетов часто не являются строго ортонормированными, поскольку их элементы бесконечно дифференцируемы и экспоненциально спадают на бесконечности. У дискретных вейвлетов эти проблемы легко снимаются, что обеспечивает более точную реконструкцию сигналов.

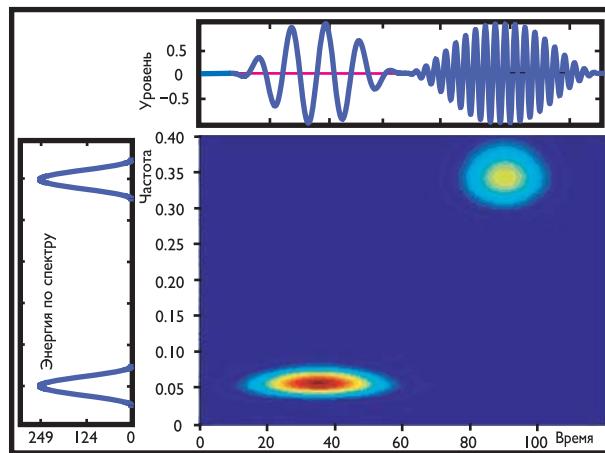


Рис. 9.5. Преобразование вейвлетом Морле сигнала в виде двух модулированных гауссианов

9.3. Кратномасштабный вейвлет-анализ

Представление формы сигнала путем суммирования его грубой аппроксимации с добавлением детализирующих локальных уточнений на различных временных интервалах — основа его кратномасштабного анализа (Multiresolution Analysis).

Часто требуется представить сигнал в виде совокупности его последовательных приближений. Например, при анализе изображений из некоторой базы данных можно сначала передать грубую его версию, а затем последовательно ее уточнять. При сжатии изображений очень часто без визуальной потери качества можно убирать из изображения незначимые мелкомасштабные детали.

Для реализации таких возможностей обычно используют ортогональные вейвлеты. Такие вейвлеты могут быть реализованы, основываясь на представлении некоторого пространства сигналов V в виде системы вложенных подпространств V_m , отличающихся друг от друга только перемасштабированием независимой переменной.

Ортогональный кратномасштабный анализ сигналов базируется на следующих исходных предпосылках:

- пространство $L^2(\mathbb{R})$ определяется как иерархически вложенные подпространства $V_m \subset L^2(\mathbb{R})$, $m = 0, \pm 1, \pm 2, \dots$, которые не пересекаются и объединение которых в пределе дает $L^2(\mathbb{R})$, при этом условием вложенности является — $V_m \subset V_{m+1} : \dots \subset V_{-1} \subset V_0 \subset V_1 \subset V_2 \subset \dots \subset V_m \subset V_{m+1} \subset \dots$.
- условие полноты разбиения определяется как: $\bigcup_{m \in \mathbb{Z}} V_m = L^2(\mathbb{R})$;
- условием ортогональности подпространств является: $\bigcap_{m \in \mathbb{Z}} V_m = \{0\}$;
- условие сохранения в подпространстве при сдвигах сигналов: $v_m(t) \in V_m \Leftrightarrow v_m(t+1) \in V_m$;
- для любого сигнала $v_m(t) \in V_m$ его масштабное преобразование по аргументу 2 перемещает этот сигнал в соседнее подпространство: $v_m(t) \in V_m \Leftrightarrow v_m(2t) = v_{m+1}(t) \in V_{m+1}$, $v_m(t) \in V_m \Leftrightarrow v_m(t/2) = v_{m-1}(t) \in V_{m-1}$;

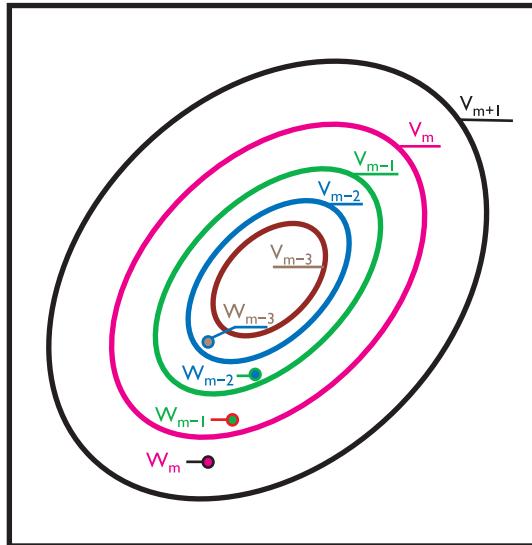


Рис. 9.6. Разбиение пространства $L^2(R)$ на систему вложенных подпространств

- для пространства V_0 существует функция $\varphi(t) \in V_0$, целочисленные сдвиги которой по аргументу образуют ортонормированный базис пространства V_0 : $\varphi_{0,k} = \varphi(t - k)$, $k \in I$ ($k = 0, \pm 1, \pm 2, \dots$), и условие нормировки этой масштабирующей (скейлинг) функции определяется соотношением:

$$\int_{-\infty}^{\infty} \varphi(t) dt = 1. \quad (9.14)$$

Из этих условий следует, что если подпространство V_0 имеет ортонормированный базис $\varphi_{0,k}$, то и все остальные подпространства также имеют ортонормированные базисы, которые образуются масштабным преобразованием базиса $\varphi_{0,k}$:

$$\varphi_{m,k}(t) = 2^{-m/2} \varphi(2^{-m}t - k), \quad m, k \in I. \quad (9.15)$$

Если сигнал $v_m(t)$ принадлежит пространству V_m , то одновременно он входит и в пространство V_{m+1} , и вместе с ним в этом пространстве находится и сигнал $v_m(2t)$. Увеличение номера пространства позволяет изучать все более и более мелкие детали и особенности сигнала с более высокочастотными компонентами.

На рис. 9.6 в условном виде приводится разбиение пространства $L^2(R)$ на систему вложенных подпространств. Все подпространства не пересекаются и вложены друг в друга так, что объем меньшего по номеру подпространства V_{m-1} в два раза меньше объема подпространства V_m , при этом четные числовые отсчеты сигнала $v_m(t) \in V_m$ входят в подпространство V_{m-1} , а нечетные отсчеты остаются в подпространстве с границами $[V_{m-1}, V_m)$ и отмеченном на рисунке индексами W_m .

Сигналы $v_m(t)$ являются ортогональными проекциями исходного сигнала $u(t)$ на подпространства V_m , что реализует возможность анализа сигнала на различ-

ных уровнях разрешения, или масштаба: если число m мало, то функция $v_m(t)$ — грубая аппроксимация $u(t)$, в которой отсутствуют детали.

Выбор масштабирующей функции $\varphi_0(t)$ достаточно произволен, но желательно стремиться к тому, чтобы спектр функций подпространства V_0 был ограничен.

Положим, пространство V_0 состоит из сигналов, имеющих разрешение по времени, равное условной «1». Тогда в пространстве V_m сигналы имеют разрешение 2^m , и оно отличается от V_0 только переслабированием базисной функции в соответствии с (9.15).

Функцию $\varphi(t)$ представляют в виде линейной комбинации сдвигов функции $\sqrt{2}\varphi(2t - k)$ с определенными весовыми коэффициентами масштабирования h_k . В общем случае может быть задано произвольное число весовых коэффициентов K :

$$\varphi(t/2) = \sqrt{2} \sum_{k=0}^{K-1} h_k \cdot \varphi(t - k) \quad \text{или} \quad \varphi(t) = \sqrt{2} \sum_{k=0}^{K-1} h_k \cdot \varphi(2t - k), \quad (9.16)$$

а значения h_k определяются из условия ортонормальности:

$$h_k = \sqrt{2} \int_{-\infty}^{\infty} \varphi(t) \cdot \varphi^*(2t - k) dt,$$

где $\varphi(t)$ и $\varphi^*(t)$ — комплексно сопряженные функции. Поскольку условие нормировки $\varphi(t)$ определяется соотношением (9.14), сумма квадратов коэффициентов h_k равна единице:

$$\sum_{k=0}^{K-1} h_k^2 = 1. \quad (9.17)$$

Масштабирующее уравнение в частотной области и полностью определяется периодической функцией $H(\omega)$:

$$\hat{\Phi}(2\omega) = H(\omega) \cdot \hat{\Phi}(\omega). \quad (9.18)$$

Перевод сигнала из пространства V_{m+1} с более высоким разрешением в пространство V_m реализует нормированную децимацию сигнала — двукратное прореживание при уменьшении в 2 раза числа отсчетов сигнала. Это эквивалентно передаче сигнала $v_{m+1}(k) \in V_m$ через низкочастотный фильтр с импульсной характеристикой, определяемой коэффициентами h_k , и с частотой среза, равной половине частоты Котельникова–Найквиста, при прореживании частотного диапазона децимированного сигнала $v_m(k)$ в 2 раза. На рис. 9.7 приведено отображение последовательности ряда таких операций в частотной области представления пространств $V_m(\omega)$, которые имеют физический смысл разделения спектров сигналов на низкочастотную $V_{m-1}(\omega)$ и высокочастотную W_{m-1} части.

При разложении сигнала составляющие пространства V_{m+1} разделяются на низкочастотные (подпространство V_m) и высокочастотные составляющие (подпространство W_m), как показано на рис. 9.8.

В пространстве W_m должна сохраняться высокочастотная часть информации сигнала, что может быть реализовано применением специально подобранного высокочастотного фильтра, и детализирующее уравнение в частотной области полностью определяется периодической функцией $G(\omega)$:

$$\hat{\Psi}(2\omega) = G(\omega) \cdot \hat{\Psi}(\omega), \quad (9.19)$$

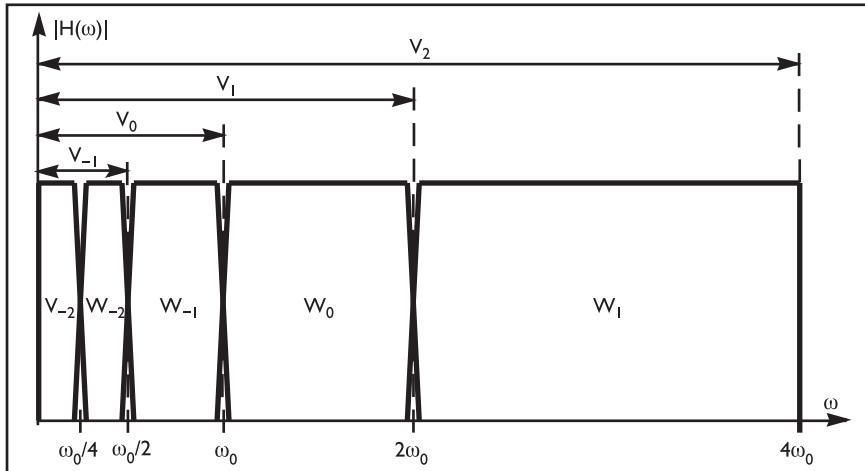


Рис. 9.7. Спектральные характеристики последовательности операций частотного разложения сигнала

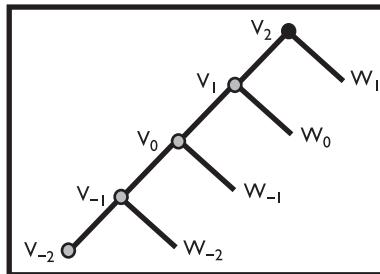


Рис. 9.8. Дерево пространств разложения сигнала

где $\hat{\Psi}(\omega)$ — Фурье-образ соответствующего вейвлета:

$$\psi(t/2) = \sqrt{2} \sum_{k=0}^{K-1} g_k \cdot \psi(t-k) \quad \text{или} \quad \psi(t) = \sqrt{2} \sum_{k=0}^{K-1} g_k \cdot \psi(2t-k), \quad (9.20)$$

$\int_{-\infty}^{\infty} \psi(t) dt = 0$, $g_k = \sqrt{2} \int_{-\infty}^{\infty} \psi(t) \cdot \psi^*(2t-k) dt$, $\psi(t)$ и $\psi^*(t)$ — комплексно сопряженные функции. Масштабированные и смешенные версии вейвлета определяются соотношением:

$$\psi_{m,k}(t) = 2^{-m/2} \psi(2^{-m}t - k), \quad m, k \in I. \quad (9.21)$$

Следует повторить, что разложение сигналов в вейвлетные ряды на заданном уровне разрешения t_0 выполняется по формуле (9.13). Это соотношение показывает возможность аппроксимации любого произвольного сигнала $u(t)$ набором простых локальных функций $\varphi_{m,k}(t)$ и $\psi_{m,k}(t)$, ортогональных на разных уровнях значений t и полностью покрывающих пространство $L^2(R)$ за счет смещений k .

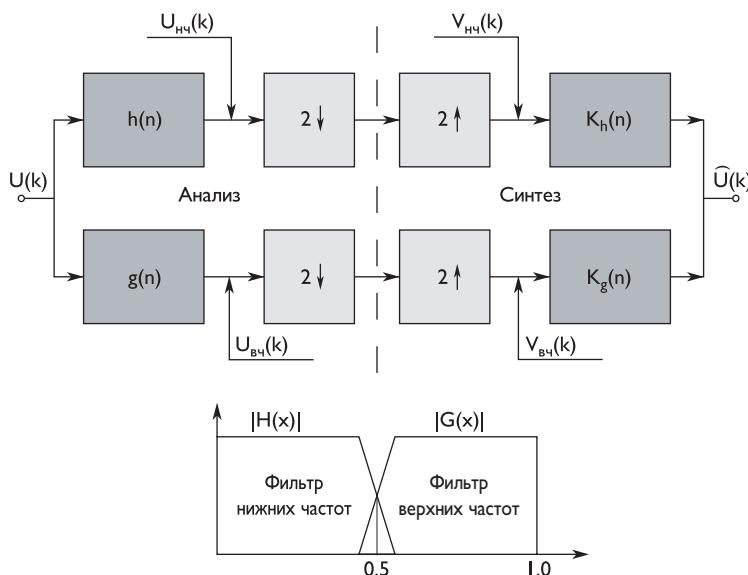


Рис. 9.9. Структурная схема двухканальной системы субполосного кодирования и декодирования

9.4. Основные сведения о субполосном кодировании

На рис. 9.9 приведена структурная схема двухканальной системы субполосного кодирования и декодирования. На вход системы кодирования (анализа) подается сигнал, представляющий собой временную последовательность отсчетов сигнала $u(k)$, $k = 0, \pm 1, \pm 2, \dots$, которую можно считать дискретными отсчетами ограниченной по спектру функции, взятыми через интервал Котельникова–Найквиста $T = \frac{1}{2}f_{\text{гр}}$, где $f_{\text{гр}}$ — граничная частота спектра сигнала $u(t)$, $f_{\text{д}} = 2f_{\text{гр}}$ — частота дискретизации.

Низкочастотный и высокочастотный фильтры анализа выделяют отсчеты двух сигналов, формируемых путем свертки:

$$v_{\text{H}^4}(k) = \sum_{n=-N_1}^{N_2} h(n) \cdot u(k-n), \quad (9.22)$$

$$v_{\text{B}^4}(k) = \sum_{m=-M_1}^{M_2} g(m) \cdot u(k-m),$$

где N_1, N_2, M_1, M_2 — произвольные числа, определяющие число отсчетов цифровой решетки низкочастотного и высокочастотного фильтров соответственно.

При изучении систем с дискретным временем обычно используют Z -преобразование, при котором последовательность $u(n)$ преобразуется следующим образом:

$$U(z) = \sum_{k=-\infty}^{\infty} u(k) \cdot z^{-k}. \quad (9.23)$$

При $z = \exp(i\pi x)$ это преобразование является обобщением дискретного преобразования Фурье, $x = f \cdot T/2$, $0 \leq x \leq 1$, T — интервал дискретизации.

Используя формулу (9.23), можно записать соотношения, определяющие Z -преобразования дискретных сигналов на выходах низкочастотного (НЧ) и высокочастотного (ВЧ) фильтров:

$$\begin{aligned} V_{\text{НЧ}}(z) &= H(z) \cdot U(z), \\ V_{\text{ВЧ}}(z) &= G(z) \cdot U(z), \end{aligned} \quad (9.24)$$

где $H(z)$ и $G(z)$ — Z -преобразования отсчетов цифровой решетки НЧ- и ВЧ-фильтров.

Отсчеты сигналов $v_{\text{НЧ}}(k)$ и $v_{\text{ВЧ}}(k)$ на выходах НЧ- и ВЧ-фильтров подвергаются децимации (устранению каждого второго, например нечетного отсчета). При синтезе сигнала в промежутках между отсчетами сигналов $u_{\text{НЧ}}(k)$ и $u_{\text{ВЧ}}(k)$ (см. рис. 9.9) вставляются нулевые отсчеты и Z -преобразования таких сигналов можно представить в следующем виде:

$$\begin{aligned} \hat{U}_{\text{НЧ}}(z) &= \frac{1}{2}[H(z) \cdot U(z) + H(-z) \cdot U(-z)], \\ \hat{U}_{\text{ВЧ}}(z) &= \frac{1}{2}[G(z) \cdot U(z) + G(-z) \cdot U(-z)]. \end{aligned}$$

Учитывая, что Z -преобразования отсчетов цифровой решетки восстанавливающих НЧ- и ВЧ-фильтров соответственно равны $K_h(z)$ и $K_g(z)$ для сигнала на выходе синтезирующей цепи, Z -преобразование можно записать так:

$$\begin{aligned} \hat{U}(z) &= \frac{1}{2}K_h(z) \cdot [H(z) \cdot U(z) + H(-z) \cdot U(-z)] + \\ &\quad + \frac{1}{2}K_g(z) \cdot [G(z) \cdot U(z) + G(-z) \cdot U(-z)]. \end{aligned} \quad (9.25)$$

Перегруппировав члены в соотношении (9.25), получим:

$$\begin{aligned} \hat{U}(z) &= \frac{1}{2}U(z) \cdot [K_h(z) \cdot H(z) + K_g(z) \cdot G(z)] + \\ &\quad + \frac{1}{2}U(-z) \cdot [K_h(z) \cdot H(-z) + K_g(z) \cdot G(-z)]. \end{aligned} \quad (9.26)$$

Точное восстановление исходного сигнала означает $\hat{u}(k) = u(k)$ и $\hat{U}(z) = U(z)$, что реализуется при выполнении условий:

$$\begin{aligned} K_h(z) \cdot H(z) + K_g(z) \cdot G(z) &= 2, \\ K_h(z) \cdot H(-z) + K_g(z) \cdot G(-z) &= 0. \end{aligned} \quad (9.27)$$

Для того чтобы фильтр с системной функцией $H(z)$ обеспечивал выделение низкочастотной составляющей сигнала, необходимо выполнение условий:

$$\begin{aligned} H(0) &= \sum_{n=-N_1}^{N_2} h_n = \text{const}, \\ H(1) &= \sum_{n=-N_1}^{N_2} (-1)^n \cdot h_n = 0. \end{aligned} \quad (9.28)$$

Фильтр с системной функцией $G(z)$ выделяет высокочастотную составляющую при

$$G(0) = \sum_{n=-M_1}^{M_2} g_m = 0,$$

$$G(1) = \sum_{n=-M_1}^{M_2} (-1)^m \cdot g_m = \text{const.} \quad (9.29)$$

Алгоритм обработки информации при субполосном кодировании можно описать несколько иначе.

Положим, на вход банка фильтров поступает один дискретный отсчет сигнала, изображенный на рис. 9.10а, тогда на выходах ФНЧ и ФВЧ формируются сигналы, показанные на рис. 9.10б и рис. 9.10в или г соответственно.

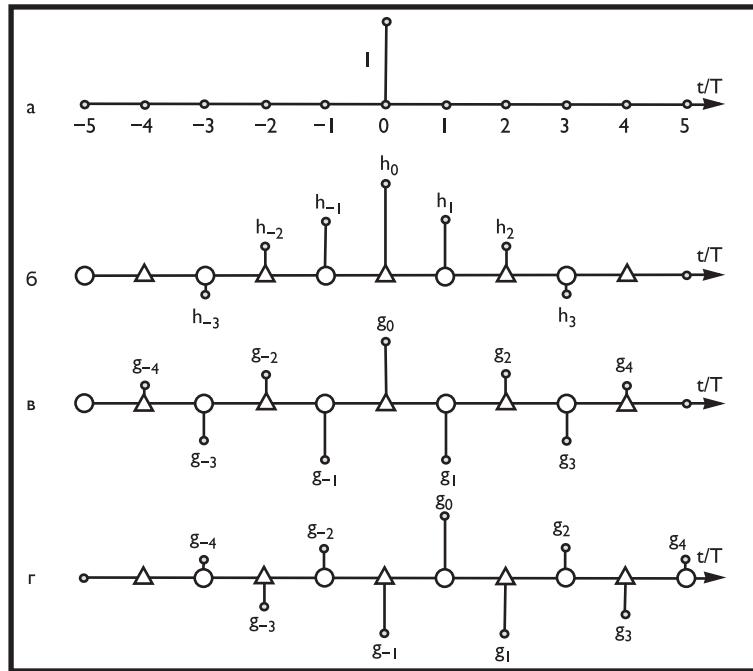


Рис. 9.10. Реакция ФНЧ (б) и ФВЧ (в, г) на дискретный отсчет входного сигнала (а)

В зависимости от того, каким образом осуществляется прореживание низкочастотного сигнала, после него выделяется сигнал, отмеченный на рис. 9.10б либо фигурой Δ , либо фигурой \circ :

$$H_{\Delta}(z) = \sum_{n=[-N_1/2]}^{[N_2/2]} h_{2n} \cdot z^{-2n}, \quad (9.30)$$

$$H_{\circ}(z) = \sum_{n=[(-N_1+1)/2]}^{[(N_2+1)/2]} h_{2n-1} \cdot z^{-(2n-1)}.$$

Если прореживание высокочастотного сигнала осуществляется синфазно, как по-

казано на рис. 9.10в, то выделяются сигналы

$$\begin{aligned} G_{\Delta}(z) &= \sum_{n=[-M_1/2]}^{[M_2/2]} g_{2n} \cdot z^{-2n}, \\ G_{\otimes}(z) &= \sum_{n=[(-M_1+1)/2]}^{[(M_2+1)/2]} g_{2n-1} \cdot z^{-(2n-1)}. \end{aligned} \quad (9.31)$$

В этом случае входной сигнал будет восстановлен при расчете характеристик восстанавливающих фильтров $Kh(z)$ и $Kg(z)$ в соответствии с системой:

$$\begin{cases} H_{\Delta}(z) \cdot Kh(z) + G_{\Delta}(z) \cdot Kg(z) = 1, \\ H_{\otimes}(z) \cdot Kh(z) + G_{\otimes}(z) \cdot Kg(z) = 1. \end{cases} \quad (9.32)$$

В случае же если прореживание высокочастотного сигнала смещено относительно прореживания низкочастотного сигнала на интервал дискретизации, как показано на рис. 9.10г, то восстановление сигнала будет осуществлено при выполнении условий:

$$\begin{cases} H_{\Delta}(z) \cdot Kh(z) + G_{\otimes}(z) \cdot Kg(z) = 1, \\ H_{\otimes}(z) \cdot Kh(z) + G_{\Delta}(z) \cdot Kg(z) = 1. \end{cases} \quad (9.33)$$

Вычитая и суммируя уравнения в соотношениях (9.32) и (9.33), можно получить соотношения, подобные (9.27) [3.105, 3.107]:

$$\begin{cases} \bar{H}(z) \cdot Kh(z) \pm \bar{G}(z) \cdot Kg(z) = 0, \\ H(z) \cdot Kh(z) + G(z) \cdot Kg(z) = 2, \end{cases} \quad (9.34)$$

где $\bar{H}(z) = \sum_{n=-N_1}^{N_2} (-1)^n \cdot h_n \cdot z^{-n}$, $\bar{G}(z) = \sum_{n=-M_1}^{M_2} (-1)^m \cdot g_m \cdot z^{-m}$. Знак «+» в первом уравнении (9.34) соответствует соотношениям (9.32), а знак «-» — соотношениям (9.33). Используя полученные соотношения, можно рассчитать ряд банков КИХ-фильтров с нечетным и четным числом отсчетов цифровой решетки.

9.4.1. КИХ-фильтры с нечетным числом отсчетов цифровой решетки и линейной (нулевой) ФЧХ

Частотные характеристики таких фильтров, учитывая, что $h_{-n} = h_n$ и $g_{-n} = g_n$, можно представить в виде:

$$\begin{aligned} H(x) &= h_0 + 2 \sum_{n=1}^N h_n \cdot \cos \pi n x, \\ G(x) &= g_0 + 2 \sum_{n=1}^M g_m \cdot \cos \pi m x, \\ \bar{H}(x) &= h_0 + 2 \sum_{n=1}^N (-1)^n \cdot h_n \cdot \cos \pi n x, \\ \bar{G}(x) &= g_0 + 2 \sum_{n=1}^M (-1)^m \cdot g_m \cdot \cos \pi m x, \\ H(0) &= \text{const}, \quad H(1) = 0, \quad G(1) = \text{const}, \quad G(0) = 0, \quad 0 \leq x \leq 1. \end{aligned} \quad (9.35)$$

В данном случае восстановление сигнала возможно только с помощью КИХ-фильтров, если используется система уравнений:

$$\begin{cases} \bar{H}(x) \cdot Kh(x) - \bar{G}(x) \cdot Kg(x) = 0, \\ H(x) \cdot Kh(x) + G(x) \cdot Kg(x) = 2, \end{cases} \quad (9.36)$$

и число $N+M = 2K+1$ — нечетное. Детерминант этой системы уравнений равен:

$$\begin{aligned} \det(x) &= [h_0 + 2 \sum_{n=1}^N h_n \cdot \cos \pi n x] \cdot [g_0 + 2 \sum_{n=1}^M (-1)^m \cdot g_m \cdot \cos \pi m x] + \\ &+ [h_0 + 2 \sum_{n=1}^N (-1)^n \cdot h_n \cdot \cos \pi n x] \cdot [g_0 + 2 \sum_{n=1}^M g_m \cdot \cos \pi m x] = \\ &= 2 \cdot \{h_0 \cdot g_0 + h_0 \sum_{k=1}^M [1 + (-1)^k] \cdot g_k \cdot \cos \pi k x + g_0 \sum_{k=1}^N [1 + (-1)^k] \cdot h_k \cdot \cos \pi k x + \\ &+ \sum_{n=1}^N \sum_{k=n-1}^{n-M} (-1)^n \cdot h_n \cdot g_{n-k} \cdot [1 + (-1)^k] \cdot \cos \pi k x + \\ &+ \sum_{n=1}^N \sum_{k=n+1}^{n+M} (-1)^n \cdot h_n \cdot g_{k-n} \cdot [1 + (-1)^k] \cdot \cos \pi k x\} = A_0 + \sum_{\forall k} A_k \cdot \cos \pi k x \end{aligned}$$

Из этого соотношения следует: коэффициенты A_k при всех нечетных значениях k равны нулю;

$$A_0 = 2 \cdot (h_0 \cdot g_0 + 2 \sum_{n=1}^{\min\{N,M\}} (-1)^n \cdot h_n \cdot g_n); \quad (9.37)$$

если $A_0 = 2$ и все значения A_{2k} при $k \neq 0$ приравнять к нулю, то восстанавливающие фильтры являются КИХ-фильтрами и

$$Kh(x) = \bar{G}(x); \quad Kg(x) = \bar{H}(x); \quad (9.38)$$

можно показать, что если

$$\begin{cases} A_{2k} = 0, & k \neq 0, \\ H(0) = h_0 + 2 \sum_{n=1}^N h_n = \sqrt{2}, & H(1) = h_0 + 2 \sum_{n=1}^N (-1)^n h_n = 0, \\ G(0) = g_0 + 2 \sum_{n=1}^M g_m = 0, & G(1) = g_0 + 2 \sum_{n=1}^M (-1)^m g_m = \sqrt{2}, \end{cases} \quad (9.39)$$

то $h_0 \cdot g_0 + 2 \sum_{n=1}^{\min\{N,M\}} (-1)^n \cdot h_n \cdot g_n = 1$ и $A_0 = 2$. $g_m, 0 \leq m \leq M$, требуется использовать $N+M+2$ уравнения, дополнительно можно приравнивать нулю четные моменты функций $H(x)$ и $G(x)$, т. е.

$$H^{(2r)}(x) \Big|_{x=0}, r = 1, 2, 3, \dots \quad \text{и} \quad G^{(2p)}(x) \Big|_{x=0}, p = 1, 2, 3, \dots$$

Нечетные производные в точке $x = 0$ равны нулю по определению функций $H(x)$ и $G(x)$. Равенство нулю четных производных функции $H(x)$ в точке $x = 0$ приводит к расширению спектра низкочастотной составляющей сигнала, равенство же нулю четных производных функции $G(x)$ при $x = 0$ обеспечивает

уменьшение высокочастотной части спектра, выделяемой составляющей сигнала. Зачастую для оптимизации формы спектра высокочастотной составляющей целесообразно подбирать значения некоторых производных $G(x)$ в точке $x = 0$.

В табл. 9.1 приведены параметры нескольких фильтров, рассчитанных по приведенной методике. Обозначение фильтров, стоящее в первом столбце, соответствует числу дискретных отсчетов фильтра при его реакции на одиничный входной сигнал — $(2N + 1)/(2M + 1)$. В ряде случаев система уравнений имеет несколько решений, наилучшие из которых зафиксированы в табл. 9.1.

Таблица 9.1. Параметры фильтров с нечетным числом отсчетов цифровой решетки и линейной ФЧХ

Тип банка	h_0	h_1	h_2	h_3	h_4	h_5	h_6	h_7	h_8
	g_0	g_1	g_2	g_3	g_4	g_5	g_6	g_7	g_8
3/5	$1/\sqrt{2}$	$1/2\sqrt{2}$							
	$3/2\sqrt{2}$	$-1/2\sqrt{2}$	$-1/4\sqrt{2}$						
5/3	$3/2\sqrt{2}$	$1/2\sqrt{2}$	$-1/4\sqrt{2}$						
	$1/\sqrt{2}$	$-1/2\sqrt{2}$							
5/7	0,91924	0,35355	-0,10607						
	0,79549	-0,36681	-0,04419	0,01326					
7/5	1,16168	0,33935	-0,22728	0,01421					
	0,66291	-0,35355	0,02210						
7/9	1,26091	0,25930	-0,27690	0,09426					
	0,65488	-0,39145	0,03901	0,03790	-0,01290				
7/11	0,79888	0,37480	-0,04589	-0,02125					
	0,85202	-0,41007	-0,09961	0,06581	0,03146	-0,00929	-0,00430		
9/7	1,01647	0,35355	-0,17678	0,0	0,02210				
	0,70711	-0,39775	0,0	0,04419					
9/11	1,19613	0,29344	-0,25647	0,06012	0,01196				
	0,65833	-0,40098	0,03252	0,04905	-0,00813	-0,00162			
9/15	1,0453	0,30266	-0,16208	0,05089	-0,00702				
	0,72798	-0,39802	-0,00855	0,04776	-0,00277	-0,00317	0,00089	-0,00012	
11/9	1,0843	0,35890	-0,23400	0,00097	0,04540	-0,00632			
	0,67017	-0,39775	0,02462	0,04419	-0,00616				
11/13	1,08054	0,37587	-0,24648	-0,01323	0,05976	-0,00908			
	0,66295	-0,39614	0,02923	0,04178	-0,00703	0,00081	-0,00012		
11/17	1,09347	0,29619	-0,19617	0,06208	0,003000	-0,00472			
	0,70220	-0,40955	0,00676	0,06203	-0,00660	-0,00612	0,00243	0,00009	-0,00014
13/11	0,93001	0,35053	-0,12210	-0,00042	0,01638	0,00344	0,00574		
	-0,74971	-0,42142	-0,02840	0,07970	0,00710	-0,01184			
13/15	1,19422	0,23649	-0,21254	0,11253	-0,03020	0,00453	-0,00081		
	0,69160	-0,40702	0,01424	0,05747	-0,00882	-0,00359	0,00234	-0,00042	

Фильтры 5/3 и 9/7 по своим характеристикам идентичны соответствующим фильтрам, приведенным в стандарте JPEG-2000 [3.108, 3.110].

В качестве примера на рис. 9.11 для сравнения приведены характеристики низкочастотного и высокочастотного фильтров для банков 5/3 и 13/11. Как следует из сравнения этих характеристик, высокочастотные составляющие сигнала на выходе банка 13/11 практически равны нулю в полосе частот до $x = 0,25$.

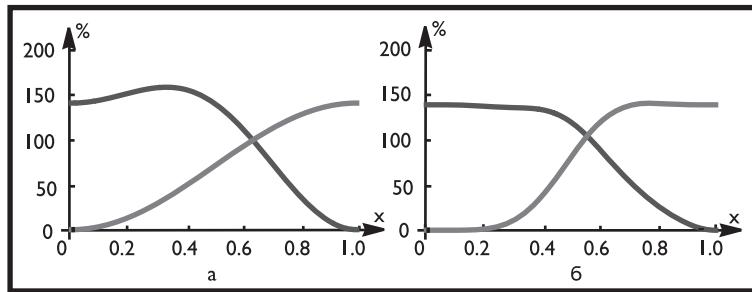


Рис. 9.11. АЧХ фильтров 5/3 (а) и 13/11 (б)

Ниже для пояснения приведена система уравнений, использованная при расчете банка фильтров 13/11. Первые пять уравнений обеспечивают выполнение условия $\det(x) = \text{const}$. Эти уравнения с последующими четырьмя уравнениями обеспечивают выполнение условий (9.39). Следующее уравнение определяет равенство нулю второго момента низкочастотного фильтра, а последние три уравнения накладывают условия на второй, четвертый и шестой моменты высокочастотного фильтра, при этом второй и четвертый моменты приравнены к нулю.

$$\left\{ \begin{array}{l} -g_5 \cdot h_5 + g_4 \cdot h_6 = 0 \\ -g_5 \cdot h_3 + g_4 \cdot h_4 - g_3 \cdot h_5 + g_2 \cdot h_6 = 0 \\ -g_5 \cdot h_1 + g_4 \cdot h_2 - g_3 \cdot h_3 + g_2 \cdot h_4 - g_1 \cdot h_5 + g_0 \cdot h_6 = 0 \\ g_4 \cdot h_0 - g_3 \cdot h_1 - g_5 \cdot h_1 + g_2 \cdot h_2 - g_1 \cdot h_3 + g_0 \cdot h_4 - g_1 \cdot h_5 + g_2 \cdot h_6 = 0 \\ g_2 \cdot h_0 - g_1 \cdot h_1 - g_3 \cdot h_1 + g_0 \cdot h_2 - g_1 \cdot h_3 + g_4 \cdot h_2 - g_5 \cdot h_3 + g_2 \cdot h_4 - g_3 \cdot h_5 + g_4 \cdot h_6 = 0 \\ h_0 + 2 \cdot h_1 + 2 \cdot h_2 + 2 \cdot h_3 + 2 \cdot h_4 + 2 \cdot h_5 + 2 \cdot h_6 = \sqrt{2} \\ h_0 - 2 \cdot h_1 + 2 \cdot h_2 - 2 \cdot h_3 + 2 \cdot h_4 - 2 \cdot h_5 + 2 \cdot h_6 = 0 \\ g_0 + 2 \cdot g_1 + 2 \cdot g_2 + 2 \cdot g_3 + 2 \cdot g_4 + 2 \cdot g_5 = 0 \\ g_0 - 2 \cdot g_1 + 2 \cdot g_2 - 2 \cdot g_3 + 2 \cdot g_4 - 2 \cdot g_5 = \sqrt{2} \\ h_1 + 4 \cdot h_2 + 9 \cdot h_3 + 16 \cdot h_4 + 25 \cdot h_5 + 36 \cdot h_6 = 0 \\ g_1 + 4 \cdot g_2 + 9 \cdot g_3 + 16 \cdot g_4 + 25 \cdot g_5 = 0 \\ g_1 + 4 \cdot 4 \cdot g_2 + 9 \cdot 9 \cdot g_3 + 16 \cdot 16 \cdot g_4 + 25 \cdot 25 \cdot g_5 = 0 \\ g_1 + 4 \cdot 4 \cdot 4 \cdot g_2 + 9 \cdot 9 \cdot 9 \cdot g_3 + 16 \cdot 16 \cdot 16 \cdot g_4 + 25 \cdot 25 \cdot 25 \cdot g_5 = 0 \end{array} \right.$$

На рис. 9.12 приведены сквозные характеристики банка фильтров 13/11 в точках до суммирования составляющих (см. рис. 9.9), соответствующие первому (а) и второму (б) уравнениям системы (9.33). В данном случае:

$$\begin{aligned} H_{\Delta}(z) \cdot Kh(z) &= H_{\Delta}(x) \cdot \bar{G}(x); & G_{\otimes}(z) \cdot Kg(z) &= G_{\otimes}(x) \cdot \bar{H}(x); \\ H_{\otimes}(z) \cdot Kh(z) &= H_{\otimes}(x) \cdot \bar{G}(x); & G_{\Delta}(z) \cdot Kg(z) &= G_{\Delta}(z) \cdot \bar{H}(x). \end{aligned}$$

Заметим, что в первом случае характеристика высокочастотной составляющей проходит через нуль при $x = 0,5$, во втором же случае через нуль в этой точке проходит характеристика низкочастотной составляющей.

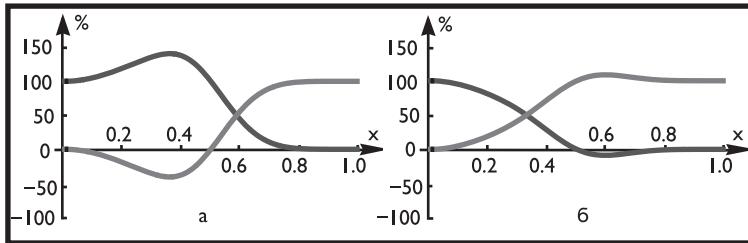


Рис. 9.12. Сквозные характеристики банка фильтров 13/11

9.4.2. КИХ-фильтры с четным числом отсчетов цифровой решетки

При четном числе отсчетов цифровой решетки низкочастотного и высокочастотного фильтров банка для того, чтобы число отсчетов справа и слева от оси ординат было одинаково, необходимо ее установить посередине между центральными отсчетами, как показано на рис. 9.13.

В этом случае положение каждого отсчета относительно оси ординат будет определяться интервалом, равным $(2n-1)T/2$, $n = \dots, -2, -1, 0, 1, 2, \dots$. Обозначим отсчеты фильтра нижних частот через $h_{(2n-1)/2}$, а фильтра верхних частот — через $g_{(2n-1)/2}$. В случае, если $h_{(2n-1)/2} = h_{-(2n-1)/2}$, а $g_{(2n-1)/2} = -g_{-(2n-1)/2}$ при $n = 1, 2, 3, \dots$, комплексные частотные характеристики соответствующих фильтров можно записать в виде:

$$H(x) = 2 \sum_{n=1}^N h_{(2n-1)/2} \cdot \cos\left(\pi \frac{2n-1}{2} x\right);$$

$$G(x) = -2j \sum_{m=1}^M g_{(2m-1)/2} \cdot \sin\left(\pi \frac{2m-1}{2} x\right). \quad (9.40)$$

В данном случае в Z -представлении величины $H_{\Delta}(z)$, $H_{\otimes}(z)$, $G_{\Delta}(z)$, $G_{\otimes}(z)$ соответственно равны:

$$H_{\Delta}(z) = \sum_{n=1}^{[N/2]} h_{(4n-1)/2} \cdot z^{\frac{4n-1}{2}} + \sum_{n=1}^{[(N+1)/2]} h_{(4n-1)/2} \cdot z^{-\frac{4n-1}{2}};$$

$$H_{\otimes}(z) = \sum_{n=1}^{[(N+1)/2]} h_{(4n-3)/2} \cdot z^{\frac{4n-3}{2}} + \sum_{n=1}^{[N/2]} h_{(4n-3)/2} \cdot z^{-\frac{4n-3}{2}};$$

$$G_{\Delta}(z) = - \sum_{m=1}^{[M/2]} g_{(4m-1)/2} \cdot z^{\frac{4m-1}{2}} + \sum_{n=1}^{[(M+1)/2]} g_{(4m-1)/2} \cdot z^{-\frac{4m-1}{2}};$$

$$G_{\otimes}(z) = - \sum_{m=1}^{[(M+1)/2]} g_{(4m-3)/2} \cdot z^{\frac{4m-3}{2}} + \sum_{m=1}^{[M/2]} g_{(4m-3)/2} \cdot z^{-\frac{4m-3}{2}}. \quad (9.41)$$

Подставив эти соотношения в систему уравнений (9.32), можно рассчитать параметры банка фильтров.

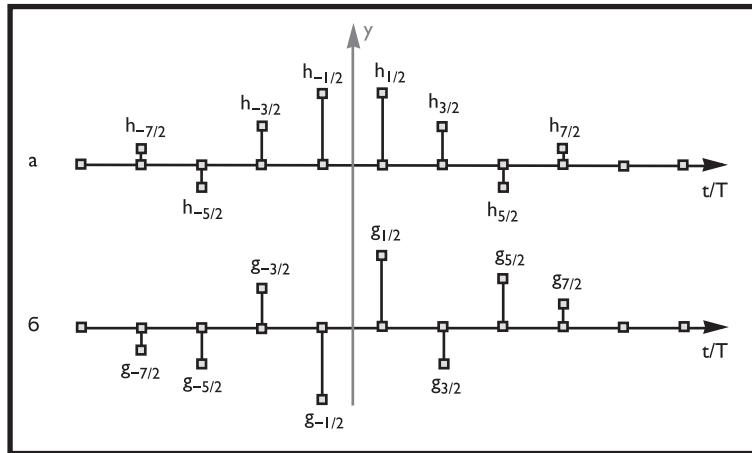


Рис. 9.13. Структура низкочастотного (а) и высокочастотного (б) КИХ-фильтров при четном количестве отсчетов цифровой решетки

Эквивалентно использование соотношения:

$$\begin{cases} \bar{H}(x) \cdot Kh(x) + \bar{G}(x) \cdot Kg(x) = 0, \\ H(x) \cdot Kh(x) + G(x) \cdot Kg(x) = 2, \end{cases} \quad (9.42)$$

где

$$\begin{aligned} \bar{H}(x) &= 2j \sum_{n=1}^N (-1)^n \cdot h_{(2n-1)/2} \cdot \sin\left(\pi \frac{2n-1}{2} x\right); \\ \bar{G}(x) &= -2 \sum_{m=1}^M (-1)^n \cdot g_{(2m-1)/2} \cdot \cos\left(\pi \frac{2m-1}{2} x\right). \end{aligned}$$

Число $N+M = 2K$ должно быть четным. Детерминант системы уравнений (9.42) можно записать в виде:

$$\begin{aligned} \det(x) &= 2 \sum_{n=1}^N \sum_{k=n-1}^{n-M} (-1)^n \cdot [(-1)^k - 1] \cdot h_{(2n-1)/2} \cdot g_{(2(n-k)-1)/2} \cdot \cos(\pi kx) + \\ &+ 2 \sum_{n=1}^N \sum_{k=n}^{n+M-1} (-1)^n \cdot [(-1)^{k+1} + 1] \cdot h_{(2n-1)/2} \cdot g_{(2(k-n)-1)/2} \cdot \cos(\pi kx) = \\ &= A_0 + \sum_{\forall k} A_k \cdot \cos \pi kx \quad (9.43) \end{aligned}$$

Исследуя это соотношение, аналогично изложенному выше, можно показать, что: коэффициенты A_k при всех нечетных значениях k равны нулю;

$$A_0 = 4 \sum_{n=1}^{\min\{N,M\}} (-1)^n \cdot h_{(2n-1)/2} \cdot g_{(2n-1)/2}; \quad (9.44)$$

если $A_0 = 2$ и все значения A_{2k} при $k \neq 0$ приравнять к нулю, то восстанавливающие фильтры являются КИХ-фильтрами и

$$Kh(x) = \bar{G}(x); \quad Kg(x) = \bar{H}(x); \quad (9.45)$$

из определения функций $H(x), \bar{H}(x), G(x), \bar{G}(x)$ следует, что $H(1) = 0, \bar{H}(0) = 0, G(0) = 0, \bar{G}(1) = 0$; можно доказать также, что если

$$\begin{cases} A_{2k} = 0, \quad k \neq 0; \\ H(0) = 2 \sum_{n=1}^N h_{(2n-1)/2} = \sqrt{2}; \\ G(1) = g_0 + 2 \sum_{m=1}^M (-1)^m g_{(2m-1)/2} = \sqrt{2}, \end{cases}$$

то $2 \sum_{n=1}^{\min\{N,M\}} (-1)^n \cdot h_{(2n-1)/2} \cdot g_{(2n-1)/2} = 1$ и $A_0 = 2$.

Таблица 9.2. Параметры КИХ-фильтры с четным числом отсчетов цифровой решетки

Тип банка	$h_{1/2}$	$h_{3/2}$	$h_{5/2}$	$h_{7/2}$	$h_{9/2}$	$h_{11/2}$	$h_{13/2}$	$h_{15/2}$	$h_{17/2}$
	$g_{1/2}$	$g_{3/2}$	$g_{5/2}$	$g_{7/2}$	$g_{9/2}$	$g_{11/2}$	$g_{13/2}$	$g_{15/2}$	$g_{17/2}$
2/2	$1/\sqrt{2}$								
	$1/\sqrt{2}$								
2/6	$1/\sqrt{2}$								
	$1/\sqrt{2}$	$-1/8\sqrt{2}$	$-1/8\sqrt{2}$						
4/4	$3/2\sqrt{2}$	$1/2\sqrt{2}$							
	$3/4\sqrt{2}$	$-3/4\sqrt{2}$							
4/8	0,79549	-0,08839							
	0,64302	-0,13037	-0,05966	0,00663					
6/6	0,92685	-0,19178	-0,02796						
	0,57159	-0,15865	-0,02313						
6/10	0,79502	0,04443	-0,13234						
	0,61023	-0,13676	-0,06635	-0,00665	0,01982				
8/8	0,85243	-0,07903	-0,12946	0,06317					
	0,59770	-0,17570	-0,04455	0,02174					
8/12	0,87834	-0,12222	-0,13234	0,08333					
	0,59247	-0,19503	-0,046422	0,03240	-0,00097	0,00061			
10/10	0,95972	-0,20193	-0,153082	0,11281	-0,01041				
	0,56320	-0,20458	-0,02744	0,03043	-0,00281				
10/14	0,72294	0,06177	0,09762	0,01435	0,00566				
	0,66927	-0,12086	-0,08869	-0,00005	0,00523	-0,00062	-0,00024		
12/12	0,98458	-0,23183	-0,16533	0,13003	-0,00284	-0,00749			
	0,55896	-0,21308	-0,02582	0,04000	-0,00054	-0,00143			
12/16	0,83729	-0,06918	-0,16953	0,085012	0,02353	-0,01692			
	0,59260	-0,19976	-0,05335	0,03983	0,00345	-0,00343	0,00061	-0,00044	
14/14	0,88664	-0,11406	-0,17114	0,10839	0,01655	-0,02133	0,00206		
	0,58656	-0,20810	-0,05111	0,04621	0,00316	-0,00602	0,00058		
16/16	0,85855	-0,12399	-0,08598	0,05468	0,02137	-0,01716	-0,00278	0,00241	
	0,60752	-0,20251	-0,07290	0,05207	0,01350	-0,01105	-0,00133	0,00116	
18/18	0,78189	0,04881	-0,03671	0,00673	0,00548	-0,00085	-0,00073	0,00006	0,00005
	0,64549	-0,17213	-0,10328	0,03688	0,02869	-0,00671	-0,00567	0,00061	0,00054

В табл. 9.2 приведены параметры нескольких ортогональных фильтров, рассчитанных по приведенной методике. Обозначение фильтров, приведенное в пер-

вом столбце, определяет число дискретных отсчетов фильтра при его реакции на одиночный входной сигнал: $2N/2M$. Существенно, что прореживание (децимация) сигналов на выходах низкочастотного и высокочастотного фильтров в данном случае должно осуществляться синфазно.

Заметим, что нечетные производные функции $H(x)$ в точке $x = 0$ равны нулю, а в точке $x = 1$ равны нулю четные производные этой функции. Противоположный характер имеют производные функции $G(x)$.

На рис. 9.14 приведены амплитудно-частотные характеристики фильтров 2/2 (а, вейвлет Хаара) и 18/18 (б).

В отличие от фильтров, рассмотренных в предыдущем разделе, в данном случае характеристики цепей после децимации имеют близкие к равномерным частотные характеристики и нелинейные фазовые характеристики. При этом АЧХ могут быть рассчитаны по формуле

$$\text{Mod } H_{\Delta}(x) = \text{Mod } H_{\otimes}(x) =$$

$$= \sqrt{\sum_{n=1}^N \sum_{m=1}^N h_{(2n-1)/2} \cdot h_{(2m-1)/2} \cdot \cos \pi x \cdot [n - 1/2 - (-1)^{n+m} \cdot (m - 1/2)]}. \quad (9.46)$$

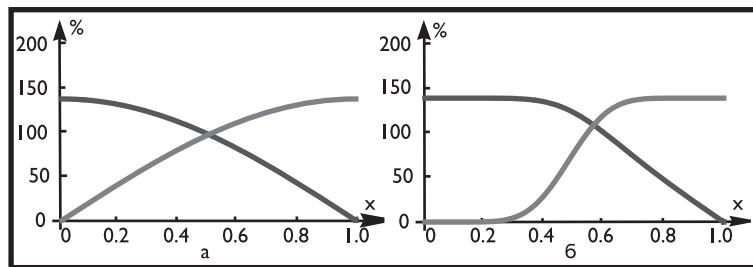


Рис. 9.14. АЧХ фильтров 2/2 (а) и 18/18 (б)

Для функций $\text{Mod } G_{\Delta}(x)$ и $\text{Mod } G_{\otimes}(x)$ формула аналогична, но вместо произведения $h_{(2n-1)/2} \cdot h_{(2m-1)/2}$ необходимо подставить $g_{(2n-1)/2} \cdot g_{(2m-1)/2}$. Для низкочастотного и высокочастотного фильтров банка 18/18 графики $\text{Mod } G_{\Delta}(x)$ и $\text{Mod } G_{\otimes}(x)$ приведены на рис. 9.15.

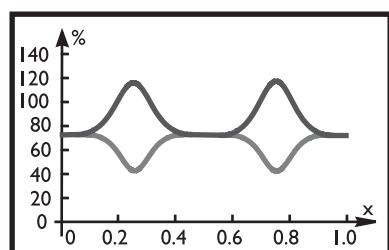


Рис. 9.15. АЧХ ФНЧ (сверху) и ФВЧ (снизу) банка 18/18 после децимации

Групповое время запаздывания (производная фазовой характеристики) этих фильтров после децимации имеет вид:

$$\begin{aligned}
 t_{gr}[H_{\Delta}(x)] &= -t_{gr}[H_{\otimes}(x)] = \\
 &= \frac{\pi \cdot T}{\text{Mod } H_{\Delta}(x)} \left[\sum_{n=1}^N \sum_{m=1}^N \frac{2n-1}{2} \cdot (-1)^{n+1} h_{(2n-1)/2} \times \right. \\
 &\quad \left. \times h_{(2m-1)/2} \cdot \cos \pi x \cdot \left[n - \frac{1}{2} - (-1)^{n+m} \cdot \left(m - \frac{1}{2} \right) \right] \right]. \quad (9.47)
 \end{aligned}$$

Формулы расчета $t_{gr}[G_{\Delta}(x)] = -t_{gr}[G_{\otimes}(x)]$ могут быть получены при замене $h_{(2n-1)/2} \cdot h_{(2m-1)/2}$ на $g_{(2n-1)/2} \cdot g_{(2m-1)/2}$ и $\text{Mod } H_{\Delta}(x)$ на $\text{Mod } G_{\Delta}(x)$.

На рис. 9.16а и б приведены относительные характеристики группового времени запаздывания фильтров 18/18 после децимации, соответствующие первому и второму уравнениям системы (9.33).

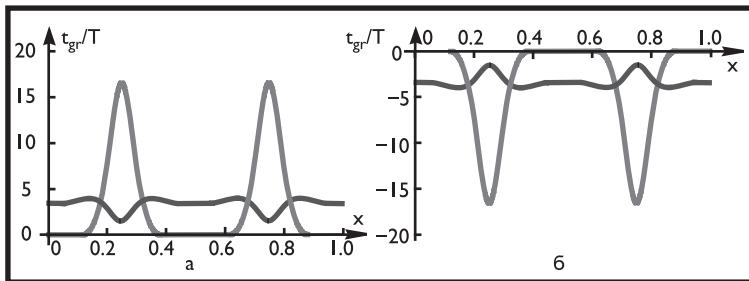


Рис. 9.16. Относительные характеристики группового времени запаздывания фильтров 18/18 после децимации (менее изменяющаяся характеристика соответствует ФНЧ, а более изрезанная — ФВЧ)

9.4.3. Квадратурно-зеркальные КИХ-фильтры

Аналогично производится расчет квадратурно-зеркальных фильтров, частотные характеристики которых в Z -представлении имеют вид:

$$\begin{aligned}
 H(z) &= \sum_{n=1}^N \left(h_{-\frac{2n-1}{2}} \cdot z^{\frac{2n-1}{2}} + h_{\frac{2n-1}{2}} \cdot z^{-\frac{2n-1}{2}} \right), \\
 G(z) &= \sum_{n=1}^N (-1)^n \left(h_{-\frac{2n-1}{2}} \cdot z^{\frac{2n-1}{2}} - h_{\frac{2n-1}{2}} \cdot z^{-\frac{2n-1}{2}} \right).
 \end{aligned} \quad (9.48)$$

В качестве примера приведены характеристики квадратурно-зеркального фильтра 10/10. Система уравнений в данном случае дает четыре равнозначных решения, приведенных в табл. 9.3. На рис. 9.17 приведены АЧХ этих фильтров.

Таблица 9.3. Характеристики квадратурно-зеркального фильтра 10/10

$h_{-9/2}$	$h_{-7/2}$	$h_{-5/2}$	$h_{-3/2}$	$h_{-1/2}$	$h_{1/2}$	$h_{3/2}$	$h_{5/2}$	$h_{7/2}$	$h_{9/2}$
0,16010	0,60383	0,72431	0,13843	-0,24229	-0,03224	0,07757	-0,00624	-0,01258	0,00334
0,00334	-0,01258	-0,00624	0,07757	-0,03224	-0,24229	0,13843	0,72431	0,60383	0,16010
0,02733	0,0295	-0,03913	0,19940	0,72341	0,63398	0,01660	-0,17533	-0,02110	0,01954
0,01954	-0,02110	-0,17533	0,01660	0,63398	0,72341	0,19940	-0,03913	0,02952	0,02733

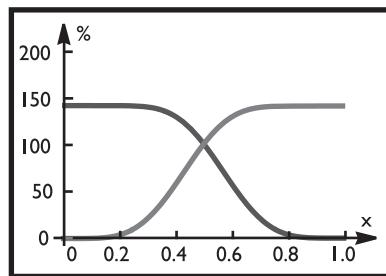


Рис. 9.17. АЧХ квадратурно-зеркального фильтра 10/10

Рассмотренные принципы расчета банка фильтров дискретного вейвлет-преобразования позволяют выбрать оптимальные варианты преобразований сигналов (в том числе и многомерных) по эффективности преобразований и требований к вычислительной мощности соответствующих цифровых систем.

Эти принципы расчета могут быть использованы также и для расчета фильтров многополосных вейвлет-преобразований.

9.4.4. Кратномасштабное дискретное вейвлет-преобразование изображений

Двумерное, как и одномерное, вейвлет-преобразование реализуется с помощью операции фильтрации и прореживания выборки. Обычно при преобразовании изображений используют разделимые масштабирующие и вейвлет-функции, и обработка двумерного сигнала $u(i, k)$ осуществляется сначала по строкам, а затем — по столбцам.

На рис. 9.18а приведена блок-схема такой процедуры. При анализе изображения сначала построчные отсчеты сигнала фильтруются двумя фильтрами (ФНЧ и ФВЧ) и производится децимация результатов фильтрации. Затем каждая из двух последовательностей полученных отсчетов, количество которых уменьшено в каждой строке в два раза, подаются на два аналогичных фильтра (ФНЧ и ФВЧ), на выходе которых выделяется четыре последовательности дискретных отсчетов, которые также подвергаются децимации.

Таким образом, после фильтрации двумерного сигнала с помощью двух ФНЧ и двойной децимации выделяется так называемая низкочастотно-низкочастотная информация, изображение которой в два раза меньше по горизонтали и вертикали по сравнению с исходным изображением (на рис. 9.18б обозначено буквами НН). После последовательной фильтрации с помощью фильтров ФНЧ–ФВЧ и децимации выделяется изображение, обозначенное на этом рисунке буквами НВ. После последовательной фильтрации с помощью фильтров ФВЧ–ФНЧ и децимации выделяется изображение, обозначенное на этом рисунке буквами ВН. И, наконец, после последовательной фильтрации с помощью двух фильтров ФВЧ и децимации выделяется изображение, обозначенное на этом рисунке буквами ВВ.

При синтезе сначала между всеми четырьмя дискретными отсчетами по столбцам вставляются нулевые отсчеты и восстанавливаются вертикальные последовательности. После соответствующего суммирования преобразованных составля-

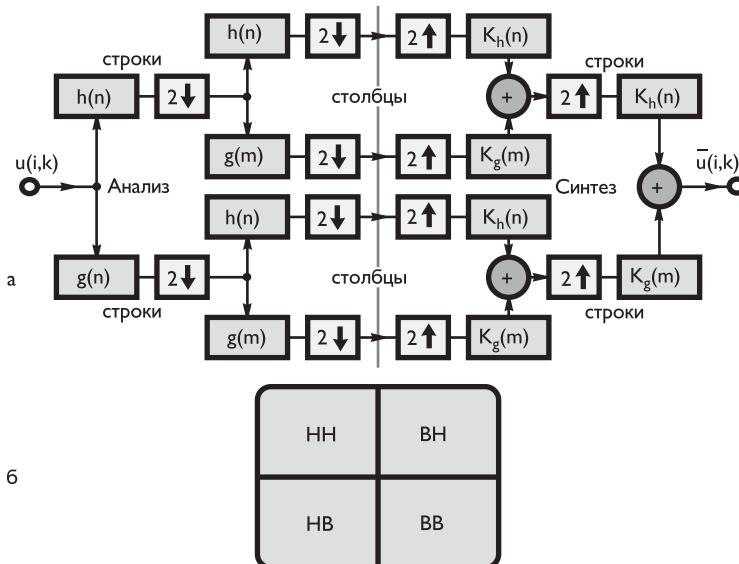


Рис. 9.18. Блок-схема вейвлет-преобразований двумерного сигнала (а) и формат преобразованного изображения (б)

ющих НН и НВ, а также ВН и ВВ по строкам между отсчетами вставляются нулевые отсчеты и восстанавливаются две горизонтальные последовательности. После их суммирования формируется двумерный сигнал $\hat{u}(i, k)$, который равен исходному сигналу $u(i, k)$ при выполнении условий, изложенных в разделах 9.4.1–9.4.3.

На рис. 9.19б приведено двумерное дискретное преобразование изображения «Залив» (рис. 9.19а). Отсчеты части изображения НН уменьшены в два раза, а отсчеты остальных частей для большей визуальной заметности увеличены в четыре раза и помещены на уровень «серого». На рис. 9.19в и г приведены соответственно двухмасштабное и трехмасштабное разложение изображения «Залив» соответственно.

9.4.5. Использование банков трехполосных КИХ-фильтров при обработке изображений

Рассмотренный в разделах 9.4.1–9.4.2 принцип обработки может быть использован также и для расчета фильтров многополосных вейвлет-преобразований.

В качестве примера приведем методику расчета характеристик трехполосных банков фильтров, содержащих фильтры нижних частот $H(x)$, средних частот $B(x)$ и верхних частот $G(x)$. В данном случае выбраны следующие соотношения:

- для фильтра нижних частот $h_n = h_{-n}$;
- для фильтра средних частот $b_n = -b_{-n}$, $b_0 = 0$;
- для фильтра верхних частот $g_n = g_{-n}$.

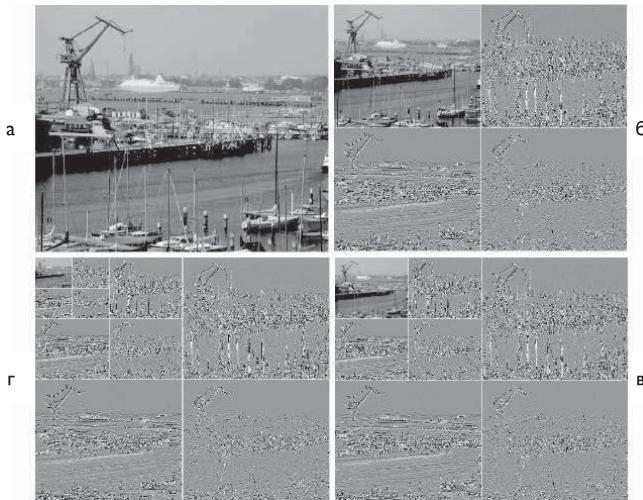


Рис. 9.19. Варианты двумерного разложения изображений

Следовательно:

$$\begin{aligned}
 H(x) &= h_0 + 2 \sum_{n=1}^N h_n \cdot \cos \pi n x; \\
 B(x) &= 2j \sum_{m=1}^M b_m \cdot \sin \pi m x; \\
 G(x) &= g_0 + 2 \sum_{k=1}^K g_k \cdot \cos \pi k x.
 \end{aligned} \tag{9.49}$$

Характеристики восстанавливющих фильтров обозначим соответственно:

$$\begin{aligned}
 K_h(x) &= \bar{h}_0 + 2 \sum_{n=1}^{\bar{N}} \bar{h}_n \cdot \cos \pi n x; \\
 K_b(x) &= 2j \sum_{m=1}^{\bar{M}} \bar{b}_m \cdot \sin \pi m x; \\
 K_g(x) &= \bar{g}_0 + 2 \sum_{k=1}^{\bar{K}} \bar{g}_k \cdot \cos \pi k x.
 \end{aligned} \tag{9.50}$$

В данном случае после каждого из трех входных фильтров при анализе производится децимация (обнуление) двух из каждого трех отсчетов, затем при синтезе эти нулевые отсчеты возобновляются и входной сигнал будет восстановлен при расчете характеристик восстанавливющих фильтров $K_h(x)$, $K_b(x)$ и $K_g(z)$ в соответствии с системой:

$$\begin{cases} H_\Theta(x) \cdot K_h(x) + B_\Theta(x) \cdot K_b(x) + G_\Theta(x) \cdot K_g(x) = 1, \\ H_\nabla(x) \cdot K_h(x) + B_\nabla(x) \cdot K_b(x) + G_\nabla(z) \cdot K_g(z) = 1, \\ H_\Delta(x) \cdot K_h(x) + B_\Delta(x) \cdot K_b(x) + G_\Delta(z) \cdot K_g(z) = 1, \end{cases} \tag{9.51}$$

где, как следует из рис. 9.20,

$$\begin{aligned}
 H_\Theta(x) &= h_0 + \sum_{n=1}^{[N/3]} h_{3n} (e^{j3\pi nx} + e^{-j3\pi nx}); \\
 H_\nabla(x) &= \sum_{n=1}^{[(N+2)/3]} h_{3n-2} e^{j(3n-2)\pi x} + \sum_{n=1}^{[(N+1)/3]} h_{3n-1} e^{-j(3n-1)\pi x}; \\
 H_\Delta(x) &= \sum_{n=1}^{[(N+1)/3]} h_{3n-1} e^{j(3n-1)\pi x} + \sum_{n=1}^{[(N+2)/3]} h_{3n-1} e^{-j(3n-2)\pi x}; \\
 B_\Theta(x) &= \sum_{n=1}^{[N/3]} b_{3n} (e^{j3\pi nx} + e^{-j3\pi nx}); \\
 B_\nabla(x) &= \sum_{n=1}^{[(N+2)/3]} b_{3n-2} e^{j(3n-2)\pi x} - \sum_{n=1}^{[(N+1)/3]} b_{3n-1} e^{-j(3n-1)\pi x}; \\
 B_\Delta(x) &= \sum_{n=1}^{[(N+1)/3]} b_{3n-1} e^{j(3n-1)\pi x} - \sum_{n=1}^{[(N+2)/3]} h_{3n-1} e^{-j(3n-2)\pi x}; \\
 G_\Theta(x) &= g_0 + \sum_{n=1}^{[N/3]} g_{3n} (e^{j3\pi nx} + e^{-j3\pi nx}); \\
 G_\nabla(x) &= \sum_{n=1}^{[(N+2)/3]} g_{3n-2} e^{j(3n-2)\pi x} + \sum_{n=1}^{[(N+1)/3]} g_{3n-1} e^{-j(3n-1)\pi x}; \\
 G_\Delta(x) &= \sum_{n=1}^{[(N+1)/3]} g_{3n-1} e^{j(3n-1)\pi x} + \sum_{n=1}^{[(N+2)/3]} g_{3n-2} e^{-j(3n-2)\pi x}.
 \end{aligned}$$

В табл. 9.4 и 9.5 приведены параметры нескольких банков трехполосных КИХ-фильтров.

В столбце 2 табл. 9.4 приводится значение параметра Z , рассчитанного в соответствии с условием:

$$Z = H(0) = B(x) \rightarrow \max = G(1).$$

На рис. 9.21 приведены АЧХ трехполосных КИХ-фильтров типа 3/3/3 (а) и 9/9/9 (б).

При применении трехполосных фильтров для обработки дискретных изображений вдоль и поперек строк изображение разбивается на девять частей, как показано на рис. 9.22:

- верхнее слева изображение подвергается низкочастотной фильтрации по горизонтали и вертикали (НН),
- следующие за ним по горизонтали части сначала подвергаются низкочастотной фильтрации по строкам, а затем среднечастотной (НС) и высокочастотной (НВ) фильтрации по столбцам,
- во втором сверху ряду сначала элементы изображения подвергаются по строкам среднечастотной фильтрации, а затем, по столбцам, — низкочастотной (СН), среднечастотной (СС) и высокочастотной (СВ) фильтрации,

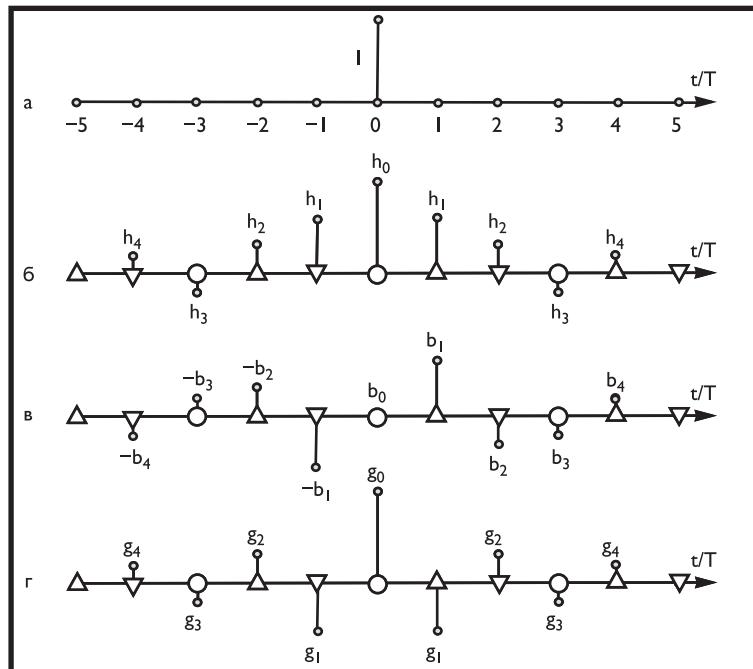


Рис. 9.20. Реакция фильтров низких (б), средних (в) и высоких (г) частот на дискретный отсчет входного сигнала (а)

Таблица 9.4. Параметры фильтров низких, средних и высоких частот при трехполосном вейвлет-преобразовании

Тип фильтра	Z	h_0 b_0 g_0	h_1 b_1 g_1	h_2 b_2 g_2	h_3 b_3 g_3	h_4 b_4 g_4	h_5 b_5 g_5
3/3/3	1,5874	0,73970 0 0,73970	0,39685 0,73970 -0,39685				
5/5/5	1,6361	1,02259 0 0,58856	0,40904 0,81807 -0,40904	-0,10226 -0,22039 0,11476			
7/7/7	1,8429	1,26670 0 0,57590	0,43193 0,74779 -0,43193	-0,17277 -0,29911 0,17277	0,02880 -0,04985 -0,02880		
7/7/11	1,8663	1,30505 0 0,56622	0,431916 0,75103 -0,43620	-0,18595 -0,31743 0,18815	0,03465 -0,03873 -0,03236	-0,00469	0,00200
9/9/9	1,8989	1,30222 0 0,54974	0,44634 0,77110 -0,43508	-0,17672 -0,35037 0,20048	0,02837 -0,02065 -0,03963	0,00032 -0,00210 -0,00063	

— в нижнем ряду сначала элементы изображения подвергаются по строкам высокочастотной фильтрации, а затем, по столбцам, — низкочастотной (ВН), среднечастотной (ВС) и высокочастотной (ВВ) фильтрации.

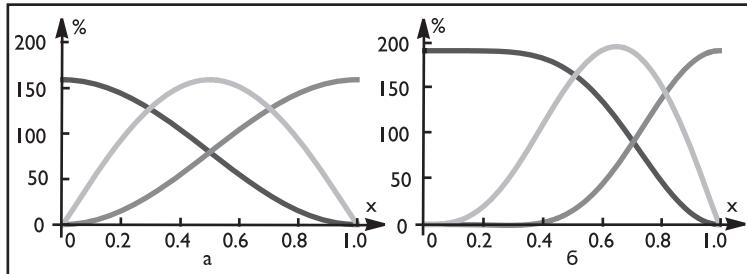


Рис. 9.21. АЧХ трехполосных КИХ-фильтров 3/3/3 (а) и 9/9/9 (б)

Таблица 9.5. Параметры восстанавливающих фильтров при трехполосном вейвлет-преобразовании

Тип фильтра	h_0 b_0 g_0	h_1 b_1 g_1	h_2 b_2 g_2	h_3 b_3 g_3	h_4 b_4 g_4	h_5 b_5 g_5	h_6 b_6 g_6	h_7 b_7 g_7	h_8 b_8 g_8
3/3/3	0,62996 0 0,62996	0,62996 0,62996 -0,62996							
5/5/5	0,61866 0 0,62417	0,48148 0,65902 -0,83656	0,12971 0,17754 -0,22536	-0,00374 -0,00511 0,00649					
7/7/7	0,54262 0 0,54262	0,43065 0,79600 -0,94744	0,17226 0,31840 -0,37898	0 0	-0,04306 -0,04306	-0,01723 -0,01723			
7/7/11	0,53576 0 0,53072	0,42245 0,81348 -0,98393	0,17240 0,34977 -0,42356	0,00004 -0,00181 -0,00255	-0,04120 0,00726 -0,04275	-0,01756 0,00311 -0,01820	-0,00001 -0,00018 -0,00016	-0,00008 -0,00006	
9/9/9	0,53050 0 0,56451	0,41425 0,81327 -1,01036	0,17103 0,36356 -0,46891	-0,00258 -0,01274 0,01942	-0,04070 -0,00434 -0,02835	-0,01802 -0,00136 -0,01359	0,00064 0,00005 0,00049	0,00007 0,00005	

На рис. 9.23 в качестве примера приведена такая обработка изображения «Залив». Здесь величина отсчетов НН изображения уменьшена в три раза, а всех остальных участков изображения увеличена в пять раз и располагается на уровне «серого».

HH	CH	BH
HC	CC	BC
HB	CB	BB

Рис. 9.22. Разбиение изображения при трехполосной фильтрации

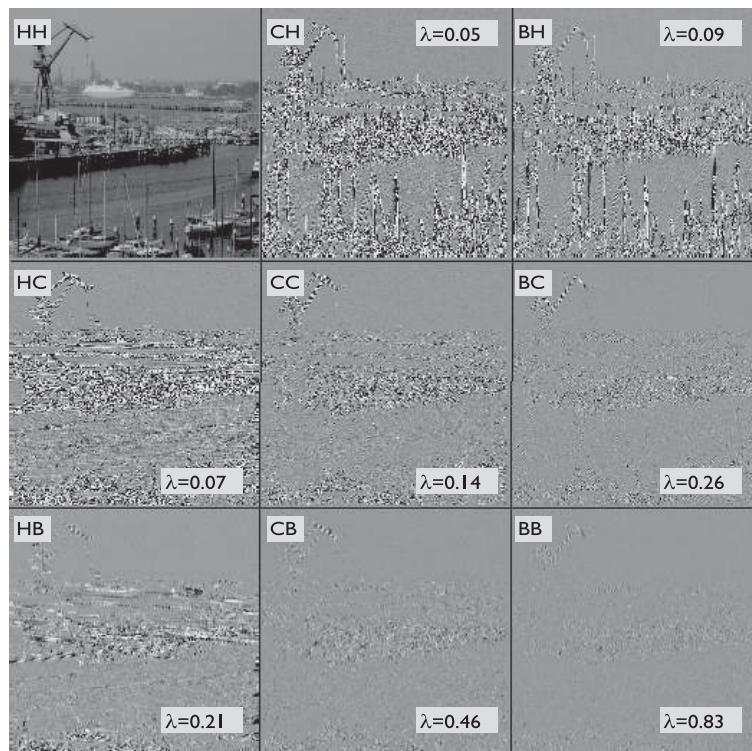


Рис. 9.23. Обработка изображения «Залив» с помощью трехполосного банка фильтров

Литература к части III

- 3.1. Цифровая обработка телевизионных и компьютерных изображений. 2-е изд., перераб. и доп. / Под ред. Ю.Б. Зубарева и В.П. Дворковича. М.: НАТ, 1997.
- 3.2. *Oliver B.M., Pierce J.R. and Shannon C.E.* The Philosophy of PCM// Proceedings of IRE. V. 36. Oct., 1948.
- 3.3. *Goodall W.M.* Television by Pulse Code Modulation// Bell System Technical Journal. V. 30. Jan., 1951.
- 3.4. *Катермоул К.В.* Принципы импульсно-кодовой модуляции/ Пер. с англ. под ред. В.В. Маркова. М.: Связь, 1974.
- 3.5. Цифровое кодирование телевизионных изображений/ Под ред. И.И. Цукермана. М.: Связь, 1981.
- 3.6. *Игнатьев Н.К.* Дискретизация и ее приложения. М.: Связь, 1980.
- 3.7. Справочник по телевизионной технике. Т. 1–2 / Пер. с англ. под ред. С.И. Катаева. М.–Л.: Госэнергоиздат, 1962.
- 3.8. Телевизионная техника: Справочник/ Под общ. ред. Ю.Б. Зубарева и Г.Л. Глориозова. М.: Радио и связь, 1994.
- 3.9. *Пэдхем Ч., Сродерс Д.* Восприятие света и цвета. М.: Мир, 1978.
- 3.10. *Bhushan A.K.* Efficient Transmission and Coding of Colour Components. M.S. Thesis/ Massachusetts Institute of Technology, Cambridge, MA. June, 1977.
- 3.11. *Frei W., Jaeger P.A. and Probst P.A.* Quantization of Pictoral Color Information// Nachrichtentech. Z. V. 61. 1972.
- 3.12. *Marti B.* Preliminary Processing of Color Images// CCETT ATA/T/3/73. Sept.5, 1973.
- 3.13. *Stenger L.* Quantization of TV Chrominance Signals Considering the Visibility of Small Color Differences // IEEE Trans. on Communications. V. COM-25, 11. Nov., 1977.
- 3.14. *Harrison C. W.* Experiments with Linear Prediction in Television// Bell System Technical Journal. V. 31. July, 1952.
- 3.15. *Habibi A.* Comparison of Nth order DPCM encoder with Linear Transformations and Block Quantization Techniques// IEEE Trans. on Communication Technology. V. COM-19. Dec., 1971.
- 3.16. *Graham R.E.* Predictive Quantizing of Television Signals// IRE Wescon Convention Record. V. 2, pt. 4. 1958.
- 3.17. *Netravali A.N., Haskell B.G.* Digital pictures: Representation and Compression. Plenum Press, N.Y., 1991.
- 3.18. *Прэтт У.* Цифровая обработка изображений. Т. 1–2. М.: Мир, 1982.
- 3.19. *Погрибной В.А.* Дельта-модуляция в цифровой обработке сигналов. М.: Радио и связь, 1990.

- 3.20. Венедиктов М.Д., Женевский Ю.П., Марков В.В., Эйдус Г.С. Дельта-модуляция. Теория и применение. М.: Связь, 1976.
- 3.21. Стил Р. Принципы дельта-модуляции. М.: Связь, 1979.
- 3.22. Geddes W.K.E. Picture Processing by Quantization of the Time Derivative// BBC Eng. Div., Techn. Rep., T-114. 1963.
- 3.23. Коннер Д., Брейнард Р., Лимб. Дж. Внутрикадровое кодирование изображений для передачи по цифровому каналу// В. кн.: Обработка изображений при помощи цифровых вычислительных машин/ Пер. с англ. под ред. Д.С. Лебедева. М.: Мир, 1973.
- 3.24. Musmann H.G. Predictive Coding// Chapter in the book Image Transmission Techniques, edited by W.K. Pratt, Academic Press, 1979.
- 3.25. Umeda M., Omori T. Quantizing Characteristics of DPCM for Video Signal// J. Inst. Telev. Japan. V. 28, № 8. 1972.
- 3.26. Umeda M. Video Signal Quantization by Differential. PCM Technique// Rev. Electr. Commun. Lab. V. 22, № 11. 1974.
- 3.27. Sabatier J. Le codage differential de composantes du signaux de television couleur// Acta Electronica. V. 19, № 3. 1976.
- 3.28. Haskell B.G. Entropy Measurements for Nonadaptive and Adaptive, Frame-to-Frame, Linear Predictive Coding of Video Telephone Signals// Bell Systems Technical Journal. V. 54, № 6. August, 1975.
- 3.29. Netravali A.N. and Rubinstein C. Quantization of Color Signals// Proc. IEEE, № 8. 1977.
- 3.30. Max J. Quantizing for Minimum Distortion// IEEE Trans. V. JT-6, № 3. 1960.
- 3.31. Kretz F. Subjectively Optimal Quantization of Pictures// IEEE Trans. on Commun. № 11. 1975.
- 3.32. Kretz F., Boudeville J., Sallio P. Qualite subjective en television numerique. Deuxieme partie: Criteres objectifs de qualite et exemple d'optimisation d'un codage numerique// Revue de radiodissusion-television. № 52. 1978.
- 3.33. Kretzmer E.R. Reduced Alphabet Representation of Television Signals// IRE Nat. Conv.
- 3.34. Цифровое телевидение/ Под ред. М.И. Кривошеева. М.: Связь, 1980.
- 3.35. Харатишвили Н.Г. Цифровое кодирование с предсказанием непрерывных сигналов. М.: Радио и связь, 1986.
- 3.36. Ярославский Л.П. Введение в цифровую обработку изображений. М.: Сов. радио, 1979.
- 3.37. Залманзон Л.А. Преобразования Фурье, Уолша, Хаара и их применение в управлении, связи и других областях. М.: Наука, 1989.
- 3.38. Ахмед Н., Rao K.R. Ортогональные преобразования при обработке цифровых сигналов. М.: Связь, 1980.
- 3.39. Оппенгейм А.В., Шафер Р.В. Цифровая обработка сигналов. М.: Связь, 1979.



- 3.40. *Andrews H.C.* Computer Techniques in Image Processing// Academic Press., N.J. 1970.
- 3.41. *Andrews H.C.* Two-Dimensional Transforms// Topics in Applied Physics: Picture Processing and Digital Filtering/V/ 6/ *Huang T.*, Ed., Springer Verlag, N.J. 1975.
- 3.42. *Bellman R.* Introduction to Matrix Analysis// McGraw-Hill, N.J. 1960.
- 3.43. *Andrews H.C., Caspary K.* Generalized Technique for Spectral Analysis// IEEE Trans. Computers, C-19. January, 1970.
- 3.44. Обработка изображений и цифровая фильтрация/ Под ред. Т. Хуанга; пер. с англ. Е.З. Сороки и В.А. Хлебородова. М.: Мир, 1979.
- 3.45. *Anderson G.B., Huang T.S.* Piecewise Fourier Transformation for Picture Bandwidth Compression// IEEE Trans. V. COM-19, № 26. 1971.
- 3.46. *Блейхут Р.* Быстрые алгоритмы цифровой обработки сигналов/ Пер. с англ. И.И. Грушко. М.: Мир, 1989.
- 3.47. Быстрые алгоритмы в цифровой обработке изображений. Преобразования и медианные фильтры/ Под ред. Т.С. Хуанга; пер. с англ. под ред. Л.П. Ярославского. М.: Радио и связь, 1984.
- 3.48. *Нуссбаумер Г.* Быстрое преобразование Фурье и алгоритмы вычисления сверток/ Пер. с англ. под ред. В.М. Амербаева и Т.Э. Кренкеля. М.: Радио и связь, 1985.
- 3.49. *Гольденберг Л.М., Матюшкин Б.Д., Поляк М.Н.* Цифровая обработка сигналов: Справочник. М.: Радио и связь, 1985.
- 3.50. *Brigham E.O.* The Fast Fourier Transform// Prentice-Hall, Englewood Cliffs, N.J. 1984.
- 3.51. *Haar A.* Zur Theorie der Orthogonalen Funktionen-System: Inaugural Dissertation. Vfth. Annalen, 5, 1955.
- 3.52. *Прэтт У., Эндрюс Х., Кайн Дж.* Кодирование изображений с использованием преобразования Адамара/ Пер. с англ.// ТИИЭР. Т. 56. № 1. 1969.
- 3.53. *Райзер Г.Дж.* Комбинаторная математика/ Пер. с англ. М.: Мир, 1966.
- 3.54. *Хармут Х.Ф.* Передача информации ортогональными функциями/ Пер. с англ. М.: Связь, 1975.
- 3.55. *Jain A.K.* A Fast Karhunen-Loeve Transform for Finite Discrete Images: Proceedings of the National Electronics Conference. Chicago, Illinois. October, 1974.
- 3.56. *Means R.W., Whitehouse H.J., Speiser J.M.* Television Encoding Using a Hybrid Discrete Cosine Transform and a Differential Pulse Code Modulator in Real Time: Proc. of the National Telecom. Conf. San Diego, California. Dec., 1974.
- 3.57. *Ahmed N., Natarajan T., Rao K.R.* On Image Proceeding and a Discrete Cosine Transform // IEEE Trans. Computers, C-23, 1. Jan., 1974.
- 3.58. *Wintz P.A.* Transform Picture Coding// Proc. IEEE. V. 60. July, 1972.
- 3.59. *Лоули Д., Максвелл А.* Факторный анализ как статистический метод. М.: Мир, 1967.

- 3.60. Обработка изображений при помощи ЦВМ/ Пер. с англ. под ред. Д.С. Лебедева. М.: Мир, 1973.
- 3.61. *Enomoto H., Shibata K.* Orthogonal Transform Coding System for Television Signals// IEEE Trans. Electromagnetic Compatibility, EMC-13, 3. Aug. 1971.
- 3.62. *Pratt W.K., Welch L.R., Chen W.H.* Slant Transform for Image Coding// Proceedings Symposium on Applications of Walsh Functions. March, 1972.
- 3.63. *Pratt W.K., Chen W.H., Welch L.R.* Slant Transform Image Coding// IEEE Trans. Commun. COM-22, 8. Aug., 1974.
- 3.64. *Netravali A.N. and Rubinstein C.* Quantization of Color Signals// Proc. IEEE. № 8. 1977.
- 3.65. *Max J.* Quantizing for Minimum Distortion// IEEE Trans. V. JT-6. № 3. 1960.
- 3.66. *Huang J., Schultheiss P.M.* Block Quantization of Correlated Gaussian Random Variables// IEEE Trans. 1963.
- 3.67. *Lohscheller H.* A Subjectively Adapted Image Communication System// IEEE Trans. Commun. COM-32. Dec., 1984.
- 3.68. МККР. Параметры кодирования сигналов цифрового телевидения для студий: Рекомендация 601.
- 3.69. *Wallace G.K.* The JPEG Still Picture Compression Standard// Communication of the ACM. V. 34. № 4. April, 1991.
- 3.70. *King N. Ngan, Hee C. Koh., Wai C. Wong.* Hybrid Image Coding Scheme Incorporating Human Visual System Characteristics// Optical Engineering. V. 30. № 7. 1991.
- 3.71. *Natarajan T.R. and Ahmed N.* On Interframe Transform Coding// IEEE Trans. Commun. V. COM-25. № 11. Nov., 1977.
- 3.72. *Knauer S.C.* Real-Time Video Compression Algorithm for Hadamard Transform Processing// IEEE Trans. Electromagnetic Compatibility. V. EMC-18. № 1. Feb., 1976.
- 3.73. *Habibi A.* Hybrid Coding of Pictoral Data// IEEE Trans. Communications. V. COM-22. № 5. May, 1974.
- 3.74. *Roese J.A.* Hybrid Transform/ Predictive Image Coding// Charter 5 of Image Transmission Techniques/ W.K.Pratt ed. Academic Press, N.Y., 1979.
- 3.75. *Clarke R.J.* Hybrid Intraframe Transform Coding of Image Data// IEE Proc. V. 131 part F. № 1. Feb., 1984.
- 3.76. *King N. Ngan, Hee C. Koh, Wai C. Wong.* Hybrid Image Coding Scheme Incorporating Human Visual System Characteristics// Optical Engineering. V. 30. № 7. 1990.
- 3.77. *Gray R.M.* Vector Quantization ASSP Magazine. Nov. 1984
- 3.78. *Linde Y. Buzo A. and Gray R.* An Algorithm for Vector Quantization Designs// IEEE Trans. Commun. COM-28. Jan., 1980.
- 3.79. *Equitz W.H.* Fast Algorithms for Vector Quantization Picture Coding// M.S.Thesis, MIT. June, 1984.



- 3.80. Gersho A. Ramamurthi B. Image Coding using Vector Quantization: Proc. of Int. Conf. ASSP. Paris, 1982.
- 3.81. Proceedings of the 2-nd Data Compression Conference (March 1992 Snowbird Utah USA)// IEEE Computer Society Press, 1992 - XIII.
- 3.82. *Sproull R.F., Sutherland I.E.* A Comparison of Codebook Generation Techniques for Vector Quantization, in the Proceedings of the 2-nd Data Compression Conference (Mar. 92 Snowbird Ut)// IEEE Computer Society Press, 1992 - XIII.
- 3.83. *Wu X.* Vector Quantizer Design by Constrained Global Optimization, in the Proceedings of the 2-nd Data Compression Conference (Mar. 92 Snowbird Ut)// IEEE Computer Society Press, 1992 - XIII.
- 3.84. *Musmann H.G., Pirsch P. and Grallert H.-J.* Advances in Picture Coding// Proceedings of IEEE. April, 1975.
- 3.85. *Burt P.J., Adelson E.H.* The Laplacian Pyramid as a Compact Image Code// IEEE Trans. Commun. V. COM-31, 1983.
- 3.86. *Baaziz N., Labit C.* Pyramid Transform for Digital Images// IRISA, Campus de Beaulieu. № 526. 1990.
- 3.87. *Adelson E.H., Simoncelli E.* Orthogonal Pyramid Transforms for Image Coding// SPIE. V. 45. 1987.
- 3.88. *Barnsley M., Sloan A.* Method and Apparatus for Processing Digital Data/ US Patent #5065447
- 3.89. *Barnsley M., Hurd L.P.* Fractal Image Compression. A.K.Peters Ltd. Wellesley Massachusetts, 1993.
- 3.90. *Monroe D.M., Dudbridge F.* Fractal Block Coding of Images// Electronics Letters. V. 28. № 11. 1992
- 3.91. *Fisher Y.* Fractal Image Compression/ SIGGRAPH'92 Course Notes
- 3.92. *Mandelbrot B.* Les Objets Fractals: Form, Hasard et Dimension Flammarion. Paris, 1975.
- 3.93. *Mandelbrot B.B.* The Fractal Geometry of Nature. W.H.Freeman, N. Y., 1983.
- 3.94. *Barnsley M.* Fractals Everywhere. Academic Press Boston, 1988; San Diego, 1989; San Diego, 1993.
- 3.95. *Федер Е.* Фракталы. М.: Мир, 1991.
- 3.96. *Barnsley M.F., Sloan A.D.* A Better Way to Compress Images// BYTE. V. 13. 1988.
- 3.97. *Ансон Л., Барнсли М.* Фрактальное сжатие изображений// Мир ПК. № 4. 1992.
- 3.98. *Anson L.F.* Fractal Image Compression// BYTE. V. 18. № 11. 1993.
- 3.99. *Воробьев В.И., Грибунин В.Г.* Теория и практика вейвлет-преобразования. ВУС. Сю-Петербург, 1999.
- 3.100. *Петухов А.П.* Введение в теорию базисов всплесков: Учеб. пособие. СПб.: СПбГТУ, 1999.

- 3.101. *Добеши И.* Десять лекций по вейвлетам/ Пер. с англ. Мищенко Е.В.; под ред. А.П. Петухова. М.: РХД, 2001.
- 3.102. *Primer A.* Introduction to Wavelets and Wavelet Transforms. Prentice Hall. Upper Saddle River, New Jersey 07458. <http://www.prenhall.com>
- 3.103. *Дьяконов В.П.* Вейвлеты. От теории к практике. М.: Солон-Р., 2002.
- 3.104. *Блаттер К.* Вейвлет-анализ. Основы теории/ Пер. с нем. Т.Э. Кренкеля; под. ред. А.Г. Кюркчана, М.: Техносфера, 2006.
- 3.105. *Шарк Г.-Г.* Применение вейвлетов для ЦОС/ Пер. с англ. Смирновой Н.И.; под. Ред. Кюркчана А.Г. М.: Техносфера, 2007.
- 3.106. *Гонсалес Р., Будс Р.* Цифровая обработка изображений/ Пер. с англ. под ред. Чичиа П.А. М.: Техносфера, 2005.
- 3.107. *Давыдов А.В.* Вейвлеты. Вейвлетный анализ сигналов/ Персональный сайт проф. Давыдова. <http://prodav.narod.ru/wavelet/index.html>
- 3.108. *Дворкович В.П., Дворкович А.В.* Расчет банков фильтров дискретного вейвлет-преобразования и анализ их характеристик // ЦОС. № 2. 2006.
- 3.109. *ITU-T Recommendation.* T.800// Information Technology — JPEG 2000 Image Coding System: Core Coding System, 2004.
- 3.110. *ITU-T Recommendation* T.801/ Information Technology — JPEG 2000 Image Coding System: Extensions, 2004.

ЧАСТЬ IV

ВНУТРИКАДРОВАЯ И МЕЖКАДРОВАЯ ОБРАБОТКА ИЗОБРАЖЕНИЙ

Введение

Повышение визуального качества восстановленного изображения при заданном коэффициенте сжатия или повышение коэффициента сжатия при заданном визуальном качестве восстановленного изображения для статических изображений и опорных кадров в динамической последовательности изображений связано с изменением коэффициентов квантования отдельных их блоков или преобразований этих блоков и влияет на эффективность всего алгоритма компрессии в целом.

Основная идея повышения качества внутрикадрового кодирования изображений состоит в подборе коэффициентов квантования блоков таким образом, чтобы визуальное качество воспроизведения блоков по всему изображению было примерно одинаковым. При применении же одинакового для всех блоков коэффициента квантования, что зачастую используется в упрощенных кодирующих устройствах, качество восстановленного изображения в блоках существенно зависит от содержания блоков (структуры данного участка изображения).

Таким образом, требуется выработать критерий оценки искажений визуального качества — допустимых искажений в данном блоке при заданном параметре качества, причем этот критерий должен хорошо согласовываться с визуальным восприятием искажений.

Наличие жесткой взаимосвязи между коэффициентом сжатия информации об изображении и используемым коэффициентом квантования определяет недостатки покадрового сжатия изображений при использовании алгоритмов обработки по стандарту JPEG [4.1–4.3]. Это приводит к тому, что при насыщенности

изображения мелкими деталями на ровном поле возможно получить высокое качество восстановленного изображения только при малых коэффициентах сжатия и при больших потоках цифровых данных в случае передачи динамических изображений. При больших коэффициентах сжатия визуальное качество восстанавливаемых изображений оказывается неудовлетворительным: мелкие детали изображений не воспроизводятся, в областях переходов яркости и цвета возникают сильно искажающие изображение эхо-сигналы, возникает так называемый блокинг-эффект.

Эффективным методом повышения качества воспроизведения изображений является направленное пространственное предсказание при внутрикадровом кодировании. Эта методика повышает качество сигнала, применяемого для предсказания, а также позволяет использовать для предсказания соседние области, которые были закодированы не с помощью внутрикадровой методики.

Деблокинговая адаптивная фильтрация в цикле кодирования и декодирования – хорошо известный метод повышения качества восстановленного видео, и при правильном проектировании может повысить как субъективное, так и объективное качество.

При обзоре различных методов кодирования следует упомянуть также о методах, использующих преобразование изображений на несколько составляющих меньшего разрешения (*multiresolution representation*).

Один из вариантов расщепления, рассмотренный в части III, носит название «пирамида Лапласа» [4.4–4.6].

Наиболее эффективный метод представления изображений в виде набора составляющих меньшего разрешения основан на использовании так называемого вейвлет-разложения [4.7, 4.8]. Особенno большое развитие получила практика применения вейвлет-преобразования для решения задач сжатия и обработки изображений, распознавания образов и др. Дискретизация вейвлет-преобразования, изложенная И. Добеши [4.9], перекинула мост между математиками и разработчиками систем цифровой обработки сигналов. Вейвлеты составляют основу мощного метода обработки сигналов – теории кратномасштабного анализа, реализующей представление и анализ сигналов, в частности изображений, в различных масштабах, т. е. при различных разрешениях.

Кратномасштабный анализ изображений реализуется путем свертки видеосигналов с импульсными характеристиками ряда фильтров и последующей децимацией результатов этих преобразований.

Совокупность набора фильтров с дециматорами называется банком фильтров, реализующих анализ исходного сигнала. Набор сформированных в результате преобразований сигналов несет в себе информацию о спектральных составляющих исходного сигнала при некотором пространственном (временном) масштабе.

При синтезе сигнала (его реконструкции) выполняется прореживание субполосных сигналов, их фильтрация соответствующими восстанавливающими фильтрами и сложение.

У обработки изображений с использованием вейвлетов [4.10–4.12] много общего с кодированием изображений с помощью рассмотренных в части III преобразований.

В изображениях имеются детали самых различных размеров, и, следовательно, желательно, чтобы преобразование позволяло анализировать изображение на *различных масштабах*, что реально обеспечивается при кратномасштабном анализе. При этом желательно, чтобы преобразование делило входной сигнал на локальные частотные области, обеспечивая способность корректно анализировать *ориентированные типичные для изображения структуры*, такие как контуры, линии и др.

Кроме частотной локализации желательно, чтобы преобразование обеспечивало *пространственную локализацию*, что особенно важно, когда информация о местоположении деталей изображения является существенной. Локальность не должна быть блочной, как при использовании, например, ДКП, поскольку это ведет к потере свойства локальности в частотной области.

Вейвлеты являются примером функций, хорошо локализованных как в частотной, так и в пространственной областях.

ГЛАВА 10

ВНУТРИАДРОВОЕ КОДИРОВАНИЕ ИЗОБРАЖЕНИЙ

Известно, что субъективная чувствительность глаза к уровню яркости участка изображения зависит от структуры соседних участков изображения: чем больше они насыщены мелкими деталями и чем больше их контраст, тем менее заметными оказываются ошибки, возникающие при кодировании изображения с потерей информации. Поскольку структурные свойства изображений неоднородны, в большинстве обрабатываемых изображений можно найти участки как с более, так и с менее выраженной заметностью ошибок кодирования. Применение единого по всем блокам квантования коэффициентов какого бы то ни было группового кодирования приводит в этих условиях к тому, что в некоторых местах восстановленного изображения проявляются сильно заметные искажения, в то время как в иных местах можно было бы допустить и большие ошибки без ущерба для визуального качества.

10.1. Изменения коэффициентов квантования блоков в JPEG-подобных алгоритмах видеокомпрессии

Для уменьшения искажений изображений при использовании JPEG-подобных алгоритмов видеокомпрессии необходимо изменять от блока к блоку коэффициент глубины квантования, на который умножаются коэффициенты матрицы квантования, либо выбирать матрицу коэффициентов квантования исходя из особенностей каждого блока.

Основной проблемой при использовании этого резерва улучшения характеристик сжатия изображения является то, что в данном случае связь визуального качества восстановленного изображения и его отличий от исходного в смысле формально выражаемых величин (например, среднеквадратичной ошибки) перестает быть однозначной.

Данная проблема решается с помощью формирования поля допустимых искажений (допусков), где каждому пикселу исходного изображения соответствует значение максимально допустимой погрешности передачи его значения.

В [4.13, 4.14] предлагаются два варианта определения допустимых изменений уровней каждого пикселя изображения при анализе рельефа обрабатываемых блоков изображения и выборе коэффициентов квантования — оценка модуля градиента яркости каждого пикселя и вычисление максимума вариаций уровней

в некоторой задаваемой окрестности каждого пикселя:

$$\Delta A_{\max i,j} = \Delta A_0 + k \frac{1}{2} \sqrt{(X_{i+1,j} - X_{i-1,j})^2 + (X_{i,j+1} - X_{i,j-1})^2}, \quad (10.1)$$

$$\Delta A_{\max i,j} = \Delta A_0 + k \max_{di,dj=-r,r} |X_{i+di,j+dj} - X_{i,j}|, \quad (10.2)$$

где $\Delta A_{\max i,j}$ — допустимая ошибка величины яркости пикселя $X_{i,j}$; $\Delta A_0, k$ — эмпирические константы.

Кодер определяет такие параметры квантования каждого блока коэффициентов ДКП, чтобы ошибки в восстановленном изображении были максимальны, но не превышали допустимой величины: $|\tilde{X}_{i,j} - X_{i,j}| \leq \Delta A_{\max i,j}$.

После двумерного дискретного косинусного преобразования и квантования элементов образов блоков компонент изображения производится восстановление искаженного изображения, вычисление разностей исходного и восстановленного изображений, сравнение этих разностей с формируемыми допустимыми отклонениями значений каждого из пикселов компонент изображения. По результатам сравнения в случае, если изменения значений пикселов превышают допустимые отклонения или значительно меньше допусков, производится итерационное изменение коэффициента квантования, обеспечивающее необходимое уменьшение или увеличение ошибок преобразования.

В том случае, когда требуется уменьшить количество операций при обработке изображения, вычисление разностей исходного и восстановленного изображений можно производить не по всем пикселям блока, а только по нескольким, имеющим наименьшие допуска, и только по этим значениям производят вычисление требуемого коэффициента квантования.

Недостатками подобного способа являются необходимость определения коэффициента квантования спектральных компонент ДКП каждого блока итерационным методом, что резко увеличивает объем вычислительных операций, и отсутствие возможности адаптации алгоритма цифровой обработки к пропускной способности используемого канала цифровой передачи информации. В связи с этим был предложен усовершенствованный алгоритм [4.14].

Технический результат, который реализуется при осуществлении этого решения, выражается в повышении скорости цифровой обработки изображений и повышении визуального качества восстановленного изображения при заданном коэффициенте сжатия или повышении коэффициента сжатия при заданном визуальном качестве восстановленного изображения, в реализации коэффициента сжатия изображения, обеспечивающего его передачу в канале с заданной пропускной способностью.

На рис. 10.1 изображен вариант структурной схемы устройства для реализации такой цифровой обработки изображений. На рис. 10.2 приведена типичная зависимость ошибки восстановления от коэффициента квантования блока изображения.

Поясним принцип работы устройства на рис. 10.1. Кадры цветного изображения обычно представляют собой либо двумерные массивы скалярных компонент при дискретизации полного цветового сигнала, кодируемого, например, по системам SECAM, PAL, NTSC, либо двумерный массив векторов, состоящий из трех скалярных компонент, соответствующий величинам цветовых составляющих R, G, B.

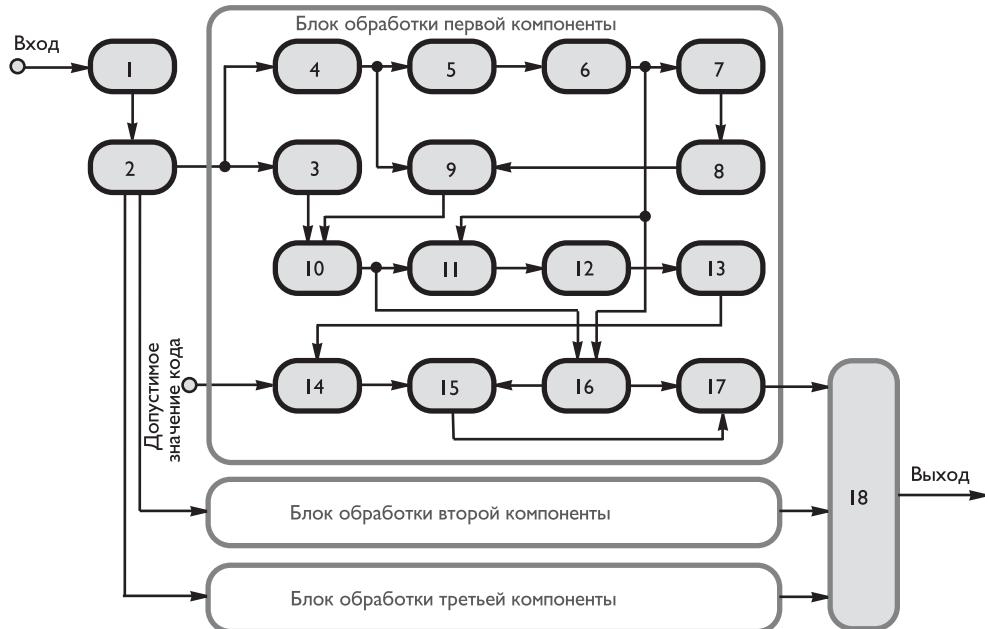


Рис. 10.1. Структурная схема устройства для реализации адаптивного квантования: 1 — блок цифровой памяти кадров изображений, 2 — блок разделения кадра на компоненты, 3 — формирователь допусков, 4 — схема памяти блоков, 5 — преобразователь ДКП, 6 — схема матрицирования, 7 — первая схема квантования и округления, 8 — преобразователь ОДКП, 9 — вычислитель ошибок, 10 — вычислитель коэффициентов квантования, 11 — вторая схема квантования и округления, 12 — схема оценки кода каждого блока кадра, 13 — сумматор кодов оценки, 14 — схема сравнения, 15 — перемножитель, 16 — схема памяти, 17 — схема квантования и округления данных всех блоков кадра, 18 — блок кодирования

Этот массив данных запоминается в блоке 1 цифровой памяти кадров изображений и через блок 2, в котором создаются компоненты изображения (обычно, яркостная составляющая Y и цветоразностные сигналы U, V), поступающие на соответствующие блоки их цифровой обработки, по структурной схеме не отличающиеся друг от друга.

В одном из таких блоков, например обработки составляющей яркости, с помощью формирователя 3 допусков создается рельеф допустимых отклонений значений пикселов каждого кадра.

Один из известных эффективных вариантов вычисления такого рельефа определяется соотношением:

$$G(i, j) = K_0 \sum_{m=-r}^{m=r} \sum_{n=-r}^{n=r} a(m, n) \cdot \{ [X(i + m + 1, j + n) - X(i + m - 1, jn)]^2 + [X(i + m, j + n + 1) - X(i + m, j + n - 1)]^2 \}^{1/2}, \quad (10.3)$$

где $X(i, j)$ — значение пикселя в точке (i, j) , $a(m, n) = 1/\text{round}(\sqrt{m^2 + n^2})$ — коэффициенты взвешивания, K_0 — коэффициент пропорциональности.

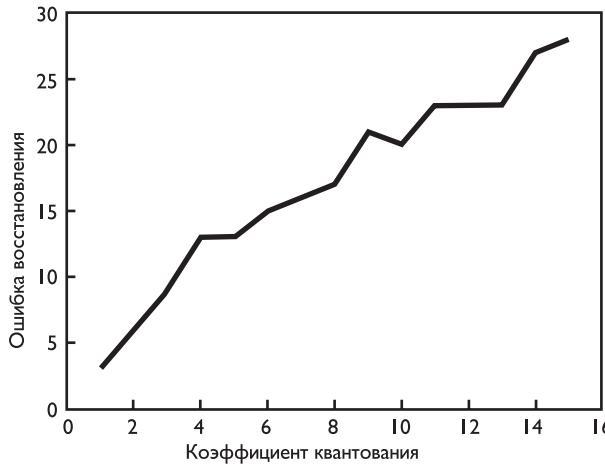


Рис. 10.2. Типичная зависимость ошибки восстановления от коэффициента квантования

Сформированные таким образом допустимые отклонения определяют возможные изменения величин пикселов компоненты при ее восстановлении после преобразования, обеспечивающего сжатие изображений.

Выделенные схемой 4 коды каждого блока размером $N \times N$ пикселов (например, согласно стандарту JPEG — 8×8) проходят последовательную обработку в нескольких блоках. Сначала данные поступают на преобразователь ДКП 5, формирующий из дискретных значений пикселов $X(i, j)$, $0 \leq i, j \leq N$, спектральные компоненты дискретного косинусного преобразования блока $F(u, v)$, $0 \leq u, v \leq N$. Затем они проходят схему 6 матрицирования, в которой каждая компонента ДКП делится на соответствующее число $Q(u, v)$, учитывающее влияние визуальной заметности изменений различных спектральных компонент ДКП. Далее следуют первая схема 7 квантования и округления, формирующая преобразованные спектральные компоненты, $\text{round}[(1/K^*) \cdot F(u, v)/Q(u, v)]$, где K^* — заранее заданный коэффициент квантования, и преобразователь ОДКП, формирующий из «восстановленных» спектральных компонент косинусного дискретного преобразования:

$$F(u, v) = K^* \cdot Q(u, v) \cdot \text{round}[(1/K^*) \cdot F(u, v)/Q(u, v)], \quad (10.4)$$

измененные вследствие обработки значения пикселов компоненты $\bar{X}(i, j)$. После этого следует модуль вычислителя ошибок 9, на второй вход которого поступают исходные значения уровней пикселов этого же блока.

В результате на выходе вычислителя 9 формируются отклонения преобразованных вследствие матрицирования и квантования с заданным коэффициентом K^* значений пикселов от исходных величин: $\Delta(i, j) = |\bar{X}(i, j) - X(i, j)|$.

Ввиду того, что величина этих отклонений значений пикселов практически линейно зависит от коэффициента квантования K^* в достаточно большом диапазоне его значений (см. рис. 10.2), на основании полученных данных можно рассчитать требуемый коэффициент квантования $K_{\text{доп}}$, обеспечивающий такое преобразование пикселов блока, что изменения их значений будут находиться в зонах допустимых отклонений $\Delta_{\text{доп}}(i, j) = G(i, j)$, формируемых в схеме 3.

Расчет величины коэффициента квантования $K_{\text{Доп}}$ в блоке 10 может осуществляться различным образом. Один из вариантов расчета связан с определением средних значений реальных и допустимых отклонений в блоке:

$$\Delta = \frac{1}{N^2} \sum_{i=0}^{i=N-1} \sum_{j=0}^{j=N-1} \Delta(i, j), \Delta_{\text{Доп}} = \frac{1}{N^2} \sum_{i=0}^{i=N-1} \sum_{j=0}^{j=N-1} \Delta_{\text{Доп}}(i, j), \quad (10.5)$$

и по этим значениям вычисляют величину $K_{\text{Доп}}$ по формуле $K_{\text{Доп}} = K \cdot \Delta_{\text{Доп}} / (\Delta + \lambda)$, где $\lambda < \Delta_{\text{Доп}}$ — некоторая малая величина.

Другой вариант, требующий меньше вычислительных операций, связан с определением величин $\Delta_{\text{Доп}}$ и Δ только путем усреднения нескольких минимальных значений среди $\Delta_{\text{Доп}}(i, j)$ и соответствующих им по положению в блоке значений $\Delta(i, j)$.

Вычисленный блоком коэффициент квантования используется для вычислений во второй схеме квантования и округления 11 спектральных компонент ДКП каждого блока кадра:

$$L(u, v) = \text{round}[(1/K_{\text{Доп}}) \cdot F(u, v)/Q(u, v)], \quad 0 \leq u, v \leq N. \quad (10.6)$$

По значениям этих величин каждого из M блоков кадра в схеме 12 производится Z-упорядочивание спектральных компонент ДКП и оценка величины кода (например, кода Хаффмана) $\text{Size}(m)$, $1 \leq m \leq M$, определяющего цифровой поток каждого блока.

Коды $\text{Size}(m)$ поступают на сумматор 13, где накапливаются величины кодов всех блоков кадра $\text{Size}_{\Sigma} = \sum_{m=1}^M \text{Size}(m)$, и далее — на схему сравнения 14, в которой значение Size_{Σ} сравнивается с допустимым значением кода $\text{Size}_{\Sigma\text{Доп}}$.

Если величина $q = \text{Size}_{\Sigma} / \text{Size}_{\Sigma\text{Доп}} > 1$, то формируемый поток оказывается больше пропускной способности канала передачи информации.

Для того чтобы устранить это несоответствие, в перемножителе 15 коэффициенты квантования всех блоков кадра, запомненные схемой памяти 16, умножаются на рассчитанный коэффициент $K_{\text{Доп}}^*(m) = q \cdot K_{\text{Доп}}(m)$, $1 \leq m \leq M$ и подаются на схему 17 квантования и округления данных, на второй вход которой поступают также спектральные компоненты ДКП всех блоков кадра, предварительно запомненные в схеме памяти 16.

Таким образом, на выходе схемы формируются коды цифровых данных всех блоков кадра:

$$L_m(u, v) = \text{round}\{[1/K_{\text{Доп}}^*(m)] \cdot F_m(u, v)/Q(u, v)\}, \quad 0 \leq u, v \leq N, \quad 1 \leq m \leq M. \quad (10.7)$$

Эти результаты обработки одной из компонент изображения поступают на блок 18 кодирования, на который подаются результаты обработки других компонент изображения.

В случае, если цифровой поток вновь окажется не соответствующим пропускной способности канала связи, операции вычисления коэффициента $q = \text{Size}_{\Sigma\text{вых}} / \text{Size}_{\Sigma\text{Доп}}$, где $\text{Size}_{\Sigma\text{вых}}$ — цифровой поток, полученный в результате указанных выше преобразований, и уточнения коэффициентов квантования могут быть повторены. Выход блока 18 является выходом устройства, реализующего данный принцип цифровой обработки изображений.



Рис. 10.3. Изображение «Залив», сжатое с помощью стандартного алгоритма JPEG, коэффициент сжатия 16

Исследования сложного тестового изображения «Залив», содержащего большое количество мелких деталей с высоким контрастом, показали, что при применении адаптивного выбора коэффициентов глубины квантования коэффициентов ДКП получается более высокое визуальное качество восстановленного изображения при той же степени сжатия, либо при сохранении качества восстановленного изображения возможно получение большей степени сжатия (примерно в 1,5 раза — см. рис. 10.3 и рис. 10.4).

На рис. 10.5 приведен пример кадра, закодированного программно-реализованным кодером видеоконференцсвязи с величиной выходного битового потока 58 кбитов/с, на котором демонстрируется использование переменных коэффициентов квантования.

10.2. Направленное пространственное внутрикадровое предсказание

В стандарте кодирования динамических изображений H-264/AVC [4.16, 4.17] поддерживается несколько типов пространственного внутрикадрового предсказания яркости, обозначаемых Intra_4 × 4, Intra_8 × 8 или Intra_16 × 16, а также предсказание цветности.

Режим Intra_4 × 4 основан на предсказании каждого блока яркости 4 × 4, и подходит для кодирования частей изображения с существенными деталями, режим Intra_8 × 8 использует предсказание каждого блока яркости 8 × 8. Режим Intra_16 × 16 осуществляет предсказание блока яркости 16 × 16 целиком и более подходит для гладких областей изображения. В дополнение к этим типам предсказания яркости вводится отдельный режим предсказания цветности.



Рис. 10.4. Изображение «Залив», сжатое с помощью алгоритма с адаптивным выбором коэффициентов квантования, коэффициент сжатия 16

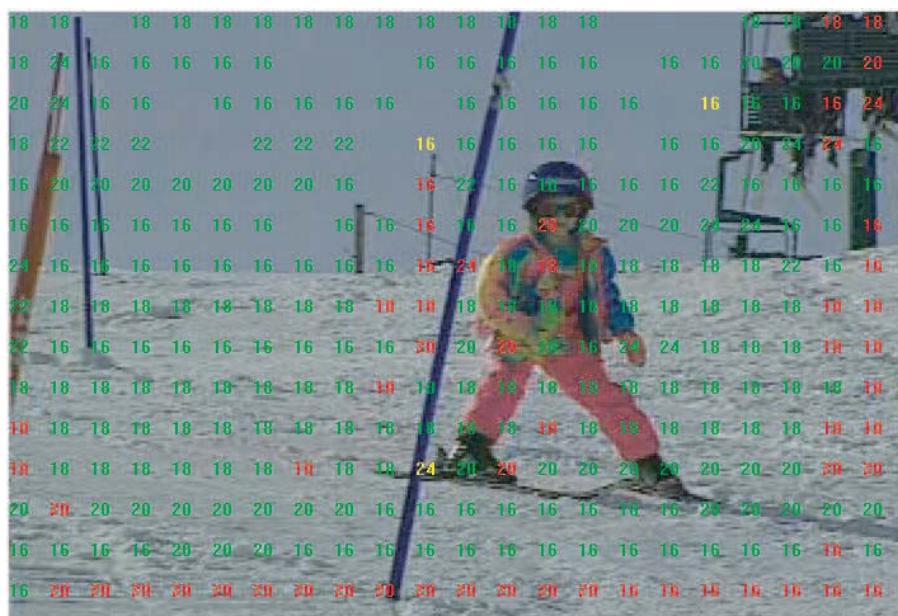


Рис. 10.5. Адаптивный выбор коэффициентов квантования при кодировании динамиче- ской последовательности по стандарту H.261 (поток 58 кбитов/с, 10 кадров/с)

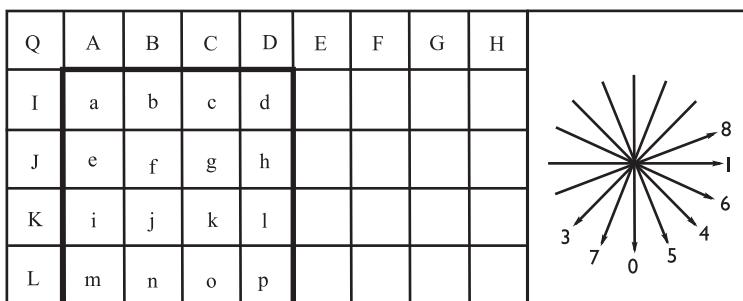


Рис. 10.6. Слева: предсказание Intra 4×4 для отсчетов a–p блока с использованием отсчетов A–Q; справа: 8 «направлений предсказания» для предсказания Intra 4×4

Внутрикадровое предсказание в H.264/AVC всегда проводится в пространственной области с помощью ссылок на соседние отсчеты ранее закодированных блоков, которые находятся слева и/или выше предсказываемого блока.

10.2.1. Режим предсказания пикселов яркости блоков 4×4

При использовании режима Intra 4×4 каждый блок 4×4 предсказывается по пространственно соседним отсчетам, как показано на рис. 10.6 слева. 16 отсчетов блока 4×4 , помеченные буквами a–p, предсказываются с использованием ранее декодированных отсчетов соседних блоков, отмеченных буквами A–Q.

Для каждого блока 4×4 используется один из 9 режимов предсказания. В дополнение к DC-предсказанию (одно значение используется для предсказания всего блока 4×4), имеются 8 режимов «направленного» предсказания, как показано на рис. 10.6 справа. Эти режимы полезны для предсказания направленных структур в изображении, таких как границы, расположенные под различными углами.

На рис. 10.7 показаны пять из девяти режимов предсказания Intra 4×4 . Для режима 0 (вертикальное предсказание) отсчеты над блоком 4×4 копируются в блок, как показано стрелками. Режим 1 (горизонтальное предсказание) работает аналогично вертикальному предсказанию за исключением того, что копируются отсчеты слева от блока 4×4 . Для режима 2 (DC-предсказание) прилегающие отсчеты усредняются. Оставшиеся шесть режимов являются режимами диагонального предсказания и называются предсказаниями диагонально-вниз-влево, диагонально-вниз-вправо, вертикально-вправо, горизонтально-вниз, вертикально-влево и горизонтально-вверх. Как показывают их названия, они подходят для предсказания текстур со структурами указанных направлений. Первые два режима диагонального предсказания также проиллюстрированы. Когда отсчеты E–H для предсказания диагонально-вниз-влево отсутствуют (потому что они еще не декодированы или, например, не принадлежат внутрикодированному макроблоку в режиме ограниченного внутрикадрового предсказания), они заменяются значением отсчета D.

Режим предсказания по вертикали отсчетов сигнала яркости — направление обозначено цифрой 0 (Intra 4×4 Vertical). Этот режим может исполь-

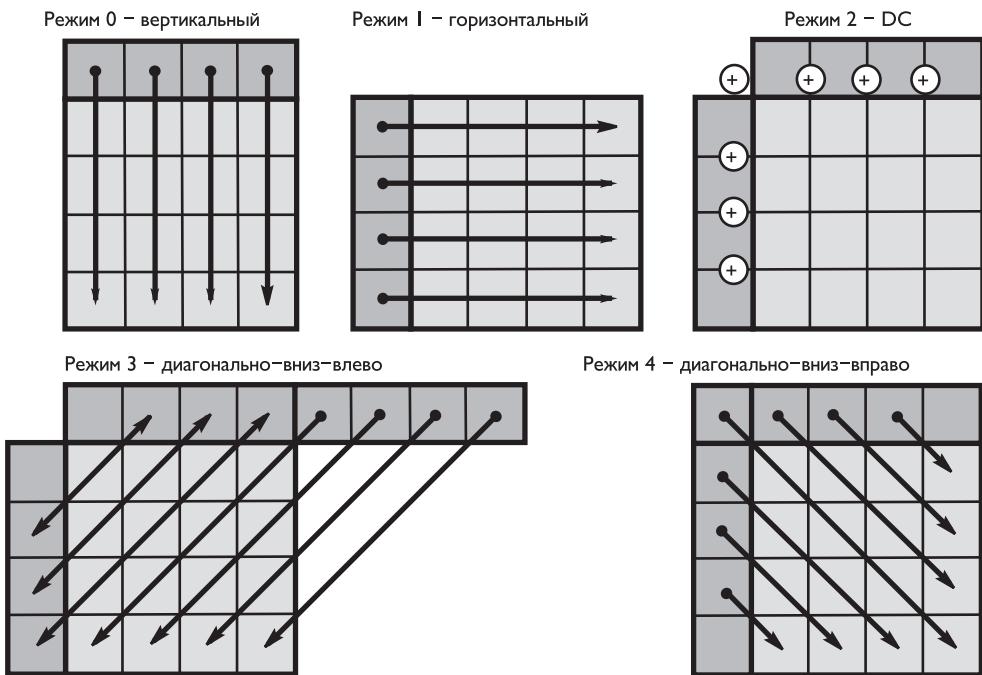


Рис. 10.7. Пять из девяти режимов предсказания Intra_4 × 4

зоваться, если доступны значения пикселов $Y(i, -1)$ при $i = 0, \dots, 3$ (например, эти пиксели не выходят за границы изображения), и реализуется в соответствии с соотношением (см. рис. 10.7, режим 0):

$$\hat{Y}(i, j) = Y(i, -1) \text{ при } i, j = 0, \dots, 3. \quad (10.8)$$

Режим предсказания по горизонтали отсчетов сигнала яркости — направление обозначено цифрой 1 (Intra_4 × 4_Horizontal). Этот режим может использоваться, если доступны значения пикселов $Y(-1, j)$ при $i = 0, \dots, 3$, и реализуется в соответствии с соотношением (см. рис. 10.7, режим 1):

$$\hat{Y}(i, j) = Y(-1, j) \text{ при } i, j = 0, \dots, 3. \quad (10.9)$$

Режим предсказания средней составляющей DC обозначается цифрой 2 (Intra_4 × 4_DC). Может быть использован один из следующих вариантов вычисления DC: если доступны значения пикселов $Y(i, -1)$ при $i = 0, \dots, 3$ и $Y(-1, j)$ при $j = 0, \dots, 3$, то предсказанные значения пикселов блока находят следующим образом (см. рис. 10.7, режим 2):

$$\hat{Y}(i, j) = \left[\left(\sum_{n=0}^3 Y(n, -1) + \sum_{n=0}^3 Y(-1, n) + 4 \right) / 8 \right] \text{ при } i, j = 0, \dots, 3; \quad (10.10)$$

если недоступны значения пикселов $Y(-1, j)$ при $j = 0, \dots, 3$, а имеются значения пикселов $Y(i, -1)$ при $i = 0, \dots, 3$, то предсказанные значения пикселов блока определяются так:

$$\hat{Y}(i, j) = \left[\left(\sum_{n=0}^3 Y(n, -1) + 2 \right) / 4 \right] \text{ при } i, j = 0, \dots, 3; \quad (10.11)$$

если недоступны значения пикселов $Y(i, -1)$ при $i = 0, \dots, 3$, но имеются значения пикселов $Y(-1, j)$ при $j = 0, \dots, 3$, то предсказанные значения пикселов блока определяются так:

$$\hat{Y}(i, j) = \left[\left(\sum_{n=0}^3 Y(-1, n) + 2 \right) / 4 \right] \text{ при } i, j = 0, \dots, 3 \quad (10.12)$$

и, наконец, если некоторые значения пикселов $Y(i, -1)$ при $i = 0, \dots, 3$ и некоторые значения пикселов $Y(-1, j)$ при $j = 0, \dots, 3$ недоступны, то предсказанные значения пикселов блока следует рассчитывать так:

$$\hat{Y}(i, j) = 2^{(B_Y - 1)} \text{ при } i, j = 0, \dots, 3, \quad (10.13)$$

где B_Y — число битов на отсчет сигнал яркости. Здесь и далее значение $[M]$ равно целой части числа M .

Режим предсказания диагонально-вниз-влево отсчетов сигнала яркости — направление обозначено цифрой 3 (Intra_4 × 4_Diagonal_Down_Left). Этот режим может использоваться, если имеются значения пикселов $Y(i, -1)$ при $i = 0, \dots, 7$. Предсказанные значения пикселов определяются следующим образом (см. рис. 10.7, режим 3):

$$\hat{Y}(i, j) = \begin{cases} \left[\frac{Y(6, -1) + 3Y(7, -1) + 2}{4} \right], & i = j = 3; \\ \left[\frac{Y(i+j, -1) + 2Y(i+j+1, -1) + Y(i+j+2, -1) + 2}{4} \right], & i, j = 0, \dots, 3, i = j \neq 3. \end{cases} \quad (10.14)$$

Режим предсказания диагонально-вниз-вправо отсчетов сигнала яркости — направление обозначено цифрой 4 (Intra_4 × 4_Diagonal_Down_Right). Этот режим может использоваться, если доступны значения пикселов $Y(i, -1)$ при $i = 0, \dots, 3$ и $Y(-1, j)$ при $j = -1, \dots, 3$.

Предсказанные значения пикселов при $i, j = 0, \dots, 3$ определяются следующим образом:

$$\hat{Y}(i, j) = \begin{cases} \left[\frac{Y(i-j-2, -1) + 2Y(i-j-1, -1) + Y(i-j, -1) + 2}{4} \right], & i > j; \\ \left[\frac{Y(-1, j-i-2) + 2Y(-1, j-i-1) + Y(-1, j-i) + 2}{4} \right], & i < j; \\ \left[\frac{Y(0, -1) + 2Y(-1, -1) + Y(-1, 0) + 2}{4} \right], & i = j. \end{cases} \quad (10.15)$$

Режим предсказания вертикально-вправо отсчетов сигнала яркости — направление обозначено цифрой 5 (Intra_4 × 4_Vertical_Right). Этот режим может использоваться, если доступны значения пикселов $Y(i, -1)$ при $i = 0, \dots, 3$ и $Y(-1, j)$ при $j = -1, \dots, 3$. Если обозначить $n = 2i - j$ ($i, j = 0, \dots, 3$), то предсказанные значения пикселов определяются следующим образом:

$$\hat{Y}(i, j) = \begin{cases} \left[\frac{Y(i-[j/2]-1, -1) + Y(i-[j/2], -1) + 1}{2} \right], & n = 0, 2, 4, 6; \\ \left[\frac{Y(i-[j/2]-2, -1) + 2Y(i-[j/2]-1, -1) + Y(i-[j/2], -1) + 2}{4} \right], & n = 1, 3, 5; \\ \left[\frac{Y(-1, 0) + 2Y(-1, -1) + Y(0, -1) + 2}{4} \right], & n = -1; \\ \left[\frac{Y(-1, j-1) + 2Y(-1, j-2) + Y(-1, j-3) + 2}{4} \right], & n = -2, -3. \end{cases} \quad (10.16)$$

Режим предсказания горизонтально-вниз отсчетов сигнала яркости — направление обозначено цифрой 6 (Intra_4 × 4_Horizontal_Down). Этот режим может использоваться, если доступны значения пикселов $Y(i, -1)$ при $i = 0, \dots, 3$ и $Y(-1, j)$ при $j = -1, \dots, 3$. Если обозначить $n = 2j - i$ ($i, j = 0, \dots, 3$), то предсказанные значения пикселов определяются следующим образом:

$$\hat{Y}(i, j) = \begin{cases} \left[\frac{Y(-1, j-[i/2]-1) + Y(-1, j-[i/2]) + 1}{2} \right], & n = 0, 2, 4, 6; \\ \left[\frac{Y(-1, j-[i/2]-2) + 2Y(-1, j-[i/2]-1) + Y(-1, j-[i/2]) + 2}{4} \right], & n = 1, 3, 5; \\ \left[\frac{Y(-1, 0) + 2Y(-1, -1) + Y(0, -1) + 2}{4} \right], & n = -1; \\ \left[\frac{Y(i-1, -1) + 2Y(i-2, -1) + Y(i-3, -1) + 2}{4} \right], & n = -2, -3. \end{cases} \quad (10.17)$$

Режим предсказания вертикально-влево отсчетов сигнала яркости — направление обозначено цифрой 7 (Intra_4 × 4_Vertical_Left). Этот режим может использоваться, если имеются значения пикселов $Y(i, -1)$ при $i = 0, \dots, 7$.

Предсказанные значения пикселов определяются при $i = 0, \dots, 3$ следующим образом:

$$\hat{Y}(i, j) = \begin{cases} \left[\frac{Y(i+[j/2], -1) + Y(i+[j/2]+1, -1) + 1}{2} \right], & j = 0, 2; \\ \left[\frac{Y(i+[j/2], -1) + 2Y(i+[j/2]+1, -1) + Y(i+[j/2]+2, -1) + 2}{4} \right], & j = 1, 3. \end{cases} \quad (10.18)$$

Режим предсказания горизонтально-вверх отсчетов сигнала яркости — направление обозначено цифрой 8 (Intra_4 × 4_Horizontal_Up). Этот режим может использоваться, если доступны значения пикселов $Y(-1, j)$ при $j = -1, \dots, 3$.

Если обозначить $n = i + 2j$ ($i, j = 0, \dots, 3$), то предсказанные значения пикселов определяются следующим образом:

$$\hat{Y}(i, j) = \begin{cases} \left[\frac{Y(-1, j+[i/2]) + Y(-1, j+[i/2]) + 1}{2} \right], & n = 0, 2, 4; \\ \left[\frac{Y(-1, j+[i/2]) + 2Y(-1, j+[i/2]+1) + Y(-1, j+[i/2]+2) + 2}{4} \right], & n = 1, 3; \\ \left[\frac{Y(-1, 2) + 3Y(-1, 3) + 2}{4} \right], & n = 5; \\ [Y(-1, 3)], & n > 5. \end{cases} \quad (10.19)$$

10.2.2. Режим предсказания пикселов яркости блоков 8×8

Процесс фильтрации опорных пикселов для предсказания блоков 8×8

Процесс предсказания пикселов блоков 8×8 (Intra_8 × 8) производится после фильтрации значений опорных пикселов $Y(-1, j)$ при $j = -1, \dots, 7$ и $Y(i, -1)$ при $i = 0, \dots, 15$, если эти значения возможно использовать для предсказания.

Если все $Y(i, -1)$ при $i = 0, \dots, 7$ могут быть использованы для предсказания, то применяется нижеследующая фильтрация.

Фильтрованное значение $Y'(0, -1)$ определяется так: если $Y(-1, -1)$ можно использовать для предсказания, то

$$Y'(0, -1) = [(Y(-1, -1) + 2Y(0, -1) + Y(1, -1) + 2) / 4]; \quad (10.20)$$

иначе

$$Y'(0, -1) = [(3Y(0, -1) + Y(1, -1) + 2) / 4]. \quad (10.21)$$

Фильтрованное значение $Y'(i, -1)$ при $i = 1, \dots, 7$ определяется так:

$$Y'(i, -1) = [(Y(i - 1, -1) + 2Y(i, -1) + Y(i + 1, -1) + 2) / 4]. \quad (10.22)$$

Если все $Y(i, -1)$ при $i = 7, \dots, 15$ могут быть использованы для предсказания, то применяется нижеследующая фильтрация:

$$Y'(i, -1) = \begin{cases} [(Y(i - 1, -1) + 2Y(i, -1) + Y(i + 1, -1) + 2) / 4], & i = 8, \dots, 14; \\ [(Y(14, -1) + 3Y(15, -1) + 2) / 4], & i = 15. \end{cases} \quad (10.23)$$

Если пиксели $Y(-1, -1)$ могут быть использованы для предсказания блоков 8×8 , то фильтрованные значения $Y'(-1, -1)$ определяются в зависимости от возможности использования для предсказания пикселов $Y(0, -1)$ и $Y(-1, 0)$: если пиксель $Y(0, -1)$ может быть использован для предсказания, а $Y(-1, 0)$ не может быть использован для предсказания, то

$$Y'(-1, -1) = [(3Y(-1, -1) + Y(0, -1) + 2) / 4]; \quad (10.24)$$

если пиксель $Y(0, -1)$ не может быть использован для предсказания, а $Y(-1, 0)$ может быть использован для предсказания, то

$$Y'(-1, -1) = [(3Y(-1, -1) + Y(-1, 0) + 2) / 4]; \quad (10.25)$$

если пиксели $Y(0, -1)$ и $Y(-1, 0)$ могут быть использованы для предсказания, то

$$Y'(-1, -1) = [(Y(0, -1) + 2Y(-1, -1) + Y(-1, 0) + 2) / 4]. \quad (10.26)$$

Если все пиксели $Y(-1, j)$ могут быть использованы для предсказания блоков 8×8 , то фильтрованные значения $Y'(-1, 0)$ определяются в зависимости от возможности использования для предсказания пикселов $Y(-1, -1)$: если $Y(-1, -1)$ может быть использован для предсказания, то

$$Y'(-1, 0) = [(Y(-1, -1) + 2Y(-1, 0) + Y(-1, 1) + 2) / 4], \quad (10.27)$$

если $Y(-1, -1)$ не может быть использован для предсказания, то

$$Y'(-1, 0) = [(3Y(-1, 0) + Y(-1, 1) + 2) / 4]. \quad (10.28)$$

Фильтрованные значения пикселов $Y'(-1, j)$ при $i = 1, \dots, 7$ определяются так:

$$Y'(-1, j) = \begin{cases} [(Y(-1, j-1) + 2Y(-1, j) + Y(-1, j+1) + 2)/4], & j = 1, \dots, 6; \\ [(Y(-1, 6) + 3Y(-1, 7) + 2)/4], & j = 7. \end{cases} \quad (10.29)$$

Описание режимов предсказания блоков 8×8

Режим предсказания по вертикали отсчетов сигнала яркости — направление обозначается цифрой 0 (Intra_8 × 8_Vertical). Этот режим может использоваться, если значения пикселов $Y(i, -1)$ при $i = 0, \dots, 7$ доступны (например, они не выходят за границы изображения), и реализуется в соответствии с соотношением:

$$\hat{Y}(i, j) = Y'(-1, j) \text{ при } i, j = 0, \dots, 7. \quad (10.30)$$

Режим предсказания по горизонтали отсчетов сигнала яркости — направление обозначено цифрой 1 (Intra_8 × 8_Horizontal). Этот режим может использоваться, если доступны значения пикселов $Y(-1, j)$ при $j = 0, \dots, 7$, и реализуется в соответствии с соотношением:

$$\hat{Y}(i, j) = Y'(-1, j) \text{ при } i, j = 0, \dots, 7. \quad (10.31)$$

Режим предсказания средней составляющей DC обозначается цифрой 2 (Intra_8 × 8_DC). Используется несколько вариантов вычисления DC: если доступны значения пикселов $Y(i, -1)$ при $i = 0, \dots, 7$ и $Y(-1, j)$ при $j = 0, \dots, 7$, то предсказанные значения пикселов блока находят следующим образом:

$$\hat{Y}(i, j) = \left[\left(\sum_{n=0}^7 Y'(n, -1) + \sum_{n=0}^7 Y'(-1, n) + 8 \right) / 16 \right] \text{ при } i, j = 0, \dots, 7; \quad (10.32)$$

если не все значения пикселов $Y(-1, j)$ при $j = 0, \dots, 7$ доступны, а значения пикселов $Y(i, -1)$ при $i = 0, \dots, 7$ доступны, то предсказанные значения пикселов блока определяются так:

$$\hat{Y}(i, j) = \left[\left(\sum_{n=0}^7 Y'(n, -1) + 4 \right) / 8 \right] \text{ при } i, j = 0, \dots, 7; \quad (10.33)$$

если не все значения пикселов $Y(i, -1)$ при $i = 0, \dots, 7$ доступны, а значения пикселов $Y(-1, j)$ при $j = 0, \dots, 7$ доступны, то предсказанные значения пикселов блока определяются так:

$$\hat{Y}(i, j) = \left[\left(\sum_{n=0}^7 Y'(-1, n) + 4 \right) / 8 \right] \text{ при } i, j = 0, \dots, 7; \quad (10.34)$$

наконец, если некоторые значения пикселов $Y(i, -1)$ при $i = 0, \dots, 7$ и некоторые значения пикселов $Y(-1, j)$ при $j = 0, \dots, 7$ недоступны, то предсказанные значения пикселов блока следует рассчитывать так:

$$\hat{Y}(i, j) = 2^{(B_Y-1)} \text{ при } i, j = 0, \dots, 7, \quad (10.35)$$

где B_Y — число битов на отсчет сигнала яркости.

Режим предсказания диагонально-вниз-влево отсчетов сигнала яркости — направление обозначено цифрой 3 (Intra_8 × 8_Diagonal_Down_Left).

Этот режим может использоваться, если имеются значения пикселов $Y(i, -1)$ при $i = 0, \dots, 15$. Предсказанные значения пикселов определяются следующим образом:

$$\hat{Y}(i, j) = \begin{cases} \left[\frac{Y'(14, -1) + 3Y'(15, -1) + 2}{4} \right], & i = j = 7; \\ \left[\frac{Y'(i+j, -1) + 2Y'(i+j+1, -1) + Y'(i+j+2, -1) + 2}{4} \right], & i, j = 0, \dots, 7, i = j \neq 7. \end{cases} \quad (10.36)$$

Режим предсказания диагонально-вниз-вправо отсчетов сигнала яркости — направление обозначено цифрой 4 (Intra_8 × 8_Diagonal_Down_Right). Этот режим может использоваться, если доступны значения пикселов $Y(i, -1)$ при $i = 0, \dots, 7$ и $Y(-1, j)$ при $j = -1, \dots, 7$. Предсказанные значения пикселов при $i, j = 0, \dots, 7$ определяются следующим образом:

$$\hat{Y}(i, j) = \begin{cases} \left[\frac{Y'(i-j-2, -1) + 2Y'(i-j-1, -1) + Y'(i-j, -1) + 2}{4} \right], & i > j; \\ \left[\frac{Y'(-1, j-i-2) + 2Y'(-1, j-i-1) + Y'(-1, j-i) + 2}{4} \right], & i < j; \\ \left[\frac{Y'(0, -1) + 2Y'(-1, -1) + Y'(-1, 0) + 2}{4} \right], & i = j. \end{cases} \quad (10.37)$$

Режим предсказания вертикально-вправо отсчетов сигнала яркости — направление обозначено цифрой 5 (Intra_8 × 8_Vertical_Right). Этот режим может использоваться, если доступны значения пикселов $Y(i, -1)$ при $i = 0, \dots, 7$ и $Y(-1, j)$ при $j = -1, \dots, 7$. Если обозначить $n = 2i - j$, $m = i - [j/2]$, $l = j - 2i - 1$ ($i, j = 0, \dots, 7$), то предсказанные значения пикселов определяются следующим образом:

$$\hat{Y}(i, j) = \begin{cases} \left[\frac{Y'(m-1, -1) + Y'(m, -1) + 1}{2} \right], & n = 0, 2, 4, 6, 8, 10, 12, 14; \\ \left[\frac{Y'(m-2, -1) + 2Y'(m-1, -1) + Y'(m, -1) + 2}{4} \right], & n = 1, 3, 5, 7, 9, 11, 13; \\ \left[\frac{Y'(-1, 0) + 2Y'(-1, -1) + Y'(0, -1) + 2}{4} \right], & n = -1; \\ \left[\frac{Y'(-1, l) + 2Y'(-1, l-1) + Y'(-1, l-2) + 2}{4} \right], & n = -2, -3, -4, -5, -6, -7. \end{cases} \quad (10.38)$$

Режим предсказания горизонтально-вниз отсчетов сигнала яркости — направление обозначено цифрой 6 (Intra_8 × 8_Horizontal_Down). Этот режим может использоваться, если доступны значения пикселов $Y(i, -1)$ при $i = 0, \dots, 7$ и $Y(-1, j)$ при $j = -1, \dots, 7$. Если обозначить $n = 2j - i$, $m = j - [i/2]$, $l = i - 2j - 1$ ($i, j = 0, \dots, 7$), то предсказанные значения пикселов определяются следующим образом:

$$\hat{Y}(i, j) = \begin{cases} \left[\frac{Y'(-1, m-1) + Y'(-1, m) + 1}{2} \right], & n = 0, 2, 4, 6, 8, 10, 12, 14; \\ \left[\frac{Y'(-1, m-2) + 2Y'(-1, m-1) + Y'(-1, m) + 2}{4} \right], & n = 1, 3, 5, 7, 9, 11, 13; \\ \left[\frac{Y'(-1, 0) + 2Y'(-1, -1) + Y'(0, -1) + 2}{4} \right], & n = -1; \\ \left[\frac{Y'(l, -1) + 2Y'(l-1, -1) + Y'(l-2, -1) + 2}{4} \right], & n = -2, -3, -4, -5, -6, -7. \end{cases} \quad (10.39)$$

Режим предсказания вертикально-влево отсчетов сигнала яркости — направление обозначено цифрой 7 (Intra_8 × 8_Vertical_Left). Этот режим может

использоватьсяся, если доступны значения пикселов $Y(i, -1)$ при $i = 0, \dots, 15$. Если обозначить $m = i + [j/2]$, то предсказанные значения пикселов определяются при $i = 0, \dots, 7$ следующим образом:

$$\hat{Y}(i, j) = \begin{cases} \left[\frac{Y'(-m, -1) + Y'(-m+1, -1) + 1}{2} \right], & j = 0, 2, 4, 6; \\ \left[\frac{Y'(-m, -1) + 2Y'(-m+1, -1) + Y'(-m+2, -1) + 2}{4} \right], & j = 1, 3, 5, 7. \end{cases} \quad (10.40)$$

Режим предсказания горизонтально-вверх отсчетов сигнала яркости — направление обозначено цифрой 8 (Intra_8 × 8_Horizontal_Up). Этот режим может использоваться, если доступны значения пикселов $Y(-1, j)$ при $j = -1, \dots, 7$.

Если обозначить $n = i + 2j$, $m = j + [i/2]$ ($i, j = 0, \dots, 7$), то предсказанные значения пикселов определяются следующим образом:

$$\hat{Y}(i, j) = \begin{cases} \left[\frac{Y'(-1, m) + Y'(-1, m+1) + 1}{2} \right], & n = 0, 2, 4, 6, 8, 10, 12; \\ \left[\frac{Y'(-1, m) + 2Y'(-1, m+1) + Y'(-1, m+2) + 2}{4} \right], & n = 1, 3, 5, 7, 9, 11; \\ \left[\frac{Y'(-1, 6) + 3Y'(-1, 7) + 2}{4} \right], & n = 13; \\ [Y'(-1, 7)], & n > 13. \end{cases} \quad (10.41)$$

10.2.3. Режим предсказания пикселов яркости блоков 16×16

При использовании режима Intra_16 × 16 предсказывается целиком яркостная компонента макроБЛОКА. Поддерживаются 4 режима предсказания. Режим 0 (вертикальное предсказание), режим 1 (горизонтальное предсказание) и режим 2 (DC-предсказание) определены аналогично соответствующим режимам предсказания Intra_4 × 4 за исключением того, что вместо 4 соседей на каждой стороне для предсказания блока 4 × 4 используются 16 соседних отсчетов на каждой стороне для предсказания блока 16 × 16. Четвертый режим предсказания — так называемое плоскостное (plane) предсказание.

Режим предсказания по вертикали отсчетов сигнала яркости — направление обозначено цифрой 0 (Intra_16 × 16_Vertical). Этот режим может использоваться, если доступны значения пикселов $Y(i, -1)$ при $j = 0, \dots, 15$ (например, они не выходят за границы изображения), и реализуется в соответствии с соотношением:

$$\hat{Y}(i, j) = Y(i, -1) \text{ при } i, j = 0, \dots, 15. \quad (10.42)$$

Режим предсказания по горизонтали отсчетов сигнала яркости — направление обозначено цифрой 1 (Intra_16 × 16_Horizontal). Этот режим может использоваться, если доступны значения пикселов $Y(-1, j)$ при $i = 0, \dots, 15$, и реализуется в соответствии с соотношением:

$$\hat{Y}(i, j) = Y(-1, j) \text{ при } i, j = 0, \dots, 15. \quad (10.43)$$

Режим предсказания средней составляющей DC обозначается цифрой 2 (Intra_16 × 16_DC). Используется несколько вариантов вычисления DC: если доступны значения пикселов $Y(i, -1)$ при $i = 0, \dots, 15$ и $Y(-1, j)$ при $j = 0, \dots, 15$, то предсказанные значения пикселов блока находят следующим

образом:

$$\hat{Y}(i, j) = \left[\left(\sum_{n=0}^{15} Y(n, -1) + \sum_{n=0}^{15} Y(-1, n) + 16 \right) / 32 \right] \text{ при } i, j = 0, \dots, 15; \quad (10.44)$$

если недоступны значения пикселов $Y(-1, j)$ при $j = 0, \dots, 15$, а значения пикселов $Y(i, -1)$ при $i = 0, \dots, 15$ доступны, то предсказанные значения пикселов блока определяются так:

$$\hat{Y}(i, j) = \left[\left(\sum_{n=0}^{15} Y(n, -1) + 8 \right) / 16 \right] \text{ при } i, j = 0, \dots, 15; \quad (10.45)$$

если недоступны значения пикселов $Y(i, -1)$ при $i = 0, \dots, 15$, а значения пикселов $Y(-1, j)$ при $j = 0, \dots, 15$ доступны, то предсказанные значения пикселов блока определяются так:

$$\hat{Y}(i, j) = \left[\left(\sum_{n=0}^{15} Y(-1, n) + 8 \right) / 16 \right] \text{ при } i, j = 0, \dots, 15; \quad (10.46)$$

наконец, если недоступны некоторые значения пикселов $Y(i, -1)$ при $i = 0, \dots, 15$ и некоторые значения пикселов $Y(-1, j)$ при $j = 0, \dots, 15$, то предсказанные значения пикселов блока следует рассчитывать так:

$$\hat{Y}(i, j) = 2^{(B_Y-1)} \text{ при } i, j = 0, \dots, 15, \quad (10.47)$$

где B_Y — число битов на отсчет сигнала яркости.

Режим плоскостного предсказания обозначается цифрой 3 (Intra_16 × 16_Plane). Этот режим может использоваться, если доступны значения пикселов $Y(i, -1)$ при $i = 0, \dots, 15$ и $Y(-1, j)$ при $j = 0, \dots, 15$. Предсказанные значения пикселов определяются следующим образом:

$$\hat{Y}(i, j) = \text{Clip}_Y \{ Z(i, j) = [(a + b \cdot (i - 7) + c \cdot (j - 7) + 16) / 32] \}, i, j = 0, \dots, 15, \quad (10.48)$$

где $a = 16 (Y(-1, 15) + Y(15, -1))$, $b = [(5H + 32) / 64]$, $c = [(5V + 32) / 64]$,

$$H = \sum_{n=0}^7 (n + 1) \cdot (Y(8 + n, -1) - Y(6 - n, -1)),$$

$$V = \sum_{n=0}^7 (n + 1) \cdot (Y(-1, 4 + n) - Y(-1, 2 - n)),$$

$$\text{Clip}_Y \{ X \} = \begin{cases} X, & 0 \leq X \leq 2B_Y - 1, \\ 0, & X < 0, \\ 2B_Y - 1, & X > 2B_Y - 1. \end{cases}$$

10.2.4. Режим предсказания пикселов блоков цветности

Пиксели блоков цветности предсказываются с помощью методики, аналогичной предсказанию яркости в режиме Intra_16×16, так как цветность обычно является гладкой функцией на достаточно больших областях.

Число пикселов цветности вдоль и поперек раstra в каждом блоке изображения (4×4 , 8×8 или 16×16) зависит от используемого формата цветности, варианты которого приведены в табл. 10.1.

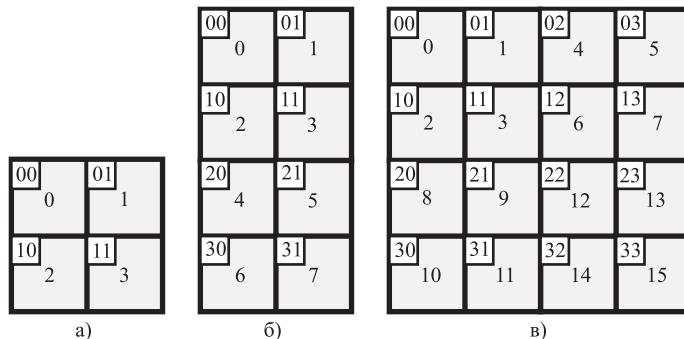


Рис. 10.8. Форматы блоков цветности 4×4 при различных индексах: а — $I_C = 1$; б — $I_C = 2$; в — $I_C = 3$

Таблица 10.1. Изменение количества пикселов цветности в блоках изображения

Индекс формата цветности I_C	Формат изображения	Коэффициент уменьшения количества пикселов цветности вдоль раstra C_H	Коэффициент уменьшения количества пикселов цветности поперек раstra C_V
0	Монохромное	—	—
1	4 : 2 : 0	2	2
2	4 : 2 : 2	2	1
3	4 : 4 : 4	1	1

При этом в каждом блоке изображения (блоке сигнала яркости) размером $N \times N$ размещается вдоль и поперек раstra соответственно $N_{CH} = N/C_H$ и $N_{CV} = N/C_V$ пикселов цветности Cr и Cb.

На рис. 10.8 приведены форматы блоков цветности 4×4 для различных форматов изображения. Формат блока цветности при $I_C = 3$ совпадает с форматом блока яркости 4×4 .

Режим предсказания средней составляющей DC блока цветности обозначается цифрой 0 (Intra_Chroma_DC). Используется несколько вариантов вычисления DC блока цветности. Для всех блоков цветности размером 4×4 индекс каждого блока цветности определяется как $Bl_C = 0, \dots, (2^{I_C+1} - 1)$.

В зависимости от индекса формата цветности I_C местоположение верхнего левого пикселя блока цветности 4×4 с индексом Bl_C определяется следующим образом: если $I_C = 1$ или 2 , то

$$i_0 = 4(Bl_C - 2[Bl_C/2]); \quad j_0 = 4[Bl_C/2];$$

если $I_C = 3$, то

$$i_0 = 8[Bl_C/4 - 2[Bl_C/8]] + 4[Bl_C/4 - 2[Bl_C/8] - 2[(Bl_C/4 - 2[Bl_C/8])/2]]; \\ j_0 = 8[Bl_C/8] + 4[(Bl_C/4 - 2[Bl_C/8])/2].$$

При $(i_0, j_0) = (0, 0)$ или $i_0, j_0 > 0$, значения предсказания пикселов при $(i_0 + i, j_0 + j)$, $i, j = 0, \dots, 3$ определяются следующим образом: если все пиксели $C(i_0 + i, -1)$ при $i = 0, \dots, 3$ и $C(-1, j_0 + j)$ при $j = 0, \dots, 3$ могут быть использованы для

предсказания цветности, то

$$\hat{C}(i_0 + i, j_0 + j) = \left[\left(\sum_{i'=0}^3 C(i_0 + i', -1) + \sum_{j'=0}^3 C(-1, j_0 + j') + 4 \right) \middle/ 8 \right], \quad i, j = 0, \dots, 3; \quad (10.49)$$

если какие-либо пиксели $C(i_0 + i, -1)$ при $i = 0, \dots, 3$ не могут быть использованы для предсказания, а пиксели $C(-1, j_0 + j)$ при $j = 0, \dots, 3$ могут быть использованы для предсказания, то

$$\hat{C}(i_0 + i, j_0 + j) = \left[\left(\sum_{j'=0}^3 C(-1, j_0 + j') + 2 \right) \middle/ 4 \right], \quad i, j = 0, \dots, 3; \quad (10.50)$$

если пиксели $C(i_0 + i, -1)$ при $i = 0, \dots, 3$ могут быть использованы для предсказания, а какие-либо пиксели $C(-1, j_0 + j)$ при $j = 0, \dots, 3$ не могут быть использованы для предсказания, то

$$\hat{C}(i_0 + i, j_0 + j) = \left[\left(\sum_{i'=0}^3 C(i_0 + i', -1) + 2 \right) \middle/ 4 \right], \quad i, j = 0, \dots, 3; \quad (10.51)$$

если некоторые пиксели $C(i_0 + i, -1)$ при $i = 0, \dots, 3$ и некоторые пиксели $C(-1, j_0 + j)$ при $j = 0, \dots, 3$ не могут быть использованы для предсказания цветности, то

$$\hat{C}(i_0 + i, j_0 + j) = 2^{(B_C - 1)}, \quad i, j = 0, \dots, 3, \quad (10.52)$$

где B_C — количество битов на отсчет сигнала цветности.

Режим предсказания по горизонтали отсчетов сигнала цветности — направление обозначено цифрой 1 (Intra_Chroma_Horizontal). Этот режим может использоваться, если пиксели $C(-1, j)$ при $j = 0, \dots, M_{VC} - 1$ доступны для предсказания. Значения предсказанных пикселов находятся следующим образом:

$$\hat{C}(i, j) = C(-1, j), \quad i = 0, \dots, M_{HC} - 1, \quad j = 0, \dots, M_{VC} - 1, \quad (10.53)$$

где M_{HC} и M_{VC} — максимальное количество пикселов блока цветности вдоль и поперек раstra.

Режим предсказания по вертикали отсчетов сигнала цветности — направление обозначено цифрой 2 (Intra_Chroma_Vertical). Этот режим может использоваться, если пиксели $C(i, -1)$ при $i = 0, \dots, M_{HC} - 1$ доступны для предсказания. Значения предсказанных пикселов находятся следующим образом:

$$\hat{C}(i, j) = C(i, -1), \quad i = 0, \dots, M_{HC} - 1, \quad j = 0, \dots, M_{VC} - 1. \quad (10.54)$$

Режим плоскостного предсказания отсчетов сигнала цветности обозначается цифрой 3 (Intra_Chroma_Plane). Этот режим может использоваться, если пиксели $C(i, -1)$ при $i = 0, \dots, M_{HC} - 1$ и пиксели $C(-1, j)$ при $j = 0, \dots, M_{VC} - 1$ доступны для предсказания.

Пусть $i_C = \begin{cases} 4, & I_C = 3, \\ 0, & I_C \neq 3, \end{cases}$ $j_C = \begin{cases} 0, & I_C = 1, \\ 4, & I_C \neq 1, \end{cases}$ тогда

$$\hat{C}(i, j) = \text{Clip}_C [(a + b \cdot (i - 3 - i_C) + c \cdot (j - 3 - j_C) + 16) / 32], \quad (10.55)$$

$$\begin{aligned}
 i &= 0, \dots, M_{HC} - 1, \quad j = 0, \dots, M_{VC} - 1, \\
 a &= 16 (C(-1, M_{VC} - 1) + C(M_{HC} - 1, -1)); \\
 b &= [(34 - 29 \cdot (I_C = 3) \cdot H + 32) / 64]; \\
 c &= [(34 - 29 \cdot (I_C \neq 1) \cdot V + 32) / 64]; \\
 H &= \sum_{i'=0}^{3+i_C} (i' + 1) \cdot (C(4 + i_C + i', -1) - C(2 + i_C - i', -1)); \\
 V &= \sum_{j'=0}^{3+j_C} (j' + 1) \cdot (C(-1, 4 + j_C + j') - C(-1, 2 + j_C - j')); \\
 \text{Clip}_C \{X\} &= \begin{cases} X, & 0 \leq X \leq 2B_C - 1, \\ 0, & X < 0, \\ 2B_C - 1, & X > 2B_C - 1. \end{cases}
 \end{aligned}$$

10.3. Уменьшение влияния блокинг-эффекта

При кодировании изображений с помощью JPEG-подобных алгоритмов обычно появляются искажения в виде характерного разбиения всего изображения на квадратные блоки пикселов с заметными границами. Эти искажения, называемые блокинг-эффектом, возникают вследствие деления изображения на блоки с последующим их независимым кодированием, в котором используется ДКП и квантование коэффициентов. Характерной особенностью ДКП с учетом квантования коэффициентов является возникновение ненулевых ошибок на границах блоков, которые идентифицируются глазом как скачки яркости от одного блока к другому. Подобно формированию блокинг-эффекта возникает так называемый мозаичный эффект, который обусловливается не резкими переходами между различными блоками, а заметным глазу различием яркости в соседних блоках. Можно также определить мозаичный эффект как остаточный блокинг-эффект после низкочастотной фильтрации изображения. Несмотря на то, что переходы между блоками становятся плавными, глаз воспринимает изображение как разбитое на блоки. Мозаичный эффект также возникает при слишком грубом квантовании коэффициентов ДКП, при котором изображения внутри соседних блоков сильно отличаются друг от друга.

Специально разработанные алгоритмы обработки изображений используются для сглаживания блочных артефактов. Фильтры, реализующие такие алгоритмы, сглаживая края блоков, улучшают визуальное восприятие изображения.

В зависимости от структуры блокинг-эффекта и его интенсивности алгоритм обработки блоков изображения может включать ряд операций. Возможно, например, выполнение следующих операций:

- фильтрация четырех вертикальных границ блока 16×16 сигнала яркости, состоящего из 16 блоков 4×4 пиксела яркости (a, b, c, d – рис. 10.9a);
- фильтрация четырех горизонтальных границ блока 16×16 сигнала яркости (e, f, g, h – рис. 10.9a);

- фильтрация двух вертикальных границ блока 8×8 сигнала цветности, состоящего из четырех блоков 4×4 пикселя цветности (i, j — рис. 10.9б);
- фильтрация двух горизонтальных границ блока 8×8 сигнала цветности (k, l — рис. 10.9б).

Каждая фильтрующая операция затрагивает до трех пикселов с обеих сторон границы. На рис. 10.9в, г приведено по четыре пикселя с каждой стороны от вертикальной и горизонтальной границы примыкающих друг к другу блоков P и Q (p_3, p_2, p_1, p_0 и q_0, q_1, q_2, q_3).

Выбор параметров используемого фильтра зависит от параметров границ блоков и градиентов значений пикселов вдоль границ. Параметры границ блоков bS выбираются по правилам [4.16, 4.18], приведенным в табл. 10.2. Чем больше величина bS , тем интенсивней должна быть фильтрация возникающего блокинг-эффекта. Значение $bS = 0$ означает отсутствие фильтрации.

Группа пикселов из множества (p_2, p_1, p_0 и q_0, q_1, q_2) подвергается фильтрации, если $bS > 0$ и $|p_0 - q_0| < \alpha$, $|p_1 - p_0| < \beta$, $|q_1 - q_0| < \beta$, где α и β — пороги, задаваемые стандартом [4.16–4.19].

Невыполнение условий фильтрации приводит к отключению фильтра, например когда изменение уровней пикселов вдоль границы блока имеет значительный градиент. Эти пороги зависят от усредненного коэффициента квантования блоков P и Q :

$$Q_{cp} = \lfloor (Q_P + Q_Q + 1)/2 \rfloor, \quad (10.56)$$

и от некоторых смещений этого коэффициента Q_A и Q_B для α и β соответственно.

Когда параметр Q_{cp} мал, то малый градиент изменений уровней приграничных пикселов, скорее всего, вызван особенностями изображения, а не эффектом блочности. Следовательно, нет необходимости реализовать фильтрацию перехода, т. е. пороги α и β должны быть малыми. Если же параметр Q_{cp} велик, эффект блочного искажения оказывается более вероятным, что отражается в увеличении значений порогов α и β , и большее число граничных пикселов подвергается фильтрации.

Величины α и β возрастают с ростом усредненного коэффициента квантования и зависят от числа битов на отсчет сигнала яркости B_Y и сигнала цветности B_C : при обработке сигнала яркости $\alpha\alpha' \cdot 2^{(BY-8)}$, $\beta\beta' \cdot 2^{(BY-8)}$; при обработке сигнала цветности $\alpha\alpha' \cdot 2^{(BC-8)}$, $\beta\beta' \cdot 2^{(BC-8)}$.

В стандарте [4.17] задано изменение смещений $-6 \leq Q_A, Q_B \leq 6$, реализующее уменьшение или увеличение эффективности фильтрации, а значения величин α', β' сведены в табл. 10.3 в зависимости от некоторых аргументов D_A и D_B , вычисляемых по формулам:

$$D_A = \min \{ \max(0, Q_{cp} + Q_A), 51 \}; \quad D_B = \min \{ \max(0, Q_{cp} + Q_B), 51 \}. \quad (10.57)$$

Процесс фильтрации краев при $bS < 4$

При этом производится фильтрации значений примыкающих друг к другу пикселов pi и qi ($i = 0 \dots 2$). Результирующие значения пикселов $p'0$ и $q'0$ определяются следующим образом:

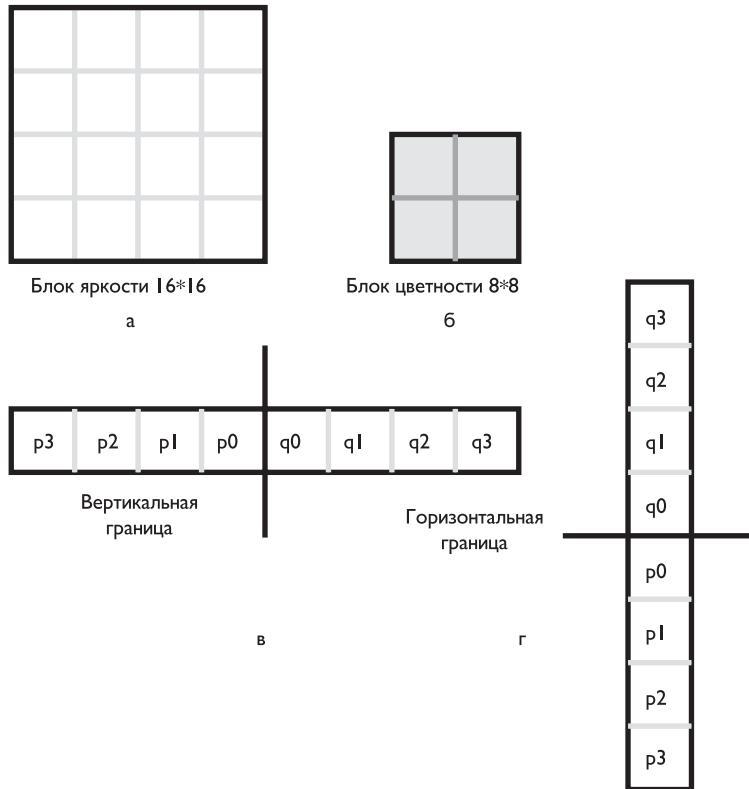


Рис. 10.9. Порядок фильтрации краев блоков яркости 16×16 (а) и цветности 8×8 (б); примеры примыкания пикселов к вертикальным (г) и горизонтальным (д) границам

Таблица 10.2. Правила выбора типа границ блоков

Описание типа границ	Величина bS
При кодировании блоков Р и/или Q границей является граница между блоками яркости 16×16 или блоками цветности 8×8	4
При кодировании блоков Р и/или Q границей не является граница между блоками яркости 16×16 или блоками цветности 8×8	3
Ни блок Р, ни блок Q не кодируются непосредственно, но используются кодированные коэффициенты	2
Ни блок Р, ни блок Q не кодируются, но имеют различные ссылки	1
Во всех прочих случаях	0

$$\begin{aligned}
 p'0 &= \min \{ \max(0, p0 + \Delta), 2^{BY} - 1 \}, \\
 q'0 &= \min \{ \max(0, q0 + \Delta), 2^{BY} - 1 \}, \\
 \text{либо} &
 \end{aligned} \tag{10.58}$$

$$\begin{aligned}
 p'0 &= \min \{ \max(0, p0 + \Delta), 2^{BC} - 1 \}, \\
 q'0 &= \min \{ \max(0, q0 + \Delta), 2^{BC} - 1 \},
 \end{aligned}$$

Таблица 10.3. Зависимости α' , β' от величин D_A , D_B

	D_A для α' ; D_B для β'																	
1, ..., 15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	
α'	0	4	4	5	6	7	8	9	10	12	13	15	17	20	22	25	28	32
β'	0	2	2	2	3	3	3	4	4	4	6	6	7	7	8	8	9	

	D_A для α' ; D_B для β'																		
	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51
α'	36	40	45	50	56	63	71	80	90	101	113	127	144	162	182	203	226	255	255
β'	9	10	10	11	11	12	12	13	13	14	14	15	15	16	16	17	17	18	18

где $\Delta = \min \{ \max(-t_C, [(4(q0-p0)+(p1-q1)+4)/8]), t_C \}$, а порог t_C определяется по формулам при обработке сигнала яркости

$$t_C = t_{C0} + \begin{cases} 1, & \text{при } |p2 - p0| < \beta \\ 0, & \text{иначе} \end{cases} + \begin{cases} 1, & \text{при } |q2 - q0| < \beta \\ 0, & \text{иначе} \end{cases},$$

$t_{C0} = t'_{C0} \cdot 2^{(BY-8)}$, при обработке сигнала цветности $t_C = t_{C0} + 1$ и $t_{C0} = t'_{C0} \times 2^{(BC-8)}$.

Переменная t'_{C0} определена в табл. 10.4 в зависимости от значений D_A и bS. Результирующие значения фильтрации пикселя p'1 определяются следующим образом: при обработке сигнала яркости и при

$$|p2 - p0| < \beta - p'1 = p1 + \min \{ \max(-t_{C0}, [(p2 + (p0 + q0 + 1)/2 - -2 \cdot p1)/2]), t_{C0} \}, \quad (10.59)$$

при обработке сигнала цветности или при $|p2 - p0| \geq \beta - p'1 = p1$. Результирующие значения фильтрации пикселя q'1 определяются следующим образом: при обработке сигнала яркости и при

$$|q2 - q0| < \beta - q'1 = q1 + \min \{ \max(-t_{C0}, [(q2 + (p0 + q0 + 1)/2 - -2 \cdot q1)/2]), t_{C0} \}, \quad (10.60)$$

при обработке сигнала цветности или при $|q2 - q0| \geq \beta - q'1 = q1$.

Результирующие значения фильтрации пикселов p'2 и q'2 всегда устанавливаются равными значениям входных пикселов:

$$p'2 = p2; \quad q'2 = q2. \quad (10.61)$$

Таблица 10.4. Значения t'_{C0} в зависимости от D_A и bS

	D_A														
	0...16	17...20	21,22	23...26	27...30	31,32	33	34	35,36	37	38,39	40	41	42	
bS = 1	0	0	0	1	1	1	2	2	2	3	3	4	4	4	
bS = 2	0	0	1	1	1	2	2	2	3	3	4	5	5	6	
bS = 3	0	1	1	1	2	3	3	4	4	5	6	7	8	9	

	D_A														
	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56
bS = 1	4	5	6	6	7	8	9	10	11	13	14	15	16	17	18
bS = 2	6	7	8	8	10	11	12	13	15	17					
bS = 3	9	10	11	13	14	16	18	20	23	25					

Процесс фильтрации краев при $bS = 4$

При этом производится фильтрации значений примыкающих друг к другу пикселов p_i и q_i ($i = 0 \dots 3$). Результатирующие значения фильтрации пикселов $p'i$ ($i = 0 \dots 2$) находят следующим образом: при обработке сигнала яркости, если выполняются условия $|p2 - p0| < \beta$ и $|p0 - q0| < (\alpha/4 + 2)$, то

$$\begin{aligned} p'0 &= (p2 + 2 \cdot p1 + 2 \cdot p0 + 2 \cdot q0 + q1 + 4)/8, \\ p'1 &= (p2 + p1 + p0 + q0 + 2)/4, \\ p'2 &= (2 \cdot p3 + 3 \cdot p2 + p1 + p0 + q0 + 4)/8; \end{aligned} \quad (10.62)$$

при обработке сигнала цветности и если не выполняются условия, указанные в (10.61), то

$$\begin{aligned} p'0 &= (2 \cdot p1 + p0 + q1 + 2)/4, \\ p'1 &= p1, \\ p'2 &= p2. \end{aligned} \quad (10.63)$$

Результатирующие значения фильтрации пикселов $q'i$ ($i = 0 \dots 2$) находят следующим образом:

при обработке сигнала яркости, если выполняются условия $|q2 - q0| < \beta$ и $|p0 - q0| < (\alpha/4 + 2)$, то

$$\begin{aligned} q'0 &= (p1 + 2 \cdot p0 + 2 \cdot q0 + 2 \cdot q1 + q2 + 4)/8, \\ q'1 &= (p0 + q0 + q1 + q2 + 2)/4, \\ q'2 &= (2 \cdot q3 + 3 \cdot q2 + q1 + q0 + p0 + 4)/8. \end{aligned} \quad (10.64)$$

при обработке сигнала цветности и если не выполняются условия, указанные в (10.63), то

$$\begin{aligned} q'0 &= (2 \cdot q1 + q0 + p1 + 2)/4, \\ q'1 &= q1, \\ q'2 &= q2. \end{aligned} \quad (10.65)$$

На рис. 10.10 приведены примеры результатов фильтрации изображения при различных параметрах коэффициента квантования (изображения взяты из книги [4.17]).



Рис. 10.10. Пример фильтрации блочного эффекта в изображениях при двух значениях коэффициента квантования Q_{cp} : а — исходное изображение; б и в — изображения при отсутствии фильтрации; $Q_{cp}=36$ и $Q_{cp}=32$ соответственно; г и д — субъективно заметные улучшения изображений б и в соответственно при включенной процедуре фильтрации

ГЛАВА 11

ОБРАБОТКА ИЗОБРАЖЕНИЙ ПРИ КРАТНОМАСШТАБНОМ АНАЛИЗЕ

Важной проблемой при кратномасштабной обработке изображений (их кодировании и сжатии) является выбор вейвлет-базиса. В общем случае это — сложная задача, не решенная до настоящего времени. Известен ряд критериев, определяющих качество того или иного вейвлет-базиса, среди которых наиболее важными являются гладкость, точность восстановления изображения, частотная избирательность фильтров. Одним из возможных подходов к решению этой проблемы является оценка изменений распределения энергии по высокочастотным компонентам в зависимости от выбранного для обработки вейвлет-базиса [4.19].

Как уже отмечалось в главе 9, наиболее часто используется последовательное разложение изображения по столбцам и строкам. Такой вид обработки соответствует использованию разделимых двумерных фильтров, импульсная характеристика которых есть тензорное произведение импульсных характеристик соответствующих одномерных фильтров. При использовании двухполосных вейвлет-фильтров частотная плоскость изображения разбивается на четыре области.

В этом случае используется последовательное применение преобразования сначала к строкам, а затем к столбцам изображения. При этом сначала формируются 2 области (левая — низкочастотная и правая — высокочастотная), а затем каждая из них разбивается еще на две области (рис. 11.1). Одна из этих областей НН — низкочастотная (в левом верхнем углу) и три высокочастотных области: НВ (в нижнем левом углу), ВН (в верхнем правом углу), ВВ (в нижнем правом углу).

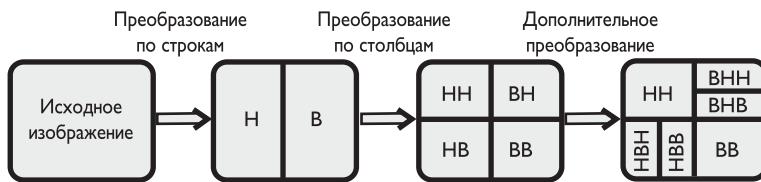


Рис. 11.1. Структурные схемы вейвлет-разложения изображения на четыре области и шесть областей

Если необходимо, данная процедура может быть снова применена к НН-компоненте изображения, в результате чего вместо нее будет сформировано еще 4 частотных компонента. Для оценки изменений распределения энергии по высокочастотным компонентам в этом нет необходимости. В данном случае рационально дополнительное преобразование, примененное к НВ- и ВН-компонентам изображения. В результате чего все изображение разделяется на 6 частотных областей [4.20–4.22]: одну низкочастотную и пять высокочастотных, как показано на рис. 11.1.

11.1. Оценка энергетических характеристик изображения в зависимости от вида вейвлет-базиса разложения

Оценка энергетических характеристик различных областей вейвлет-разложения изображения производилась путем исследования зависимостей средней величины яркости пикселя в той или иной области изображения от выбранного базиса, при этом среднее значение яркости в каждой области оценивалась соотношением:

$$\bar{Y}_k = \sqrt{\frac{\sum_{i_k=0}^{(N_k-1) \times (M_k-1)} Y_{i_k}^2}{N_k \times M_k}}, \quad (11.1)$$

где $N_k \times M_k$ — размер k -ой выделенной области изображения.

Для устранения зависимости энергетической характеристики от конкретного изображения и его размера использовалась относительная величина (относительно суммарной яркости всех компонент), например для компоненты НВН она определялась следующим образом:

$$\bar{I}_{\text{НВН}} = \frac{\bar{Y}_{\text{НВН}}}{\bar{Y}_{\text{НН}} + \bar{Y}_{\text{НВН}} + \bar{Y}_{\text{НВВ}} + \bar{Y}_{\text{ВНН}} + \bar{Y}_{\text{ВНВ}} + \bar{Y}_{\text{ВВ}}}. \quad (11.2)$$

Для исследования энергетических характеристик преобразования от типа вейвлет-базиса было использовано 7 тестовых изображений (рис. 11.2), их названия и основные параметры приведены в табл. 11.1.



Рис. 11.2. Тестовые изображения: а — «Лена», б — «Барбара», в — «Золотой холм», г — «Залив», д, е, ж — 206, 444, 1165 кадры из фильма «Матрица» соответственно

Каждое из этих изображений имеет свои отличительные особенности:

- изображение «Лена» — классическое тестовое изображение, обладающее плавными цветовыми переходами, наклонными границами;
- изображение «Барбара» — черно-белое изображение, множество полосок на

- скатерти, платке и брюках, здесь часто возникает муар и другие артефакты;
- изображение «Золотой холм» — черно-белое изображение, содержащее множество мелких деталей как на переднем, так и на заднем планах;
 - изображение «Залив» — классическое тестовое изображение, содержащие большое количество вертикально ориентированных элементов.

Таблица 11.1. Типы тестовых изображений

Название изображения	Исходный размер	Разрешение
«Лена»	786 486 байтов	512 × 512
«Барбара»	786 486 байтов	512 × 512
«Золотой холм»	786 486 байтов	512 × 512
«Залив»	786 486 байтов	512 × 512
Кадры из фильма «Матрица»	0206	1 244 214 байтов
	0444	1 244 214 байтов
	1165	1 244 214 байтов
		720 × 576

Три последних изображения являются яркими примерами телевизионных изображений, спектр которых содержит небольшое число высокочастотных элементов:

- изображение 206-го кадра из фильма «Матрица» характеризуется плавными цветовыми переходами и невысокой контрастностью;
- изображение 444-го кадра из фильма «Матрица» имеет большое количество вертикально и горизонтально ориентированных деталей и также невысокую контрастность изображения;
- изображение 1165-го кадра из фильма «Матрица» характеризуется большим числом мелких прямолинейных элементов, произвольно ориентированных в пространстве.

При оценке зависимостей относительных энергетических величин высокочастотных областей преобразованных изображений $\bar{I}_{\text{НВН}}$, $\bar{I}_{\text{НВВ}}$, $\bar{I}_{\text{ВНН}}$, $\bar{I}_{\text{ВНВ}}$, $\bar{I}_{\text{ВВ}}$ использовались КИХ-фильтры нижних/верхних частот, обозначаемые количеством отсчетов цифровой решетки соответственно (см. главу 9): 2/2; 3/5; 4/8; 5/7; 6/6; 7/9; 8/8; 9/11; 10/10.

На рис. 11.3–11.9 приведены результаты этих исследований. В каждом из этих рисунков имеются два графика: на левом графике (а) кривые характеризуют изменения усредненных яркостей соответствующих областей разделенного изображения, а на правом графике (б) — распределение усредненных яркостей по высокочастотным компонентам после обратного вейвлет-преобразования до проведения операции суммирования отдельных компонент. По оси абсцисс на этих графиках приводятся длины используемых КИХ-фильтров нижних частот — от двух до десяти.

Оценивая результаты исследований, можно сделать ряд выводов:

- с ростом числа отсчетов цифровой решетки низкочастотного и высокочастотного фильтров, т. е. с увеличением ширины полосы пропускания ФНЧ и уменьшением соответствующей полосы ФВЧ, наблюдается небольшое уменьшение энергии в высокочастотных компонентах только при переходе от фильтров с 2 или 3 отсчетами цифровой решетки к фильтрам, имеющим

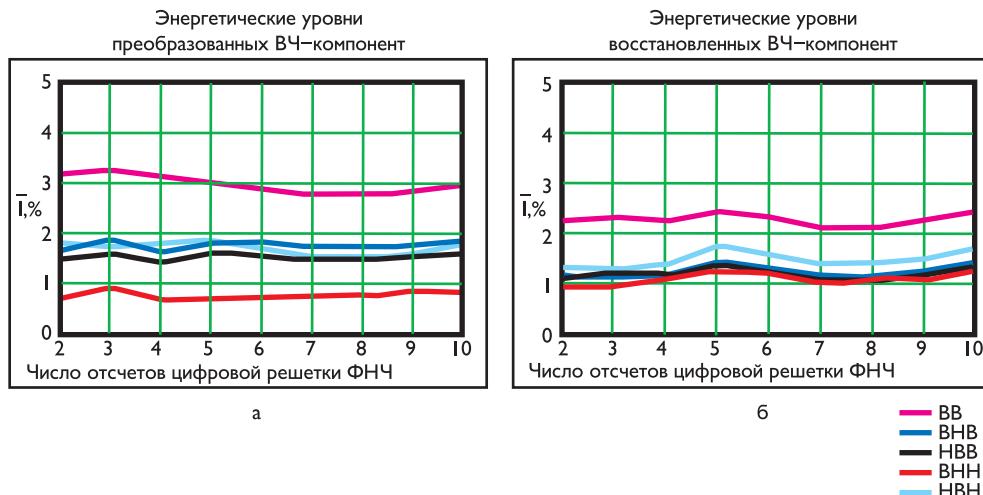


Рис. 11.3. Распределение энергии по высокочастотным компонентам изображения «Лена» после прямого преобразования — а и восстановления составляющей — б

большее количество таких отсчетов, дальнейшее уменьшение мало заметно или вообще отсутствует;

- по мере увеличения числа отсчетов цифровой решетки фильтров, очевидно, требуется увеличение времени на выполнение операции дискретного вейвлет-преобразования, поэтому в реальных системах сжатия изображений необходим компромисс между выбранным размером носителя вейвлет-базиса и скоростью обработки информации;
- из полученных результатов следует, что в различных высокочастотных компонентах содержатся разные части энергии изображения, и к этим компонентам возможно применять различные маски квантования.

11.2. Выбор маски квантования высокочастотных компонент вейвлет-преобразования изображений

Разбиение изображения на одну низкочастотную область и большое количество высокочастотных областей позволяет сконцентрировать основную энергию в НН-области и обеспечивает возможность квантования высокочастотных компонент со значительно меньшим количеством уровней.

После вейвлет-преобразования в каждой из высокочастотных областей изображения содержится различное количество энергии. Поэтому к каждой из этих областей изображения возможно применение своей маски квантования, незначительно ухудшающей качество изображения.

Соответственно для описания случайной величины — яркости пикселов в каждой из высокочастотных компонент вейвлет-преобразования — можно использовать одномерную плотность распределения вероятностей, с достаточной степе-

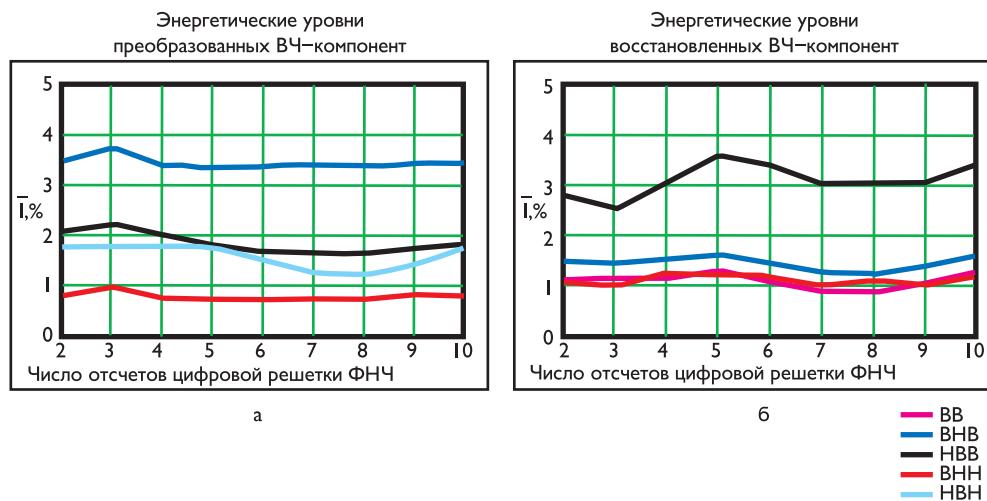


Рис. 11.4. Распределение энергии по высокочастотным компонентам изображения «Барбара» после прямого преобразования — а и восстановления составляющей — б

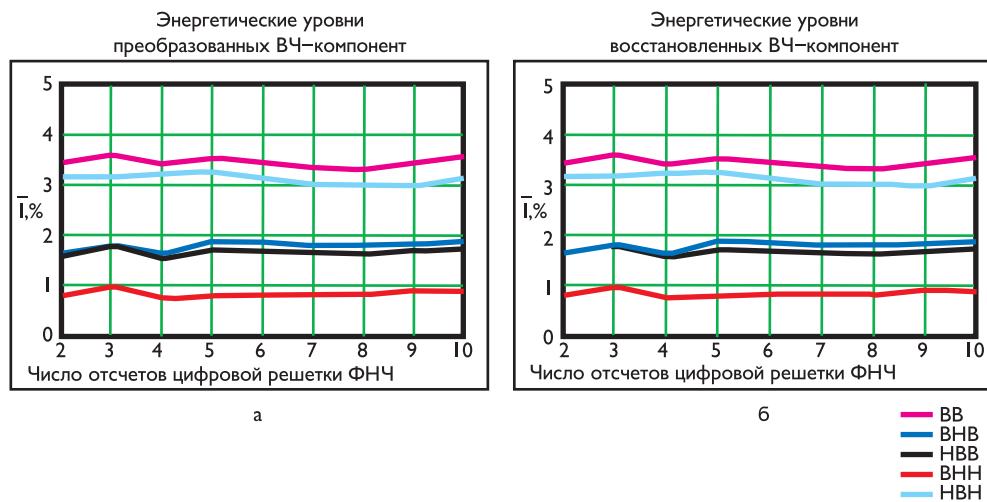


Рис. 11.5. Распределение энергии по высокочастотным компонентам изображения «Золотой холм» после прямого преобразования — а и восстановления составляющей — б

нюю точности подчиняющейся закону Лапласа:

$$w(x) = \frac{\lambda}{2} e^{-\lambda|x|}. \quad (11.3)$$

Величину λ можно найти, зная среднеквадратичное отклонение (СКО) или усредненное значение модулей дискретных значений (Mod) данного распределения, т. к. по определению для непрерывного случайного процесса с нулевым математическим ожиданием:

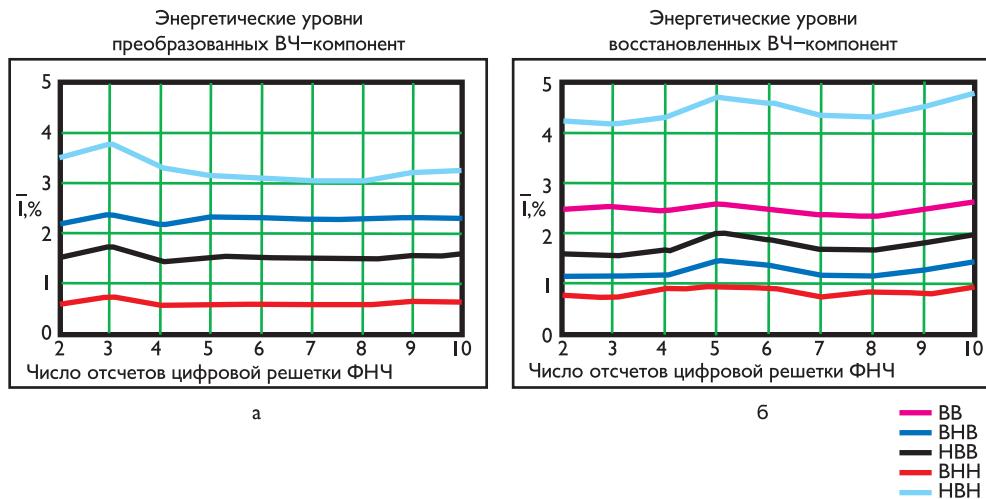


Рис. 11.6. Распределение энергии по высокочастотным компонентам изображения «Залив» после прямого преобразования — а и восстановления составляющей — б

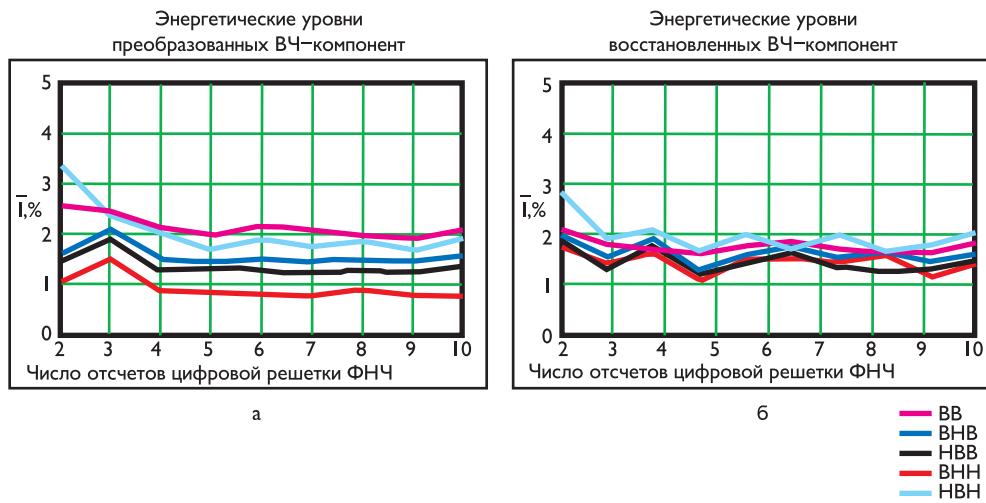


Рис. 11.7. Распределение энергии по высокочастотным компонентам изображения 206-го кадра из фильма «Матрица» после прямого преобразования — а и восстановления составляющей — б

$$\text{СКО} \cong \sqrt{\frac{1}{M \times N} \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} Y_{nm}^2}, \quad \text{Mod} \cong \frac{1}{M \times N} \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} |Y_{nm}|, \quad (11.4)$$

где Y_{nm} и $M \times N$ — соответственно значения и количество отсчетов в исследуемой области изображения.

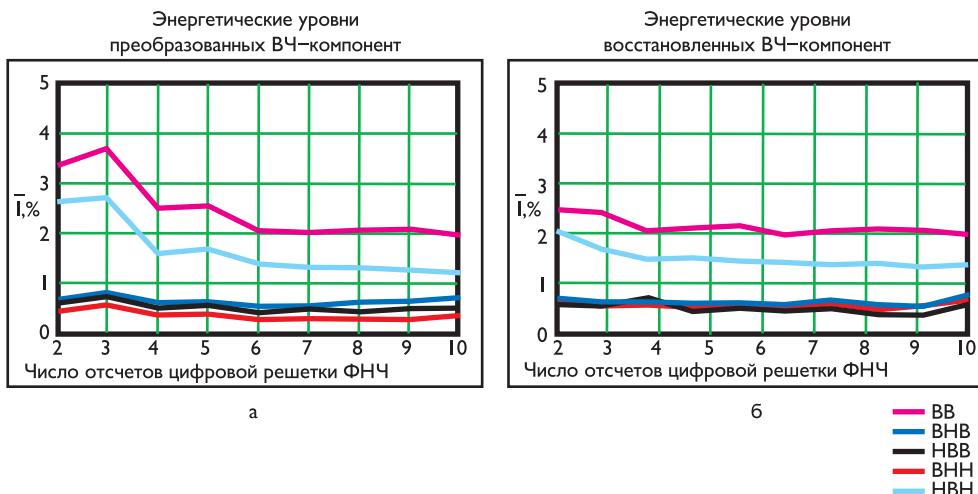


Рис. 11.8. Распределение энергии по высокочастотным компонентам изображения 444-го кадра из фильма «Матрица» после прямого преобразования — а и восстановления составляющей — б

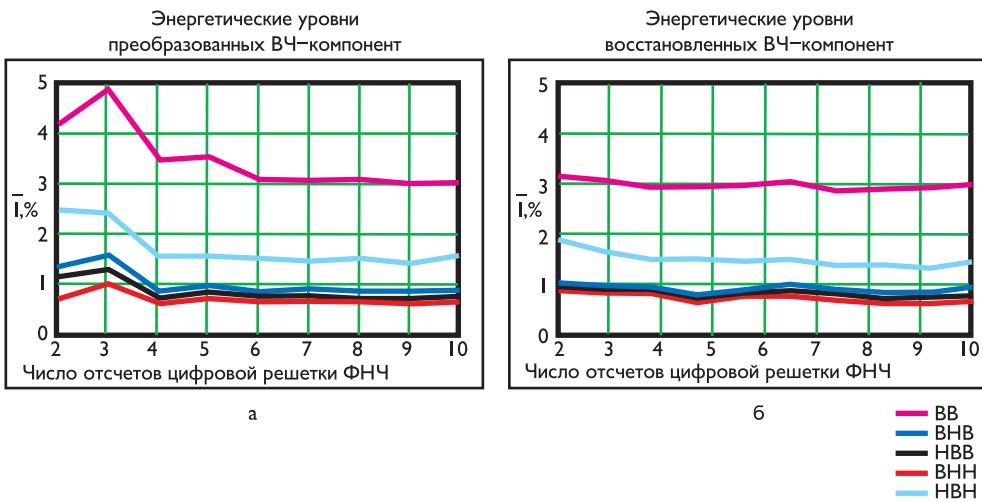


Рис. 11.9. Распределение энергии по высокочастотным компонентам изображения 1165-го кадра из фильма «Матрица» после прямого преобразования — а и восстановления составляющей — б

Используя формулы (11.3) и (11.4), нетрудно показать, что:

$$\lambda = \frac{\sqrt{2}}{\text{СКО}} = \frac{1}{\text{Mod}}. \quad (11.5)$$

Выбор количества уровней квантования проводился экспериментальным путем на основе:

- практического отсутствия визуального улучшения качества изображения при переходе к большему числу уровней квантования;

- уменьшения отношения сигнала–шум (PSNR) не более чем на 0,8 дБ для каждой квантованной и восстановленной высокочастотной компоненты (т. е. после обратного вейвлет-преобразования) по сравнению с неквантованной при общем уменьшении PSNR восстановленного изображения не более чем на 3 дБ.

Учитывая, что обычно используется 8-разрядная шкала для сигналов изображения:

$$\text{PSNR} = 20 \log \frac{256}{\text{СКО}}. \quad (11.6)$$

В табл. 11.2 приведены результаты расчета параметра λ для пяти различных высокочастотных областей использованных тестовых изображений при вейвлет-преобразовании с помощью КИХ-фильтров с числовой решеткой 10/10.

Таблица 11.2. Параметр λ для различных областей тестовых изображений

Изображение		НВН	НВВ	ВНН	ВНВ	ВВ
«Лена»	Y	0,151	0,161	0,093	0,14	0,35
	Cr	0,645	0,73	0,516	0,68	1,262
	Cb	0,593	0,675	0,479	0,632	1,202
«Барбара»	Y	0,168	0,177	0,024	0,047	0,314
«Золотой холм»	Y	0,081	0,148	0,077	0,135	0,319
«Залив»	Y	0,084	0,165	0,041	0,095	0,362
	Cr	1,181	1,762	0,752	1,682	3,101
	Cb	1,01	1,536	0,646	1,453	2,84
Кадры из фильма «Матрица»	206	Y	0,482	0,669	0,42	0,62
	444	Y	0,198	0,47	0,114	0,44
	1165	Y	0,212	0,382	0,092	0,301
						1,01

Число уровней квантования выбиралось из ряда $0, 3, 7, 15, \dots, (2^n - 1)$. В результате исследования большого количества различных изображений была сформирована приведенная в табл. 11.3 маска выбора числа уровней квантования в зависимости от параметра λ .

Таблица 11.3. Маска числа уровней квантования вейвлет-преобразований изображений

Параметр λ	$\lambda > 1$	$1 \geq \lambda > 0,4$	$0,4 \geq \lambda > 0,15$	$0,15 \geq \lambda > 0,05$	$0,05 \geq \lambda > 0,02$	$0,02 \geq \lambda$
Число уровней квантования	0	3	7	15	31	63

Значение $\lambda < 0,02$ свойственно в основном текстурам (искусственным изображениям) и поэтому редко встречается при естественных и телевизионных изображениях.

Каждый из диапазонов значений λ может быть разбит на более мелкие поддиапазоны, для которых формируются свои маски квантования при заданном числе уровней квантования, что также может позволить несколько увеличить величину PSNR.

Соответственно, выбор параметров неравномерного квантования осуществлялся на основе метода Ллойда–Макса [4.23] применительно к усредненному

значению коэффициента λ . Полученные маски квантования для количества уровней квантования, равного 3, 7 и 15, представлены в табл. 11.4. В этой таблице приведена только положительная часть пороговых уровней и уровней квантования (отрицательный участок маски квантования симметричен).

Таблица 11.4. Маски уровней квантования

Количество уровней	Номер маски	Значения λ	Пороговые уровни и уровни квантования							
			[-1,1]	[2, ...]						
3	1	0,6 ... 1	[-1,1]	[2, ...]						
			0	3						
7	2	0,6 ... 0,4	[-2,2]	[3, ...]						
			0	3						
7	3	0,4 ... 0,24	[-2,2]	[3,5]	[6,10]	[11, ...]				
			0	3	7	16				
15	4	0,24 ... 0,15	[-3,3]	[4,7]	[8,14]	[15, ...]				
			0	4	9	23				
15	5	0,15 ... 0,08	[-3,3]	[4,6]	[7,11]	[12,18]	[19,27]	[28,38]	[39,51]	[52, ...]
			0	4	8	14	23	33	45	60
15	6	0,08 ... 0,05	[-3,3]	[4,7]	[8,14]	[15,24]	[25,35]	[36,52]	[53,69]	[70, ...]
			0	4	9	19	30	43	61	82

Естественно, квантование высокочастотных компонент вейвлет-преобразований изображений приводит к уменьшению объема информативных составляющих и уменьшению величины PSNR, что иллюстрируется табл. 11.5.

Таблица 11.5. Результаты квантования высокочастотных областей тестовых изображений

Изображение	Уменьшение объема информации		PSNR, дБ
«Лена»		4,12	34
«Барбара»		4,17	34,5
«Золотой холм»		4,17	36
«Залив»		4,24	36
Кадры из фильма «Матрица»	0206	6,32	32,3
	0444	6,0	35,8
	1165	5,5	38,8

11.3. Обработка тестовых изображений двухполосными вейвлет-фильтрами

На рис. 11.10а–г приведены результаты разложения тестовых изображений «Лена», «Барбара», «Золотой холм» и «Залив», на 4 и 6 областей каждое. Величины коэффициентов области НН уменьшены в 2 раза, а величины коэффициентов всех высокочастотных областей, расположенных на уровне серого, увеличены в 4 раза.

Из приведенных рисунков видно, что разбиение на 6 областей позволяет сконцентрировать энергию в более низкочастотных компонентах, что обеспечивает квантование высокочастотных компонент с меньшим количеством уровней.

После вейвлет-преобразования в каждой из высокочастотных компонент изображения содержится различное количество энергии. Поэтому к каждой из компонент изображения возможно применение своей маски квантования, незначительно ухудшающей качество изображения.

В табл. 11.6 приведены результаты уменьшения объема информации после преобразования и квантования при применении разбиения изображений на 4 и на 6 областей при условии $PSNR \approx 37$ дБ.

Следует отметить, что в данном случае рассмотрено только уменьшение объема информации, связанное с вейвлет-разложением и последующим квантованием, без учета последующего кодирования.

Таблица 11.6. Изменение объема информации при квантовании высокочастотных областей

Название изображения	Уменьшение объема информации	
	Разбиение на 4 области	Разбиение на 6 областей
«Лена»	1,68	2,05
«Барбара»	1,60	1,94
«Золотой холм»	1,68	2,05
«Залив»	1,60	1,94

Разложение изображения на 6 областей приводит к некоторому увеличению числа выполняемых операций в кодирующем устройстве, число же операций, выполняемых декодером, возрастает незначительно.

Таким образом, вейвлет-разложение на 6 областей позволяет дополнитель но уменьшить примерно на 20% объем информации по сравнению с разложением на 4 области при применении одной и той же маски квантования и одинакового уровня $PSNR$.

11.4. Обработка тестовых изображений трехполосными вейвлет-фильтрами

Для повышения эффективности преобразования изображений возможно производить разделение сигналов при помощи трехполосных фильтров (см. главу 9), которые делят сигнал на три составляющие (низкочастотную, среднечастотную и высокочастотную).

В этом случае вейвлет-преобразование выполняется по стандартной схеме: сначала преобразованию подвергаются все строки изображения, в результате чего формируются 3 области (низкочастотная, среднечастотная и высокочастотная); затем вейвлет-преобразование применяется ко всем столбцам полученных коэффициентов, в результате чего получается 9 областей. Структурная схема вейвлет-разложения изображения и обозначение полученных областей представлены на рис. 11.11.

Для вейвлет-преобразования по данной схеме может использоваться большой набор банков вейвлет-фильтров с разной цифровой решеткой импульсных

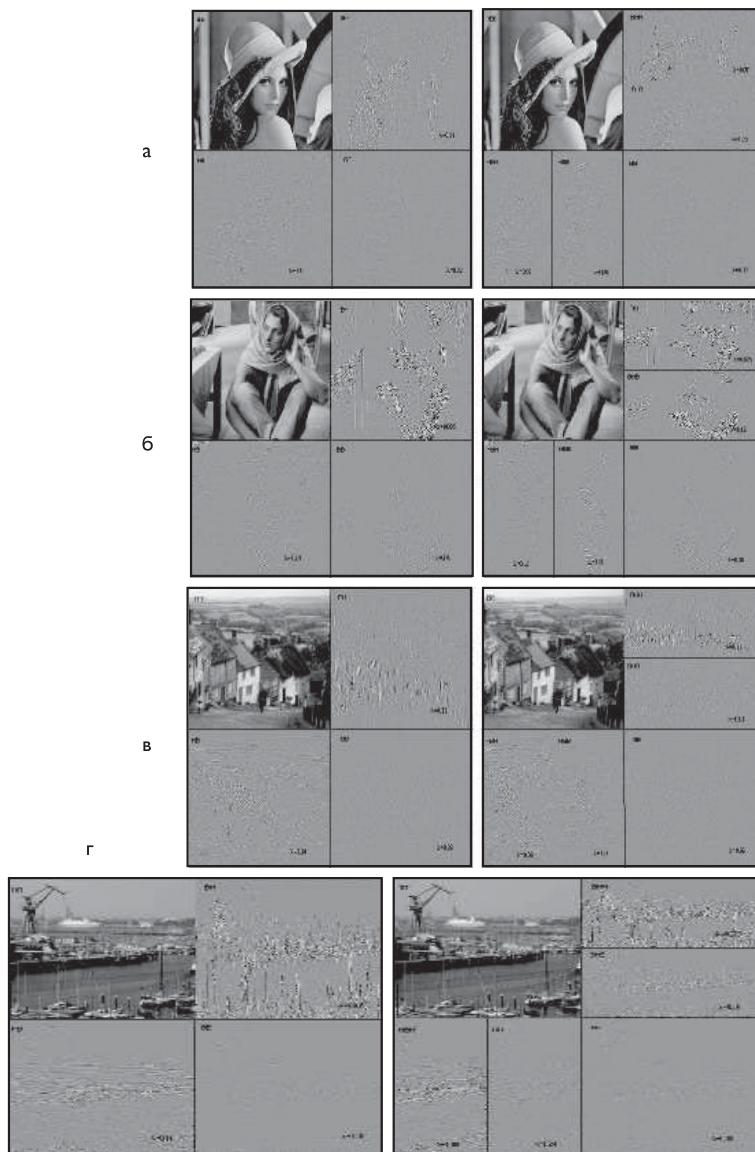


Рис. 11.10. Разложение тестовых изображений «Лена» (а), «Барбара» (б), «Золотой холм» (в), «Залив» (г) на 4 и 6 областей

характеристик (ИХ) низкочастотных (ФНЧ), среднечастотных (ФСЧ) и высокочастотных (ФВЧ) фильтров. Следует отметить, что с ростом числа отсчетов в ИХ увеличивается ширина полосы пропускания НЧ-фильтра и уменьшается соответствующая полоса ВЧ-фильтра, при этом наблюдается небольшое уменьшение энергии в среднечастотных и высокочастотных компонентах.

С другой стороны, по мере увеличения числа отсчетов в ИХ растет количество вычислительных операций при выполнении дискретного вейвлет-преобразования.

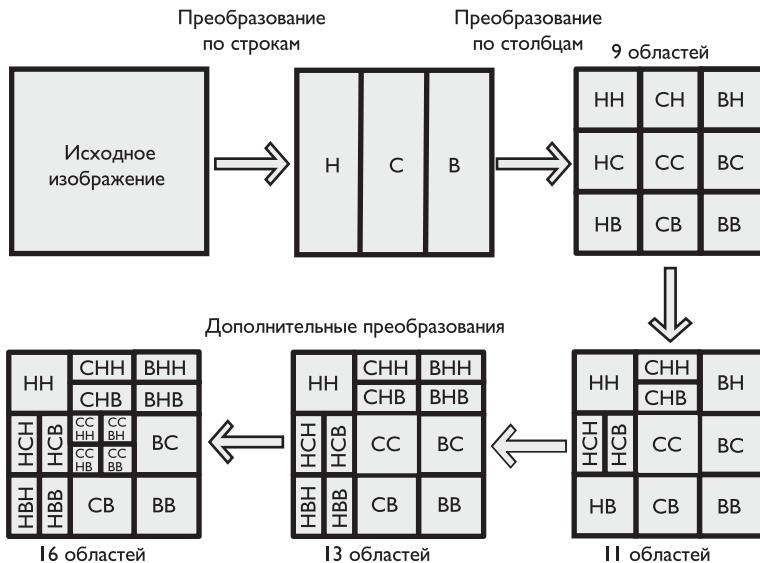


Рис. 11.11. Структурная схема вейвлет-разложения изображения на 9 областей

ния, поэтому в реальных системах сжатия изображений при помощи вейвлет-преобразования необходим компромисс между длиной ИХ вейвлет-базиса и требуемым объемом вычислений.

На рис. 11.12а–г приведены результаты разложения четырех тестовых изображений на 9 областей. Величина НН-коэффициентов уменьшена в 3 раза, величина всех высокочастотных коэффициентов, расположенных на уровне серого, увеличена в 5 раз.

Из приведенных рисунков видно, что разбиение на 9 компонент позволяет сконцентрировать основную энергию изображения в низкочастотной и среднечастотной компонентах, что обеспечивает квантование высокочастотных компонент со значительно меньшим количеством уровней.

В табл. 11.7 приведены результаты уменьшения объема информации после преобразования и квантования при применении разбиения на 9 компонент по сравнению с разбиением на 4 компоненты при условии $PSNR \approx 37$ дБ.

Таблица 11.7. Изменение объема информации при разбиении изображения на 4 и 9 областей

Название изображения	Уменьшение объема информации	
	Разбиение на 4 области	Разбиение на 9 областей
«Лена»	1,68	2,32
«Барбара»	1,60	2,16
«Золотой холм»	1,68	2,16
«Залив»	1,60	2,32

Таким образом, вейвлет-разложение на 9 компонент позволяет дополнительно уменьшить примерно на 40% объем информации по сравнению с разложением на 4 компоненты при применении одной и той же маски квантования и одинаковой величины $PSNR$.

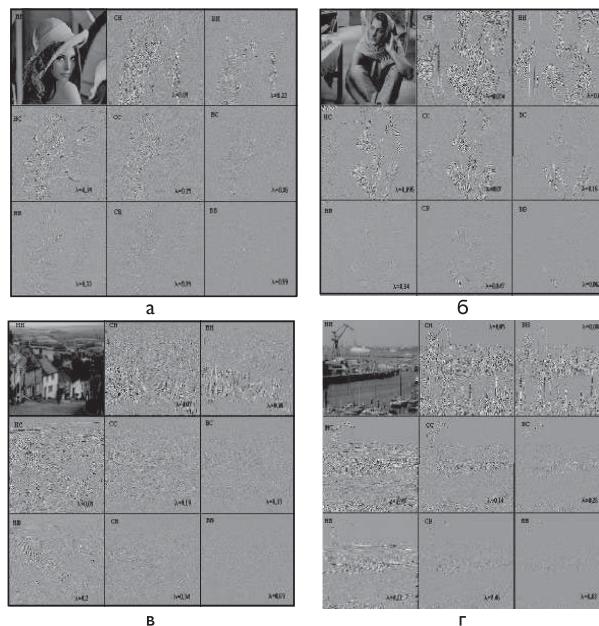


Рис. 11.12. Разложение тестовых изображений «Лена» (а), «Барбара» (б), «Золотой холм» (в), «Залив» (г) на 9 областей

Для еще большей эффективности преобразования изображений высокочастотные и среднечастотные компоненты трехполосного вейвлет-преобразования можно дополнить двухполосным вейвлет-преобразованием, при этом в зависимости от глубины преобразования возможно разложение на 11, 13 или 16 компонент (см. рис. 11.11).

На рис. 11.13а–в приведены вейвлет-разложения изображения «Залив» с разбиением на 11, 13 и 16 областей. Величина коэффициентов НН-области уменьшена в 3 раза, величина коэффициентов всех остальных областей, расположенных на уровне серого, увеличена в 5 раз.

В табл. 11.8 приведены средние значения параметра λ для компонент по четырем тестовым изображениям при разложении на 4 и 6 областей.

Таблица 11.8. Средние значения параметра λ по 4 и 6 областям

Количество областей	Средние значения λ			
	НВ		ВН	
4	0,13	0,14		0,42
6	НВН	НВВ	ВНН	ВНВ
	0,09	0,20	0,10	0,22
	0,42			

В следующей табл. 11.9 приведены средние значения параметра λ по четырем тестовым изображениям при разложении на 9, 11, 13 и 16 областей соответственно.

В табл. 11.10 приведены результаты уменьшения объема информации, содержащейся в преобразованном изображении при применении разбиения на 4, 6, 9, 11, 13 и 15 компонент соответственно при условии $PSNR \geq 37$ дБ.

Таблица 11.9. Средние значения параметра λ по 9, 11, 13 и 16 областям

Количество областей	Среднее значение λ для всех областей														
	HC		CH		HB		BH		CC		CB	BC	BB		
9	0,09		0,08		0,15		0,13		0,18		0,36	0,32	0,7		
11	HCH	HCB	HCH	HCB	0,15		0,13		0,18		0,36	0,32	0,7		
	0,06	0,16	0,05	0,15											
13	0,06	0,16	0,05	0,15	HBH	HBB	BHH	BHB	0,18		0,36	0,32	0,7		
					0,13	0,25	0,12	0,24							
16	0,06	0,16	0,05	0,15	0,13	0,25	0,12	0,24	HH	BH	HB	BV	0,36	0,32	0,7
									0,14	0,35	0,33	0,62			

Таблица 11.10. Уменьшение объема информации при вейвлет-преобразовании изображений

Название изображения	Уменьшение объема информации в зависимости от количества областей в разбиении					
	4	6	9	11	13	16
«Лена»	1,68	2,05	2,32	2,51	2,61	2,84
«Барбара»	1,60	1,94	2,16	2,34	2,46	2,63
«Золотой холм»	1,68	2,05	2,16	2,29	2,42	2,67
«Залив»	1,60	1,94	2,32	2,49	2,58	2,81

При этом под уменьшением объема информации понимается отношение количества битов, требуемых для передачи исходных тестовых монохромных изображений, к количеству битов после вейвлет-преобразования и последующего квантования высокочастотных областей.

Таким образом, разложение изображения на большое количество областей приводит к некоторому увеличению числа выполняемых операций в кодирующем устройстве по сравнению со стандартным вейвлет-разложением на 4 компоненты. Число же операций, выполняемых декодером, в силу малого числа ненулевых отсчетов в среднечастотных и высокочастотных компонентах возрастает незначительно.

При стандартном вейвлет-разложении изображений на 4 компоненты и последующем квантовании высокочастотных областей объем информации в среднем может быть уменьшен в 1,64 раза. Дополнительная обработка двух высокочастотных областей этих вейвлет-преобразований позволяет дополнительно уменьшить объем информации на 25%.

При вейвлет-разложении изображений на 9, 11, 13 и 16 областей объем информации в среднем уменьшается по сравнению с разложением на 4 компоненты на 35%, 45%, 50% и 60% соответственно.

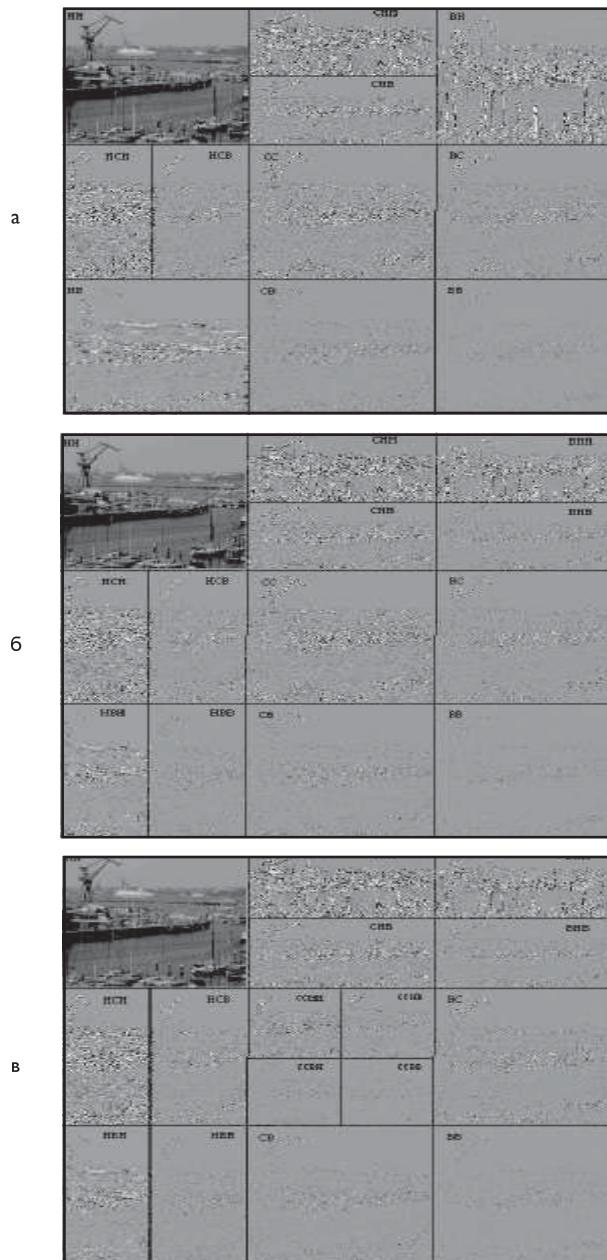


Рис. 11.13. Разложение изображения «Залив» на 11 (а), 13 (б) и 16 (в) областей

ГЛАВА 12

МЕТОДЫ АНАЛИЗА И КОМПЕНСАЦИИ ДВИЖЕНИЯ В ДИНАМИЧЕСКИХ ИЗОБРАЖЕНИЯХ

Можно привести по крайней мере две проблемы, не позволяющие практически использовать трехмерную обработку последовательности кадров динамических изображений.

Во-первых, в процессе развития сюжета во времени отдельные объекты перемещаются в плоскости кадра из-за собственного движения или перемещения камеры. При этом соответствовать друг другу по положению на объекте и, следовательно, иметь наибольшую корреляцию величин яркости будут несоответствующие друг другу по координатам в кадре пиксели. В таких случаях применение трехмерных преобразований к прямоугольным блокам уже не представляется оптимальным.

Вторая проблема связана с происходящей время от времени сменой сюжета. При этом изменения в изображении могут быть глобальными, во всем пространстве кадра (например, когда режиссер переключает трансляцию с одной камеры на другую), или локальными, охватывающими лишь несколько ограниченных участков изображения, которые могут быть связаны со скачкообразными перемещениями объектов в сюжете или резкими изменениями их яркости.

Самое значительное распространение в настоящее время получил метод гибридной обработки движущихся изображений, в котором используются двумерное групповое кодирование (например, ДКП) в пространстве и анализ движения внутри трехмерного блока по оси времени.

Такой способ позволяет осуществлять так называемую компенсацию движения, т. е. ставить в соответствие предыдущему по времени двумерному блоку блок, смещенный в пространстве на вектор $(\Delta i, \Delta j)$, называемый вектором движения, так что учитывается плоскопараллельное перемещение объекта в изображении за время смены кадра Δt . При этом если Δi и Δj — целые числа, то $b(i + \Delta i, j + \Delta j, t + \Delta t) = b(i, j, t)$, где b — яркости пикселов.

Точное равенство возможно лишь при условии, что движение объекта исчерпывается плоскопараллельным смещением, его освещенность постоянна, градиент яркости фона равен нулю и все объекты в каждом блоке движутся с одной и той же скоростью, выражаемой целыми числами пикселов за смену кадра по обеим пространственным осям. Вообще говоря, вектор движения может быть и нецелочисленным. В общем случае приходится решать задачу минимизации меры различия анализируемых блоков:

$$\min_{\Delta i, \Delta j} \sum_{i, j \in \text{block}} D[b(i + \Delta i, j + \Delta j, t + \Delta t) - b(i, j, t)],$$

где мера $D(\Delta b)$ может быть произвольной.

Как правило, используют квадратичную меру $D(\Delta b) = (\Delta b)^2$ или абсолютную меру $D(\Delta b) = |\Delta b|$, причем последняя предпочтительна с точки зрения экономии вычислительной мощности.

Эта задача может быть решена с помощью ряда различных подходов, каждый из которых предполагает различные возможности реализации.

Заметим, что в результате применения методов анализа движения получается глобальный минимум, локализуемый с точностью до одного пикселя. Такой вектор движения можно уточнять дальше различными способами. Например, стандартами MPEG предусматривается возможность передавать значения векторов движения, определенных с точностью до половины и четверти пикселя. При этом яркость смещенного на долю пикселя элемента определяется с помощью линейной интерполяции, одномерной или двумерной. Вообще говоря, найденный вектор движения можно далее уточнить с любой наперед заданной точностью, передискретизируя изображение (например, с помощью преобразования Фурье).

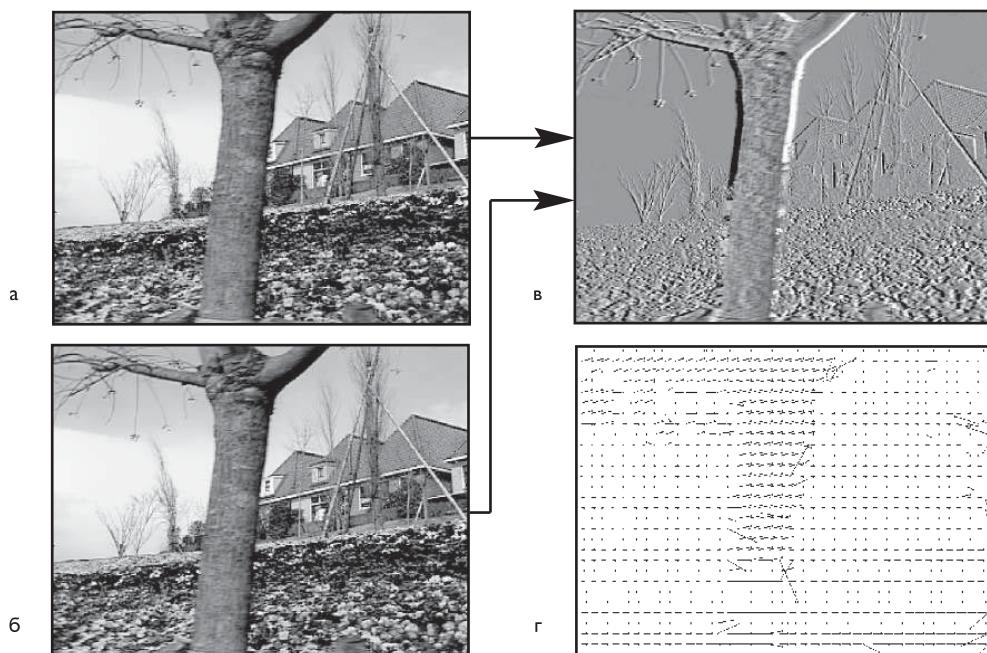


Рис. 12.1. К анализу движения: а и б — опорный и текущий кадры, в — разностный сигнал без учета компенсации движения, г — вектора движения

На рис. 12.1 и 12.2 приведены иллюстрации результатов анализа и компенсации движения с различной точностью:

- рис. 12.2а, б — опорный кадр (в котором производится поиск эквивалентных блоков) и текущий кадр (движение блоков которого анализируется) соответственно;
- рис. 12.2в — прямой разностный сигнал текущего и опорного кадров полученный без учета движения (нулевые вектора движения);
- рис. 12.2г — схема векторов движения блоков текущего кадра;

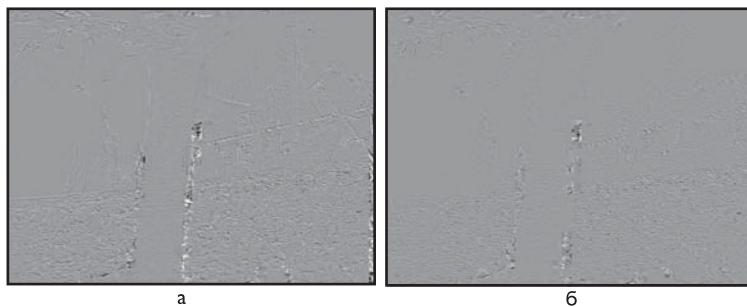


Рис. 12.2. Разностные сигналы с компенсацией движения с точностью до полпикселя — а и с точностью до действительного значения — б

- рис. 12.2а — разностный сигнал текущего и опорного кадров, полученный при компенсации движения с точностью до половины пикселя;
- рис. 12.2б — разностный сигнал текущего и опорного кадров, полученный при компенсации движения с дальнейшим уточнением векторов движения.

Из сравнения изображений на рис. 12.2а и б видно, что эффективная компенсация движения резко уменьшает объем информации о динамическом изображении.

Дополнительно несколько сократить объем вычислений позволяет отказ от выполнения процедуры поиска движения для блоков, вся информация о которых сосредоточена в нулевых (DC) членах косинусного преобразования блоков. Такие блоки можно определить заранее (до запуска процедуры поиска движения), вычисляя их дисперсии.

12.1. Основные методы анализа движения

Известно большое количество схем анализа движения, отличающихся принципом, необходимым числом вычислительных операций, а также степенью сжатия информации за счет компенсации движения.

Согласно стандартам MPEG все кадры видеопоследовательности делятся на 3 основных типа: I, P и B. I-кадр передается без использования предсказания движения (обычно он встречается один раз в группе кадров, и является опорным для кодирования и восстановления остальных кадров). P-кадр кодируется с предсказанием движения в одну сторону («вперед»), и для его восстановления необходим I- или другой опорный P-кадр. Для кодирования P-кадра все изображение разбивается на макроблоки размером, например, 16×16 пикселов, и каждому макроблоку ставится в соответствие наиболее «похожий» участок изображения из опорного кадра, сдвинутый на вектор движения. Кодером передаются только разности сигналов яркости и цветности между макроблоком и соответствующим участком изображения опорного кадра. В случае B-кадра исключение межкадровой избыточности производится с предсказанием во времени в обе стороны, для чего используются два вектора движения и два опорных кадра (I- или P-типа). Такой способ позволяет увеличить сжатие примерно в 2 раза по сравнению с кодированием, только с предсказанием во времени в одну сторону.

В обоих способах для текущего и опорного кадров необходимо определить набор векторов движения макроблоков.

12.1.1. Метод полного перебора

Наиболее простым и универсальным является алгоритм поиска векторов движения макроблоков на основе полного перебора. Данный метод может использоваться в качестве эталонного для оценки качества других методов. Недостатком метода является низкое быстродействие. Для поиска вектора движения $\vec{V} = (V_x, V_y)$ рассматривается норма разницы сигналов яркости макроблока в текущем кадре и участка опорного кадра со сдвигом на вектор движения:

$$H(V_x, V_y) = \sum_{x-x_0, y-y_0=0, \dots, 15} |F(x, y, t) - F(x - V_x, y - V_y, t - 1)|. \quad (12.1)$$

Здесь F — значение яркости, t — временной индекс кадра, (x, y) — пространственные координаты пикселов в кадре, (x_0, y_0) — координаты левого верхнего угла макроблока, суммирование производится по всем пикселям макроблока. Значение \vec{V} , для которого сумма абсолютных разностей $H \leq H_0$ имеет наименьшее значение, принимается за искомый вектор. H_0 — малая величина, характеризующая максимально допустимую точность компенсации движения. Если $H > H_0$, то принимается, что эквивалентный участок изображения в опорном кадре отсутствует и данный макроблок кодируется без компенсации движения.

Векторы движения определяются в заданной ограниченной окрестности $-N \leq V_x, V_y \leq N$. Для нахождения вектора движения одного макроблока необходимо выполнить операции вычитания, взятия модуля и сложения, в сумме их количество равно $3 \cdot 256 \cdot (2N + 1)^2$ операций. На один пиксель макроблока количество операций составляет $3 \cdot (2N + 1)^2$, что уже при типичном значении $N = 15$ равно значительной величине $3 \cdot 10^3$ операций/пиксель.

Поскольку поиск векторов движения является наиболее трудоемкой с точки зрения вычислительных затрат частью алгоритма, эффективность алгоритма анализа движения является критическим фактором, влияющим на возможность эффективной реализации всего алгоритма кодирования. Как было указано выше, методы полного перебора не могут обеспечить поиск векторов движения в сколь-нибудь большом окне поиска, так как объем операций, необходимых для проведения такого поиска, оказывается неприемлемо большим. Поэтому в литературе были опубликованы упрощенные алгоритмы поиска, часть из которых реализована в существующих версиях кодеров систем обработки динамических изображений [4.24].

12.1.2. Логарифмический, комбинированный по двум направлениям, трехшаговый, иерархический методы

Для уменьшения количества операций первоначально были предложены различные простые методы, основанные на ограниченном рассмотрении возможного числа кандидатов в вектора движения.

Так, в *логарифмическом методе* поиск осуществляется в небольшой окрестности с шагом изменения значения вектора движения, равным 2, до тех пор, пока

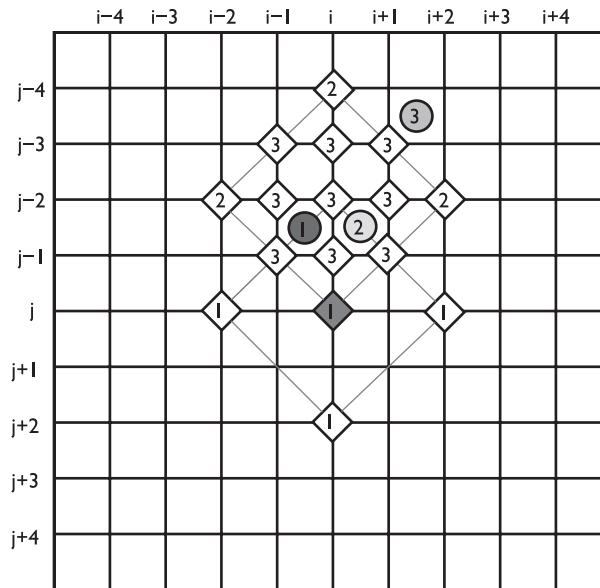


Рис. 12.3. Схема поиска векторов движения в логарифмическом методе

не будет найден локальный минимум H , после чего значение вектора движения уточняется с шагом в один пиксель в окрестности этого минимума, как показано на рис. 12.3 [4.25]. На этом рисунке цифрами обозначен порядок, в котором рассматриваются кандидаты в векторе движения и вычисляется контрольная сумма H .

Методика *комбинированного поиска по двум направлениям* показана на рис. 12.4. В этом методе проводится полный перебор по горизонтали, начиная с центра макроБЛОКА, в результате чего находится точка 1, являющаяся локальным минимумом в строке. Затем поиск ведется по вертикали и находится точка 2, локальный минимум в столбце. Процедура поиска продолжается с чередованием направлений и с уменьшением диапазона поиска вплоть до нахождения конечно-го оптимума [4.26].

В близком к логарифмическому *3-х шаговом методе* поиск векторов движения, как показано на рис. 12.5, ведется первоначально по всей окрестности с шагом в 4 пикселя, затем вокруг найденного минимума — с шагом в 2 пикселя и, наконец, — с шагом в один пиксель [4.27].

В *шерархическом методе* первоначально проводится поиск векторов движения для специальным образом передискретизированного (уменьшенного) изображения, обладающего более грубым разрешением, после чего значения найденных векторов движения уточняются при переходе к следующему уровню более высокого разрешения [4.28], и т. д., пока не будет достигнут уровень разрешения в 1 пиксель. При этом количество необходимых операций для определения вектора зависит от размера окна поиска лишь логарифмически и значительно меньше по сравнению с методом полного перебора. С другой стороны, необходимо отметить, что указанные простые методы часто приводят к существенным ошибкам вследствие достаточно сложной зависимости $H(\vec{V})$. Эта функция может

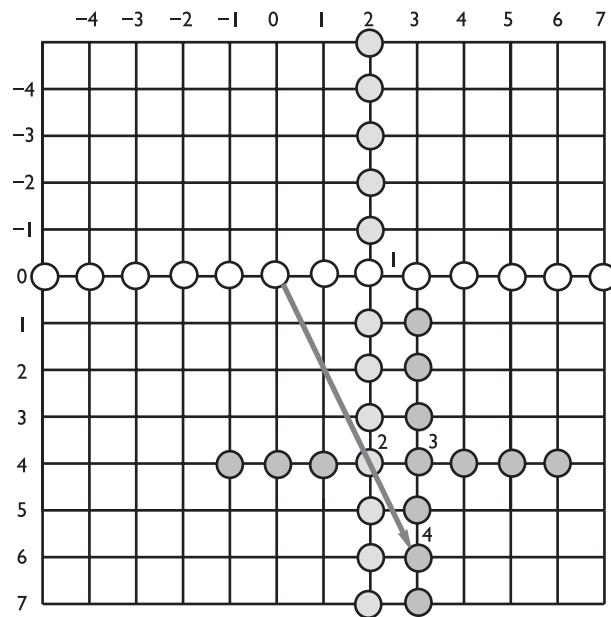


Рис. 12.4. Схема комбинированного поиска векторов движения по двум направлениям

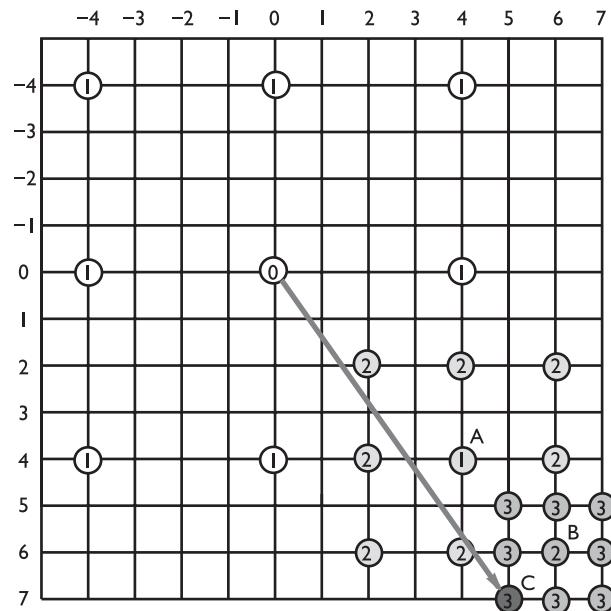


Рис. 12.5. Схема 3-шагового поиска векторов движения

иметь несколько локальных экстремумов, что усложняет применение и снижает практическую эффективность данных методов.

12.1.3. Методы, основанные на оптическом уравнении

Идея данных методов заключается в моделировании движения в плоскости кадра стандартным уравнением в частных производных гиперболического типа, используемым в физике для описания процессов переноса различных сред [4.29]:

$$\partial_t F + \vec{V} \cdot \nabla F = S, \quad (12.2)$$

где $\nabla F = (\frac{\partial F}{\partial x}, \frac{\partial F}{\partial y}) \equiv (\partial_x F, \partial_y F)$ есть пространственный градиент функции, $\partial_t F$ — производная по времени.

В этом уравнении, называемом «оптическим уравнением», под «переносимой средой» понимается яркость изображения F , а член S в правой части моделирует источник, определяющий изменения яркости, не сводимые лишь к пространственному движению. Задача состоит в определении поля векторов движения из (12.2) на основе знания сигнала яркости в двух соседних кадрах. Например, за вектор движения можно принять вектор, минимизирующий правую часть уравнения (12.2) по всей площади макроблока:

$$\sum_{\tilde{x}, \tilde{y}=0,15} S^2 = \sum_{\tilde{x}, \tilde{y}=0,15} (\partial_t F + \vec{V} \cdot \nabla F)^2 \Rightarrow \min, \quad \tilde{x} = x - x_0, \quad \tilde{y} = y - y_0. \quad (12.3)$$

Соответствующая система уравнений, определяющая экстремум, имеет вид:

$$\begin{cases} \sum_{\tilde{x}, \tilde{y}=0, \dots, 15} (\partial_t F \cdot \partial_x F) + V_x \sum_{\tilde{x}, \tilde{y}=0, \dots, 15} (\partial_x F)^2 + V_y \sum_{\tilde{x}, \tilde{y}=0, \dots, 15} (\partial_x F \cdot \partial_y F) = 0; \\ \sum_{\tilde{x}, \tilde{y}=0, \dots, 15} (\partial_t F \cdot \partial_y F) + V_y \sum_{\tilde{x}, \tilde{y}=0, \dots, 15} (\partial_y F)^2 + V_x \sum_{\tilde{x}, \tilde{y}=0, \dots, 15} (\partial_x F \cdot \partial_y F) = 0. \end{cases} \quad (12.4)$$

В (12.3) и (12.4) суммирование производится по всем пикселям макроблока.

Для практического применения системы уравнений (12.4) дифференциальные операторы необходимо заменить на разностные. Простейшими формулами могут быть:

$$\begin{aligned} \partial_t F &= F(x, y, t) - F(x, y, t-1), \\ \partial_x F &= (F(x+1, y, t) - F(x-1, y, t))/2, \\ \partial_y F &= (F(x, y+1, t) - F(x, y-1, t))/2. \end{aligned}$$

Для анализа движения предлагались также другие разностные формулы различной степени сложности на основе оптического уравнения (12.2) [4.30]. Количество операций для определения вектора движения на один пиксель можно оценить как $10Z$ арифметических операций (умножений и сложений), где Z — среднее количество арифметических операций, необходимое для вычисления одного значения производной в системе уравнений (12.4). Для приведенных выше простейших формул производных это составит около 17 операций на пиксель.

Принципиальный недостаток метода заключается в физическом принципе причинности, согласно которому воздействие в каждой точке текущего кадра зависит только от возмущений, пришедших с конечной скоростью распространения из ограниченной пространственной области опорного кадра. Поэтому при большой величине вектора движения формулы для производных должны включать значения F из достаточно большой области опорного кадра. Однако увеличение значений F в формулах для производных приводит к росту вычислительной сложности (увеличивается значение Z), и метод в итоге перестает быть эффективным.

тивным. Заметим, что данное ограничение носит тот же характер, что и ограничение на устойчивость разностных схем для численного решения уравнений типа (12.2). Различные эксперименты демонстрируют, что данный метод имеет преимущество по сравнению с методом полного перебора лишь в случае, когда векторы движения имеют малые значения в пределах нескольких пикселов.

12.1.4. Использование временной и пространственной корреляции для поиска векторов движения

При достаточно большой частоте смены кадров вектора движения мало меняются от кадра к кадру. Согласно 2-му закону Ньютона ускорение тела есть функция только координат и скоростей:

$$\frac{d\vec{W}_i}{dt} = f(\{\vec{X}_k, \vec{W}_k\}), \quad (12.5)$$

откуда

$$\vec{W}_i(t) = \vec{W}_i(t - \Delta t) + \int_{t - \Delta t}^t f(\{\vec{X}_k(\tau), \vec{W}_k(\tau)\}) d\tau. \quad (12.6)$$

В этих уравнениях $\{\vec{X}_k\}$ представляют собой набор координат, а $\{\vec{W}_k\}$ — векторов движения физических объектов, скорости которых обозначены иначе, в отличие от векторов движения макроблоков \vec{V} . Решение дифференциального уравнения (12.5) при известных начальных координатах и скоростях, отвечающих некоторому предыдущему моменту времени $t - \Delta t$, с математической точки зрения корректно и представляет собой решение задачи Коши с начальными условиями. Поэтому вектор скорости в новый момент времени $\vec{W}(t)$ является функцией от $\vec{W}(t - \Delta t)$ и не зависит от более ранних значений $\vec{W}(t - 2\Delta t)$, $\vec{W}(t - 3\Delta t)$ и т. д. Более того, считая величину ускорения ограниченной, имеем $|\vec{W}(t) - \vec{W}(t - \Delta t)| = O(\Delta t)$ при $\Delta t \rightarrow 0$. Предполагая, что поле векторов движения макроблоков является оптической проекцией поля векторов движения физических объектов, получим, что последняя оценка справедлива также для \vec{V} : $|\vec{V}(t) - \vec{V}(t - \Delta t)| = O(\Delta t)$ при $\Delta t \rightarrow 0$.

Сказанное позволяет моделировать изменения векторов движения макроблоков от кадра к кадру случайными величинами, зависящими от своих значений лишь в предыдущем кадре и имеющими малые изменения в новом кадре. Такую последовательность случайных чисел можно считать цепью Маркова 1-го порядка. На основе этой модели был предложен быстрый алгоритм поиска векторов движения, в котором в качестве истинного вектора движения выбирается из нескольких отобранных кандидатов тот, который наиболее близок к одному из отобранных кандидатов в векторы движения макроблока в предыдущий момент времени (рис. 12.6) [4.31]. Рассматриваемые отобранные значения векторов движения вычисляются по минимальному значению H в соотношении (12.1) среди небольшого числа случайных векторов движения, которые имеют заданный закон распределения вокруг вектора движения в предыдущем кадре. В качестве закона распределения обычно используется распределение Гаусса. Следует отметить, что вследствие значительного ограничения множества всех рассмат-

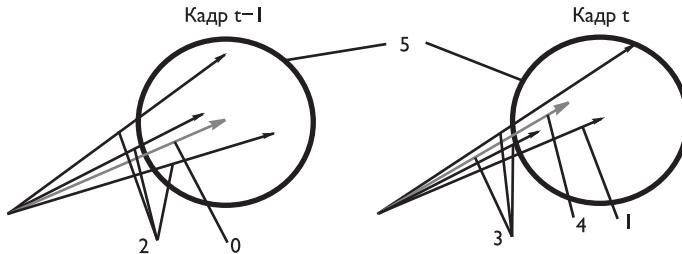


Рис. 12.6. Схема поиска векторов движения по модели Маркова 1-го порядка. Цифрами обозначены: 0 — вектор движения макроблока в кадре с номером $t - 1$, 1 — вектор движения макроблока в кадре t , 2 — различные кандидаты на вектор движения макроблока в кадре $t - 1$, 3 — различные кандидаты на вектор движения макроблока в кадре t , 4 — выбранный вектор движения макроблока в кадре t , 5 — окно поиска векторов движения

риваемых векторов движения данный метод все же приводит к относительно большой величине остаточного сигнала после компенсации движения, превышающей в среднем на 50% величину, найденную при полном переборе векторов движения.

12.1.5. Использование временной и пространственной корреляции векторов с оптимизацией возмущения битового потока

В связи со значительным возрастанием в настоящее время потребностями кодирования и передачи видеоизображений по сетям с низкой скоростью битового потока масштаба 10–30 кбитов/с (например, при передаче видео по коммутируемым телефонным линиям или радиотелефону) были предложены алгоритмы компенсации движения, учитывающие малую величину потока [4.32, 4.33]. Оказывается, что при таких низких значениях битового потока кодирование передаваемых векторов движения может занимать до 50% всей передаваемой информации. Чтобы уменьшить эту величину, было предложено заменить критерий (12.1) на функционал [4.32]:

$$H = \sum_{x-x_0, y-y_0=0, \dots, 15} |F(x, y, t) - F(x - V_x, y - V_y, t - 1)| + \lambda R(\{\vec{V}\}). \quad (12.7)$$

Последний в (12.7) член $R > 0$ представляет собой битовый размер кода, необходимый для кодирования векторов движения, λ — параметр. Идея метода заключается в поиске такой совокупности векторов движения, которая минимизирует одновременно разностный сигнал после компенсации движения и объем кода, необходимый для передачи векторов движения.

При кодировании кодами переменной длины последний значительно возрастает, если вектора движения отличаются друг от друга, и уменьшается, если среди векторов движения много одинаковых. Поскольку строгая минимизация (12.7) требует варьирования всех векторов движения одновременно, что фактически превышает реальные возможности процессора, выполняется частичная минимизация.

Поиск векторов движения осуществляется последовательно — для каждого последующего макроблока определяется минимум (12.7) при уже найденных зна-

чениях векторов предыдущих макроблоков. В работе [4.33] предложена другая, более строгая, теоретическая модель распределения битов между векторами движения и ошибками после компенсации движения, основанная на динамическом программировании и регулярности разбиения кадра на макроблоки.

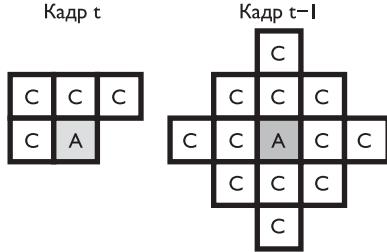


Рис. 12.7. Область поддержки макроблока в текущем и опорном кадрах: А — исходный макроблок, С — соседние макроблоки, образующие область поддержки

каждого вектора движения производится в два этапа. На первом этапе определяется промежуточное (предиктивное) значение вектора движения из анализа уже найденных значений векторов в текущем и опорном кадрах для небольшой соседней с данным макроблоком области, называемой областью поддержки макроблока (рис. 12.7). Вычисление промежуточного вектора осуществляется на основе статистического анализа векторов в области поддержки. На втором этапе найденное значение уточняется методом полного перебора в небольшой окрестности вокруг промежуточного вектора. Такой метод примерно на порядок уменьшает число операций по сравнению с методом полного перебора при достаточной корреляции векторов движения соседних макроблоков [4.32].

12.1.6. Метод полного перебора с преобразованием Фурье

В работе [4.34] был предложен простой алгоритм, позволяющий ускорить поиск векторов движения в 4–5 раз. Особенностью метода является то, что найденные векторы движения точно совпадают с векторами движения, найденными методом полного перебора, т. е. в этом смысле метод можно считать точным. Рассмотрим критерий (12.1), в котором в качестве нормы разности сигналов яркости используется не модуль, а квадрат разности:

$$H = \sum_{\tilde{x}, \tilde{y}=0, \dots, 15} (F(x, y, t) - F(x - V_x, y - V_y, t - 1))^2 = \sum_{\tilde{x}, \tilde{y}=0, \dots, 15} F^2(x, y, t) - 2 \sum_{\tilde{x}, \tilde{y}=0, \dots, 15} F(x, y, t)F(x - V_x, y - V_y, t - 1) + \sum_{\tilde{x}, \tilde{y}=0, \dots, 15} F^2(x - V_x, y - V_y, t - 1). \quad (12.8)$$

Первое слагаемое в правой части соотношения (12.8) не зависит от величины вектора движения и не требует минимизации. Расчет последнего слагаемого может

Для оптимизации по времени в работе [4.32] первым вычисляется последний член в соотношении (12.7), и расчет первого слагаемого не выполняется, если второе слагаемое превышает наилучшее значение H среди уже рассмотренных векторов движения. Такой порядок вычислений позволил примерно на 80% снизить общий объем вычислений по сравнению с методом полного перебора.

Основное достигнутое уменьшение объема операций при минимизации (12.7) основано на корреляции соседних векторов движения. В этом смысле метод близок по сущности к рассмотренному выше алгоритму на основе модели Маркова. С учетом корреляции поиск

быть существенно оптимизирован, если учесть, что для макроблока размером, например, 16×16 пикселов изменение значения вектора движения на один пиксел эквивалентно лишь добавлению к раннее рассчитанной сумме 16 новых и вычитанию 16 старых значений квадратов F , что требует только $2 \cdot 16 / 256 = 1/8$ операций на пиксел.

Наибольшую сложность представляет вычисление 2-го слагаемого в правой части (12.8), которое, тем не менее, может быть оптимизировано с использованием свертки, вычисляемой с применением Фурье-преобразования. В результате при использовании быстрого преобразования Фурье для вычисления свертки при $N \gg 1$ общее количество действий на макроблок оказывается порядка $25(2N + 16)^2(\log_2(2N + 16))/2$ [4.34], что при окне поиска ± 15 пикселов дает выигрыш в 5 раз по сравнению с методом полного перебора.

12.1.7. Методы фазовой корреляции

Отдельную группу быстрых алгоритмов поиска векторов движения представляют методы фазовой корреляции. Данные методы также основаны на преобразовании Фурье. При Фурье-преобразовании пространственный сдвиг изображения соответствует добавлению к фазе каждой двумерной Фурье-компоненты величины, пропорциональной значению вектора движения.

Поэтому при выполнении обратного преобразования от разницы фаз спектральных Фурье-компонент двух соседних кадров на корреляционной поверхности образуется пик, координаты которого совпадают с координатами вектора движения [4.35]. Следует отметить, что вследствие принципа неопределенности Фурье-преобразования, согласно которому вклад в Фурье-спектр вносят элементы изображения одновременно всей пространственной области, метод фазовой корреляции не позволяет однозначно определить, какому из объектов соответствует найденный вектор движения, и требуется дополнительный анализ для определения этого соответствия.

С другой стороны, главным преимуществом метода по сравнению с методом полного перебора является то, что метод может использоваться в достаточно большой области изображения, в которой происходит движение нескольких различных объектов, что позволяет существенно снизить вычислительные затраты. Если вектора движения ищутся одновременно для группы соседних макроблоков, число которых равно $n \cdot n$, то количество действий на пиксел определяется главным образом числом операций на вычисление двух прямых и одного обратного Фурье-преобразований и составляет $3 \cdot 10 (1 + N/8n)^2 \log_2(16n + 2N) \ll 3 \cdot (2N + 1)^2$, если $n \gg 1$ и размер окна поиска каждой из компонент вектора движения равен $\pm N$.

В случае поиска векторов движения в области большого размера на корреляционной поверхности должно возникать несколько пиков, положение которых соответствует различным векторам движения каждого из объектов. При увеличении n все большее число различных движущихся объектов оказывается в пределах рассматриваемой области и сложность определения векторов движения значительно возрастает. Причиной этого является наличие шумов на корреляционной поверхности, обусловленных частью сигнала, не сводящегося лишь к простому движению, возможность сложного движения с масштабированием, а также

уменьшение относительной площади каждого из движущихся объектов. Если объект имеет сложную структуру и его площадь мала, вклад в корреляционный сигнал может оказаться недостаточным для правильного вычисления вектора движения.

Для улучшения анализа движения нескольких объектов было предложено использовать предварительное нелинейное преобразование исходного изображения [36]. В частности, если движущийся объект имеет мелкие детали низкой контрастности (так называемую текстуру на изображении), вероятность правильного определения вектора движения улучшается при использовании следующего комплексного нелинейно-дифференциального преобразования:

$$F \rightarrow g + ih, \quad (12.9)$$

где

$$\begin{cases} g(x, y) = \frac{1}{2}(F + s(\frac{\partial F}{\partial x})) \equiv \frac{1}{2}\{F(x, y) + s[F(x+1, y) - F(x-1, y)]\}, \\ h(x, y) = \frac{1}{2}(F + s(\frac{\partial F}{\partial y})) \equiv \frac{1}{2}\{F(x, y) + s[F(x, y+1) - F(x, y-1)]\}. \end{cases} \quad (12.10)$$

Функция $s(\Lambda)$ определяется по знаку аргумента:

$$\begin{cases} s(\Lambda) = 255, & \Lambda \geq 0, \\ s(\Lambda) = 0, & \Lambda < 0. \end{cases}$$

На рис. 12.8 показаны 2 области размером 128×128 пикселов из соседних кадров (рис. 12.1), на которых происходит движение с разными скоростями — участка дерева с текстурой и заднего фона. После применения преобразования (12.9) изображение на рис. 12.8 имеет характерный вид, похожий на «отпечатки пальцев». Несмотря на то, что новое изображение «отпечатков пальцев» отличается от исходного, оно сохраняет информацию о положении объектов в рассматриваемой области и, как показывают эксперименты, улучшает вероятность определения векторов движения.

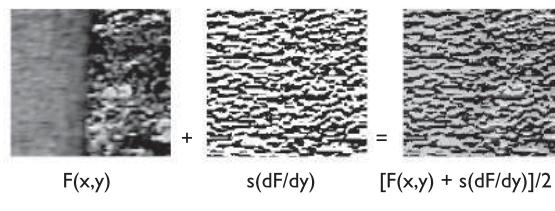
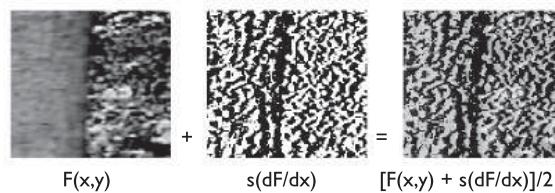
В стандартном методе фазовой корреляции на корреляционной поверхности присутствует только пик от движения заднего фона, а пик от движения дерева не превышает шумов (рис. 12.9а), тогда как при использовании преобразования (12.9) отчетливо наблюдаются оба пика (рис. 12.9б), дающих правильные значения векторов движения двух имеющихся объектов — участков с деревом и задним фоном.

12.1.8. Оптимизация поиска векторов движения по стандарту MPEG-4 (метод быстрого поиска MVFAST)

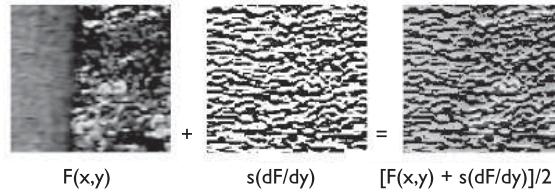
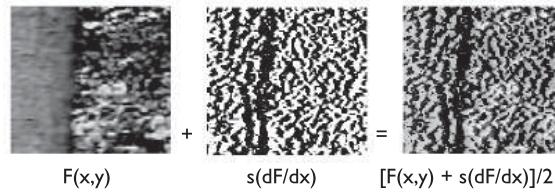
Во второй модели оптимизации стандарта кодирования MPEG-4 описана методика быстрого анализа движения [4.37–4.41]. Эта методика сочетает высокое быстродействие и достаточно высокую эффективность поиска векторов движения объектов, смещение которых близко к плоскопараллельному с небольшой скоростью или с небольшим ускорением. В этой методике поиск вектора движения состоит из нескольких шагов.

1) Проверка нулевого вектора.

Большое количество макроблоков в видеопоследовательностях с небольшим движением имеют вектор движения, равный нулю. Такие макроблоки могут быть



a



б

Рис. 12.8. Пример различных входных сигналов для области 128×128 точек, с помощью которых определялись вектора движения двумя методами фазовой корреляции: (а) — опорный кадр, (б) — текущий кадр

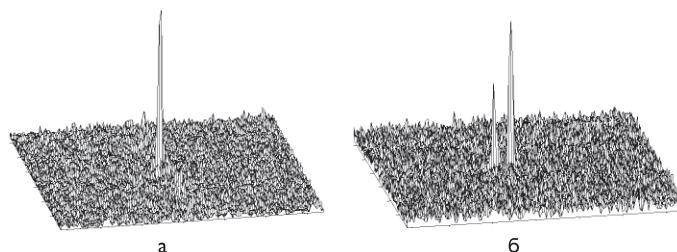


Рис. 12.9. Корреляционная поверхность без использования преобразования (12.9) — а, с преобразованием (12.9) — б

обнаружены на основе суммы абсолютных разностей (SAD) в соответствующих областях изображений. Поиск вектора будет прерван, если SAD, полученный в точке $(0, 0)$ меньше, чем некоторое пороговое значение.

2) Определение активности движения.

При определении активности движения для данного макроблока (Current MB) используются вектора трех соседних макроблоков. Соответствующие макроблоки, вектора которых используются при поиске движения, показаны на рис. 12.10.

Пусть $V = \{V_0, V_1, V_2, V_3\}$, где $V_0 = (0, 0)$, $V_i (i \neq 0)$ — вектор движения i -го соседнего макроблока (MB_i). Длина вектора $V_i = (x_i, y_i)$ определяется как $l_{vi} = |x_i| + |y_i|$. Пусть $L = \max \{l_{vi}\}$ для всех V_i . Активность движения данного макроблока определяется следующим образом:

- низкая активность движения, если $L \leq L_1$;
- средняя активность движения, если $L_1 < L \leq L_2$;
- высокая активность движения, если $L > L_2$,

где L_1 и L_2 — целые постоянные, например $L_1 = 1$ и $L_2 = 2$. Пример распределения векторов приведен на рис. 12.11. В данном случае $l_{v1} = 2$, $l_{v2} = 1$, $l_{v3} = 6$ и, следовательно, $L = \max \{l_{v1}, l_{v2}, l_{v3}\} = 6$.

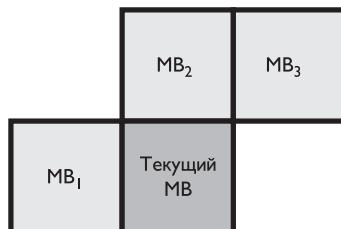


Рис. 12.10. Соседние макроблоки, использующиеся для поиска движения: MB_1 , MB_2 и MB_3

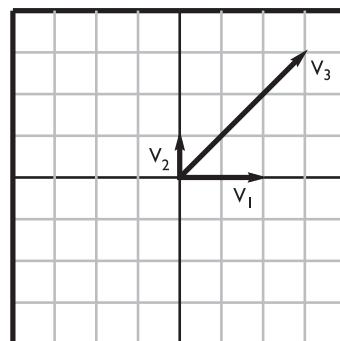


Рис. 12.11. Пример распределения векторов движения соседних макроблоков

3) Центр поиска.

Выбор центра поиска зависит от активности движения для данного макроблока. Если активность движения низкая или средняя, то центр поиска совпадает с точкой $(0,0)$ (остается первоначальным). В противном случае центр поиска перемещается на вектор из множества V (см. выше), для которого сумма абсолютных разностей (SAD) минимальна.

4) Стратегия поиска.

Чтобы получить вектор движения, поиск выполняется вокруг центра поиска. Используются две стратегии поиска, и их выбор зависит от активности движения. Если активность движения низкая или высокая, то выбирается модель поиска по «малому бриллианту» (SDS, small diamond search, рис. 12.12б). В противном случае выбирается модель поиска по «большому бриллианту» (LDS, large diamond search, рис. 12.12а).

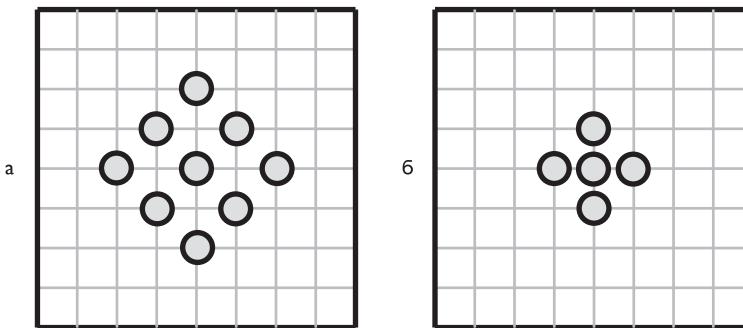


Рис. 12.12. Модели поиска по «большому бриллианту» (LDS, а) и «малому бриллианту» (SDS, б)

4а) Модель поиска SDS.

Шаг 1: Центр решетки малого бриллианта помещается в центр поиска, и подсчитывается SAD для всех его точек. Если у центральной точки SAD минимален, то центр представляет собой вектор движения; в противном случае осуществляется переход к шагу 2.

Шаг 2: Центр решетки малого бриллианта перемещается в точку, которой соответствует минимальный SAD, подсчитывается SAD для всех точек нового положения решетки. Этот шаг повторяется рекурсивно, пока центр решетки не совпадет с точкой минимального SAD.

4б) Модель поиска LDS.

Шаг 1: Центр решетки большого бриллианта помещается в центр поиска, и подсчитывается SAD для всех его точек. Если у центральной точки SAD минимален, то осуществляется переход к шагу 3; в противном случае происходит переход к шагу 2.

Шаг 2: Центр решетки большого бриллианта перемещается в точку, которой соответствует минимальный SAD, подсчитывается SAD для всех точек нового положения решетки. Этот шаг повторяется рекурсивно, пока центр решетки не совпадет с точкой минимального SAD. После этого выполняется шаг 3.

Шаг 3: Осуществляется переключение на модель поиска SDS.

В табл. 12.1 суммируется методика выбора центра поиска и стратегии поиска в зависимости от активности движения.

Таблица 12.1. Выбор центра и стратегии поиска движения в зависимости от активности макроблока

Активность движения	Центр поиска	Стратегия поиска
Низкая	Первоначальный (0,0)	SDS
Средняя	Первоначальный (0,0)	LDS
Высокая	Центр соответствует вектору из множества V с минимальной SAD	SDS

Описанная выше методика может быть усовершенствована приведенным ниже образом [4.42]. К множеству векторов V добавляется еще один вектор, равный

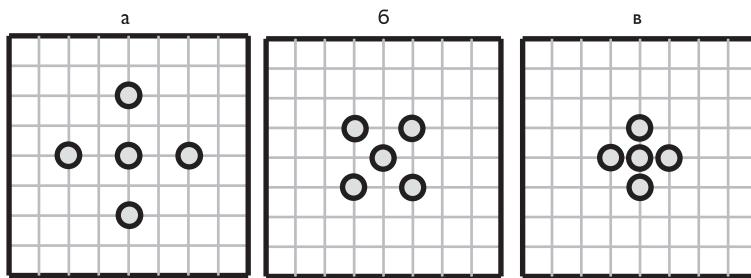


Рис. 12.13. Модели поиска крест (а), диагональ (б) и уточнение (в)

медианному значению трех векторов соседних макроблоков. Кроме того, добавляется еще одна стратегия поиска. Фактически стратегия LDS делится на две части — крест (cross) и диагональ (diagonal), стратегия SDS называется уточнением (refine) (см. рис. 12.13).

Переключение между стратегиями поиска происходит аналогично шагам, описанным выше. От модели «крест» происходит переключение к модели «диагональ», а затем к модели «уточнение».

Начальный выбор стратегии поиска модифицируется следующим образом. Модель «крест» выбирается, если

- медианный вектор — нулевой;
- хотя бы один из трех векторов соседних макроблоков «похож» на медианное значение (под похожестью двух векторов $A = (A_x, A_y)$ и $B = (B_x, B_y)$ понимается выполнение условий $|A_x - B_x| < 4$ и $|A_y - B_y| < 4$);
- похожи все три вектора соседних макроблоков.

В противном случае выбирается модель «уточнение».

Поиск движения также может прерываться, если найдена точка, для которой SAD меньше заданного порогового значения. Это может заметно повысить скорость поиска движения, но снизить его эффективность.

12.2. Повышение эффективности анализа движения по опорным точкам

В [4.43] изложен надежный и достаточно быстрый алгоритм анализа движения деталей в динамических изображениях, обеспечивающий увеличение зоны поиска векторов движения и, вследствие этого, повышение качества воспроизведения быстро перемещающихся деталей. В качестве основного применения алгоритма предполагалось его использование в составе кодеров, использующих алгоритмы компрессии семейства MPEG. Этот алгоритм позволил за счет предварительного анализа характера изображения в макроблоке достичь ускорения по сравнению со стандартным алгоритмом полного перебора в 15–30 раз. Величина ускорения зависит от требований к точности компенсации движения, и уже при отличии от максимального качества компенсации движения — меньше, чем на 10%, составляет величину порядка 15 раз.

Этот способ анализа векторов движения деталей в динамических изображениях включает следующие стадии:

- преобразование последовательности кадров изображений в цифровую форму;
- запоминание дискретных отсчетов яркости текущего и соседнего по времени (опорного) кадров;
- разбиение текущего кадра на макроБЛОКи и поиск вектора движения каждого из макроБЛОКОВ текущего кадра относительно опорного кадра посредством минимизации по рассматриваемому множеству векторов движения контрольной суммы данного макроБЛОКА.

Контрольная сумма — это сумма норм попиксельной разности уровней в текущем и опорных кадрах, причем все множество пикселов рассматриваемого макроБЛОКА разбивается на участки, в каждом из которых выбирается только один пиксель, и контрольная сумма вычисляется лишь по выбранным пикселам. При этом упомянутые пиксели в каждом из участков выбираются таким образом, что их уровни в смежных участках наиболее отличаются друг от друга.

Дополнительно в этом способе для каждого текущего рассматриваемого значения вектора движения вычисление контрольной суммы по выбранным упомянутым пикселям производится в порядке убывания отклонения значения сигнала в каждом пикселе от среднего значения по всей совокупности упомянутых пикселов. Вычисление контрольной суммы прерывается в случае, когда она превышает минимальное значение, найденное среди всех уже рассмотренных векторов движения.

Ограничением этого способа является разбиение макроБЛОКА на несколько строго зафиксированных участков и выбор по заданному алгоритму характерного пикселя без учета структуры сигнала в остальных участках, что является лишь одним из возможных способов выборов опорных пикселов.

В [4.43] рассмотрен также второй способ выбора опорных пикселов макроБЛОКА с помощью использования порядковой статистики для всех пикселов макроБЛОКА, что также является лишь одним из возможных способов выборов опорных пикселов. Кроме того, поиск векторов движения производится по опорным пикселям макроБЛОКА только для исходного разрешения кадра, что ограничивает технический результат по уменьшению числа вычислительных операций.

При дальнейшем развитии этого нового способа анализа движения была поставлена задача сокращения вычислительных операций при более общем возможном выборе структуры опорных точек, характеризующих рельеф («скелет») значений макроБЛОКА, снижение вычислительной сложности метода, что позволяет уменьшить сложность вычисления векторов движения, повышение быстродействия кодирующих устройств и, вследствие этого, возможность производить анализ движения деталей динамических изображений в больших областях, сократив при этом объем сжатой информации и повысив качество воспроизведения быстродвижущихся деталей [4.44].

Этот способ поиска векторов движения деталей в динамических изображениях реализуется следующими действиями:

- преобразование последовательности кадров изображений в цифровую форму;
- запоминание дискретных отсчетов яркости текущего и опорного кадров;

- разбиение текущего кадра на макроблоки;
- поиск вектора движения каждого из макроблоков текущего кадра относительно опорного кадра посредством минимизации по рассматриваемому множеству векторов движения контрольной суммы данного макроблока, являющейся суммой норм попиксельной разности уровней в текущем и опорных кадрах;
- выбор среди множества пикселов каждого макроблока небольшого числа пикселов, характеризующих рельеф значений макроблока;
- вычисление упомянутой контрольной суммы лишь по выбранным опорным пикселям, при этом координаты выбранных пикселов в макроблоке определяют с использованием значений всех пикселов макроблока.

Кроме того, перед выбором опорных пикселов производят передискретизацию исходного и опорного кадров с уменьшением пространственного разрешения по вертикали и горизонтали в заданное число раз посредством применения фильтра к исходному и опорному кадрам. Далее для каждого получившегося макроблока меньшего размера производят выбор опорных пикселов и находят один или несколько наилучших векторов движения по отношению к опорному кадру меньшего разрешения посредством минимизации контрольной суммы с использованием выбранных опорных пикселов. Значение полученных одного или нескольких векторов движения увеличивают в соответствии с исходным разрешением кадра. В окрестностях одного или нескольких полученных векторов производят поиск вектора движения макроблока в кадре исходного разрешения с точностью до целых либо половинных значений пикселов посредством минимизации контрольной суммы с использованием пикселов макроблока исходного разрешения.

Дополнительно для выбора опорных пикселов производят переупорядочивание пикселов каждой из строк макроблока в порядке возрастания их значений и отбор нескольких равноотстоящих друг от друга пикселов в порядке возрастания получившихся номеров пикселов. Для отобранных таким образом в макроблоке пикселов производят переупорядочивание по столбцам в порядке возрастания их значений и выбор нескольких равноотстоящих друг от друга пикселов в порядке возрастания получившихся номеров пикселов. При этом для каждого из выбранных пикселов запоминают их координаты в исходном макроблоке.

Для выбора опорных пикселов в каждой из строк макроблока выбирают несколько первых пикселов в порядке убывания абсолютного отклонения значений пикселов от их среднего значения по строке. Среди выбранных таким образом пикселов в каждом из столбцов выбирают нескольких первых пикселов в порядке убывания абсолютного отклонения значений пикселов от их среднего значения по столбцу, при этом для каждого из выбранных пикселов запоминают их координаты в исходном макроблоке.

Для выбора опорных пикселов каждый макроблок также разбивают на несколько областей, в каждой из которых выбирают в качестве опорного только один пикセル, имеющий максимальное либо минимальное значение внутри данной области. При этом если в данной области выбран пикセル с максимальным значением, то в соседней с ним области выбирают пикセル с минимальным значением, и наоборот, причем для каждого из выбранных пикселов запоминают их координаты в исходном макроблоке.



Рис. 12.14. Текущий — (а) и опорный — (б) кадры видеопоследовательности «Сад цветов»

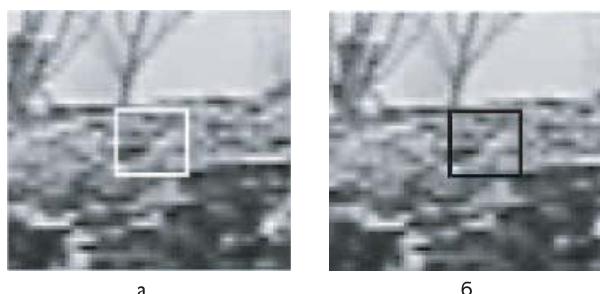


Рис. 12.15. Увеличенное изображение одного макроблока из текущего кадра (а — помечен белой рамкой) и соответствующий ему участок изображения в опорном кадре (б — помечен черной рамкой)

И, наконец, для каждого текущего рассматриваемого значения вектора движения вычисление контрольной суммы с использованием выбранных опорных пикселов производят в порядке убывания отклонения значения сигнала в каждом пикселе от среднего значения по всей совокупности опорных пикселов. Дальнейшее вычисление контрольной суммы прерывают в случае, когда она превышает К-е минимальное значение контрольной суммы, найденной среди всех уже рассмотренных векторов движения.

Алгоритм анализа движения деталей в динамических изображениях можно проиллюстрировать на примере определения вектора смещения одного из макроблоков динамического изображения «Сад цветов» (рис. 12.14а — текущий, б — опорный кадры).

На рис. 12.15а приведена в увеличенном масштабе часть изображения текущего кадра с обведенным макроблоком, вектор движения которого определяется в зоне опорного кадра, изображенной на рис. 12.15б. Макроблок имеет размер 16×16 пикселов, зона поиска движения — 64×64 пикселя. На рис. 12.15б обозначено положение смещенного макроблока в опорном кадре.

В случае использования стандартного метода поиска вектора движения в указанной зоне поиска требуется вычислительная мощность в $3 \cdot 49^2 = 7203$ операции на каждый пиксель макроблока. Рельеф сигнала яркости макроблока изображен на рис. 12.16, а в табл. 12.2 приведены цифровые значения яркости пикселов этого макроблока.

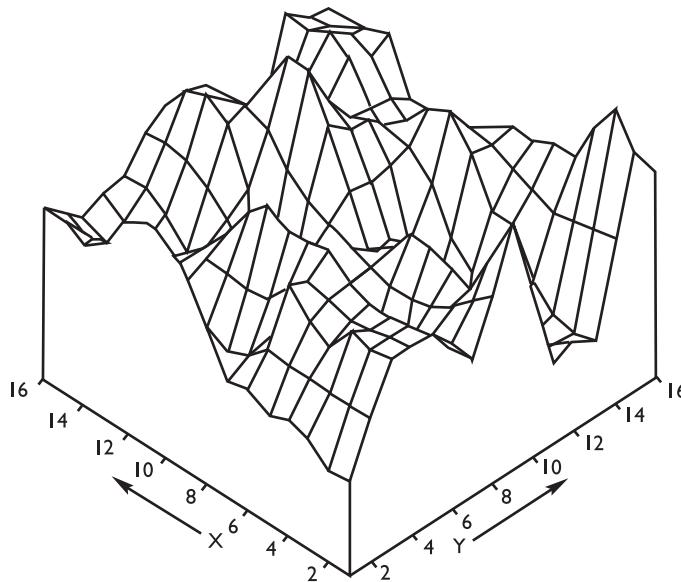


Рис. 12.16. Рельеф сигнала яркости выбранного макроботка

Таблица 12.2. Уровни яркости пикселов в выбранном макроботке

$y \setminus x$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	82	<u>81</u>	97	102	94	105	100	132	171	189	<u>196</u>	181	151	140	151	151
2	126	125	127	125	127	127	112	104	117	137	<u>150</u>	144	125	117	125	135
3	167	165	168	142	153	<u>174</u>	115	62	64	87	118	121	86	84	106	117
4	166	163	158	154	161	165	147	126	123	127	125	114	<u>101</u>	103	115	128
5	166	160	156	156	164	183	178	174	186	168	138	115	100	119	131	131
6	140	140	135	128	125	132	<u>145</u>	153	147	128	117	126	143	157	159	157
7	120	122	116	104	75	80	117	136	123	80	82	139	186	203	195	170
8	164	149	<u>131</u>	120	115	115	120	126	134	143	160	182	<u>191</u>	187	178	170
9	221	<u>179</u>	136	142	147	151	125	111	152	197	235	<u>237</u>	204	172	166	163
10	142	120	100	90	87	85	88	109	148	182	195	188	170	150	132	125
11	70	<u>61</u>	55	39	<u>23</u>	24	41	91	151	180	171	141	139	126	95	78
12	78	63	61	64	69	80	103	136	159	163	153	148	153	151	139	131
13	68	61	68	85	120	124	161	188	179	157	134	143	178	186	177	180
14	146	143	138	134	140	148	<u>158</u>	162	151	133	<u>124</u>	140	166	174	170	166
15	210	236	210	169	172	161	163	148	124	110	97	127	174	169	<u>155</u>	161
16	181	189	176	145	120	113	<u>121</u>	124	112	94	86	97	112	<u>110</u>	99	94

В макроботке выбирается несколько опорных пикселов, характеризующих рельеф (скелет) макроботка. Рассмотрим пример, когда число опорных пикселов равняется 16.

Для выбора опорных пикселов все пиксели макроботка переупорядочивают по строкам в порядке возрастания их значений (табл. 12.3). После переупорядочивания производят выбор пикселов с номерами столбцов $x = \{1, 6, 11, 16\}$ (табл. 12.4).

Таблица 12.3. Уровни яркости пикселов в выбранном макроблоке, переупорядоченные в строках в порядке возрастания их значений

$y \setminus x$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	<u>81</u>	82	94	97	100	102	105	132	140	151	151	151	171	181	189	<u>196</u>
2	104	112	117	117	125	125	125	125	126	127	127	127	135	137	144	<u>150</u>
3	62	64	84	86	87	106	115	117	118	121	142	153	165	167	168	<u>174</u>
4	<u>101</u>	103	114	115	123	125	126	127	128	147	154	158	161	163	165	166
5	100	115	119	131	131	138	156	156	160	164	166	168	174	178	183	186
6	117	125	126	128	128	132	135	140	140	143	<u>145</u>	147	153	157	157	159
7	75	80	80	82	104	116	117	120	122	123	136	139	170	186	195	203
8	115	115	120	120	126	<u>131</u>	134	143	149	160	164	170	178	182	187	<u>191</u>
9	111	125	136	142	147	151	152	163	166	172	<u>179</u>	197	204	221	235	<u>237</u>
10	85	87	88	90	100	109	120	125	132	142	148	150	170	182	188	195
11	<u>23</u>	24	39	41	55	<u>61</u>	70	78	91	95	126	139	141	151	171	180
12	61	63	64	69	78	80	103	131	136	139	148	151	153	153	159	163
13	61	68	68	85	120	124	134	143	157	161	177	178	179	180	186	188
14	<u>124</u>	133	134	138	140	140	143	146	148	151	<u>158</u>	162	166	166	170	174
15	97	110	124	127	148	<u>155</u>	161	161	163	169	169	172	174	210	210	236
16	86	94	94	97	99	<u>110</u>	112	112	113	120	<u>121</u>	124	145	176	181	189

После выбора столбцов производят переупорядочивание выбранных пикселов по столбцам в порядке возрастания значений (табл. 12.5). После переупорядочивания по столбцам производят конечный выбор опорных пикселов с номерами строк $y = \{1, 6, 11, 16\}$ (табл. 12.6). При выборе опорных пикселов их координаты запоминают в исходном макроблоке, выбранные опорные пиксели обозначены подчеркиванием (табл. 12.2–12.5).

Для поиска вектора движения $\vec{V} = (V_x, V_y)$ в способе рассматривают сумму нормы разницы сигналов выбранных пикселов для двух макроблоков в текущем и опорном кадрах SAD1 со сдвигом на вектор движения:

$$\text{SAD 1} = \sum_{\substack{(x,y) - \text{координаты} \\ \text{выбранных точек} \\ \text{в соответствующих} \\ \text{участках макроблока}}} |F(x, y, t) - F(x - V_x, y - V_y, t - \Delta t)|. \quad (12.11)$$

Таблица 12.4. Переупорядоченные уровни яркости пикселов в выбранном макроблоке в выбранных столбцах

$x \setminus y$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	<u>81</u>	104	62	<u>101</u>	100	117	75	115	111	85	<u>23</u>	61	61	<u>124</u>	97	86
2	102	125	106	125	138	132	116	<u>131</u>	151	109	<u>61</u>	80	124	140	<u>155</u>	<u>110</u>
3	151	127	142	154	166	<u>145</u>	136	164	<u>179</u>	148	126	148	177	<u>158</u>	169	<u>121</u>
4	<u>196</u>	<u>150</u>	<u>174</u>	166	186	159	203	<u>191</u>	<u>237</u>	195	180	163	188	174	236	189

Таблица 12.5. Переупорядоченные уровни яркости пикселов в выбранном макроблоке в выбранных столбцах, переупорядоченные в столбцах в порядке возрастания значений

$x \setminus y$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	<u>23</u>	61	61	62	75	<u>81</u>	85	86	97	100	<u>101</u>	104	111	115	117	<u>124</u>
2	<u>61</u>	80	102	106	109	<u>110</u>	116	124	125	125	<u>131</u>	132	138	140	151	<u>155</u>
3	<u>121</u>	126	127	136	142	<u>145</u>	148	148	151	154	<u>158</u>	164	166	169	177	<u>179</u>
4	<u>150</u>	159	163	166	174	<u>174</u>	180	186	188	189	<u>191</u>	195	196	203	236	<u>237</u>

Таблица 12.6. Значения пикселов, выбранных в качестве опорных

$x \setminus y$	1	2	3	4
1	23	81	101	124
2	61	110	131	155
3	121	145	158	179
4	150	174	191	237

В другом варианте выбора опорных пикселов из исходных пикселов макроблока (рис. 12.16) выбирают в каждой строке такие пиксели, которые имеют максимальное абсолютное отклонение своих значений от среднего значения уровня по строке (табл. 12.7). После этого выбирают в каждом сформированном столбце такие пиксели, которые имеют максимальное абсолютное отклонение своих значений от среднего значения уровня по столбцам (табл. 12.8). Данные выбранные пиксели используют в качестве опорных, их положения в исходном макроблоке показаны подчеркиванием (табл. 12.9).

Таблица 12.7. Значения пикселов в выбранных строках

$x \setminus y$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	82	112	168	165	119	157	<u>75</u>	115	<u>221</u>	85	<u>24</u>	163	85	166	<u>210</u>	86
2	81	144	174	166	<u>186</u>	157	186	115	111	182	<u>23</u>	<u>64</u>	<u>68</u>	170	110	176
3	189	104	<u>64</u>	103	115	159	195	187	<u>235</u>	188	171	<u>63</u>	<u>68</u>	124	97	181
4	196	150	<u>62</u>	101	100	117	203	191	<u>237</u>	195	180	<u>61</u>	<u>61</u>	174	236	189

Таблица 12.8. Значения пикселов, выбранных в качестве опорных

$x \setminus y$	1	2	3	4
1	75	210	221	24
2	186	68	64	23
3	68	64	63	235
4	237	62	61	61

Таблица 12.9. Положения выбранных опорных пикселов в исходном макроблоке (опорные пиксели обозначены подчеркиванием)

$y \setminus x$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	82	81	97	102	94	105	100	132	171	189	196	181	151	140	151	151
2	126	125	127	125	127	127	112	104	117	137	150	144	125	117	125	135
3	167	165	168	142	153	174	115	<u>62</u>	<u>64</u>	87	118	121	86	84	106	117
4	166	163	158	154	161	165	147	126	123	127	125	114	101	103	115	128
5	166	160	156	156	164	183	178	174	<u>186</u>	168	138	115	100	119	131	131
6	140	140	135	128	125	<u>132</u>	145	153	<u>147</u>	128	117	126	143	157	159	157
7	120	122	116	104	<u>75</u>	80	117	136	123	80	82	139	186	203	195	170
8	164	149	131	120	115	115	120	126	134	143	160	182	191	187	178	170
9	<u>221</u>	179	136	142	147	151	125	111	152	197	<u>235</u>	<u>237</u>	204	172	166	163
10	142	120	100	90	87	85	88	109	148	182	195	188	170	150	132	125
11	70	61	55	39	<u>23</u>	<u>24</u>	41	91	151	180	171	141	139	126	95	78
12	78	<u>63</u>	<u>61</u>	64	69	80	103	136	159	163	153	148	153	151	139	131
13	<u>68</u>	<u>61</u>	<u>68</u>	85	120	124	161	188	179	157	134	143	178	186	177	180
14	146	143	138	134	140	148	158	162	151	133	124	140	166	174	170	166
15	<u>210</u>	236	210	169	172	161	163	148	124	110	97	127	174	169	155	161
16	181	189	176	145	120	113	121	124	112	94	86	97	112	110	99	94

При поиске векторов движения для выбора опорных пикселов необходимо выполнить порядка $(16 \cdot (16 + 1) \cdot 16/2 + \sqrt{M} \cdot (16 + 1) \cdot 16/2)/256 \approx 10$ обращений к одному пикселу макроблока, где M — число опорных пикселов (в данном случае $M = 16$), для поиска вектора движения с использованием SAD1 необходимо выполнить порядка $\frac{M}{256} 3 \cdot (2N + 1)^2$ операций. При окне поиска $N > 10$ количество операций, необходимых для поиска опорных пикселов, пренебрежимо мало по сравнению с числом операций, необходимых для вычисления SAD1 и поиска вектора движения. Поэтому достигается ускорение поиска векторов приблизительно в $256/M = 16$ раз.

Эффективность этого способа поиска векторов движения иллюстрируется рисунках 12.17а–г. На рис. 12.17а показаны векторы движения, полученные стандартным эталонным алгоритмом с использованием всех пикселов макроблока. На рис. 12.17б, в приведены векторы движения, определенные в соответствии с описанными выше двумя вариантами обработки. Как видно из этих рисунков, векторы движения в большинстве случаев совпадают. Чтобы дать количественную оценку способа, рассмотрим результаты кодирования динамической последовательности кадров (табл. 12.10) в рамках стандарта MPEG-2 с использованием разработанного алгоритма поиска движения. Поскольку в рамках стандарта возможно использование векторов движения с точностью до половинных значений пикселов, рассмотрим два случая кодирования — без уточнения полученных векторов, и с уточнением в пределах $\pm 0,5$ пикселя с использованием интерполяции между пикселами. Для уточнения использована контрольная сумма (12.1) со всеми 256 пикселями макроблока. Уточнение векторов до половины пикселя требует порядка $3 \cdot 9 = 27$ дополнительных операций на каждый пиксель. Как следует из анализа результата кодирования, предложенные способы дают существенное ускорение поиска векторов движения (в 16 раз при 16 опорных точках), при этом коэффициент сжатия ухудшается не более 1–3% в случае использования точности векторов до 0,5 пикселя и 4–10% при точности векторов в один пиксель.

На рис. 12.17г приведены результаты использования еще одного, третьего варианта поиска векторов движения, предусматривающего пространственную передискретизацию исходного изображения и предварительный поиск векторов движения по передискретизированному изображению.

Изложение способа анализа с передискретизацией изображения можно проиллюстрировать на примере того же макроблока динамического изображения (рис. 12.14). В отличие от описанных выше принципов поиска, в данном варианте перед выбором опорных точек для каждого макроблока проводят передискретизацию исходного и опорного кадров. Рассмотрим случай, когда передискретизацию производят с уменьшением пространственного разрешения в 2 раза с использованием усреднения соседних пикселов:

$$F^{(d)}(x, y) = \frac{1}{4}(F(2x, 2y) + F(2x + 1, 2y) + F(2x, 2y + 1) + F(2x + 1, 2y + 1)).$$

Здесь $F^{(d)}(x, y)$ — значение пикселя после передискретизации. Отсчеты пикселов макроблока, показанного на рис. 12.15 и 12.16, после передискретизации будут иметь значения, представленные в табл. 12.11.

Согласно одному из вариантов способа вычисления векторов движения, передискретизированный макроблок размером 8×8 пикселов делят на несколько

областей, в каждой из которых выбирают одно максимальное либо минимальное значение в чередующемся порядке в качестве опорного пикселя. Рассмотрим случай, когда число опорных пикселов равно 16. В табл. 12.12 и 12.13 представлены 16 областей макроблока и значения отсчетов в них, опорный пиксель в каждой области отмечен подчеркиванием.

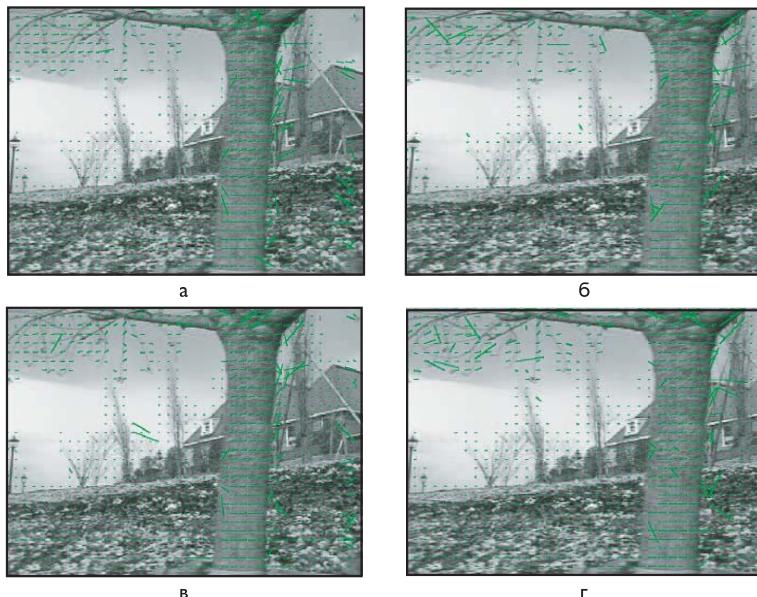


Рис. 12.17. Векторы движения, найденные стандартным способом — а и тремя вариантами эффективной реализации способа — б, в и г соответственно

Таблица 12.10. Размер кода MPEG-2 (в байтах) для последовательности «Сад цветов», 97 кадров с разрешением 640×480 пикселов

Конечная точность векторов движения	Первый вариант	Второй вариант	Третий вариант	Способ полного перебора с использованием 256 пикселов макроблока
0,5 пикселя	3 317 551	3 375 686	3 295 558	3242122
1 пиксель	3 747 245	3 767 395	3 699 264	3545362

Таблица 12.11. Значения пикселов в выбранном макроблоке после передискретизации

y/x	1	2	3	4	5	6	7	8
1	103	112	113	112	153	168	133	148
2	165	155	163	112	100	119	93	116
3	151	143	151	162	157	124	129	144
4	138	117	96	124	120	140	191	178
5	165	117	117	108	169	213	174	146
6	68	54	49	92	163	153	142	110
7	104	106	133	167	155	135	176	173
8	204	175	141	139	110	101	141	127

Таблица 12.12. Разбиение макроблока на участки и порядок выбора опорных пикселов

$y \setminus x$	1	2	3	4	5	6	7	8
1	max	min	max	min	max	min	max	min
2								
3	min	max	min	max	min	max	min	max
4								
5	max	min	max	min	max	min	max	min
6								
7	min	max	min	max	min	max	min	max
8								

Таблица 12.13. Положение выбранных пикселов
(опорные пиксели отмечены подчеркиванием)

$y \setminus x$	1	2	3	4	5	6	7	8
1	103	112	113	<u>112</u>	153	<u>168</u>	133	148
2	<u>165</u>	155	163	112	100	119	<u>93</u>	116
3	151	143	151	<u>162</u>	157	124	<u>129</u>	144
4	138	<u>117</u>	96	124	<u>120</u>	140	<u>191</u>	178
5	<u>165</u>	117	117	108	169	<u>213</u>	174	146
6	68	54	<u>49</u>	92	163	153	142	<u>110</u>
7	<u>104</u>	106	133	<u>167</u>	155	135	<u>176</u>	173
8	204	175	141	139	110	<u>101</u>	141	127

В процессе минимизации нормы SAD1 в окне поиска $\pm N/2$ пикселов по передискретизованным изображениям определяют K векторов $V_{1d}, V_{2d}, \dots, V_{Kd}$, дающих наименьшие значения SAD1:

$$\text{Min} = SAD1(V_{1d}) \leq SAD1(V_{2d}) \leq SAD1(V_{3d}) \leq \dots$$

После нахождения наилучших векторов $V_{1d}, V_{2d}, V_{3d}, \dots$, соответствующих передискретизированым опорному и текущему кадрам, значения векторов увеличиваются в соответствии с отношением разрешений исходного и передискретизированного кадров, в данном случае — в 2 раза. Далее в небольшой окрестности (например, ± 1 пиксел) от каждого из полученных значений векторов $(2 \cdot V_{1d}, 2 \cdot V_{2d}, 2 \cdot V_{3d}, \dots)$ производят минимизацию контрольной суммы и определяют наилучший вектор движения. Данный вектор принимается за конечный вектор движения макроблока в случае использования точности векторов, равной 1 пикселу.

При использовании точности векторов движения, равной половине пикселя, как это, например, реализуется в стандарте MPEG-2, последний найденный выше вектор уточняется в окрестности $\pm 0,5$ пикселя, либо сразу в малых окрестностях (например, ± 1 пиксел) векторов $2 \cdot V_{1d}, 2 \cdot V_{2d}, 2 \cdot V_{3d}, \dots$ с точностью до 0,5 пикселя.

Для поиска векторов движения на этапе их оценки в передискретизированном кадре число операций на один пиксель передискретизированного кадра составляет $3 \frac{16}{64} (2 \cdot \frac{N}{Z} + 1)^2$, где Z — отношение исходного и передискретизированного разрешений (в данном случае $Z = 2$), или $\frac{1}{4} 3 \frac{16}{64} (2 \cdot \frac{N}{Z} + 1)^2$ на пиксель в кадре исходного разрешения.

Уточнение векторов движения в окрестности каждого из найденных векторов занимает небольшое количество операций — порядка $3K(2 \cdot 1 + 1)^2$ на пиксель в кадре исходного разрешения, где K — количество наилучших векторов, и не зависит

от N . При $K \leq 3$ и $N > 10$ последним числом можно пренебречь. В этом случае ускорение по сравнению с эталонным способом полного перебора составляет $\frac{1}{4}3\frac{16}{64}(2 \cdot \frac{N}{Z} + 1)^2 / 3\frac{16}{64}(2 \cdot \frac{N}{Z} + 1)^2 \approx 16Z^2 = 64$ при $Z = 2$.

Дополнительно скорость анализа векторов движения макроблоков увеличивается за счет определенного порядка вычисления контрольной суммы SAD1.

При этом вычисляют среднее значение выбранных пикселов макроблока:

$$F_{\text{cp}} = \frac{1}{M} \sum_{\substack{(x,y) \text{ — координаты} \\ \text{выбранных точек} \\ \text{в соответствующих} \\ \text{участках макроблока}}} F(x, y, t).$$

Затем вычисляют модули разностей $X(x, y) = |F(x, y, t) - F_{\text{cp}}|$, где (x, y) — координаты выбранных точек в соответствующих участках макроблока, и контрольную сумму вычисляют последовательно по координатам (x, y) , для которых величины $X(x, y)$ располагаются в убывающей последовательности.

В случае если текущий выбранный вектор $\vec{V} = (V_x, V_y)$ не соответствует истинному вектору движения, для которого контрольная сумма минимальна, рассчитываемая контрольная сумма статистически достаточно быстро возрастает, и в этом случае обычно не требуется производить вычисления по всем выбранным точкам макроблока. При этом в среднем по кадру скорость анализа движения макроблоков дополнительно возрастает в два раза. Вычисление контрольной суммы прекращается, если ее текущее значение превышает K -е минимальное значение контрольной суммы среди уже рассмотренных значений векторов движения. При этом в среднем по кадру скорость анализа движения макроблоков дополнительно возрастает в полтора-два раза.

Эффективность способа анализа движения по этому методу иллюстрируется в табл. 12.14. На рис. 12.17г приведены вектора движения, полученные с использованием способа при $K = 3$, а в табл. 12.14 даны результаты кодирования тестовой динамической последовательности кадров в рамках стандарта MPEG-2 ($K=1, 2$ и 3). Из таблицы следует, что этот способ при $Z = 2$ уступает по коэффициенту сжатия наилучшему эталонному способу перебора по всем пикселям не более 1% уже при $K=2$ и 3 , и превосходит другие способы, не использующие передискретизацию. При этом коэффициент ускорения анализа движения по сравнению с эталонным способом составляет 64 раза при $Z = 2$.

Таблица 12.14. Размер кода MPEG-2 (в байтах) для последовательности «Сад цветов», 97 кадров с разрешением 640×480 пикселов (с фильтрацией)

Конечная точность векторов движения	$K = 1$	$K = 2$	$K = 3$
0,5 пикселя	3 308 449	3 276 854	3269046
1 пиксель	3 648 403	3 601 394	3590763

Заметим, что использование фильтрации улучшает результат. Например, если бы передискретизация проводилась без фильтрации, а только за счет прореживания по формуле $F^{(d)}(x, y) = F(2x, 2y)$, то результат по коэффициенту сжатия был бы хуже на 1–2% (табл. 12.15).

Таблица 12.15. Размер кода MPEG-2 (в байтах) для последовательности «Сад цветов», 97 кадров с разрешением 640×480 пикселов (с прореживанием)

Конечная точность векторов движения	$K = 1$	$K = 2$	$K = 3$
0,5 пикселя	3 350 937	3 297 776	3 283 745
1 пиксель	3 688 898	3 624 819	3 606 289

12.3. Дополнительные возможности компенсации движения деталей в динамических изображениях

Принципиальной особенностью рассмотренных выше методов является то, что улучшение межкадрового сжатия достигается за счет усложнения модели движения. После компенсации движения в рамках стандарта MPEG-2 внутрикадровое сжатие приводит фактически к изменению истинных значений пикселов изображения (обусловленных квантованием коэффициентов спектра) без изменения положения этих пикселов в кадре. С другой стороны, можно незначительно изменить содержание кодируемого кадра таким образом, чтобы данное изменение было незаметно для глаза и улучшало компенсацию движения в рамках той или иной используемой модели. При этом дополнительное сжатие может быть достигнуто за счет небольших (в пределах долей пикселя) пространственных смещений деталей исходного изображения [4.45].

Поясним сказанное примером. На рис. 12.18 изображен одномерный импульсный сигнал, движущийся с постоянной скоростью вдоль оси ординат. Предположим, что данный сигнал должен быть закодирован и передан в рамках стандарта MPEG. При дискретности передаваемого вектора движения после компенсации движения значение разностного сигнала отлично от нуля (кривая 4), и требуется определенное количество битов для передачи разностного сигнала. С другой стороны, сдвинутый сигнал в пределах долей пикселя (кривая 3) визуально мало отличается от исходного и не требует передачи дополнительной информации после компенсации движения. Поэтому выгоднее передать сдвинутый в пределах долей пикселя сигнал (кривая 3) вместо исходного сигнала (кривая 1).

Можно выделить две принципиально различные возможности пространственной передискретизации элементов изображения в предлагаемом способе:

- любым способом в пределах чувствительности человеческого зрения к небольшим пространственным перемещениям пикселов;
- определенным способом, который отвечает некоторой физической модели движения.

Можно предложить алгоритм, в котором каждый пиксель изображения перемещается независимо от других пикселов так, чтобы значение разностного сигнала после компенсации движения в рамках данной модели движения было минимально для данного пикселя. Зрительное восприятие такого сигнала будет характеризоваться небольшим пространственным дрожанием пикселов, на которое должно быть наложено ограничение, чтобы дрожание было незаметно для человеческого зрения.

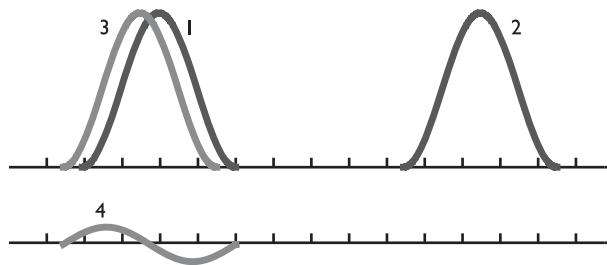


Рис. 12.18. Компенсация движения одиночного импульса: 1 — значение сигнала в текущий момент времени, 2 — значение сигнала в предыдущий момент времени, 3 — сдвинутое положение сигнала, обеспечивающее нулевой сигнал после компенсации движения, 4 — значение сигнала после компенсации движения с точностью до целых значений пикселов

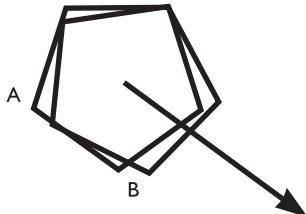


Рис. 12.19. А — реальное положение объекта, В — отображаемое положение объекта

В другом варианте, которого, вероятно, следует придерживаться, предполагается, что пространственные перемещения объектов соответствуют их физическому движению. Представим, что вместо отображения элементов изображения в строго определенный момент времени (совокупность таких элементов образует кадр изображения в обычном понимании) элементы изображения отображаются на экране декодера в моменты времени, отличающиеся друг от друга на небольшую временную величину, соответствующую разнице между истинным и передаваемым дискретным значением вектора движения (рис. 12.19).

В таком понимании приходим к новой модели кадра изображения, элементы которого (в стандарте MPEG такими элементами могут быть макроблоки) имеют временную неопределенность в воспроизведении в пределах долей временного интервала между кадрами. Появление временной неопределенности, с одной стороны, оправдано свойством инерции человеческой глаза, а с другой стороны, расширяет возможности по улучшению компенсации движения в рамках заданного стандарта и, следовательно, увеличению коэффициента сжатия.

Проанализируем метод пространственной передискретизации элементов изображения в рамках стандарта MPEG-2, адекватный сформулированной модели. Ограничимся случаем, когда передискретизации подвергаются только В-кадры. Изменение пространственного положения объектов в этих кадрах, в отличие от кадров типа I и P, не будет влиять на компенсацию движения и кодирование остальной последовательности кадров. Поэтому такой способ должен всегда характеризоваться положительным эффектом по увеличению коэффициента сжатия.

Важно отметить, что в рамках этой модели стандартный критерий искажений, определяемый по среднеквадратичному значению шума, становится больше неприменим. Действительно, после пространственной передискретизации значение среднеквадратичной ошибки может быть велико, несмотря на то, что сигнал отличается от исходного лишь незначительным сдвигом по координатам (см. рис. 12.19).

Разработка адекватного критерия визуальной «заметности» вносимых искажений должна включать критерий пространственного отклонения деталей изображения от исходных значений в кадре. Можно предположить, что, если, как отмечалось выше, данные изменения связаны с физическими перемещениями объектов, заметность искажений должна значительно уменьшаться вследствие инерции зрения. Разработка адекватного критерия качества является темой отдельного исследования и должна учитывать свойства инерции зрительного восприятия.

Основные уравнения

Рассмотрим видеопоследовательность кадров, содержащую кадры В-типа стандарта MPEG. В случае кодирования макроблока В-кадра на кодере вычисляется величина разности

$$C_{x,y} = Y_{x,y}^{(n)} - X_{x,y}, \quad (12.12)$$

где $C_{x,y} = Y_{x,y}^{(n)} - X_{x,y}$ при предсказании движения в одну сторону и

$$X_{x,y} = \frac{1}{2} \left(Y_{x+V_{1x},y+V_{1y}}^{(n-\Delta 1)} + Y_{x+V_{2x},y+V_{2y}}^{(n-\Delta 2)} \right)$$

при предсказании движения в обе стороны; $Y_{x,y}^{(n)}$ — значение сигнала яркости пикселя в макроблоке в кадре с номером n , имеющего пространственные координаты (x, y) ; $X_{x,y}^{(n)}$ — предсказанное значение пикселя с учетом компенсации движения; Δ — расстояние до опорного кадра; (V_x, V_y) — координаты вектора движения.

При кодировании с векторами движения с точностью до полпикселя значения (V_x, V_y) вычисляются с помощью линейной интерполяции.

Будем считать, что вектор движения макроблока известен и определен, например, методом полного перебора и, соответственно, значения $X_{x,y}^{(n)}$ вычислены.

Сущность предлагаемого метода сводится к замене истинного значения $Y_{x,y}^{(n)}$ некоторым другим значением $Y_{x,y}^{(n)}$, соответствующим небольшому смещению изображения во времени. Целью замены является уменьшение модуля разности $C_{x,y}$. Будем считать, что значения пикселов $Y_{x,y}^{(n)}$ можно получить исходя из значений $Y_{x,y}^{(n)}$ с использованием интерполяции между пикселями:

$$Y_{x,y}^{new(n)} = Y_{x+\Delta V_x, y+\Delta V_y}^{(n)}, \quad (12.13)$$

где $\Delta V_x, \Delta V_y$ — координаты вектора небольшого смещения макроблока. Величина смещения $\vec{V} = (\Delta V_x, \Delta V_y)$ должна определяться из условия минимизации разностного сигнала:

$$\sum_{x,y} (Y_{x,y}^{new(n)} - X_{x,y})^2 \Rightarrow \min, \quad (12.14)$$

Суммирование здесь производится по всем пикселям макроблока.

Пусть ориентация вектора \vec{V} соответствует случаю, изображеному на рисунке 12.20, на котором обозначено $Y_0 = Y_{i,k}$, $Y_1 = Y_{i+1,k}$, $Y_2 = Y_{i,k+1}$, $Y_3 = Y_{i+1,k+1}$. Для вычисления промежуточного значения сигнала яркости в точке \vec{V} будем использовать интерполяцию по четырем соседним точкам. Предпола-

гая функцию достаточно гладкой, в первом приближении можно считать

$$Y(\Delta \vec{V}) = Y_0 + \Delta \vec{V} \cdot \nabla Y + O((\Delta \vec{V} \cdot \nabla Y)^2),$$

где ∇Y — градиент функции.

В пренебрежении членами второго порядка малости имеем:

$$Y(\Delta \vec{V}) \approx Y^{(1)} = Y_0 + x_1(\vec{e}_1 \cdot \nabla Y) + x_2(\vec{e}_2 \cdot \nabla Y) = Y_0 + x_1(Y_1 - Y_0) + x_2(Y_2 - Y_0), \quad (12.15)$$

$$Y(\Delta \vec{V}) \approx Y^{(2)} = Y_0 + y_1(\vec{e}_1 \cdot \nabla Y) + y_2(\vec{e}_3 \cdot \nabla Y) = Y_0 + y_1(Y_1 - Y_0) + y_2(Y_3 - Y_0), \quad (12.16)$$

$$Y(\Delta \vec{V}) \approx Y^{(3)} = Y_0 + z_1(\vec{e}_2 \cdot \nabla Y) + z_2(\vec{e}_3 \cdot \nabla Y) = Y_0 + z_1(Y_2 - Y_0) + z_2(Y_3 - Y_0), \quad (12.17)$$

где координаты x, y, z определены как координаты вектора смещения в различных базисах:

$$\begin{aligned} \Delta \vec{V} &= x_1 \vec{e}_1 + x_2 \vec{e}_2 = y_1 \vec{e}_1 + y_2 \vec{e}_3 = z_1 \vec{e}_2 + z_2 \vec{e}_3, \\ y_2 &= x_2, \quad y_1 = x_1 - x_2, \quad z_1 = x_2 - x_1, \quad z_2 = x_1 \end{aligned}$$

(см. рис. 12.20).

В соотношениях (12.15)–(12.17) использовалась приближенная формула для производных: $\vec{e}_k \cdot \nabla Y \approx Y_k - Y_0$.

Для того чтобы включить в интерполяционную формулу значения функции во всех четырех точках, рассмотрим усредненное между (12.15)–(12.17) значение с различным весом для каждого выражения:

$$\begin{aligned} Y(\Delta \vec{V}) &\approx qY^{(1)} + \frac{1-q}{2}Y^{(2)} + \frac{1-q}{2}Y^{(3)} = \\ &= Y_0 + \left(qx_1 + \frac{1-q}{2}y_1 \right) (Y_1 - Y_0) + \\ &+ \left(qx_2 + \frac{1-q}{2}z_1 \right) (Y_2 - Y_0) + \frac{1-q}{2}(y_2 + z_2)(Y_3 - Y_0) \end{aligned} \quad (12.18)$$

Рис. 12.20. Положение действительной поправки к целочисленному значению вектора движения макроблока

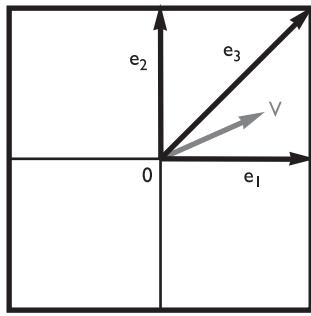
q и $(1-q)/2$ — веса, с которыми выражения (3.15)–(3.17) входят в конечную формулу. В последней формуле выбраны вследствие симметрии (см. рис. 12.20) одинаковые веса для выражений (12.15)–(12.17). Выражая коэффициенты через Декартовы координаты вектора (x_1, x_2) , имеем

$$Y(\Delta \vec{V}) \approx Y_0 + \alpha_1(Y_1 - Y_0) + \alpha_2(Y_2 - Y_0) + \alpha_3(Y_3 - Y_0), \quad (12.19)$$

$$\alpha_1 = \frac{1+q}{2}x_1 - \frac{1-q}{2}x_2, \quad (12.20)$$

$$\alpha_2 = \frac{1+q}{2}x_2 - \frac{1-q}{2}x_1, \quad (12.21)$$

$$\alpha_3 = \frac{1-q}{2}(x_1 + x_2). \quad (12.22)$$



В формулах (12.15)–(12.17) независимыми величинами являются только координаты вектора (x_1, x_2) , по которым необходимо произвести минимизацию в (12.14). С другой стороны, в (12.18) присутствует новый параметр q , поэтому минимизацию можно проводить по трем независимым параметрам (x_1, x_2, q) . Нетрудно видеть, что в общем случае якобиан преобразования (12.19)

$$\partial(\alpha_1, \alpha_2, \alpha_3)/(x_1, x_2, q) \neq 0,$$

поэтому величины $(\alpha_1, \alpha_2, \alpha_3)$ являются независимыми и, следовательно, минимизацию можно проводить по величинам $(\alpha_1, \alpha_2, \alpha_3)$.

После нахождения оптимальных значений $(\alpha_1, \alpha_2, \alpha_3)$ необходимо вычислить, какому вектору движения они соответствуют. Формулы обратного преобразования имеют вид:

$$x_1 = \alpha_1 + \alpha_3, \quad (12.23)$$

$$x_2 = \alpha_2 + \alpha_3, \quad (12.24)$$

$$q = \frac{\alpha_1 + \alpha_2}{\alpha_1 + \alpha_2 + 2\alpha_3}. \quad (12.25)$$

Координаты вектора движения (x_1, x_2) определяются простыми выражениями (12.23), (12.24).

Необходимо наложить условие, чтобы вектор движения находился внутри рассматриваемой области. Величины ΔV_x , ΔV_y не должны превышать доли пикселя, $0 \leq x_1, x_2 \leq p$, откуда

$$0 \leq \alpha_1 + \alpha_3 \leq p, \quad (12.26)$$

$$0 \leq \alpha_2 + \alpha_3 \leq p, \quad (12.27)$$

где p — диапазон допустимого размера смещения элементов изображения при передискретизации, $0 \leq p \leq 1/2$. Величины $(\alpha_1, \alpha_2, \alpha_3)$ находятся из условия:

$$H = \sum (X_{0i} - Y_{0i} - \alpha_1(Y_{1i} - Y_{0i}) - \alpha_2(Y_{2i} - Y_{0i}) - \alpha_3(Y_{3i} - Y_{0i}))^2 \Rightarrow \min. \quad (12.28)$$

В (12.28) $X_0 = X_{i,k}$ — значение сигнала яркости в текущем кадре суммирование производится по всем точкам макроБЛОКА (индексу i).

Необходимыми условиями минимума являются:

$$\partial H / \partial \alpha_1 = \partial H / \partial \alpha_2 = \partial H / \partial \alpha_3 = 0,$$

откуда получаем линейную систему из трех уравнений для определения $(\alpha_1, \alpha_2, \alpha_3)$:

$$a_{11}\alpha_1 + a_{12}\alpha_2 + a_{13}\alpha_3 = -s_1, \quad (12.29)$$

$$a_{21}\alpha_1 + a_{22}\alpha_2 + a_{23}\alpha_3 = -s_2, \quad (12.30)$$

$$a_{31}\alpha_1 + a_{32}\alpha_2 + a_{33}\alpha_3 = -s_3, \quad (12.31)$$

где $a_{lm} = \sum (X_{0,i} - Y_{l,i})(Y_{0,i} - Y_{l,i})$, $s_l = \sum (X_{0,i} - Y_{0,i})(Y_{0,i} - Y_{l,i})$, $m = 1, 2, 3$.

Система (12.29)–(12.31) решается стандартными методами линейной алгебры.

В вырожденном случае (детерминант матрицы равен нулю) передискретизацию проводить не будем (положим $\alpha_1 = \alpha_2 = \alpha_3 = 0$). Поскольку вектор смещения может иметь 4 различные ориентации, необходимо выбрать наилучшее значение среди значений, соответствующих четырем квадратам на рис. 12.20.

После определения вектора смещения для данного макроБЛОКА производится замена значений попиксельной яркости согласно (12.19).

При кодировании сигнала цветности необходимо учитывать ту модель цветности, которая используется в данном приложении. Например, при использовании распространенной модели 4:2:0 пространственное разрешение цветоразностных сигналов Cr и Cb в 4 раза меньше пространственного разрешения сигнала яркости. В этом случае передискретизация сигналов цветности производится с половинными значениями векторов движения, найденными выше, например (см. рис. 12.20):

$$Y_{Cr}(\Delta \vec{V}) \approx Y_{Cr}^{(1)} = Y_{Cr0} + \frac{x_1}{2}(Y_{Cr1} - Y_{Cr0}) + \frac{x_2}{2}(Y_{Cr2} - Y_{Cr0}), \quad (12.32)$$

$$Y_{Cb}(\Delta \vec{V}) \approx Y_{Cb}^{(1)} = Y_{Cb0} + \frac{x_1}{2}(Y_{Cb1} - Y_{Cb0}) + \frac{x_2}{2}(Y_{Cb2} - Y_{Cb0}). \quad (12.33)$$

Результаты экспериментов

Тестирование рассмотренного алгоритма осуществлялось на модели программного кодера MPEG-2. Для сравнения результатов выбирались постоянные значения коэффициентов квантования для всех макроблоков в кадре. Выбор типа макроблока в В-кадре осуществлялся после процедуры пространственной передискретизации. Для этого пространственная передискретизация проводилась независимо для трех значений $X_{x,y}$, соответствующих типу кодирования макроблоков с предсказанием по времени вперед, назад либо в обе стороны. Решение от выборе типа кодирования осуществлялось по минимальному значению мощности сигнала после компенсации движения:

$$f = \sum_{i,k} C_{i,k}^2. \quad (12.34)$$

Макроблок кодировался без компенсации движения, если $f > g$,

$$g = \sum_{i,k} (Y_{i,k} - \langle Y_{i,k} \rangle)^2, \quad \langle Y_{i,k} \rangle = \frac{1}{256} \sum_{i,k} Y_{i,k}. \quad (12.35)$$

При кодировании каждый 15-й кадр являлся опорным, на каждый Р-кадр приходилось два В-кадра. Результаты вычислений приведены в табл. 12.16 и 12.17 для различных тестовых последовательностей. Выбирались такие последовательности или их участки, в которых отсутствует смена сюжета.

Таблица 12.16. Последовательность «Футбол», разрешение 352×240 пикселов, формат цветности $4 : 2 : 0$, 105 кадров, 25 кадров/с

Коэффициент квантования	Размер файла (байты)			Выигрыш по сжатию [%]		
	$p = 0$	$p = 0,25$	$p = 0,5$	$p = 0$	$p = 0,25$	$p = 0,5$
4	4 717 155	4 539 104	4 433 831	0	3,9	6,4
8	2 665 627	2 575 045	2 508 722	0	3,5	6,2
16	1 365 542	1 337 365	1 313 466	0	2,1	3,9

Таблица 12.17. Последовательность «Теннис», разрешение 352×240 пикселов, формат цветности $4 : 2 : 0$, 60 кадров, 25 кадров/с

Коэффициент квантования	Размер файла (байты)			Выигрыш по сжатию [%]		
	$p = 0$	$p = 0,25$	$p = 0,5$	$p = 0$	$p = 0,25$	$p = 0,5$
4	1 555 862	1 499 434	1 456 107	0	3,8	6,8
8	830 824	798 822	773 192	0	4,0	7,5
16	413 488	403 713	396 726	0	2,4	4,2



Рис. 12.21. Восстановленный кадр последовательности «Футбол» при различном значении максимального вектора смещения при передискретизации

Для рассмотренных видеопоследовательностей эффект дополнительного сжатия составляет порядка 4–7% и зависит от максимально допустимого значения p вектора смещения, увеличиваясь с увеличением p . Визуальный анализ закодированных последовательностей показывает, что при $p = 0,25$ искажения сигнала незаметны для глаза.

При $p = 0,5$ становятся заметными небольшие искажения, которые проявляются как слабое дрожание элементов изображения с частотой, равной частоте смены кадров.

На рис. 12.21 показаны восстановленные кадры последовательности «Футбол» при различных значениях p . Из этого рисунка видно, что статические изображения без применения и с применением нового алгоритма практически не отличаются друг от друга.

Таким образом, метод передискретизации приводит к дополнительному эффекту сжатия порядка 4–7% и возрастает с увеличением максимального значения допустимого вектора смещения. С другой стороны, при больших значениях вектора смещения ($p \geq 0,5$) становятся заметными мелкомасштабное дрожание элементов изображения. Поэтому практически приходится ограничиваться значением $p = 0,25$, при котором эффект дополнительного сжатия составляет около 4%.

12.4. Анализ смены сюжета в динамических изображениях

Смена сюжета в видеопоследовательности — это резкое изменение содержание сцены в соседних кадрах [4.45]. Смена сюжета происходит при переключении источника видео (камеры), при соединении фрагментов видео (редактирование) и т. д.

При кодировании видеопоследовательностей с использованием предсказания движения по соседним кадрам следует учитывать смену сюжета по ряду причин:

- предсказание на основе кадров другого сюжета неэффективно (возможны только случайные совпадения), но на предсказание тратятся вычислительные ресурсы;
- учет смены сюжета позволяет повысить качество декодируемого видео при регулировании скорости выходного потока кодирования;

- использование опорных кадров при смене сюжета повышает качество и удобство работы при последующем редактировании закодированного видеоматериала.

Анализ смены сюжета можно проводить по результатам блочного анализа движения: если предсказывается небольшая часть блоков, то произошла смена сюжета.

Этот алгоритм легко встраивается в процедуру кодирования, уже содержащую блочный анализ движения, но имеет некоторые недостатки. При этом алгоритме за смену сюжета будет также приниматься быстрое панорамирование, масштабирование, изменение освещенности и пр. Кроме того, на анализ движения (впоследствии ненужный из-за смены сюжета) расходуются вычислительные ресурсы.

В результате проведения исследований был разработан и предложен быстрый, эффективный и вычислительно простой метод поиска смены сюжета. Он основан на анализе изменения гистограммы распределения яркости в соседних кадрах.

Метод состоит из нескольких шагов, описанных ниже.

R_0	R_1	R_2	R_3
R_4	R_5	R_6	R_7

Рис. 12.22. Разбиение изображения на неперекрывающиеся области

Изображение разбивается на несколько областей. Области могут быть как перекрывающиеся, так и неперекрывающиеся; они могут в совокупности охватывать либо кадр целиком, либо большую часть кадра. В приведенном ниже примере кадр разбит на восемь прямоугольных неперекрывающихся областей $R_i, i = 0, \dots, N - 1, N = 8$, охватывающих все изображение, как показано на рис. 12.22.

Для каждой области j кадра i подсчитывается гистограмма распределения яркостей

пикселов $H_{i,j} = \{h_{i,j,k}\}, j = 0, \dots, (N - 1), k = 0, \dots, (256/S - 1)$. Шаг вычисления гистограммы S больше единицы (в приведенном примере шаг гистограммы $S = 8$). Яркости пикселов распределены в диапазоне от 0 до 255, $h_{i,j,k}$ — процентная доля пикселов области R_j кадра i , имеющих яркость в диапазоне $[k \cdot S, (k + 1) \cdot S)$.

Для каждой области подсчитывается сумма квадратов разностей значений гистограмм текущего и предшествующего кадров: $D_{i,j} = \sum_k (h_{i,j,k} - h_{i-1,j,k})^2$.

Определяется количество областей, в которых $D_{i,j} > T$, где T — пороговое значение (в примере используется $T = 500$).

Если количество областей, в которых порог превышен, не менее половины, то детектируется смена сюжета.

Ниже приводятся примеры, иллюстрирующие эффективность работы этого алгоритма. В примерах рассмотрены различные сюжеты телевизионного качества. При обработке изображений существенно меньшего разрешения или меньшей частоты кадров нужно изменять пороговое значение T определения смены сюжета.

На рис. 12.23 приведены два последовательных кадра со значительным движением фона и гистограммы яркости участков изображений и значения $D_{i,j}$. Смена сюжета не зафиксирована (порог не превышен ни на одном из участков). На графике гистограммы по горизонтали отложен номер группы суммирования

(всего 32 группы, так как шаг вычисления гистограммы равен 8), а по вертикали — процент пикселов в этой группе. Красные прямоугольники — гистограмма текущего кадра, зеленые — предыдущего.

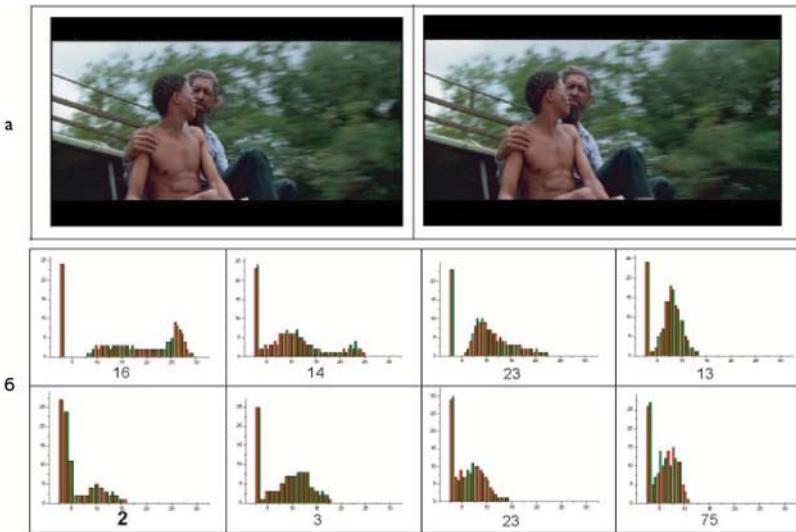


Рис. 12.23. Видеосюжет со значительным движением фона — а и гистограммы яркости участков кадров и соответствующие им значения сумм квадратов разностей для данного видеосюжета — б

На рис. 12.24 представлены два последовательных кадра со значительным движением крупного объекта фона и соответствующие графики. Смена сюжета не зафиксирована (порог не превышен ни на одном из участков).

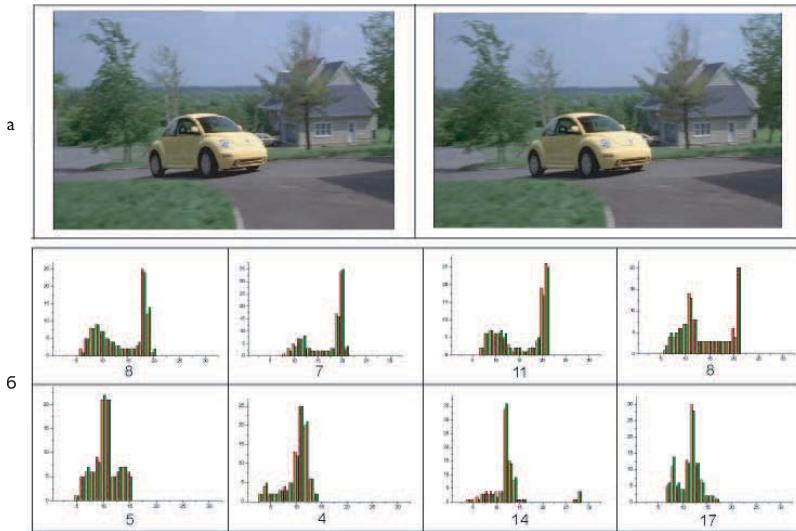


Рис. 12.24. Видеосюжет со значительным движением крупного объекта и фона — а и гистограммы яркости участков кадров с соответствующими им значениями сумм квадратов разностей — б

На рис. 12.25 изображены два последовательных кадра со значительным изменением яркости кадра и соответствующие графики. Смена сюжета не зафиксирована (порог превышен на двух из восьми участков).

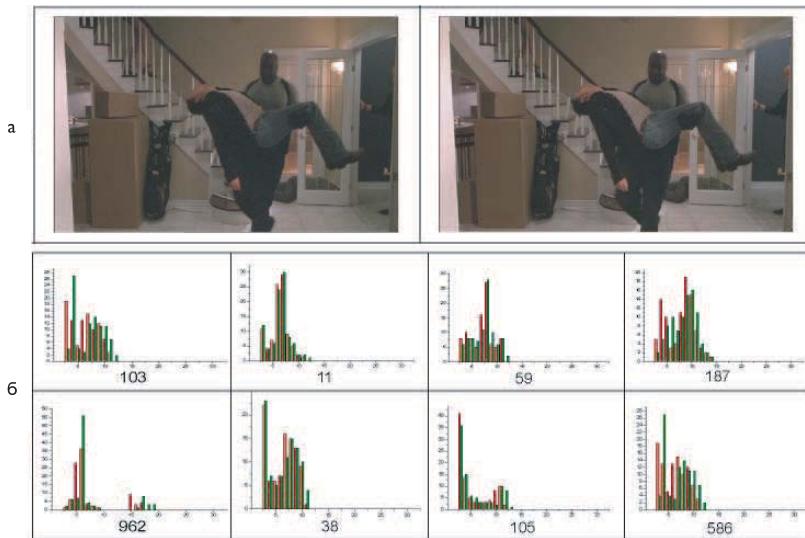


Рис. 12.25. Видеосюжет со значительным изменением яркости кадра — а и гистограммы яркости участков кадров с соответствующими им значениями сумм квадратов разностей — б

На рис. 12.26 приведены два последовательных кадра со сменой плана съемки и соответствующие графики. Смена сюжета зафиксирована (порог превышен на семи из восьми участков).



Рис. 12.26. Видеосюжет со сменой плана съемки — а и гистограммы яркости участков кадров с соответствующими им значениями сумм квадратов разностей — б

На рис. 12.27 представлены два последовательных кадра со сменой сюжета и соответствующие графики. Смена сюжета зафиксирована (порог превышен на четырех из восьми участков).

Таким образом, представленные иллюстрации показывают, что принцип анализа смены сюжета в последовательности кадров, основанный на разбиении кадров на несколько областей и сравнении гистограмм в каждой области, весьма прост в реализации и его использование обеспечит повышение эффективности кодирования динамических изображений.



Рис. 12.27. Последовательность кадров со сменой сюжета — а и гистограммы яркости участков кадров с соответствующими им значениями сумм квадратов разностей — б

Литература к части IV

- 4.1. Цифровая обработка телевизионных и компьютерных изображений. 2-е изд., перераб. и доп. / Под ред. Ю.Б. Зубарева и В.П. Дворковича. М.: НАТ, 1997.
- 4.2. ITU-T Recommendation T.81 / Information Technology – Digital Compression and Coding of Continuous-Tone Still Images: Requirements and Guidelines, 1994.
- 4.3. ITU-T Recommendation T.84 / Information Technology – Digital Compression and Coding of Continuous-Tone Still Images: Extensions, 1997.
- 4.4. *Andelson E.H., Simonselli E.* Orthogonal Pyramid Transforms for Image Coding// SPIE. 1987. V.45.
- 4.5. *Baaziz N. Labit C.* Pyramid Transform for Digital Images// IRISA, Campus de Beaulieu. — 1990. № 526.
- 4.6. *Burt P., Andelson E.* The Laplasian Pyramid as a Compact Image Code// IEEE Trans. Commun. — 1983. V.31. № 4.
- 4.7. *Antonini M., Darlaud M., Mathieu P., Daubechies I.* Image Coding Wavelet Transform// IEEE Trans. Image Proc. 1992. Vol. 1. № 2.
- 4.8. *Mallat S.* A theory for multiresolution signal decomposition: the wavelet representation// IEEE Pattern Anal. and Machine Intel. July, 1989. Vol. 11. № 7.
- 4.9. *Добеши И.* Десять лекций по вейвлетам/ Пер. с англ. Е.В. Мищенко; под ред. А.П. Петухова. М.: РХД, 2001.
- 4.10. *Госалес Р., Вудс Р.* Цифровая обработка изображений/ Пер. с англ. П.А. Чочия. М.: Техносфера, 2005.
- 4.11. *Воробьев В.И., Грибулин В.Г.* Теория и практика вейвлет-преобразований. СПб.: Военный университет связи, 1999.
- 4.12. *Петухов А.П.* Введение в теорию базисов всплесков. СПб.: Издательство СПбГТУ, 1999.
- 4.13. *Дворкович В.П., Мохин Г.Н., Нечепаев В.В., Дворкович А.В.* Способ по-кадрового сжатия изображений // Патент РФ № 2122295, 20.11.1998.
- 4.14. *Дворкович А.В., Дворкович В.П., Мохин Г.Н., Соколов А.Ю.* Способ цифровой обработки динамических изображений// Патент РФ № 2182746, 20.05.2002.
- 4.15. Joint Video Team of ITU-T and ISO/IEC JTC 1, «Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC),» Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, JVT-G050, March, 2003.
- 4.16. *Wiegand Thomas, Sullivan Gary J.* Gisle Bjontegaard, and Ajay Luthra. Overview of the H.264/ AVC Video Coding Standard // IEEE Transactions on Circuits and Systems for Video Technology. July, 2003.
- 4.17. Richardson Iain E. G. H.264 and MPEG-4 Video Compression. Video Coding for Next-generation Multimedia. The Robert Gordon University. Aberdeen, UK, 2003.



- 4.18. *List P., Joch A., Lainema J., Bj G.øntegaard, M. Karczewicz.* Adaptive Deblocking Filter// IEEE Transactions on Circuits and Systems for Video Technology. Vol. 13. № 7. July, 2003.
- 4.19. *Дворкович В.П., Дворкович А.В.* Расчет банков фильтров дискретного вейвлет-преобразования и анализ их характеристик// Цифровая обработка сигналов. 2006. № 2.
- 4.20. *Дворкович В.П., Гильманшин А.В.* Новый подход к использованию двумерных вейвлет-фильтров при обработке изображений: Доклад на 10-й Международной конференции «Цифровая обработка сигналов и ее применение». Москва, март 2008.
- 4.21. *Дворкович В.П., Гильманшин А.В.* Использование трехполосных фильтров вейвлет-фильтров для обработки изображений: Доклад на 10-й Международной конференции «Цифровая обработка сигналов и ее применение», Москва, март 2008.
- 4.22. *Дворкович В.П., Гильманшин А.В.* Новый подход к использованию двумерных вейвлет-фильтров при обработке изображений// Доклад на 10-й Международной конференции «Цифровая обработка сигналов и ее применение», Москва, март 2008.
- 4.23. *Max J.* Quantizing for Minimum Distortion// IEEE Trans. V. JT-6. 1960. № 3.
- 4.24. *Rao K.R., Hwang J.J.* Techniques and Standards for Image, Video and Audio Coding. Prentice Hall PTR, NJ, 1996.
- 4.25. *Jain J.R., Jain A.K.* Displacement measurement and its application in interframe image coding// IEEE Trans. Commun. 1981. V.COM-29, pp. 1799–1808.
- 4.26. *Chen M.J., Chen L.G., Chieuh T.D.* One dimensional full-search motion estimation algorithm for video coding// IEEE Trans. Circuits and Systems for Video Technol. 1994. V.4, pp. 504–509.
- 4.27. *Koga T. et. al.* Motion compensation interframe coding for video conferencing, NTC '81, National Telecomm. Conf. 1981, pp. G5.3.1–G5.3.5.
- 4.28. *Vos L.D.* VLSI — architecture for the hierarchical block matching algorithm for HDTV Applications // SPIE/VCIP. Vol.1360, pp.398–409. Cambridge, MA, Nov, 1990.
- 4.29. *Ландау Л.Д., Либшиц Е.М.* Теоретическая физика: Учебное пособие: В 10 т. Т. VI. Гидродинамика. 3-е изд., перераб. М.: Наука, 1986. С.15.
- 4.30. *Tekalp M.* Digital Video Processing. Prentice Hall PTR. Prentice-Hall, Inc, USA, 1995.
- 4.31. *Kim S., Kuo C.-C.J.* Fast motion vector estimation with a Markov model for MPEG. Proc. of The Int. Society for Optical Engineering. Digital Video Compression: Algorithms and Technologies. 1995. 7–10 Feb., 1995. San Jose, California, Vol. 2419. P. 210.
- 4.32. *Koga T. et al.* Motion compensated interframe coding for video conferencing. NTC 1981: National Telecom. Conf. P. G.5.3.1–G.5.3.5. New Orleans, LA, Nov.-Dec., 1981.

- 4.33. *Schuster G.M., Katsaggelos A.K.* A Theory of Optimal Bit Allocation Between Displacement Vector Field and Displaced Frame Difference// IEEE Journal On Selected Areas in Communication. Vol.15. Num.9. Dec., 1997. P. 1739.
- 4.34. *Дворкович В.П., Нечепаев В.В.* Компенсация движения с использованием преобразования Фурье: Материалы 1-й межд. конф. «Цифровая обработка сигналов и ее применения», 30 июня — 30 июля 1998 г., Россия, Москва, МЦНТИ. Т. III. С. 149.
- 4.35. *Watkinson J.* Textbook for Engineers on Motion Compensation, Snell & Wilcox Ltd, 1994.
- 4.36. *Дворкович В.П., Нечепаев В.В., Соколов А.Ю.* Предварительная обработка изображения для анализа движения методом фазовой корреляции: Материалы 1-й межд. конф. «Цифровая обработка сигналов и ее применения», 30 июня —30 июля 1998 г., Россия, Москва, МЦНТИ. Т. III. С. 143.
- 4.37. *Janusz Konrad, Dufaux Frederic.* Digital Equipment Corp., Improved Global Motion Estimation for N3, ISO/IEC JTC1/SC29/WG11, MPEG97/M3096. San Jose. February, 1998.
- 4.38. *Wei Qi, Yuzhuo Zhong, Shiqiang Yang.* A New Robust Global Motion Estimation Approach. Seoul Meeting of ISO/IEC JTC1/SC29/WG11, MPEG99/M4423. Seoul. March, 1999.
- 4.39. *Wei Qi, Yuzhuo Zhong, Shiqiang Yang.* Improved Method for Global Motion Estimation. Vancouver Meeting of ISO/IEC JTC1/SC29/WG11, MPEG99/M4687. Vancouver. July, 1999.
- 4.40. ISO/IEC WG11. MPEG Video Group, Encoder Optimization Core Experiment Descriptions, ISO/IEC/JTC1/SC29/WG11, No.MPEG00/N3523. Beijing. July, 2000.
- 4.41. ISO/IEC WG11. Optimization Model. Version 3.0. — ISO/IEC/JTC1/SC29/WG11, № 3675.- ISO. Sidney. July, 2001. 17 p.
- 4.42. ITU-T — Coding of Moving Pictures and Audio — MPEG-4 Video Optimization Model OM 2.0, 2000.
- 4.43. *Дворкович А.В., Дворкович В.П., Зубарев Ю.Б., Соколов А.Ю.* Способ анализа векторов движения деталей в динамических изображениях// Патент РФ № 2137194, 10.09.1999.
- 4.44. *Дворкович А.В., Дворкович В.П., Соколов А.Ю.* Способ поиска векторов движения деталей в динамических изображениях// Патент РФ № 2182727, 20.07.2000.
- 4.45. *Дворкович А.В.* Разработка и исследование высокоеффективных систем цифровой обработки динамических изображений и оценки ее качества: Дисс: на соискание ученой степени доктора технических наук. М.: МГТУ им. Н.Э. Баумана, 2007.

ЧАСТЬ V

ОСНОВНЫЕ СТАНДАРТЫ ЦИФРОВОГО КОДИРОВАНИЯ ВИДЕОИНФОРМАЦИИ

Введение

Разработки стандартов эффективного цифрового кодирования статических и динамических изображений ориентированы на решение проблем сокращения объема данных, требуемого для обработки, запоминания, хранения и передачи видеинформации. Основой такого кодирования является удаление статистической и визуальной избыточности. С математической точки зрения это равнозначно преобразованию некоторого двумерного или многомерного массива данных в статистически некоррелированный массив, принимая во внимание способность визуального восприятия малозаметных изменений структуры восстанавливаемых изображений или некоторых их приближений.

Создание в последние 15–20 лет ключевых международных стандартов сжатия статических и динамических изображений наглядно продемонстрировало возможности эффективного практического применения теоретических разработок в данной области цифровой обработки информации.

Настоящий раздел посвящен изложению основных методов сжатия видеинформации, на базе которых разработаны стандарты кодирования статических и динамических изображений. Для хранения видеинформации могут использоваться методы сжатия без потерь, использующие устранение только статистической избыточности изображений.

Кодирование изображений с потерями применяется, например, в видеотелефонии, видеоконференцсвязи, телевизионном вещании, при цифровой реализации которых некоторое изменение изображений является приемлемым компромиссом, обеспечивающим повышение степени сжатия информации.

Одним из объективных критериев оценки верности закодированного, а затем восстановленного изображения является величина среднеквадратичного отклонения разности пикселов исходного изображения и полученного в результате цифровых преобразований:

$$\text{СКО} = \sqrt{\frac{1}{M \cdot N} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (\bar{F}(x, y) - F(x, y))^2},$$

где $M \cdot N$ — размеры изображения; $\bar{F}(x, y)$, $F(x, y)$ — значения уровней восстановленного и исходного изображения в точке (x, y) .

Часто используют величину пикового отношения сигнала/шум, выраженную в децибелах (PSNR): $\text{PSNR} = 20 \lg \left(\frac{F_{\max}}{\text{СКО}} \right)$, где F_{\max} — максимальное значение яркости (цветности) пикселя.

Если для сигнала яркости используется 8-битовая шкала, то $F_{\max} = 2^8 - 1 = 255$.

Хотя приведенные объективные критерии верности воспроизведения кодированных изображений являются простым и удобным принципом оценки потерь информации, все же качество изображения с использованием субъективной оценки является более предпочтительным.

ГЛАВА 13

СТАНДАРТЫ КОДИРОВАНИЯ СТАТИЧЕСКИХ ИЗОБРАЖЕНИЙ

Наиболее широко используемые стандарты сжатия полутоночных и цветных статических изображений разрабатывались группой JPEG (Joint Photographic Expert Group), функционирующей при Международной организации по стандартам (ISO) и сотрудничающей с сектором стандартизации телекоммуникаций Международного союза электросвязи (ITU-T, прежнее название МККТТ, CCITT).

Исходными требованиями при разработке этих стандартов были [5.1, 5.2]:

- возможность адаптивного выбора параметров для оптимизации соотношения сжатие–качество восстановленного изображения в самом широком диапазоне;
- универсальность процедуры обработки изображений любых размеров с любой цветовой гаммой, не должно быть ограничений по сложности сюжета и статистическим свойствам;
- приемлемая сложность вычислений при обработке изображений, реализуемых на компьютерах различной мощности с соответствующим программным обеспечением;
- наличие следующих операционных режимов программы сжатия:
 - **пространственно-поступательная обработка**, при которой изображение, возможно разбитое на блоки, обрабатывается слева направо и сверху вниз;
 - **обработка с последовательным углублением четкости**, при которой наблюдатель получает на экране сначала грубое приближение, а затем программа осуществляет циклическую обработку изображения, с каждым циклом повышая его четкость, добавляя высокочастотные компоненты;
 - **возможность сжатия без потерь**, гарантирующего полное восстановление информации о каждом пикселе при обратной обработке, даже если степень сжатия оказывается небольшой;
 - **обеспечение многоуровневого кодирования**, при котором четкость изображения связана не только с пространственно-частотными диапазонами (например, номерами компонент унитарного преобразования), но и с уровнем точности передачи (или количеством битов двоичного представления величин передаваемых компонент).

В соответствии с этими требованиями стандарты не должны содержать указаний выполнения жесткой последовательности операций обработки, но могут предполагать их выбор по желанию пользователя.

Для сжатия статических монохромных и полноцветных статических изображений наиболее широко используются стандарты ISO (рабочая группа — JTS1/

SC29) — JPEG и JPEG-2000 [5.3]. Информация о разработке этих стандартов приведена в табл. 13.1.

Таблица 13.1. Стандарты сжатия статических изображений

	Название	Информация о стандарте	
		Стандарты ISO	Год публикации
Цифровое сжатие и кодирование неподвижных изображений JPEG	ISO/IEC 10918-1	1994	Требования и руководящие указания
	ISO/IEC 10918-2	1995	Сертификация нормативных тестов на процедуры кодирования и декодирования
	ISO/IEC 10918-3	1997	Стандарт на расширения для JPEG
	ISO/IEC 10918-4	1999	Регистрация профилей JPEG — профилей, цветовых пространств, типов сжатия и др.
Система кодирования изображений JPEG-2000	ISO/IEC 15444-1	2004	Основная система кодирования — определяет базовый набор свойств и синтаксис потокового кода в JPEG-2000
	ISO/IEC 15444-2	2004	Расширения — добавляет новые свойства в основную систему кодирования
	ISO/IEC 15444-3	2007	Motion JPEG2000 — определяет файловый формат, содержащий последовательность изображений, закодированных в соответствии с основной системой кодирования для сжатия видео
	ISO/IEC 15444-4	2004	Нормативные тесты — определяют процедуру тестирования кодирования и декодирования с использованием части 1 стандарта JPEG-2000
	ISO/IEC 15444-6	2003	Комплексный формат файлов — определяет файловый формат JPM для компрессии сложных изображений
	ISO/IEC 15444-9	2005	Интерактивные инструментальные программные средства, интерфейсы прикладного программирования и протоколы
	ISO/IEC 15444-12	2008	Основной формат медиа файлов

13.1. Стандарт JPEG и его модификация

JPEG — один из наиболее полных и достаточно мощных алгоритмов сжатия полутоновых и цветных изображений. Исходное изображение в соответствии со



Рис. 13.1. Структурная схема преобразований 24-битовых изображений в алгоритме JPEG

стандартом может быть многокомпонентным, где каждая i -я компонента представляет собой прямоугольный массив данных с размерностью $X_i \times Y_i$, причем X_i и Y_i не могут превышать величину 2^{16} [5.4–5.10].

В стандарте JPEG оговаривается, что сжатие изображений может осуществляться либо с помощью ДКП двумерных блоков 8×8 пикселов, либо с помощью ДИКМ.

Разрядность величин каждого элемента всех компонент перед подачей на кодер должна быть одинакова и равна N битам:

- 8 или 12 для кодера с дискретным косинусным преобразованием (ДКП);
- от 2 до 16 для дифференциального кодера (ДИКМ).

Сами величины элементов должны быть целыми числами от 0 до 2^{N-1} .

Ко всем компонентам изображения во время обработки должно применяться **одно и то же преобразование**. ДИКМ может осуществлять предсказание с помощью восьми различных масок (приведенных в табл. 13.2) по одному, двум или трем отсчетам.

Таблица 13.2. Варианты масок предсказания при ДИКМ

Номер маски	Формула предсказания $\tilde{X}(i, j)$
0	0
1	$X(i - 1, j)$
2	$X(i, j - 1)$
3	$X(i - 1, j - 1)$
4	$X(i - 1, j) + X(i, j - 1) - X(i - 1, j - 1)$
5	$X(i - 1, j) + [X(i, j - 1) - X(i - 1, j - 1)]/2$
6	$X(i, j - 1) + [X(i - 1, j) - X(i - 1, j - 1)]/2$
7	$[X(i - 1, j) + X(i, j - 1)]/2$

Алгоритм кодирования с применением ДКП разработан в основном для сжатия цветных 24-битовых изображений. На рис. 13.1 приведен конвейер преобразований цветных изображений формата RGB.

После дискретизации RGB-сигналов перевод в цветовое пространство $Y_D C_R C_B$ осуществляется с применением следующих соотношений:

$$\begin{aligned}
 Y_D &= \text{round} \left[\frac{77}{256} \cdot R + \frac{150}{256} \cdot G + \frac{29}{256} \cdot B \right], \\
 C_R &= \text{round} \left[\frac{131}{256} \cdot R - \frac{110}{256} \cdot G - \frac{21}{256} \cdot B \right] + 128, \\
 C_B &= \text{round} \left[-\frac{44}{256} \cdot R - \frac{87}{256} \cdot G + \frac{131}{256} \cdot B \right] + 128,
 \end{aligned} \tag{13.1}$$

здесь $16 \leq (R, G, B, Y_D) \leq 235$, $16 \leq (C_R, C_B) \leq 240$ [5.11, 5.12].

Обратное преобразование имеет вид:

$$\begin{aligned}
 R &= \text{round} [Y_D + 1,371 \cdot (C_R - 128)], \\
 G &= \text{round} [Y_D - 0,698 \cdot (C_R - 128) - 0,336 \cdot (C_B - 128)], \\
 B &= \text{round} [Y_D + 1,732 \cdot (C_B - 128)].
 \end{aligned} \tag{13.2}$$

При таком преобразовании RGB-компоненты будут лежать в интервале 16–235 с возможным попаданием в области 0–15 и 236–255.

В стандарте JPEG используется цветовая модель 4 : 2 : 0, так что при формировании составляющих изображения C_R и C_B набираются отсчеты через строку и через столбец. Таким образом, теряется сразу 3/4 информации о цветовых составляющих изображения и автоматически получается сжатие изображения в два раза. Визуальных искажений изображения при обратном преобразовании это сжатие практически не вызывает.

Исходные значения преобразованных пикселов 24-битовых изображений могут иметь $2^8=256$ уровней. Процесс кодирования обычно начинается со сдвига диапазона значений пикселов на $2^7=128$, что приводит к выравниванию динамического диапазона и некоторому увеличению коэффициента сжатия [5.13].

Составляющая яркости изображения Y_D и сформированные составляющие цветности C_R и C_B разбиваются на матрицы 8×8 , используемые для дискретного косинусного преобразования ДКП:

$$F(u, v) = \frac{1}{4} \cdot C(u) \cdot C(v) \cdot \sum_{n=0}^7 \sum_{m=0}^7 X(n, m) \cdot \cos \frac{\pi u(2n+1)}{16} \cdot \cos \frac{\pi v(2m+1)}{16}, \tag{13.3}$$

$$\text{где } C(k) = \begin{cases} \frac{1}{\sqrt{2}}, & k = 0, \\ 1, & k \neq 0. \end{cases}$$

Конкретный выбор того или иного быстрого алгоритма вычисления ДКП оставлен на усмотрение разработчиков прикладных программ.

Результат вычисления ДКП подвергается обработке в соответствии с известной формулой:

$$F_q(u, v) = \text{round}\{F(u, v)/Q(u/v)\}, \tag{13.4}$$

где $Q(u/v)$ — таблица коэффициентов квантования, разных для различных коэффициентов ДКП и различных компонент изображения.

Выбор и применение конкретных таблиц коэффициентов квантования оставлен на усмотрение пользователей, поскольку они могут быть оптимизированы для конкретных прикладных программ. Однако стандарт JPEG требует применения для всех блоков каждого массива компонент изображения одной и той же таблицы квантования.

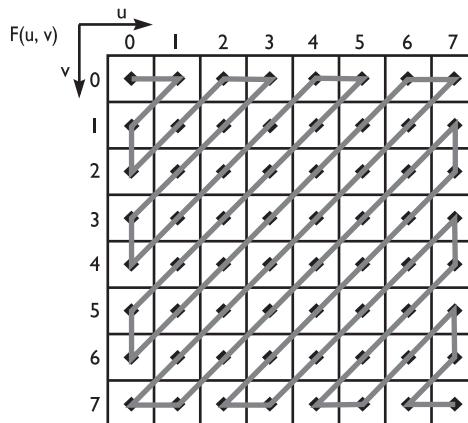


Рис. 13.2. Алгоритм Z-упорядочивания матрицы ДКП

Эмпирически получено несколько таблиц квантования, например таблицы для обработки сигналов яркости и цветности, приведенные ниже в табл. 13.3 [5.6, 5.7].

После квантования компоненты ДКП в каждом блоке должны подвергаться Z-упорядочиванию. Величина $F_q(0, 0)$ заменяется разностью с соответствующей величиной из предыдущего блока (при сканировании слева направо сверху).

На рис. 13.2 поясняется алгоритм Z-упорядочивания отсчетов ДКП преобразования.

Таблица 13.3. Таблицы коэффициентов квантования для

а) сигналов яркости

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

б) сигналов цветности

17	18	24	47	99	99	99	99
18	21	26	66	99	99	99	99
24	26	56	99	99	99	99	99
47	66	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99

В таком виде данные подаются на последовательно соединенные кодер длин серий RLE и хаффмановский кодер.

После вычисления величин $F_q(u, v)$ для восстановления изображения используется обратное преобразование в соответствии с соотношением:

$$\hat{X}(n, m) = \frac{1}{4} \cdot \sum_{u=0}^7 \sum_{v=0}^7 C(u) \cdot C(v) \cdot F_q(u, v) \cdot Q(u, v) \cdot \cos \frac{\pi u(2n+1)}{16} \times \\ \times \cos \frac{\pi v(2m+1)}{16}. \quad (13.5)$$

Одним из основных недостатков такого метода кодирования является использование единой таблицы квантования, что при относительно больших коэффициентах сжатия вызывает искажения восстанавливаемых изображений в виде нарушений воспроизведения мелких деталей, возникновения эхосигналов при резких переходах яркости и цвета, возникновения блочной структуры.

На рис. 13.3а показаны такие искажения изображения «Залив» при его обработке по стандарту JPEG с коэффициентом сжатия, равным 16.

Указанные дефекты могут быть существенно уменьшены, если применять различные таблицы квантования в зависимости от сложности рельефа в каждом блоке, подвергающемся ДКП [5.14].

Операция изменения таблицы квантования может осуществляться следующим образом.

После ДКП и квантования коэффициентов в соответствии с начально выбранной жесткой таблицей квантования на передающей стороне производится обратное преобразование пикселов блока и их сравнение с исходными. Если полученные при этом ошибки преобразования оказываются значительными, то таблица квантования изменяется на менее жесткую и процесс преобразования повторяется. При этом одновременно с данными о квантованных коэффициентах ДКП передается также номер используемой таблицы квантования.

На рис. 13.3б приведено изображение «Залив», обработанное с указанной модификацией стандарта JPEG при сжатии, также равном 16.

Как видно из сравнения рис. 13.3а и б, дефекты воспроизведения этого изображения при изменении коэффициентов квантования в зависимости от «сложности» обрабатываемых блоков существенно меньше.

Существенным недостатком стандарта JPEG является возникновение при больших степенях сжатия искажений типа блокинг-эффекта и мозаичного эффекта.

Блокинг-эффект — это характерное разбиение всего изображения на квадратные блоки 8×8 пикселов с заметными границами. Он возникает вследствие деления изображения на блоки с последующим их независимым кодированием. Характерной особенностью ДКП с учетом квантования коэффициентов является возникновение ненулевых ошибок на границах блоков, которые идентифицируются глазом как скачки яркости от одного блока к другому.

Мозаичный эффект выглядит подобно блокинг-эффекту, но обуславливается не резкими переходами между различными блоками, а заметным глазу различием яркости в соседних блоках. Можно также определить мозаичный эффект как остаточный блокинг-эффект после низкочастотной фильтрации изображения. Несмотря на то, что переходы между блоками становятся плавными, глаз воспринимает изображение как разбитое на блоки. Мозаичный эффект также возникает при слишком грубом квантовании коэффициентов ДКП, при котором изображения внутри соседних блоков сильно отличаются друг от друга.



Рис. 13.3. Искажения изображений при их обработке по стандарту JPEG (а) и при модификации этого стандарта (б)

Эти недостатки кодирования статических изображений практически исключены в стандарте JPEG-2000.

Иногда стандарт JPEG используется и при кодировании динамических изображений (так называемый алгоритм Motion-JPEG или M-JPEG). При этом каждый кадр сжимается независимо, степень сжатия таких изображений не велика.

13.2. Стандарт JPEG-2000

Более мощный и гибкий стандарт JPEG-2000 принципиально расширяет возможности стандарта JPEG при сжатии как полутоновых, так и полноцветных изображений. Он обладает целым рядом преимуществ: доступны более высокие степени сжатия при соответствующем качестве восстановленного изображения,



Рис. 13.4. Структурная схема обработки изображений по стандарту JPEG-2000

возможно использование эффективного режима сжатия без потерь, реализуется прогрессивная визуализация изображения во время загрузки, применяются масштабирование, исправление ошибок и др. Если поддержка стандарта JPEG-2000 будет встроена в популярные программы, то в скором времени он сможет полностью заменить устаревший JPEG. Наиболее важными преимуществами стандарта JPEG-2000 по сравнению с JPEG являются [5.15–5.21]:

- возможность работы с фрагментами изображения, обеспечение при необходимости разбиения изображения на квадратные тайлы и их кодирование независимо друг от друга;
- большая степень сжатия за счет использования дискретного вейвлет-преобразования и более сложного энтропийного кодирования;
- масштабируемость фрагментов изображений: JPEG-2000 обеспечивает бесшовное сжатие разных компонентов изображения благодаря разбиению на блоки, можно хранить изображения разных разрешений в одном кодовом потоке;
- эффективная организация кодового потока позволяет просматривать файл с меньшей разрешающей способностью или с меньшим качеством;
- сжатие как с потерями, так и без потерь при использовании целочисленного вейвлет-преобразования;
- возможность кодирования монохромных изображений, что позволяет обеспечить компрессию изображений с текстом;
- возможность последовательного декодирования и вывода изображения сверху вниз без необходимости буферизации всего изображения и др.

Изначально новый стандарт разрабатывался как база для будущего стандарта сжатия без потерь JPEG-LS [5.22, 5.23], но позднее этот подход был отвергнут в связи с появлением более эффективных алгоритмов.

На рис. 13.4 приведена упрощенная структурная схема конвейера обработки изображений, используемой в соответствии со стандартом JPEG-2000 [5.2, 5.13, 5.24, 5.25].

В отличие от JPEG, кодер JPEG-2000 не требует разбиения изображения на малые квадратные блоки, так как используемое в ходе работы алгоритма дискретное вейвлет-преобразование (ДВП) работает на фрагментах любого размера. Но в случае, когда объем памяти, доступный кодеру для работы, меньше, чем объем памяти, необходимый для кодирования всего изображения, используется

разбиение изображения на прямоугольные тайлы, которые кодируются независимо друг от друга.

После разбиения RGB компоненты каждого тайла перед преобразованием в цветовое пространство $Y_D C_R C_B$ (для симметрирования динамического диапазона сигнала относительно нуля и некоторого увеличения степени сжатия) сдвигаются по интенсивности в соответствии с соотношением:

$$\hat{I}(x, y) = I(x, y) - 2^{N-1}, \quad (13.6)$$

где N — количество битов на пиксел каждой компоненты.

Величина N для каждой компоненты R, G и B определяется кодером и передается декодеру так, что при восстановлении тайла выполняется обратное преобразование:

$$\bar{I}(x, y) = \hat{I}(x, y) + 2^{N-1}. \quad (13.7)$$

Следующий шаг обработки обеспечивает, аналогично стандарту JPEG, преобразование RGB-компонент в цветовое пространство $Y_D C_R C_B$ в соответствии с формулами (13.1). В режиме сжатия изображений без потерь используются несколько иные соотношения:

$$\begin{aligned} \tilde{Y}_D &= \frac{R + 2G + B}{4}; \\ \tilde{C}_R &= R - G; \\ \tilde{C}_B &= B - G. \end{aligned} \quad (13.8)$$

При этом в декодере выполняются обратные преобразования:

$$\begin{aligned} R &= \tilde{C}_R + G; \\ G &= \tilde{Y}_D - \frac{\tilde{C}_R + \tilde{C}_B}{4}; \\ B &= \tilde{C}_B + G. \end{aligned} \quad (13.9)$$

В отличие от алгоритма JPEG, в алгоритме JPEG-2000 не осуществляется прореживание матриц C_R и C_B по строкам и столбцам в целях увеличения степени сжатия. Для достижения этой цели достаточно при преобразовании матриц цветоразностных отсчетов уменьшить количество рекурсивных вейвлет-разбиений по сравнению с числом разбиений при вейвлет-преобразовании матриц яркостных отсчетов.

Далее производятся ДВП яркостной и двух цветоразностных матриц. В стандарте JPEG-2000 при преобразовании этих матриц используют разделимые масштабирующие вейвлет-функции, обработка двумерного сигнала $u(i, k)$ осуществляется сначала по строкам, а затем по столбцам. Двумерное вейвлет-преобразование реализуется с помощью операции фильтрации и прореживания выборки.

Согласно стандарту JPEG-2000, при этом применяются КИХ-фильтры с нечетным числом отсчетов цифровой решетки и линейной (нулевой) ФЧХ. Частотные характеристики ФНЧ и ФВЧ соответственно можно представить в виде [5.26]:

$$H(x) = h_0 + 2 \sum_{n=1}^N h_n \cdot \cos \pi n x, \quad G(x) = g_0 + 2 \sum_{m=1}^M g_m \cdot \cos \pi m x, \quad 0 \leq x \leq 1, \quad (13.10)$$

где $M + N$ — нечетное число. В случае, если

$$H(0) = h_0 + 2 \sum_{n=1}^N h_n = C_1, \quad H(1) = h_0 + 2 \sum_{n=1}^N (-1)^n h_n = 0,$$

$$G(1) = g_0 + 2 \sum_{m=1}^M (-1)^m g_m = C_2, \quad G(0) = g_0 + 2 \sum_{m=1}^M g_m = 0,$$

восстанавливающие НЧ и ВЧ-фильтры соответственно определяются так:

$$K_h(x) = g_0 + 2 \sum_{n=1}^M (-1)^n g_m \cos \pi mx, \quad (13.11)$$

$$K_g(x) = h_0 + 2 \sum_{n=1}^N (-1)^n h_n \cos \pi nx.$$

Если константы C_1 и C_2 равны, то $C_1 = C_2 = \sqrt{2}$. Выполнение этого условия не является обязательным, однако необходимо, чтобы $C_1 \cdot C_2 = 2$.

В табл. 13.4 представлены коэффициенты импульсных характеристик фильтров 7/9 (ФНЧ и ФВЧ), используемых при необратимом вейвлет-преобразовании в случае сжатия изображений с потерями [5.15, 5.16].

Таблица 13.4. Коэффициенты импульсных характеристик ФНЧ и ФВЧ при сжатии с потерями

n	h_n	m	g_m
0	1,115087052456994	0	0,6029490182363579
1	0,5912717631142470	1	-0,2668641184428723
2	-0,05754352622849957	2	-0,07822326652898785
3	-0,09127176311424948	3	0,01686411844287495
—	—	4	0,02674875741080976

Формы импульсных характеристик ФНЧ анализа $H(x)$ и синтеза $K_h(x)$ приведены на рис. 13.5а и б соответственно. Формы импульсных характеристик ФВЧ анализа $G(x)$ и синтеза $K_g(x)$ также приведены соответственно на рис. 13.5а и б.

На восстанавливающие фильтры $K_h(x)$ и $K_g(x)$, определенные соотношением (13.11), при синтезе изображения поступают сигналы:

$$\begin{cases} \hat{Y}_h(2k) = h_0 \cdot Y(2k) + \sum_{n=1}^N h_n \cdot [Y(2k-n) + Y(2k+n)], \\ \hat{Y}_h(2k-1) = 0, \end{cases} \quad (13.12)$$

$$\begin{cases} \hat{Y}_g(2k-1) = g_0 \cdot Y(2k-1) + \sum_{m=1}^M g_m \cdot [Y(2k-n-1) + Y(2k+n-1)], \\ \hat{Y}_g(2k) = 0. \end{cases} \quad (13.13)$$

Здесь $N = 3$, $M = 4$, $C_1 = 2$, $C_2 = 1$, а $Y(k)$ — дискретные отсчеты входного сигнала.

При сжатии информации без потерь используются более простые фильтры 5/3, параметры которых приведены в табл. 13.5.

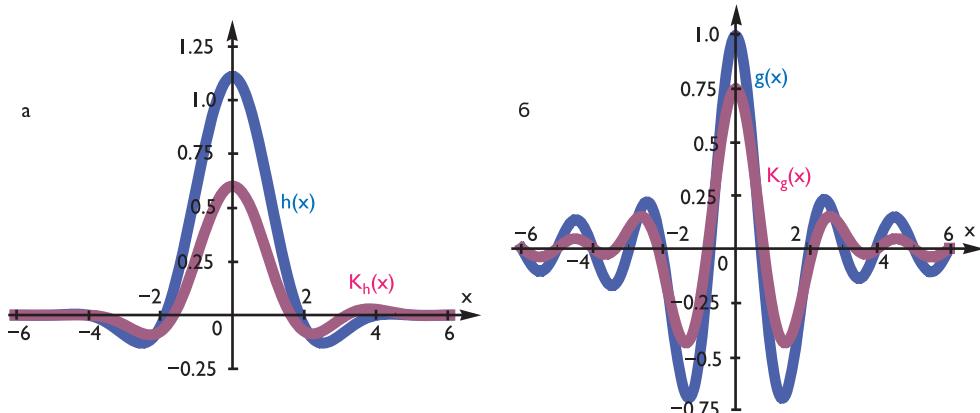


Рис. 13.5. Формы импульсных характеристик ФНЧ и ФВЧ, используемых при сжатии с потерями и при восстановлении изображений

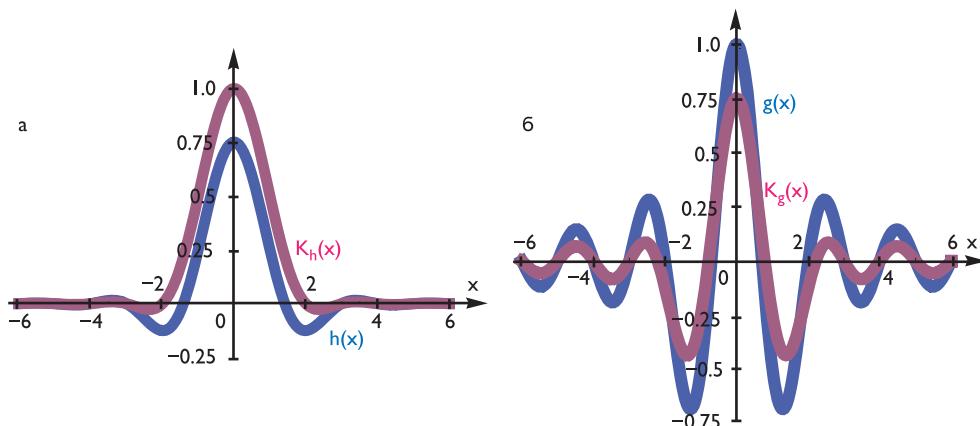


Рис. 13.6. Формы импульсных характеристик ФНЧ и ФВЧ, используемых при сжатии без потерь и при восстановлении изображений

Таблица 13.5. Коэффициенты импульсных характеристик ФНЧ и ФВЧ при сжатии без потерь

n	h_n	m	g_m
0	3/4	0	1
1	1/4	1	-1/2
2	-1/8	-	-

В данном случае $N = 2$, $M = 1$, $C_1 = 1$, $C_2 = 2$ и выполняются все соотношения (13.10)–(13.13). Формы импульсных этих характеристик ФНЧ анализа $H(x)$ и синтеза $K_h(x)$ приведены на рис. 13.6а и б соответственно. Формы импульсных характеристик ФВЧ анализа $G(x)$ и синтеза $K_g(x)$ также приведены соответственно на рис. 13.6а и б.

Для того чтобы вейвлет-преобразование можно было бы применять к крайним пикселям изображения, изображение симметрично достраивается по всем четырем сторонам на несколько пикселов. В худшем случае необходимо достраивать изображение в каждую сторону на четыре пикселя (при использовании сжатия информации с потерями).

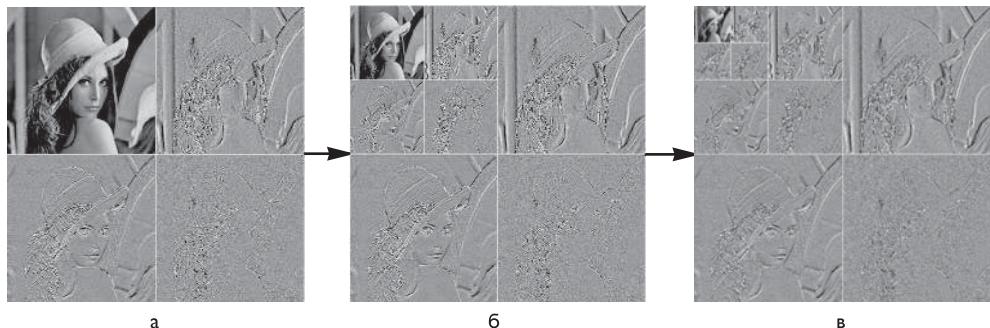


Рис. 13.7. Поэтапное дискретное вейвлет-преобразование изображения

После одного этапа дискретного вейвлет-преобразования, которое применяется сначала ко всем строкам изображения, а затем ко всем его столбцам, обрабатываемый тайл или изображение в целом делится на четыре квадранта (рис. 13.7а):

- НН — низкие частоты по строкам и столбцам;
- ВН — высокие частоты по строкам и низкие по столбцам;
- НВ — низкие частоты по строкам и высокие по столбцам;
- ВВ — высокие частоты по строкам и столбцам.

В левом верхнем квадранте НН формируется уменьшенная копия изображения, а в остальных трех — высокочастотная информация. После этого подобная операция может многократно применяться, но уже только к НН-квадранту изображения, как показано на рис. 13.7а и б. Для обычного изображения возможно использование до 8 этапов.

При сжатии изображений без потерь для хранения данных НН-, ВН- и НВ-квадрантов должно быть выделено на 1 бит больше, чем для исходной информации, а для хранения данных ВВ-квадранта — на два бита больше. При записи преобразованной информации в файл часто используют иерархическую структуру дискретного вейвлет-преобразования, помещая коэффициенты преобразований большего уровня в начало файла. Это позволяет реализовать просмотр изображения без полной распаковки файла или плавно повышать четкость изображения.

При квантовании коэффициентов дискретного вейвлет-преобразования используется система квантования с «мертвой зоной» в области нулевых значений. Интервал с расширенным диапазоном около нуля обеспечивает увеличенное количество равных нулю уровней на выходе квантователя.

На рис. 13.8а пороговые уровни и уровни квантования обозначены через d_i и r_i соответственно; они располагаются симметрично относительно нуля. Для каждой i -й области преобразования могут использоваться различные коэффициенты квантования Q_i :

$$\hat{Y}_i(n, m) = \text{sign}[Y_i(n, m)] \cdot \left\lfloor \frac{|Y_i(n, m)|}{Q_i} \right\rfloor. \quad (13.14)$$

Кодирование полученных округленных коэффициентов выполняется побочно. По стандарту JPEG-2000 непосредственно перед кодированием квадранты для того, чтобы осуществить более гибкую организацию сжатой информации, и для

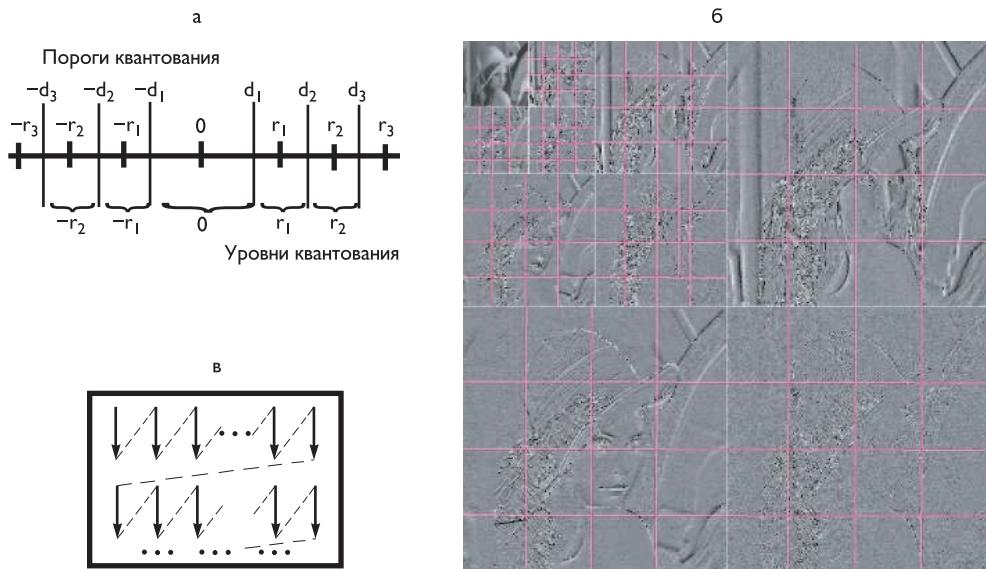


Рис. 13.8. Обработка изображений при квантовании различных квадрантов

повышения помехоустойчивости разбиваются на достаточно малые блоки (например, размером 64×64 , 32×32 и даже менее) так, чтобы все блоки одного квадранта были одинакового размера. Пример разбиения фрагментов изображения на блоки приведен на рис. 13.8б.

Каждый выделенный блок кодируется по отдельности. Алгоритм кодирования обходит матрицу коэффициентов (13.14) каждого блока полосами, как показано на рис. 13.8в. Блоки разбиваются на участки с номинальной высотой, равной 4. Далее полосы сканируются сверху вниз, а колонки в каждой полосе обходятся слева направо.

При кодировании преобразованные вейвлет-коэффициенты представляются в виде битовых плоскостей; одну из них составляют знаки коэффициентов, а остальные соответствуют различным разрядам величин коэффициентов от старшего разряда по убыванию. Может случиться так, что ряд наиболее значимых битовых плоскостей (плоскостей старших разрядов, НЗБ-плоскостей), не будут содержать единиц. В этом случае НЗБ-плоскостью становится первая по порядку плоскость, содержащая хотя бы одну единицу. Предшествующие ей пустые плоскости опускаются при кодировании, а информация об их количестве заносится в заголовок блока.

Арифметическое кодирование битовых плоскостей блоков основано на контекстно-зависимой модели. Контекст формируется как функция от значений битов, окружающих кодируемый бит. Каждая битовая плоскость, кроме НЗБ-плоскости, кодируется в **три стадии**, к НЗБ-плоскости применяется только третья стадия.

При этом кодировании вводится понятие контекстной значимости. Бит считается значимым, если он равен единице. Но незначимый бит (равный нулю) может

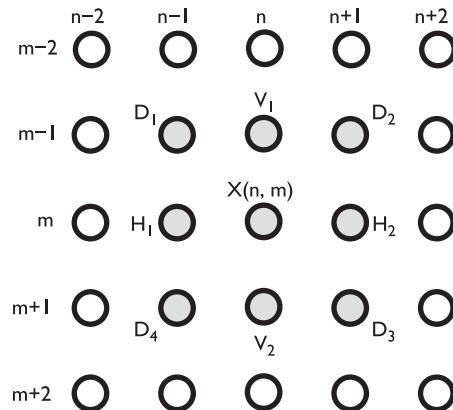


Рис. 13.9. Схема определения значимости текущего бита $X(n, m)$

быть контекстно-значимым, если его контекст $K_{SIG}(n, m)$ больше нуля. Определение контекстно-зависимой значимости бита $X(n, m)$ осуществляется таблично с помощью оценки контекстов, определяющих значимость битов по горизонтали, по вертикали и по диагоналям:

$$\begin{aligned} K_H(n, m) &= z(n-1, m) + z(n+1, m), \\ K_V(n, m) &= z(n, m-1) + z(n, m+1), \\ K_D(n, m) &= z(n-1, m-1) + z(n+1, m-1) + z(n-1, m+1) + z(n+1, m+1), \end{aligned} \quad (13.15)$$

где $z(x, y)$ — значимость бита в ячейке с координатами (x, y) . $K_{SIG}(n, m)$ равно нулю тогда и только тогда, когда все три значения $K_H(n, m)$, $K_V(n, m)$ и $K_D(n, m)$ равны нулю, что, в свою очередь, означает, что все восемь окружающих битов незначимы. Положения окружающих битов, используемых при определении значимости текущего бита, проиллюстрированы на рис. 13.9. Биты, попадающие вне границ кодового блока, считаются незначимыми.

Битовая позиция помечается как значимая после того, как в ней в первый раз появится единица (при последовательной обработке битовых плоскостей от старшей к младшей).

Во время **первого кодового прохода**, называемого проходом распространения значимости, для каждой битовой позиции выполняются следующие действия:

- если битовая позиция не помечена как значимая, а контекст больше нуля, то кодируется значение коэффициента в этой позиции и его контекст, помечается, что битовая позиция закодирована во время первого кодового прохода;
- если бит в текущей позиции впервые равен единице (т.е. в текущей битовой плоскости находится старший бит соответствующего этой позиции коэффициента), то позиция помечается как значимая и кодируется знак коэффициента.

Во время **второго кодового прохода**, называемого проходом уточнения величины, кодируются данные в битовых позициях, которые стали значимыми при

кодировании предыдущей битовой плоскости, но не были закодированы на первом кодовом проходе. При этом кодируется значение коэффициента и уточненное значение контекста.

Третий кодовый проход называется проходом очистки. На нем кодируются биты, пропущенные на двух предшествующих проходах. Так как оставшиеся биты незначимы, то используется режим группового кодирования последовательностей нулей, позволяющий уменьшить размер выходного кода.



Рис. 13.10. Сравнение эффективности сжатия изображений в соответствии со стандартами JPEG и JPEG-2000

Существенной деталью, предусмотренной стандартом, является возможность пропуска кодовых проходов, что является еще одним источником повышения эффективности за счет информационных потерь (первым, наиболее явным источником является квантование). Данная возможность активно используется для осуществления контроля над скоростью данных генерируемого выходного кода.

Важным преимуществом JPEG-2000 является возможность доступа к отдельным элементам изображения без полного его декодирования. Обеспечивается такая возможность, во-первых, разбиением исходного изображения на непересекающиеся тайлы, кодируемые как отдельные изображения, а во-вторых, представлением кода отдельного тайла в виде слоев, каждый из которых является суммарным кодом коэффициентов, соответствующих некоторой его области. Слои, в свою очередь, делятся на пакеты, содержащие код блоков коэффициентов на разных уровнях декомпозиции. Для того чтобы декодировать какую-либо область изображения, достаточно определить, каким тайлам она принадлежит и какие слои, относящиеся к этим тайлам, содержат код блоков коэффициентов, необходимых для восстановления требуемой области.

Стандарт JPEG-2000 превосходит по эффективности стандарт JPEG примерно в 2 раза при сжатии с потерями и на 5–20% при сжатии без потерь.

На рис. 13.10 представлены взятые из [5.25] результаты сжатия тестового файла с разными степенями компрессии — 20, 40, 70 и 145 (слева — в формате JPEG, справа — в формате JPEG-2000). Степень сжатия JPEG выбиралась таким образом, чтобы размер файла был таким же, как после сжатия по JPEG-2000.

ГЛАВА 14

СТАНДАРТЫ КОДИРОВАНИЯ ДИНАМИЧЕСКИХ ИЗОБРАЖЕНИЙ

Основные проблемы кодирования динамических изображений связаны с наличием большого объема информации и необходимостью ее сжатия в реальном масштабе времени. Так, например, динамические изображения стандартного телевидения с форматом 720×576 пикселов при 25 кадрах в секунду в системе RGB создают поток данных в 240 Мбитов/с. В этом случае стандартные алгоритмы сжатия каждого кадра (например, M-JPEG или M-JPEG-2000) не обеспечивают требуемой степени компрессии (при условии достаточно высокого качества восстановленных изображений) и скорости обработки информации.

По этой причине применяемые алгоритмы кодирования динамических изображений используют методы сокращения визуальной избыточности на базе учета внутrikадровых корреляционных связей, подобия отдельных деталей в изображениях, а также анализа и компенсации движения деталей в последовательности кадров.

Основные стандарты кодирования динамических изображений разрабатывались группой экспертов в области цифрового видео MPEG (Moving Pictures Experts Group), функционирующей при Международной организации по стандартам (ISO) и сотрудничающей с сектором стандартизации телекоммуникаций Международного союза электросвязи (ITU-T, прежнее обозначение CCITT).

Стандарты MPEG (ISO/IEC), как и Рекомендации ITU-T H Series определяют лишь цифровой поток закодированного сигнала, неявно задавая алгоритмы декодирования и в меньшей степени — кодирования информации. Алгоритмы сжатия динамических изображений весьма гибки, что при различных реализациях может обеспечить существенное различие в качестве декодируемой информации при одной и той же степени ее сжатия.

Информация о разработке этих стандартов приведена в табл. 14.1.

Рекомендованный ITU-T стандарт H.261 [5.27, 5.28] регламентирует передачу видеинформации при уровнях битового потока, кратных 64 кбита/с. Он предназначен для видеотелефонной связи или видеоконференций в реальном масштабе времени. Стандарт MPEG-1 [5.29, 5.30] используется при уровнях битового потока порядка 1,5 Мбитов/с и предназначен в основном для мультимедиа. Стандарт MPEG-2 [5.33–5.37], определенный также рекомендацией ITU-T H.262 [5.38], применяется при более высоких уровнях битового потока, он предназначен для высококачественной цифровой передачи телевизионной информации, в том числе телевидения высокой четкости.

Принципы формирования цифрового потока, заложенные в стандартах MPEG, используются практически во всех применяемых в настоящее время системах цифрового телевидения высокой четкости [5.39, 5.40], а один из уточняющих

Таблица 14.1. Стандарты видеокодирования

Название	Стандарты ISO/ Рекомендации ITU-T	Год публикации	Информация о стандарте		
			Размер кадров	Частота кадров, Гц	Скорость битового потока
MPEG-1 Video	ISO/IEC 11 172-2	1993	SIF (352 × 240) CIF (352 × 288)	30 25	до 1,5 Мбитов/с
H.261	ITU-T Rec. H.261	1993	QCIF (176 × 144) CIF (352 × 288)	30	$p \times 64$ кбитов/с; $1 \leq p \leq 30$
MPEG-2 Video (H.262)	ISO/IEC 13 818-2 / ITU-T Rec. H.262	1995 (2000)	CIF, 4CIF, 16CIF	25–50 30–60	3–20 Мбит/с
H.263	ITU-T Rec. H.263	1998 (2005)	subQCIF (128 × 96), CIF, 4CIF, 16CIF и др.	25–50 30–60	64– 16 384 кбита/с
MPEG-4 Video	ISO/IEC 14 496-2	1999 (2004)	от subQCIF до 4000 × 4000	до 72	5–180 000 кбитов/с
MPEG-4 AVC (H.264)	ISO/IEC 14 496-10 / ITU-T Rec. H.264	2003 (2011)	от sub-QCIF до 4 096 × 2 304	15–120	15–960 000 кбитов/с

стандартов MPEG-2 специфицирует общие требования стандарта применительно к системам цифрового телевидения различного разрешения [5.41].

Стандарт ITU-T H.263 используется в системах видеотелефонной связи при низких уровнях битового потока, например по коммутируемой телефонной линии [5.42, 5.43]. Особое место в числе прочих разработок занимает стандарт MPEG-4, который предназначен для создания контекстно ориентированных систем сжатия и универсального языка обработки видео- и аудиоинформации [5.44–5.47]. Часть 10 этого стандарта [5.48] (рекомендация ITU-T H.264 [5.49]) широко используется в современных системах цифрового телевизионного вещания.

В основу процесса обработки изображений во всех упомянутых стандартах легли процедуры внутрикадрового кодирования (например, двумерное дискретное косинусное преобразование блоков пикселов, квантование компонент ДКП с помощью таблиц коэффициентов квантования, хаффмановское или арифметическое кодирование), а также принцип компенсации движения и обработка сигнала во времени с помощью интерполяции и экстраполяции.

Первоначально последовательность кадров динамического изображения разделяется на так называемые I-, P- и B-кадры. Все они разбиваются на макроБлоки, которые являются структурным элементом для компенсации движения. В P- и B-кадрах осуществляется компенсация движения, причем в P-кадрах — относительно ближайшего предыдущего I- или P-кадра, а в B-кадрах — усредненная компенсация относительно двух ближайших предыдущего и последующего I- или P-кадров.



Изображения с внутрикадровым кодированием — I-кадры — обрабатываются без использования информации из других кадров. Они обычно используются как опорные для дальнейшего предсказания и обеспечивают точки доступа к кодированной последовательности, начиная с которых возможно декодирование, но степень сжатия их сравнительно невелика.

Изображения, кодируемые с предсказанием (P-кадры), обрабатываются более эффективно благодаря использованию для предсказания и компенсации движения предыдущих I- или P-кадров. Изображения, кодируемые с двунаправленным предсказанием (B-кадры), обеспечивают наибольшую степень сжатия, но требуют ссылок как на предыдущие, так и на последующие кадры для компенсации движения. B-кадры никогда не используются в качестве опорных для предсказания (кроме стандарта MPEG-4 AVC).

Четвертый тип кадров, определенный в спецификации MPEG-1 и не используемый в MPEG-2 (D-кадры), предназначен для того, чтобы обеспечить простой режим просмотра при ограниченном качестве. При кодировании эти кадры используют только внутрикадровую информацию и только нулевые коэффициенты косинусного преобразования каждого из блоков (так называемые DC-коэффициенты). Эти кадры не встречаются в последовательностях, содержащих другие типы кадров.

Разности между исходными величинами в P- и B-кадрах и их оценками с учетом найденных векторов движения:

$$\Delta X_P(i, j) = X_P(i, j) - X_I(i + \Delta i, j + \Delta j), \quad (14.1)$$

$$\Delta X_B(i, j) = X_B(i, j) - [X_I(i + \Delta i_1, j + \Delta j_1) + X_I(i + \Delta i_2, j + \Delta j_2)]/2 - \quad (14.2)$$

разбиваются на блоки и подвергаются процедуре дискретного косинусного преобразования.

Как и в стандарте JPEG, квантование блоков компонент ДКП производится с помощью таблиц коэффициентов квантования, которые могут быть оптимизированы эмпирическим путем и могут передаваться в заголовке к сжимаемой видеинформации вместе с другими вспомогательными параметрами.

Особенностью стандартов MPEG является то, что для квантования блоков предлагается применять переменную шкалу квантования, неравномерную в окрестности нуля, чтобы частично скомпенсировать проявление неустойчивости предсказания P- и B-кадров относительно ошибок в I-кадрах.

После указанной обработки производится упорядочение блоков проквантованных компонент и подача их на статистический кодер.

Специальным условием в обработке движущихся изображений может быть жесткое ограничение цифрового потока. Поэтому в стандартах MPEG возможно применение различных алгоритмов управления постоянством скорости передаваемого битового потока.

Сжатие видеинформации обычно осуществляется в четыре этапа:

- преобразование сигналов основных цветов R, G, B в сигнал яркости Y и цветоразностные сигналы C_B , C_R , уменьшение разрешения сигнала цветности (обычно в два или в четыре раза);
- предсказание кадра на основе предыдущих и/или последующих кадров, использование процедуры оценки движения и его компенсация;

- двумерное дискретное преобразование и квантование коэффициентов преобразования;
- статистическое кодирование данных.

Поддержание постоянной скорости передачи данных в канале связи осуществляется обычно путем изменения уровня квантования коэффициентов преобразования.

Проектирование систем осуществляется таким образом, чтобы искажения не были заметны с нормальной дистанции просмотра, равной утроенной высоте экрана, а с более близких расстояний были заметны в редких случаях.

В данном разделе рассмотрены синтаксис и структура видеопотока вышеуказанных стандартов.

14.1. H.261 — первый стандарт видеокодирования МСЭ

Рекомендация ITU-T H.261 [5.28] была разработана для передачи видеоинформации при уровнях битового потока $p \times 64$ кбита/с, где p может меняться от 1 до 30. Возможные приложения стандарта H.261 включают, например, видеотелефонную связь или видеоконференции в реальном масштабе времени. Стандарт включает как кодирование отдельных кадров в стиле JPEG, так и использование компенсации движения для устранения временной корреляции между кадрами. Таким образом, он относится к гибридным системам сжатия, использующим два метода устранения избыточности информации — в пространственной и временной областях. Методика аудиокодирования не включается в рамки стандарта H.261, но подразумевается использование одной из рекомендаций по аудиокодированию серии G, перечисленных в ITU-T H.320 [5.50] для узкополосных видеотелефонных систем (narrow-band visual telephone systems) и терминального оборудования (terminal equipment).

Форматы исходных данных — CIF и QCIF

Для того чтобы обеспечить преобразование данных различных визуальных сервисов к единому стандарту, был разработан формат CIF (common intermediate format, общий промежуточный формат). Для яркостной компоненты Y разрешение составляет 288 пикселов по вертикали и 360 пикселов по горизонтали, из которых не используются по четыре крайних пикселя слева и справа для обеспечения кратности 16. Используется цветовая модель — 4 : 2 : 0 с серединным расположением пикселов цветности. Таким образом, для яркости используется разрешение 352×288 (область значимых пикселов), а для обеих цветоразностных компонент разрешение 176×144 . Используется также формат QCIF с половинным разрешением по горизонтали и по вертикали — 176×144 для яркости и 88×72 для цветности. Чересстрочная развертка не применяется. Иногда используется специальный формат CIF-NTSC, содержащий 240 вместо 288 строк, поскольку стандарт NTSC предусматривает меньшее разрешение поперек воспроизведимого растра.

Частота кадров по умолчанию составляет $30\,000/1001 \approx 29,97$ кадров/сек, но может быть и понижена до 15 или 10 кадров/сек. Декодер должен быть способен

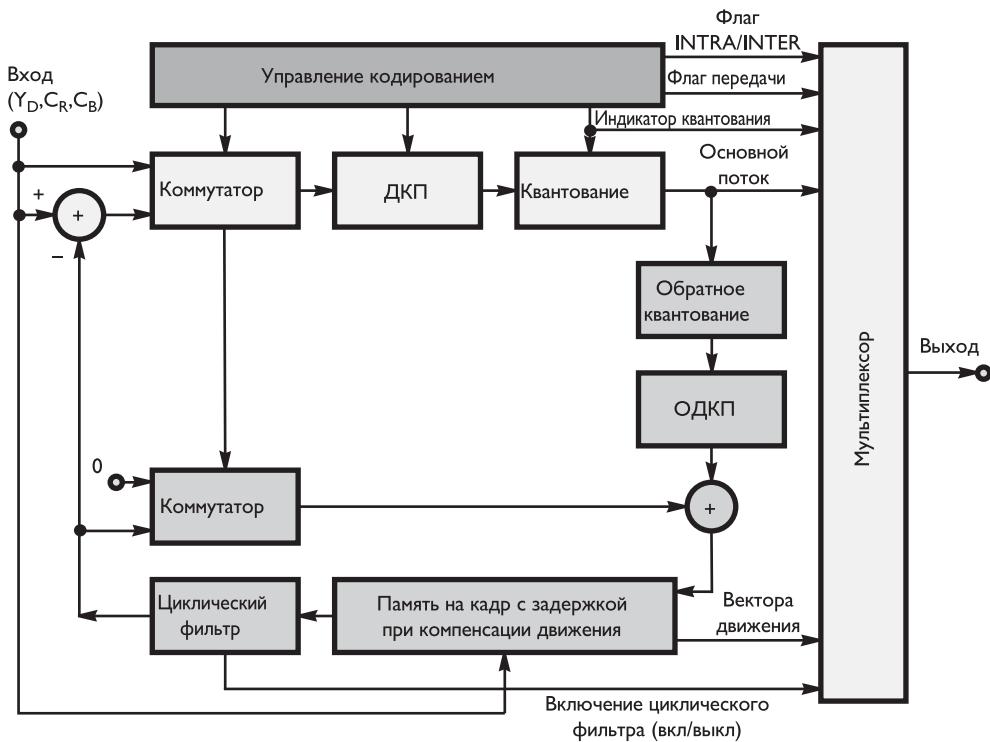


Рис. 14.1. Структурная схема кодера стандарта H.261

раскодировать поток с пропущенными кадрами, так как для увеличения сжатия предусмотрена возможность опускать при кодировании отдельные кадры вместо того, чтобы поддерживать постоянную частоту кадров.

Алгоритм кодирования

На рис. 14.1 приведена структурная схема кодирующего устройства [5.28].

Стандарт не специфицирует конкретных методов сжатия, и поиск наиболее эффективных алгоритмов сжатия из набора возможностей является задачей разработчиков кодера. Отметим, что для передачи CIF-изображения по каналу DS_0 (64 кбита/сек) степень сжатия должна превышать 300 : 1.

В целом в алгоритме кодирования можно выделить следующие этапы:

1. Входной поток подвергается предварительной обработке:

- если исходное изображение передается в виде чересстрочных полей, то из них формируются кадры с прогрессивной разверткой; кадры передискретизуются до формата CIF или QCIF;
- производится преобразование RGB в $Y_D C_R C_B$; пиксели исходного изображения представляются восемью битами (от 0 до 255);
- производится преобразование из формата цветности 4 : 4 : 4 в 4 : 2 : 0 (дву-кратная горизонтальная и вертикальная поддискретизация цветоразност-

ных компонентов); для устранения возможных искажений в виде появления ложных элементов на границе объектов (*aliasing*) или смещения позиции (*shifting*) может применяться низкочастотная префильтрация.

2. Изображение разбивается на макроблоки, для которых находятся векторы движения. Векторы движения для макроблока могут быть только целочисленными и по абсолютной величине не превышать 15 пикселов.

3. Находятся ошибки предсказания движения, которые представляются девятым битами (от -256 до $+255$).

4. Проводится анализ информации о движении и принимается решение о способе кодирования макроблока.

5. В зависимости от результатов предыдущей стадии или исходный, или разностный сигнал подвергается дискретному косинусному преобразованию.

6. Осуществляется квантование коэффициентов ДКП. Количество возможных коэффициентов квантования — один для коэффициента ДКП с индексом (0,0) (так называемый DC-коэффициент) блоков без предсказания (INTRABлоков) и 31 для всех остальных коэффициентов. Для всех коэффициентов ДКП макроблока (кроме INTRA DC) используется один коэффициент квантования. INTRA DC-коэффициент квантуется линейно с шагом 8 и без «мертвой зоны», остальные квантователи также линейны, но имеют «мертвую зону» около нуля, шаг квантования — четное число в диапазоне от 2 до 62. Восстановливаемые после обратного квантования, ОДКП и предсказания значения пикселов ограничиваются в диапазоне $0 \div 255$, т. е. значения, меньшие нуля, заменяются нулем, а значения, большие 255, заменяются на 255.

7. Z-упорядочивание и кодирование кодами переменной длины. На этом этапе необходимо строить выходной поток данных, поддерживая заданное значение битового потока, для чего требуется специальный алгоритм выбора уровня квантования. Этот алгоритм использует обратную связь между степенью заполнения буфера кодера и уровнем квантования коэффициентов: если заполнение буфера оказывается больше заданной пороговой величины, то уменьшается точность передачи данных.

Контроль скорости выходного потока кодера возможен с помощью изменения коэффициентов квантования, передачи или пропускания блоков, пропуска целых кадров, однако процедура регулирования выходит за рамки рекомендации.

Для того чтобы не происходило излишнее накопление ошибок, рекомендация предусматривает передачу макроблока в режиме без предсказания не реже, чем раз в 132 кадра.

Синтаксис видеопотока

Стандартом предусмотрено разбиение битового потока на четыре уровня [5.27]: уровень кадра, уровень группы блоков, уровень макроблока, уровень блока.

Уровень кадра состоит из заголовка и данных для групп блоков.

Заголовок кадра содержит стартовый код кадра (20 битов, 0000 0000 0000 0001 0000), временную метку кадра (5 битов — номер кадра), информацию о типе кадра (6 битов) и, возможно, дополнительную информацию (0/8/16 \cdots + 1 бит).

Кадр состоит из 12 групп блоков (396 макроблоков) — для формата CIF или из 3 групп блоков (99 макроблоков) для формата QCIF.



Группа блоков (33 макроблока) охватывает область 176×48 пикселов яркости и соответствующие им 88×24 пикселя цветоразностных сигналов.

Уровень группы блоков (GOB) состоит из заголовка и данных макроблоков. Заголовок группы блоков содержит стартовый код (16 битов, 0000 0000 0000 0001), номер группы блоков в кадре (4 бита), код коэффициента квантования (5 битов) и, возможно, дополнительную информацию (0/8/16 … + 1 бит).

Уровень макроблока (16×16 пикселов, 4 блока яркости; 8×8 пикселов, по 1 блоку цветности, всего 6 блоков) состоит из заголовка макроблока и данных блоков. Заголовок макроблока содержит адрес (код переменной длины с уникальным префиксом от 1 до 11 битов), тип (код переменной длины от 1 до 10 битов), код коэффициента квантования, если это определено типом макроблока (5 битов), горизонтальная и вертикальная компоненты вектора движения в дифференциальном виде, если это определено типом макроблока (код переменной длины от 1 до 11 битов), трафарет передаваемых блоков, если это определено типом макроблока (код переменной длины от 3 до 9 битов).

Тип макроблока определяет следующие параметры: с предсказанием (INTER) или без предсказания (INTRA); передается ли коэффициент квантования, или он не отличается от коэффициента квантования предыдущего макроблока; с компенсацией движения (передается вектор движения) или без нее; передаются ли коэффициенты блоков; используется ли фильтр при предсказании движения.

Макроблок содержит шесть блоков — четыре блока яркости и по одному цветоразностному блоку. Коэффициенты блока переупорядочены из двумерного в одномерный вид по определенному закону (так называемое Z-упорядочивание).

Уровень блока состоит из кодов коэффициентов блока (от 2 до 14 или 20 битов) и кода окончания блока (2 бита).

Коэффициенты блока в Z-порядке кодируются с помощью методики run-length: одним кодом обозначается последовательность нулей (run) и ненулевое значение после них (level). Наиболее часто встречающиеся пары кодируются кодами переменной длины, а остальные — специальным кодом (ESCAPE-код, 6 битов), за которым идут 6-битный код количества нулей (run) и 8-битный код ненулевого значения (level). Последние нули блока отбрасываются, и передается код конца блока.

Кодирование I-блоков. Процедура кодирования I-блоков похожа на методику кодирования неподвижных блоков стандарта JPEG. Однако в отличие от JPEG уровень квантования может быть переменным, и коэффициент квантования подбирается кодером.

Кодирование P-блоков. Процедура кодирования P-блоков гораздо сложнее процедуры кодирования I-блоков. Кодер должен выбрать метод кодирования макроблока. В этом случае также стандарт не предписывает какого-либо алгоритма, оставляя это на усмотрение разработчиков кодера. В описанных реализациях [5.51, 5.52] данный алгоритм осуществляется в несколько стадий.

1. Принимается решение, следует ли использовать компенсацию движения (MC/no MC decision), так как в случае отсутствия движения используется разность между текущим макроблоком и его несмещенным базовым макроблоком (формально это можно считать нулевым вектором движения). Тем самым достигается экономия за счет того, что вектор движения не передается. Для этого вычисляются сумма несмещенной разности между текущим макроблоком и его

базовым макроблоком (то есть с нулевым вектором движения):

$$D_1 = \sum |X(i, j) - X'(i, j)|/256, \quad (14.3)$$

где $X(i, j)$ — значения яркостных пикселов в текущем кадре, $X'(i, j)$ — значения яркостных пикселов в опорном кадре, суммирование проводится по всем яркостным пикселям макроблока, и та же сумма — для разности с вектором движения (n, m) :

$$D_2 = \sum |X(i, j) - X'(i + n, j + m)|/256. \quad (14.4)$$

На основе ряда численных экспериментов была получена эмпирическая кривая выбора решения (рис. 14.2), которую в силу ее простоты и малых численных затрат часто используют разработчики коммерческих реализаций кодера [5.51]. Границные точки на кривой считаются относящимися к области отсутствия компенсации.

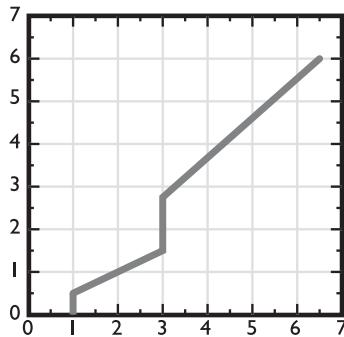


Рис. 14.2. Кривая выбора решения о компенсации движения

Кривая имеет сложную форму в области небольших значений разностей, так как любое ложное движение фона, вызванное медленным движением перемещающихся объектов, является крайне нежелательным эффектом, заметно ухудшающим визуальное качество изображения.

В Р-блоках вектор движения передается с помощью разностного кодирования, что обеспечивает значительную экономию для изображений с движением, вызванным перемещением камеры, в которых векторы движения для большинства макроблоков будут примерно одинаковыми.

2. После получения информации об оптимальном варианте компенсации движения кодер решает, следует ли ее использовать (т. е. использовать

нулевой или ненулевой вектор движения и кодировать разностный сигнал) или кодировать исходный макроблок как I-блок (INTRA/NON-INTRA mode decision). Это можно было бы сделать, сравнив количество битов, необходимое для передачи кодированной информации о макроблоке с компенсацией движения и без нее при одинаковом коэффициенте квантования. Однако ввиду значительных вычислительных затрат на эту процедуру на основе численных экспериментов была получена эмпирическая кривая выбора решения о внутрикадровом или межкадровом кодировании на основе сравнения дисперсий текущего макроблока и разностного сигнала (рис. 14.3) [5.52]. Границные точки на кривой считаются относящимися к области межкадрового предсказания.

3. Дисперсия V_1 для текущего макроблока вычисляется по следующей формуле:

$$V_1 = \sum X^2(i, j)/256^2 - \left(\sum X(i, j)/256 \right)^2, \quad (14.5)$$

а дисперсия V_2 — для разностного сигнала, полученного с учетом принятого решения о компенсации движения, т. е. с использованием вектора движения (n, m) , нулевого или ненулевого, по следующей формуле:

$$V_2 = \sum (X(i, j) - X'(i + n, j + m))^2/256. \quad (14.6)$$

Исходя из рис. 14.3, внутрикадровое кодирование имеет смысл выполнять для макроблоков с относительно меньшей дисперсией и относительно большой среднеквадратичной ошибкой предсказания.

Для устранения блокинг-эффекта, связанного с компенсацией движения, может производиться фильтрация. Спецификация фильтра установлена стандартом. Фильтрация осуществляется для каждого блока в макроблоке (как яркостных, так и цветоразностных блоков), для которого выбран режим с фильтрацией. Как показали эксперименты, фильтрация ошибок в кодере после компенсации движения дает лучшие результаты, чем постфильтрация в декодере [5.51].

4. После квантования коэффициентов ДКП некоторые блоки имеют все нулевые коэффициенты, и данный блок не требуется кодировать (coded/not coded decision). В заголовке макроблока записывается информация о том, какие именно блоки подлежат кодированию (CBP, coded block pattern).

5. В заключение принимается решение, следует ли изменять коэффициент квантования, установленный по умолчанию (quant/no quant decision). Визуальное восприятие изображения будет улучшаться, если применять переменный коэффициент квантования в зависимости от детальности в блоке и свойств зрительного анализатора при их восприятии. С другой стороны, уровень квантования должен выбираться исходя из требования обеспечения равномерности битового потока. Хотя данный вопрос является одним из наиболее существенных для улучшения визуального восприятия восстановленного изображения, его однозначное решение до сих пор не найдено. Предлагаемые методы, основанные на оценке дисперсии и активности блока, величины нулевого коэффициента ДКП и т. п., не учитывают особенности субъективного восприятия глазом, который выделяет наиболее значимые детали на фоне всего остального изображения.

Рекомендация H.261 также предусматривает специальный режим передачи статических изображений повышенного разрешения (в два раза выше по горизонтали и по вертикали, чем передаваемые кадры видеопоследовательности).

14.2. H.263. Видеокодирование для систем связи с низкой пропускной способностью

Рекомендация ITU-T H.263 определяет методику кодирования и декодирования движущихся изображений для систем передачи аудиовизуальной информации по каналам с низкой пропускной способностью [5.42]. Алгоритм кодирования аналогичен алгоритму из Рекомендации ITU-T H.261 и является гибридным. Он использует межкадровое предсказание для устранения временной избыточности и кодирование с преобразованием остатков предсказания для устранения

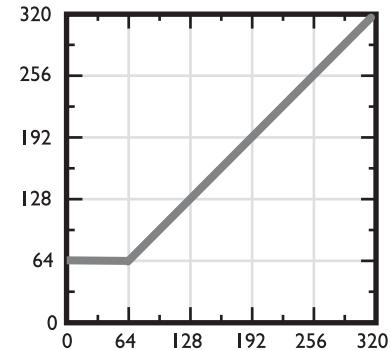


Рис. 14.3. Кривая выбора решения о внутри или межкадровом кодировании

пространственной избыточности видеосигнала. Кодер может обрабатывать кадры пяти стандартных размеров: subQCIF, QCIF, CIF, 4CIF, 16CIF, предсказание движения может осуществляться с точностью до половины пикселя (в отличие от кодера, соответствующего Рекомендации H.261, который поддерживает только два размера кадров, QCIF и CIF, и предсказание движения с точностью до пикселя). Кодек, соответствующий Рекомендации H.263, может также поддерживать ряд дополнительных возможностей, улучшающих кодирование, в том числе: неограниченные векторы движения, арифметическое кодирование, улучшенное предсказание, PB-кадры. Эти дополнительные возможности могут использоваться отдельно или вместе.

Режим с неограниченными векторами движения позволяет векторам движения указывать на области вне границ кадра. При этом отсутствующие пиксели заменяются граничными. Такой режим дает заметное повышение эффективности кодирования при движении объектов через границы кадра, особенно для малых форматов кадра.

Режим арифметического кодирования вместо кодирования кодами переменной длины позволяет сократить выходной поток данных при том же качестве восстанавливаемых изображений.

В режиме улучшенного предсказания используется метод компенсации движения с перекрывающимися блоками для составляющей яркости предсказываемых кадров. Для части макроблоков могут использоваться четыре векторы движения — по одному на каждый блок яркости. Четыре вектора обеспечивают лучшее предсказание, но занимают больше места в выходном потоке, и, следовательно, кодер должен определять целесообразность использования четырех векторов для каждого конкретного макроблока. Использование этого режима дает существенное повышение качества кодирования, особенно учитывая то, что компенсация движения с перекрытием блоков приводит к уменьшению блокинг-эффекта и других артефактов кодирования.

В режиме PB-кадров вместе с P-кадром может передаваться B-кадр — кадр, содержащий макроблоки с двунаправленным предсказанием (относительно предыдущего и текущего P-кадров). В этом режиме кадровая скорость может быть существенно увеличена при незначительном росте скорости выходного битового потока. Размер закодированного кадра не должен превышать заданной в Рекомендации величины, которая зависит от размеров кадра.

Опорная частота кадров, номинальные значения яркости и цветности, формат представления цветности такие же, как в Рекомендации H.261. В дополнение к двум стандартным размерам кодируемых кадров, описанным в Рекомендации H.261 (QCIF и CIF), добавлены еще три размера кадра — subQCIF (128×96 пикселов яркости и 64×48 пикселов цветоразностных компонент), 4CIF (704×576 и 352×288) и 16CIF (1408×1152 и 704×576).

Соотношение размеров сторон пикселя одинаково для всех форматов кадра и составляет, как и в Рекомендации H.261, $\frac{4}{3} \times \frac{288}{352}$. Соотношение размеров сторон кадра составляет 4:3 для всех форматов, кроме subQCIF.

Все декодеры должны иметь возможность декодировать кадры формата subQCIF и QCIF, остальные форматы являются необязательными. Кодер должен иметь возможность обрабатывать кадры одного из вышеперечисленных обязательных размеров.

Алгоритм кодирования в Рекомендации H.263 аналогичен алгоритму, описанному в Рекомендации H.261, однако имеются некоторые отличия.

Группы блоков состоят из $k \times 16$ строк, где k зависит от формата кадра: 1 — для subQCIF, QCIF и CIF, 2 — для 4CIF, 4 — для 16CIF. То есть группа блоков состоит из одного, двух или четырех рядов макроблоков. Количество групп блоков в кадре составляет 6 — для формата subQCIF, 9 — для QCIF, 18 — для CIF, 4CIF и 16CIF. Данные для группы блоков состоят из заголовка, который может быть пустым, и данных макроблоков, составляющих группу блоков. Кадр может быть закодирован и без разбиения на группы блоков.

Компенсация движения производится с точностью полпикселя, возможна передача одного или четырех векторов движения для макроблока, возможна передача уточнения вектора движения для макроблока B-составляющей PB-кадров. Стандартный диапазон векторов движения — $[-16; +15, 5]$, в режиме неограниченных векторов движения полный диапазон изменения векторов — $[-31, 5; +31, 5]$, но значение вектора движения не может отличаться от предсказателя на величину вне диапазона $[-16; +15, 5]$. В случае использования режима неограниченных векторов движения и/или улучшенного предсказания вектор движения может указывать за границы кодируемой части кадра.

Процедура квантования не отличается от описанной в Рекомендации H.261. В отличие от H.261 стартовые коды в потоке H.263 должны быть выровнены по границе байта.

Иерархическая структура потока H.263 аналогична структуре потока H.261, но заголовки и данные различных уровней несколько отличаются.

Уровень кадра состоит из заголовка кадра, данных групп блоков и, возможно, кода окончания видеопоследовательности и дополнительных битов для выравнивания по границе байта следующего стартового кода. Заголовок кадра содержит: стартовый код (22 бита); временную метку (8 битов); информацию о типе кадра (13 битов), содержащую размер кадра, тип кадра (I, P, PB), флаги использования дополнительных режимов кодирования; код коэффициента квантования (5 битов); флаг включения многоточечного режима (1 бит); номер потока при включенном многоточечном режиме (2 бита); временную метку B-составляющей для PB-кадра (3 бита); код уточнения коэффициента квантования B-составляющей для PB-кадра (2 бита); флаг дополнительной информации (1 бит) и, возможно, дополнительную информацию (0/8/16... битов).

Уровень группы блоков (GOB) состоит из заголовка (возможно, пустого) и данных макроблоков. Заголовок группы блоков содержит: биты выравнивания по границе байта (от 0 до 7 битов); стартовый код (17 битов); номер группы (5 битов); номер потока при включенном многоточечном режиме (2 бита); идентификатор кадра, одинаковый для всех групп блоков в кадре (2 бита); код коэффициента квантования (5 битов).

Уровень макроблока состоит из заголовка макроблока и данных блоков. Заголовок макроблока содержит: индикатор кодирования макроблока (1 бит, если он равен 1, то информация о данном макроблоке не передается); код типа макроблока и трафарета присутствия кодов блоков цветности (от 1 до 9 битов); код типа B-блоков для PB-кадров (1–2 бита); трафарет присутствия кодов B-блоков для PB-кадров (6 битов); код трафарета присутствия кодов блоков яркости (2–6 битов); код уточнения коэффициента квантования (2 бита); коды горизонтальной

и вертикальной компонент вектора движения для макроблоков с предсказанием движения (1–13 битов); коды горизонтальной и вертикальной компонент второго, третьего и четвертого векторов движения для макроблоков с четырьмя векторами движения (1–13 битов); коды горизонтальной и вертикальной компонент вектора движения В-макроблока РВ-кадра (1–13 битов).

Уровень блока содержит закодированную информацию о блоках. В макроблоках кадров I- и P-типа содержится по 6 блоков (4 блока яркости и два цветоразностных), а в РВ-кадрах — 12 (еще 6 блоков В-составляющей кадра). Нулевой коэффициент I-макроблока присутствует в потоке всегда, наличие кодов остальных коэффициентов блоков указывается в заголовке макроблока в информации о типе и/или в трафаретах присутствия блоков. Перед кодированием коэффициенты блока располагаются в Z-порядке (см. описание Рекомендации H.261). Нулевой коэффициент I-макроблока кодируется кодом постоянной длины 8 битов, остальные коэффициенты кодируются кодами переменной/постоянной длины следующим образом: каждое «событие» заменяется кодом; «событие» состоит из длины цепочки нулей (run), значения ненулевого коэффициента после цепочки нулей (level) и признака окончания блока (цепочка нулей, которой оканчивается блок, отбрасывается). Наиболее часто встречающиеся «события» кодируются кодами переменной длины, остальные — 22-битным кодом, состоящим из 7 битов признака кодирования кодами постоянной длины (ESCAPE-код), 1 бита признака окончания блока (last), 6 битов run и 8 битов level.

Векторы движения макроблоков в стандартном режиме кодирования передаются в дифференциальном виде с использованием предсказания. Предсказатель вектора движения определяется по медианному значению векторов трех соседних макроблоков (рис. 14.4).

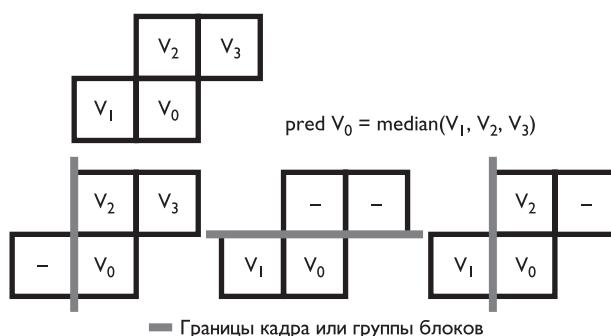


Рис. 14.4. Алгоритм предсказания вектора движения макроблоков

Векторы движения для цветоразностных компонент макроблоков в два раза меньше, чем для яркостных компонент. Для получения предсказания с точностью до полпикселя используется линейная интерполяция промежуточных значений пикселов в опорном кадре.

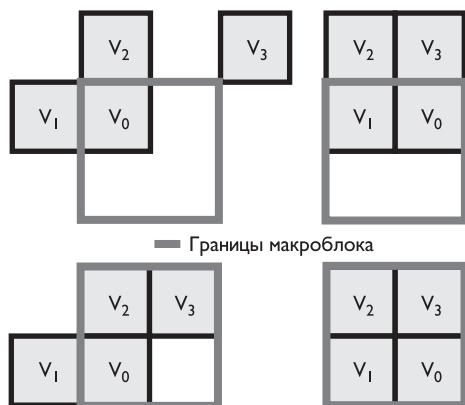
Рассмотрим подробнее дополнительные режимы, повышающие эффективность кодирования. Режим неограниченных векторов движения включает два расширения стандарта. Первое — векторы движения могут указывать за границы кодируемой части кадра. При этом отсутствующие в опорном кадре пиксели заменяются

на значения на границе. Второе — увеличение диапазона возможных значений вектора движения.

Диапазон $[-16; 15, 5]$ расширяется до $[-31, 5; 31, 5]$ при некотором ограничении: если предсказатель вектора лежит в диапазоне $[-16; 15, 5]$, то значение вектора отличается от предсказателя не более чем на $[-16; 15, 5]$.

Стандарт предусматривает уточнение векторов движения четырех блоков каждого макроблока (рис. 14.5).

Рис. 14.5. Алгоритм предсказания векторов движения четырех блоков



В режиме арифметического кодирования все процедуры кодирования кодами переменной длины заменяются на неадаптивное арифметическое кодирование нулевого порядка. Таблицы вероятностей для инициализации процессов кодирования/декодирования стандартизованы. При этом немного меняется синтаксис уровня блока — особым образом кодируются первые три «события» (last-run-level).

Режим улучшенного предсказания включает компенсацию движения с перекрытием макроблоков и использование четырех векторов движения на макроблок. В этом режиме также возможен выход вектора движения за границы кадра, но расширенный диапазон векторов движения возможен лишь при совместном использовании этого режима и режима неограниченных векторов движения. Предсказание движения с перекрытием макроблоков используется только для P-макроблоков в P- и PB-кадрах. При использовании четырех векторов движения на макроблок (по одному для каждого блока яркости) вектор движения для цветоразностных компонент получается как среднее арифметическое этих четырех векторов.

В режиме PB-кадров возможна передача кадров нового типа — PB-кадров (рис. 14.6). Такой кадр состоит из двух кадров — P-кадра и кадра с двунаправленным предсказанием, опорными для которого являются предыдущий P- или I-кадр (предсказание вперед) и текущий P-кадр (предсказание назад). В таких кадрах возможно использование I-макроблоков, но при этом внутрикодируемым является только макроблок P-кадра, а макроблок B-кадра кодируется с предсказанием. В этом случае для I-макроблоков передается вектор (вектора) движения, который относится только к B-кадру. Макроблок в PB-кадре состоит из 12 блоков — 6 блоков P-кадра и 6 блоков B-кадра.

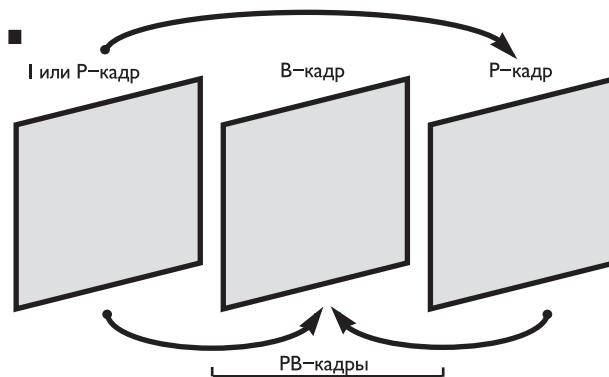


Рис. 14.6. PB-кадры в стандарте H.263

В кодере/декодере, соответствующем Рекомендации H.263, возможно также использование кодов восстановления ошибок, аналогичных приведенным в Рекомендации H.261.

В 1998 году была выпущена вторая версия Рекомендации ITU-R H.263 (известная как H.263+); третья версия этой Рекомендации разработана в 2000 году (ее зачастую обозначают H.263++ или H.263 2000) [5.43, 5.54]. В этих версиях добавлено множество новых дополнительных режимов, повышающих эффективность кодирования и увеличивающих количество возможных областей применения стандарта.

Новые дополнительные режимы можно разделить на несколько групп: использование новых типов кадров; использование новых режимов кодирования для этих кадров; определение совместимой с первой версией Рекомендации дополнительной информации, которая может быть добавлена к видеопотоку.

Новые типы кадров

Масштабируемое кодирование видео имеет потенциальную возможность улучшить передачу видео в сетях с возможной потерей данных. Оно позволяет разделить битовый поток на несколько логических каналов так, чтобы часть данных могла быть потеряна или опущена при сохранении возможности декодирования и воспроизведения. Введено три типа масштабируемости — один, обеспечивающий временную масштабируемость, и два, обеспечивающие пространственную масштабируемость или масштабируемость по отношению сигнал/шум: B-кадр, кодируемый с двунаправленным предсказанием относительно предшествующего и последующего по времени кадров; EI-кадр — кадр, имеющий в качестве опорного кадр с той же временной меткой; EP-кадр — кадр, имеющий в качестве опорных предшествующий по времени кадр и кадр с той же временной меткой.

Вводится новый тип улучшенных PB-кадров, которые более удобны и надежны при непрерывном кодировании, чем стандартные PB-кадры.

Вводится возможность использования, в дополнение к пяти стандартным размерам видеокадра, кадра с произвольными размерами по горизонтали и вертикали.

Новые режимы кодирования

Вторая версия Рекомендации вводит девять новых режимов кодирования. Улучшенное внутрикадровое кодирование, которое увеличивает эффективность кодирования I-макроблоков с помощью пространственного предсказания значений коэффициентов ДКП.

Использование деблокирующего фильтра позволяет снизить блокинг-эффект за счет фильтрации по границам блоков.

Использование слайсовой структуры кадра позволяет функционально группировать макроблоки кадра, что повышает устойчивость к ошибкам, улучшает передачу по пакетным сетям и уменьшает задержку.

Использование режима выбора опорного кадра увеличивает устойчивость к ошибкам в канале передачи, так как позволяет использовать в качестве опорного не только непосредственно предшествующий кадр.

Режим передискретизации опорного кадра позволяет передискретизировать предшествующий по времени кадр до его использования при кодировании. Это дает возможность использовать глобальную компенсацию движения, динамическое изменение пространственного разрешения, специальные эффекты.

Режим уточнения с уменьшенным разрешением позволяет поддерживать высокую кадровую скорость кодирования при интенсивном движении. В этом режиме кадр высокого разрешения уточняется с низким разрешением, что позволяет сохранять высокое разрешение в неподвижных областях изображения.

Режим независимого декодирования сегментов повышает устойчивость декодирования при ошибках в канале передачи за счет того, что испорченные данные в одной области не влияют на процесс декодирования других областей.

Режим альтернативного кодирования кодами переменной длины макроблоков с предсказанием уменьшает количество выходных битов при кодировании P-блоков с большим количеством ненулевых коэффициентов.

Режим модифицированного квантования позволяет использовать произвольно изменяющиеся от макроблока к макроблоку коэффициенты квантования, что дает возможность более точной регулировки скорости выходного потока и качества кодирования, уменьшает дефекты кодирования цветности за счет уменьшения глубины квантования цветности относительно яркости, увеличивает диапазон возможных значений коэффициентов ДКП при малых коэффициентах квантования, повышает вероятность обнаружения ошибок и упрощает декодирование за счет запрета определенных значений коэффициентов ДКП.

Дополнительная информация

Новая версия Рекомендации также позволяет добавлять сопроводительную информацию к видеопотоку. Хотя эта информация не влияет на семантику декодирования битового потока, она может повысить функциональность систем, распознающих эту информацию, не влияя на работу систем, ее не воспринимающих. Эта информация позволяет использовать команды остановки и продолжения воспроизведения, выдавать дополнительную информацию синхронно с кадрами, использовать режимы прозрачности видео.

14.3. MPEG-1 — кодирование видео- и аудиоинформации для цифровых систем с объемом данных приблизительно до 1,5 Мбитов/с

MPEG-1 стандартизует запись динамических изображений и сопровождающего звука на компакт-диски (CD-ROM, VideoCD/DVD) и передачу видеоизображений пониженной четкости по сравнительно низкоскоростным каналам связи (скорость цифрового потока до 1,5–3 Мбита/с) [5.29–5.32].

Стандарт определяет принципы обработки информации и структуру цифрового потока. Упрощенная функциональная схема декодера приведена на рис. 14.7.

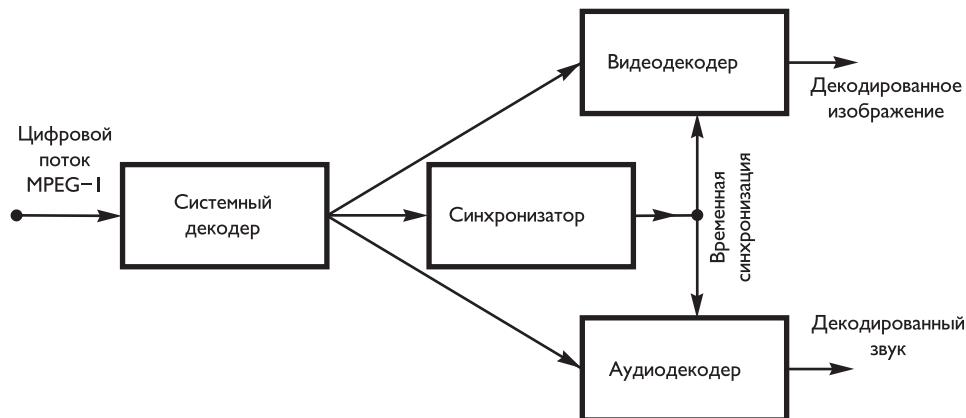


Рис. 14.7. Упрощенная функциональная схема декодера MPEG-1

Далее описаны принципы обработки видеоинформации. Обычно используются два формата динамических изображений: SIF (352×240 пикселов) при 30 кадрах в секунду и CIF (352×288 пикселов) при 25 кадрах в секунду. MPEG-1 предполагает использование цветовой схемы $YC_R C_B$ с разрешением $4 : 2 : 0$, при котором на каждые 4 пикселя сигнала Y приходится по одному пикселу цветоразностных сигналов C_R и C_B .

На рис. 14.8 приведены упрощенные структуры кодирующего и декодирующего устройств MPEG-1.

Несмотря на то что MPEG-1 применяется для кодирования изображений с пониженным разрешением при относительно малом цифровом потоке, стандарт позволяет использовать любое разрешение вплоть до 4095×4095 . Самым крупным недостатком видеокодера MPEG-1 является поддержка только прогрессивной развертки.

Синтаксис кодированного битового потока содержит шесть уровней:

- уровень кадров видеопоследовательности,
- уровень группы кадров,
- уровень кадров,
- уровень слайсов,

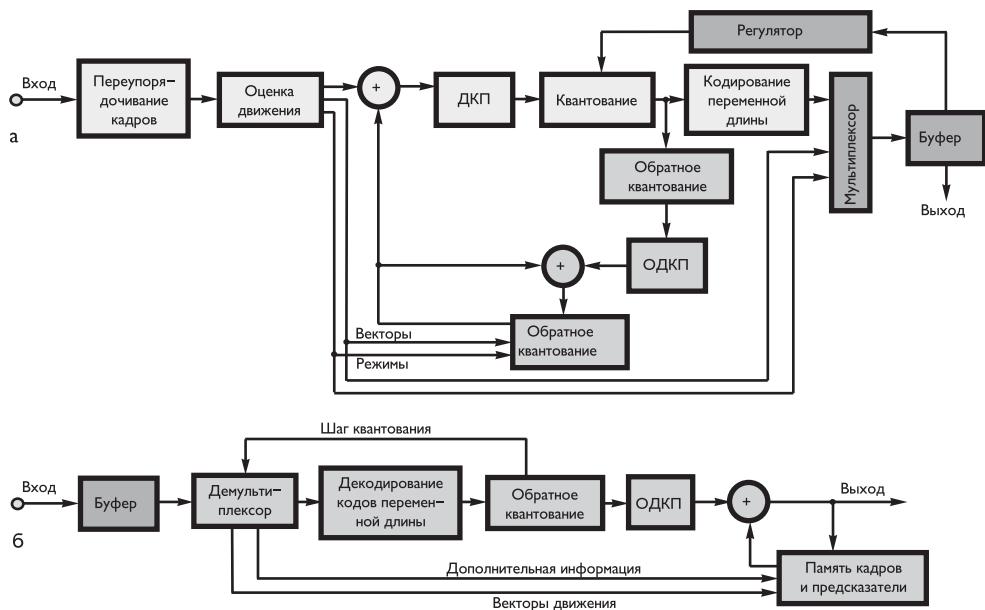


Рис. 14.8. Структуры видеокодера (а) и видеодекодера (б) стандарта MPEG-1

- уровень макроблоков,
- уровень блоков.

Первые четыре уровня имеют сходную структуру. Каждому из них предшествует заголовок.

Рис. 14.9 иллюстрирует возможную комбинацию главных типов составляющих видеопоследовательности, которые используются в стандарте.

Видеопоследовательность разбивается на группы кадров, обработка которых осуществляется совместно. Перед началом кодирования производится анализ видеинформации, в выбранной группе кадров определяются ключевые кадры, которые должны подвергаться только внутрикадровой обработке, а также кадры, при кодировании которых часть информации будет удаляться за счет межкадровой обработки. Всего выделяется три типа кадров:

- I-кадры (Intra frame), подвергающиеся только внутрикадровому кодированию;
- P-кадры (Predirected frame), в которых осуществляется односторонний анализ движения деталей; при кодировании этих кадров используется информация из предшествующего I- или P-кадра;
- B-кадры (Bidirectional predicted frame), в которых осуществляется двунаправленный анализ движения деталей; при кодировании этих кадров используется информация из предшествующего I- или P-кадра и последующего I- или P-кадра.

Наличие B-кадров в закодированной информации позволяет существенно повысить коэффициент сжатия в стандарте MPEG-1.

Каждый кадр видеопоследовательности разбивается на макроблоки, в составе которых имеются 4 блока сигнала яркости и по одному блоку двух цветоразност-

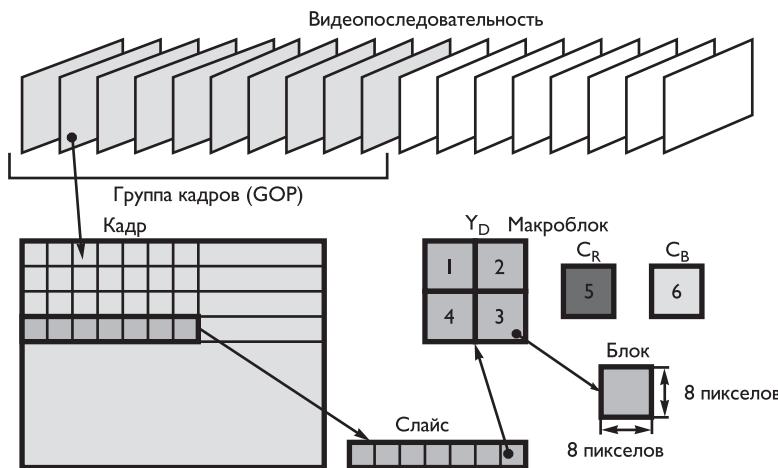


Рис. 14.9. Пример последовательности составляющих динамических изображений

ных сигналов. Размер каждого квадратного блока равен 8×8 пикселов, а размер макроблока сигнала яркости — 16×16 пикселов.

Слайс — группа следующих друг за другом макроблоков. Число макроблоков в слайсе может быть произвольным, но слайсы в изображении не должны перекрываться. В заголовке слайса задается параметр квантования макроблоков, но этот параметр может быть изменен на уровне марблока.

Все структурные элементы потока видеоданных, полученные в результате внутрикадрового и межкадрового кодирования (кроме блока и макроблока), дополняются заголовками, которые начинаются со специальных стартовых кодов. В заголовке приводится разнообразная дополнительная информация, например размеры и соотношение сторон изображения, частота кадров, тип кодирования, скорости потока, матрица квантования, формат дискретизации цветного изображения, координаты основных цветов и белого цвета, параметры матрицы для формирования яркости и цветоразностных сигналов и др.

Последовательности I-, P-, B-кадров объединяются в группы кадров (GOP — Group of Pictures), которые обычно фиксированы по длине и структуре. Каждая такая группа начинается с I-кадра и с определенной периодичностью содержит P-кадры. Для кадров разных типов реализуется различный уровень компрессии. Наименьшей степенью сжатия обладают I-кадры. P-кадр обычно составляет по размеру примерно третью часть от I-кадра, а B-кадр — восьмую часть. Поэтому суммарный поток данных в значительной степени зависит от состава GOP. В зависимости от назначения и требуемого качества записи (видеофильм, мультимедиапродукция, фильм для демонстрации в Интернете и т. д.) используется различный состав и длительность GOP.

Типичная группа кадров приведена на рис. 14.10 и имеет следующий вид: IBBPBBPBB, пронумерованная от 0 до 8. Следующий I-кадр обозначен числом 9.

MPEG-1 позволяет значительно уменьшить суммарный поток данных. Так, при обработке группы из девяти кадров, состоящей из одного I-кадра, двух P-кадров и шести B-кадров, цифровой битовый поток приблизительно составит такую же величину, как при внутрикадровом кодировании 2,5 I-кадров.

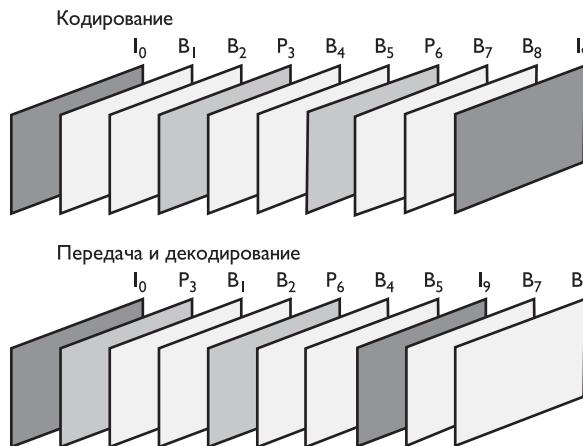


Рис. 14.10. Типичная группа кадров при кодировании и декодировании

Поскольку B-кадр нельзя декодировать, не приняв последующего P- или I-кадра, кадры декодируются не в линейной последовательности, что также показано на рис. 14.10.

Анализ движения деталей в P- и B-кадрах осуществляется путем попиксельного сравнения макроблоков с точностью до половины пикселя. При этом в P- или I-кадрах могут присутствовать макроблоки без компенсации движения (для которых эквивалентные макроблоки в других кадрах не найдены), макроблоки с компенсацией движения (вместо которых передаются только разностные составляющие) и пропущенные макроблоки (разностные сигналы для которых практически равны нулю).

Сокращение избыточности видеоинформации в стандарте MPEG-1 осуществляется практически так же, как в стандартах JPEG и H.261 [5.55, 5.56]:

- межпиксельная корреляция в блоках 8×8 пикселов существенно уменьшается с использованием дискретного косинусного преобразования (ДКП);
- квантование коэффициентов ДКП учитывает особенности человеческого зрения — невосприимчивость к высокочастотным составляющим;
- предсказание первого низкочастотного коэффициента преобразования в блоке 8×8 (среднее значение яркости блока) учитывает большую пространственную корреляцию кадра изображения в целом;
- использование Z-упорядочивания квантованных коэффициентов ДКП, кодирование повторяющихся нулевых элементов с обозначением конца блока;
- учет структуры участков изображения, пространственное маскирование осуществляются путем изменения степени квантования макроблоков и слайсов;
- использование «прямых» и «обратных» векторов движения макроблоков 16×16 пикселов существенно сокращает временную избыточность;
- применение адаптивного квантования при кодировании ошибок компенсации движения;
- предсказание векторов движения при медленном движении деталей изображения со сложным рисунком;

- индикация пропущенных макроблоков, если вектор движения и ошибка предсказания нулевые;
- устранение статистической избыточности за счет применения RLE-алгоритма и хuffmanовского кодирования.

14.4. MPEG-2. Кодирование динамических изображений и связанной с ними звуковой информации

Формат MPEG-2 был создан путем эволюционной переделки стандарта MPEG-1 под нужды заказчиков. А заказчиками данного формата являлись крупнейшие массмедиа-компании, которые сделали ставку на спутниковое телевидение и нелинейный цифровой видеомонтаж.

Стандарт был разработан для удовлетворения растущих потребностей в методах кодирования движущихся изображений и связанного с ними звука, а также других сопутствующих данных, для различных приложений, таких как цифровые устройства хранения, телевещание и связь. Использование этих стандартов для кодирования видеинформации означает, что движущиеся изображения можно обрабатывать как компьютерные данные и хранить их в различных устройствах хранения, передавать и получать их по ныне существующим и будущим сетям и распространять их по существующим и будущим каналам вещания.

Ниже перечислены приложения, которые охватываются стандартом (разумеется, приведенный перечень далеко не полон):

- кабельное ТВ, распространяемое по оптоволоконным и другим сетям;
- цифровое наземное и спутниковое телевизионное вещание (включая ТВЧ);
- кабельное цифровое распространение звука;
- цифровое звуковое вещание (наземное и спутниковое);
- электронное кино;
- электронный сбор новостей (включая спутниковые системы);
- домашний телевизионный театр;
- общение между людьми (видеоконференции, видеотелефон и др.);
- интерактивные устройства хранения (оптические диски и др.);
- новости и текущие дела;
- сервисные сетевые базы данных;
- дистанционный просмотр видео;
- цифровые видеосерверы;
- последовательные устройства хранения (цифровые видеомагнитофоны и др.).

При создании стандарта были рассмотрены требования различных типичных приложений, развиты и собраны в единый синтаксис различные необходимые алгоритмические элементы.

Основной задачей разработки стандарта MPEG-2 было улучшение алгоритмов компрессии видео и аудио, а также определение путей уменьшения скорости мультиплексированного выходного потока.



Важные изменения коснулись алгоритмов квантования, при котором стало возможным в процессе кодирования задавать точность частотных коэффициентов матрицы квантования, что непосредственно влияет на качество получаемого в результате сжатия изображения (и на размер тоже). Используя MPEG-2, пользователь может задавать следующие значения точности квантования — 8, 9, 10 и 11 битов на одно значение элемента, что делает этот формат значительно более гибким по сравнению с MPEG-1, в котором было только одно фиксированное значение — 8 битов на элемент.

Появилась возможность загрузить отдельную матрицу квантования непосредственно перед каждым кадром, что позволяет при использовании этой операции добиться более высокого качества изображения.

Дополнены и алгоритмы предсказания движения, в частности возможна обработка телевизионных изображений по полям. Основной размер блоков, на которые разбивается изображение при анализе движения, может быть 16×16 точек, как и в MPEG-1, 8×8 и 16×8 .

Из-за некоторых особенностей реализации алгоритмов предсказания движения в MPEG-2 появились некоторые ограничения на размер изображения. Стало необходимо, чтобы разрешение изображения по вертикали и горизонтали было кратно 16 в режиме покадрового кодирования, и 32 по вертикали в режиме кодирования полей.

Помимо вышеперечисленных изменений в формат MPEG-2 были введены еще несколько новых, нигде ранее не использовавшихся алгоритмов компрессии видеоданных.

Стандарт MPEG-2 призван облегчить обмен битовыми потоками между различными приложениями. Он поддерживает постоянную и переменную скорости передачи, произвольный доступ, переключение каналов, масштабируемое декодирование, редактирование битового потока, а также такие специальные функции, как быстрое воспроизведение, быстрое обратное воспроизведение, обратное воспроизведение с нормальной скоростью, медленное движение, пауза и неподвижные изображения.

14.4.1. Общие положения

Стандарт MPEG-2 состоит из трех основных частей: системной, видео и звуковой. Видеочасть стандарта описывает кодированный битовый поток для высококачественного цифрового видео. MPEG-2 является совместимым расширением MPEG-1, он поддерживает чересстрочный формат видео- и ряд развитых средств, в том числе средства для поддержки ТВЧ.

Стандарт MPEG-2 определяется в терминах расширяемых профилей, каждый из которых, являясь частным случаем стандарта, поддерживает черты, необходимые важным классам приложений.

Иерархические масштабируемые профили могут поддерживать такие приложения, как совместимое наземное многопрограммное ТВ/ТВЧ, пакетные сетевые видеосистемы, обратную совместимость с другими существующими стандартами (MPEG-1), и другие приложения, использующие многоуровневое кодирование.

Такая система сможет позволить потребителю использовать или маленький переносной приемник для декодирования стандартного ТВ, или большой стационарный приемник для декодирования ТВЧ из того же вещательного сигнала.

Звуковая часть стандарта MPEG-2 определяет низкоскоростное кодирование многоканального звука. MPEG-2 поддерживает до 5 полных широкополосных каналов плюс дополнительный низкочастотный канал и/или до семи многоязычныхcommentatorских каналов. Он также расширяет возможности кодирования mono- и стереозвуковых сигналов в MPEG-1.

Звуковой стандарт многоканального кодирования обеспечивает совместимость с MPEG-1. В нем рассматриваются также до трех несовместимых кодеков.

Системная часть стандарта MPEG-2 описывает форматы мультиплексирования звуковой, видео и другой информации, рассматривает вопросы комбинирования одного или более потоков данных в один или множество потоков, пригодных для хранения или передачи.

Системное кодирование в соответствии с синтаксическими и семантическими правилами, налагаемыми данным стандартом, обеспечивает необходимую и достаточную информацию, чтобы синхронизовать декодирование без переполнения или недополнения буферов декодера при различных условиях приема или восстановления потоков.

Таким образом, системный уровень выполняет пять основных функций:

- синхронизацию нескольких сжатых потоков при воспроизведении;
- объединение нескольких сжатых потоков в единый поток;
- инициализацию буферизации для начала воспроизведения;
- дальнейшее обслуживание буфера;
- определение временной шкалы.

Мультиплексированный поток MPEG-2 — это или программный, или транспортный поток. Оба типа потоков конструируются из пакетов пакетированных элементарных потоков (ПЭП) и пакетов, содержащих другую необходимую информацию. Оба типа потоков поддерживают мультиплексирование сжатых видео- и звуковых потоков с общей временной базой, а транспортный поток, кроме того, поддерживает мультиплексирование сжатых видео- и звуковых потоков с независимыми временными базами.

Для систем передачи и хранения с отсутствием ошибок обычно более подходит программный поток. Транспортный поток более удобен для использования в средах, где вероятны ошибки.

Мультиплексированный битовый поток MPEG-2 (или программный, или транспортный) конструируется из двух слоев:

- внешнего слоя — системного слоя и
- внутреннего слоя — слоя компрессии.

Системный слой обеспечивает функции, необходимые для использования одного или более сжатых потоков данных в системе. Видео- и звуковая части стандарта MPEG-2 определяют слой компрессии для видео- и звуковых данных.

Кодирование других видов данных не определяется стандартом, но поддерживается системным слоем при условии, что другие типы данных удовлетворяют ограничениям, описанным в системной части стандарта.

Системное кодирование осуществляется в двух формах: программный поток и транспортный поток. Каждый из них оптимизирован для различных классов приложений.

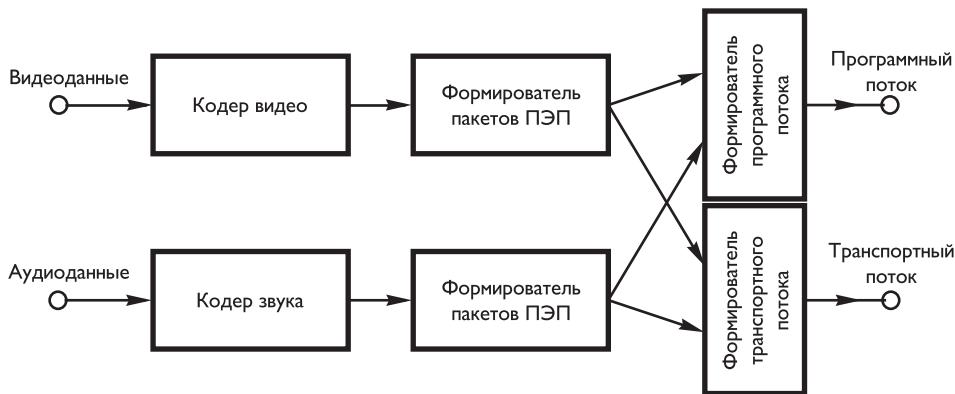


Рис. 14.11. Мультиплексирование видео- и звукового элементарных потоков

И программный, и транспортный потоки обеспечивают системный уровень кодирования, который необходим и достаточен для синхронизации декодирования и представления видео- и звуковой информации, причем буферы декодеров не будут переполняться или недополняться.

Эта информация кодируется с использованием временных меток, относящихся к кодированию и представлению видео- и звуковых данных, и временных меток, относящихся к доставке самого потока данных. Оба потока определяются как пакетно-ориентированные, мультиплексированные.

Базовый метод мультиплексирования одного видео и одного звукового элементарных потоков проиллюстрирован на рис. 14.11.

После кодеров результирующие сжатые элементарные потоки дополняются информацией системного уровня, пакетируются, образуя пакетированные элементарные потоки (ПЭП). Эти потоки могут быть затем объединены в программный или транспортный поток.

Программный поток аналогичен системному потоку MPEG-1 и сочетает один или несколько элементарных потоков с общей временной базой в едином потоке. Определение программного потока позволяет также кодировать несколько видео- и звуковых элементарных потоков в несколько программных потоков, которые имеют общую временную базу и которые, как и одиничный программный поток, могут быть декодированы с синхронизацией между различными элементарными потоками.

Программный поток предназначен для использования в условиях отсутствия ошибок и пригоден для приложений, которые могут включать программную обработку потоков. Пакеты программного потока могут иметь различную и относительно большую длину.

Транспортный поток комбинирует один или более элементарных потоков с одной или более временной базой в один поток. Элементарные потоки используют общую временную базу программы. Транспортный поток предназначен для использования в условиях вероятных ошибок, таких как хранение или передача в среде с шумом или с потерями. Пакеты транспортного потока имеют длину 188 байт.

Так как программный и транспортный потоки предназначены для различных приложений, их определения не следуют строго уровневой модели.

Возможно и разумно преобразовывать потоки из одного вида в другой, однако один из них не является подмножеством или надмножеством другого. В частности, извлечение содержания программы из транспортного потока и создание программного потока возможно и осуществляется с помощью общей информации формата пакетированных элементарных потоков, но не все поля программного потока содержатся в транспортном потоке, некоторые необходимо вывести.

Область действия синтаксических и семантических правил, устанавливаемых системным стандартом, различна:

- синтаксические правила применяются только для кодирования системного уровня и не распространяются на уровень сжатия видео- и звука;
- семантические правила применяются к потоку в целом.

Системный стандарт не определяет архитектуру или реализацию ни кодеров или декодеров, ни мультиплексоров или демультиплексоров. Однако свойства битового потока налагаются определенные требования на кодеры, декодеры, мультиплексоры и демультиплексоры.

Например, должно удовлетворяться требование минимальной точности часов (синхронизации). Несмотря на эти и другие требования, остается существенная степень свободы при создании кодеров, декодеров, мультиплексоров и демультиплексоров.

Далее приводятся описания вышеперечисленных черт стандарта MPEG-2.

14.4.2. Совместимость стандартов MPEG-1 и MPEG-2

MPEG-2 совместим с предыдущим стандартом MPEG-1 «вперед» и «вверх» по видеопотоку. Вообще говоря, при переходе к новому стандарту возможны следующие виды совместимости:

- совместимость «вперед» (forward compatibility) — декодер нового поколения должен уметь декодировать потоки кодера предыдущего поколения;
- совместимость «назад» (backward compatibility) — декодер предыдущего поколения должен уметь декодировать потоки или часть потока кодера нового поколения;
- совместимость «вверх» (upward compatibility) — декодер нового поколения с более высоким разрешением должен уметь декодировать потоки предыдущего поколения с низким разрешением;
- совместимость «вниз» (downward compatibility) — декодер предыдущего поколения с низким разрешением должен уметь декодировать потоки нового поколения с более высоким разрешением.

Совместимость кодеров предыдущего (MPEG-1) и последующего поколений (MPEG-2) достигается за счет введения синтаксиса расширений заголовков. Декодер предыдущего поколения игнорирует дополнительную информацию, содержащуюся в расширениях заголовков, которые были зарезервированы в предыдущем поколении стандарта, и способен восстанавливать данные, содержащиеся в битовом потоке. Таким образом, возможно обеспечение совместимости «вперед» и «вверх». Однако совместимость «назад» и «вниз» может быть гарантирована

лишь дополнительным аппаратным обеспечением декодера предыдущего поколения.

14.4.3. Масштабируемость в MPEG-2

Одной из характерных особенностей стандарта MPEG-2 является возможность иерархической структуры или масштабируемости битового потока, т. е. такого битового потока, низший уровень которого (базовый поток) может быть декодирован независимо от остального потока. Масштабируемый битовый поток может быть декодирован маломощным декодером за счет ухудшения разрешения и качества изображения. В стандарте MPEG-2 предусмотрены четыре основных вида масштабируемости, а также их различные сочетания.

При **пространственной масштабируемости (Spatial Scalability)** генерируются два (или более) потока с различным разрешением кадров. Базовый поток имеет низкое разрешение, полученное, как правило, передискретизацией, а дополнительный — высокое. Кодирование базового потока может осуществляться как стандартным способом MPEG-2, так и в соответствии со стандартом MPEG-1, что позволяет реализовать потоки, которые могут обрабатываться декодерами MPEG-1. Базовый поток достраивается до полного разрешения интерполирующим фильтром и используется также для формирования предсказаний в дополнительном потоке.

На рис. 14.12 приведена блок-схема кодера с двумя масштабами пространственного разрешения [5.60].

В случае использования **масштабируемости по отношению сигнал/шум (Signal to Noise Ratio — SNR Scalability)** генерируются два (или более) потока с одинаковым разрешением, но с разным качеством изображения. Например, базовый поток содержит достаточно грубо квантованную информацию, а расширенный поток содержит дополнительную информацию, которая при декодировании позволяет восстановить более высококачественное изображение, — разности коэффициентов ДКП базового потока и коэффициентов, полученных при использовании менее грубого квантования.

Таким образом, на выходе декодера формируются восстановленные коэффициенты ДКП по формуле:

$$QF[u][v] = QF_{\text{базовый}}[u][v] + QF_{\text{расширенный}}[u][v], \quad (14.7)$$

и далее поток восстанавливается обычным способом.

На рис. 14.13 и рис. 14.14 [5.60] показаны блок-схемы кодера и декодера с двумя масштабами разрешения по отношению сигнал/шум.

Базовый поток также может передаваться с использованием пространственной масштабируемости.

Было обнаружено, что поскольку компенсация движения для базового уровня осуществляется с помощью кадра, который при декодировании был восстановлен с использованием информации расширенного уровня, то при декодировании только базового уровня возможны искажения движения, получившие название дрейфа. Этот дрейф остается в приемлемых пределах, если ограничивать длину GOP-последовательности $N \leq 15$ и $M = 3$ (N — число кадров в GOP, M — количество P-кадров в GOP).

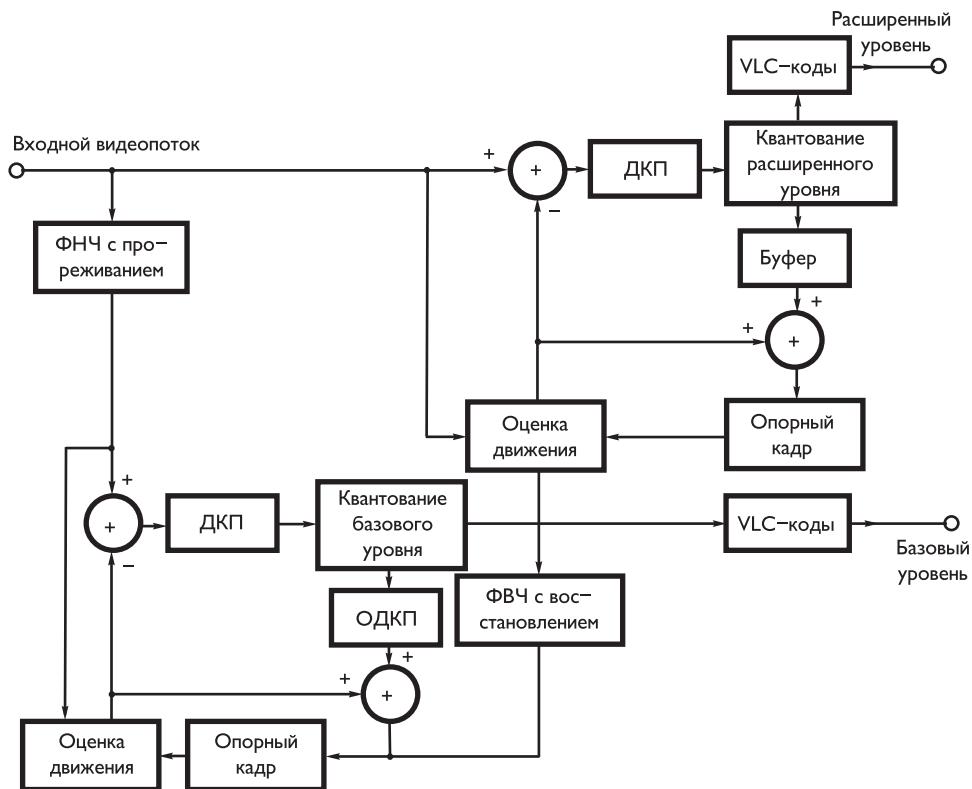


Рис. 14.12. Блок-схема кодера с двумя масштабами пространственного разрешения

При применении **временной масштабируемости** (Temporal Scalability) генерируются два (или более) потока с одинаковым пространственным, но различным временным разрешением. Базовый поток обеспечивает пониженное значение частоты кадров, а расширенный — полное временное разрешение.

Наконец, **разделение данных** (Data Partitioning) — это разбиение и передача битового потока по двум или более каналам, причем более значимые данные передаются по более помехоустойчивому каналу (например, алгоритм дробит блоки размером в 64 элемента матрицы квантования на два потока: один из них состоит из низкочастотных компонентов, а другой — из высокочастотных компонентов).

При разработке стандарта проводились исследования по возможности применения частотной масштабируемости, основная идея которой заключена в фильтрации в частотной области. Исследования показали, что она менее эффективна при предсказании по полям и дает большие ошибки при последовательном применении компенсации движения.

Возникает дрейф за счет накопления ошибок в передаче векторов движения. Данная проблема не получила удовлетворительного разрешения, и поэтому частотная масштабируемость была принята лишь частично как метод разделения данных.

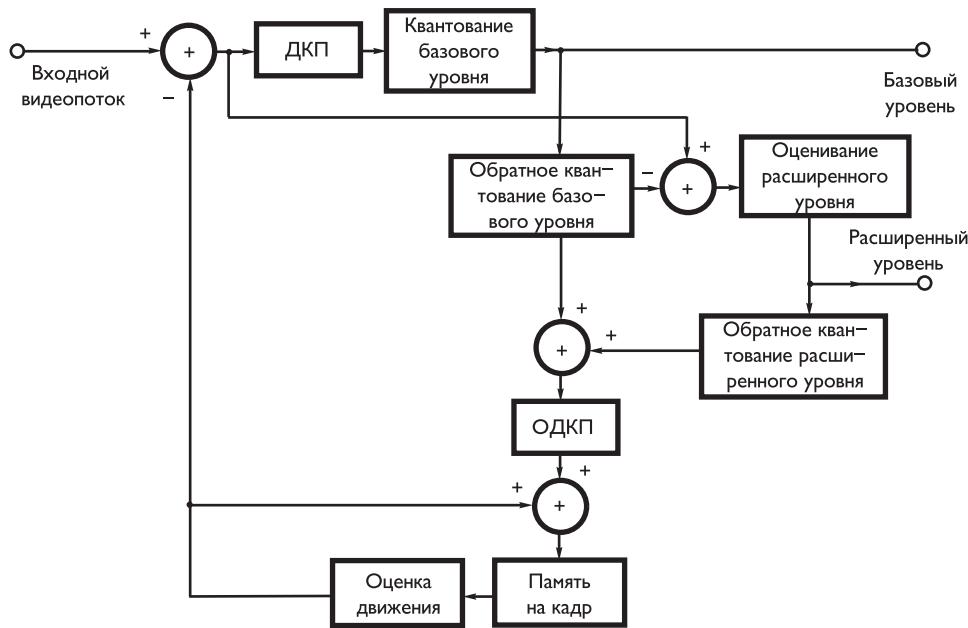


Рис. 14.13. Блок-схема кодера с масштабируемостью по отношению сигнал/шум

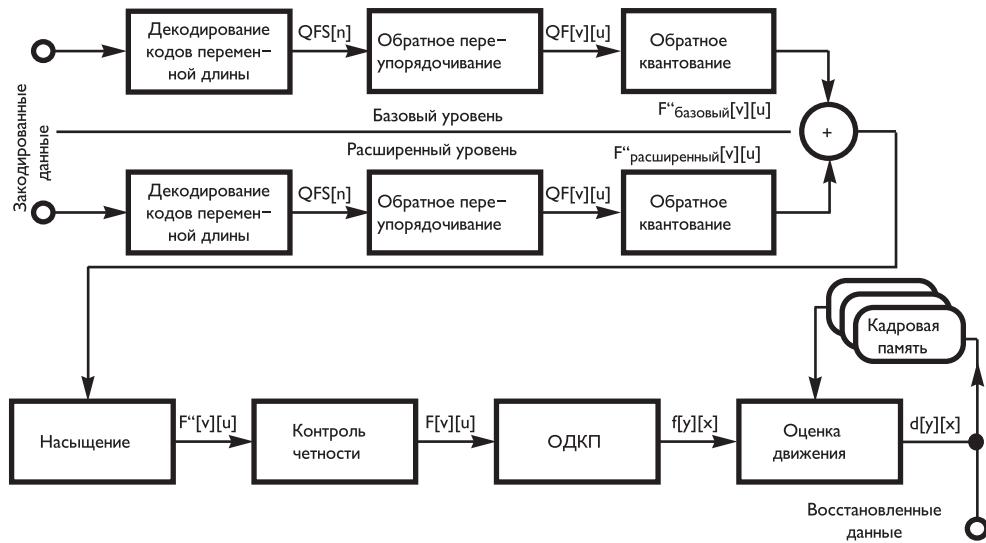


Рис. 14.14. Блок-схема декодирования потока с масштабируемостью по отношению сигнал/шум

Гибридная масштабируемость — это сочетание различных режимов масштабируемости из указанных выше.

14.4.4. Профили и уровни

Профили и уровни позволяют определить подмножества синтаксиса и семантики стандарта и, соответственно, подмножества возможностей декодера для декодирования определенного битового потока.

Профиль определяет подмножество синтаксиса битового потока, а уровень определяет набор ограничений, наложенных на параметры потока.

Цель определения опорных точек в виде профилей и уровней — облегчение обмена битовыми потоками между различными приложениями. Декодер считается совместимым с данным профилем и уровнем, если он в состоянии правильно декодировать все допустимые значения всех синтаксических элементов этого профиля и уровня. Единственное исключение из этого правила — декодер Простого профиля Главного уровня должен также иметь возможность декодировать потоки Главного профиля Низкого уровня. Битовый поток считается соответствующим профилю и уровню, если в нем не превышены допустимые диапазоны значений допустимых величин и в нем используются только разрешенные синтаксические элементы.

В стандарте определены пять основных профилей: Простой (Simple, SP), Главный (Main, MP), Мастшабируемый по отношению сигнал/шум (SNR Scalable, SNR), Пространственно масштабируемый (Spatially scalable, Spt) и Высокий (High, HP), и четыре основных уровня: Низкий (Low, LL), Главный (Main, ML), Высокий 1440 (High 1440, H-14) и Высокий (High, HL). Кроме этих основных профилей и уровней определены два дополнительных профиля: Многоракурсный (Multi-view, MVP) и профиль 4 : 2 : 2 (4 : 2 : 2p), причем последний может сочетаться только с Главным и Высоким уровнями. Сочетание профиля и уровня обычно обозначается соответствующей аббревиатурой, например, главный профиль и главный уровень — MP@ML.

SP не позволяет использовать В-кадры, MP реализует все основные черты MPEG-2, начиная от CIF формата MPEG-1 до телевидения высокой четкости (HDTV), но без масштабируемости.

Профиль 4 : 2 : 2 может обеспечить более высокое качество видео, лучшее разрешение цветности, он позволяет использовать более высокие скорости потока, чем MP@ML. Хотя 4 : 2 : 2P@ML не является частью иерархии профилей и уровней, 4 : 2 : 2P@ML декодер должен декодировать все битовые потоки, которые может декодировать MP@ML-декодер. Этот профиль обычно используется в приложениях, в которых требуется несколько циклов кодирования–декодирования, при этом деградация изображения незначительна.

Профиль MVP годится, например, для обработки стереоскопических изображений, при этом базовый поток соответствует левому изображению, а дополнительный — правому изображению стереопары. Дополнительный поток кодируется с использованием временного масштабирования.

Пять основных профилей имеют иерархическую структуру, т. е. более высокий профиль включает все синтаксические элементы более низкого профиля. Для заданного профиля набор синтаксических правил поддерживается независимо от уровня. Четыре уровня также имеют иерархическую структуру: ограничения параметров на более высоком уровне меньше или равны ограничениям на более низком уровне.

Таблица 14.2. Уровни главного профиля MPEG-2

Уровень	Размер кадра	Частота кадров, Гц	Битовый поток, Мбиты/с	Размер видеобуфера, Мбиты	Диапазон векторов движения
Низкий (LL)	352 × 288	30	4	0,475	−64...+63,5
Главный (ML)	720 × 576	30	15	1,835	−128...+127,5
Высокий 1440	1440 × 1080	60	60	7,340	−128...+127,5
Высокий (HL)	1920 × 1080	60	80	9,781	−128...+127,5

Концепция профилей и уровней помогает реализации масштабируемости в стандарте MPEG-2. Каждый из уровней определяет предельные значения основных параметров битового потока. В качестве примера в табл. 14.2 приведены параметры уровней Главного профиля (MP).

14.4.5. Транспортный и программный потоки MPEG-2

Транспортный поток предназначен для передачи или хранения одной или более программ кодированных данных и других данных в средах, где могут возникнуть существенные ошибки. Такие ошибки могут проявляться как битовые ошибки или потеря пакетов.

Транспортный поток спроектирован таким образом, что некоторые операции над транспортным потоком возможны с минимальными затратами. Среди них:

- выделение кодированных данных одной программы в транспортном потоке, их декодирование и представление результата;
- извлечение транспортных пакетов одной программы из транспортных потоков и организация другого транспортного потока с этой одной программой;
- извлечение транспортных пакетов одной или более программ из одного или более транспортного потока и организация другого транспортного потока;
- извлечение содержания одной программы из транспортного потока и организация программного потока, содержащего эту одну программу;
- преобразование программного потока в транспортный для передачи его в среде с потерями и восстановление программного потока (в некоторых случаях идентичного входному).

Транспортный поток MPEG-2 состоит из двух слоев: системного слоя и слоя компрессии.

Входной поток декодера транспортного потока имеет системный слой, «обернутый» вокруг слоя сжатия. Входные потоки видео- и звукового декодеров имеют только слой сжатия.

Операции, которые проводят декодер транспортного потока, применяются либо ко всему транспортному потоку MPEG-2 (общие операции), либо к отдельным элементарным потокам (потоковые операции).

Транспортный системный слой MPEG-2 подразделяется на два подуровня, один — для общих операций (уровень транспортных пакетов) и другой — для потоковых операций (уровень ПЭП-пакетов).

Возможно и разумно преобразование между различными типами и конфигурациями потоков MPEG-2. Есть специальные поля, определяемые синтаксисом

программного и транспортного потоков, которые облегчают эти преобразования. Не требуется, чтобы специальные реализации демультиплексоров и декодеров включали все эти функции.

Транспортный поток MPEG-2 может быть построен любым методом, который в результате даст правильный поток. Возможно строить транспортные потоки, содержащие одну или более программ, из элементарных потоков кодированных данных, из программных потоков или из других транспортных потоков, которые могут сами содержать одну или более программ.

Типичный системный декодер *программного потока* принимает программный поток MPEG-2, извлекает из потока временную информацию (информацию о временной синхронизации), демультиплексирует поток, и полученные таким образом элементарные потоки поступают на видео- и звуковой декодеры, которые на выходе дают видео- и звуковой сигналы.

В структуру декодера включен поток временной информации между системным декодером, видео- и звуковым декодерами. Видео- и звуковой декодеры синхронизированы друг с другом и с устройством цифрового хранения благодаря временной информации.

Программный поток MPEG-2 состоит также из двух слоев: системного слоя и слоя сжатия. Входные потоки видео- и звукового декодеров имеют только слой сжатия.

Операции, осуществляемые декодером программного потока, применяются либо ко всему программному потоку MPEG-2 (общие операции), либо к отдельным элементарным потокам (потоковые операции). Системный слой программного потока MPEG-2 подразделяется на два подуровня: для общих операций (слой уплотнения) и для потоковых операций (слой пакетов ПЭП).

Взаимные преобразования программного и транспортного потоков

Такие преобразования двух типов потоков MPEG-2 возможно и разумно осуществлять с помощью пакетированных элементарных потоков (ПЭП).

ПЭП-пакеты обычно могут быть прямо перенесены из одного мультиплексированного битового потока в другой. Для этого предусматривается возможность определения правильного порядка ПЭП пакетов в программе.

Другая важная для преобразования информация, а именно зависимость между элементарными потоками, имеется в таблицах и заголовках обоих потоков.

Не все транспортные потоки могут быть сформированы из программных потоков, но программный поток возможно создать из транспортного.

Пакетированный элементарный поток

Часть преобразования элементарного потока в программный поток или транспортный поток это — промежуточное преобразование в пакетированный элементарный поток (ПЭП).

ПЭП-пакеты могут иметь относительно большую длину, эта длина не фиксирована стандартом. ПЭП-не предназначен для использования в качестве самостоятельной единицы. В транспортном потоке заголовок ПЭП-выровнен с заголовком транспортного пакета.

Модель синхронизации

Для всех трех частей стандарта MPEG (система, видео, звук) используется модель синхронизации с фиксированной задержкой между окончанием ввода сигнала в кодер и окончанием выхода сигнала из декодера. Эта задержка складывается из задержки кодирования, буферизации после кодирования, мультиплексирования, передачи или хранения, демультиплексирования, буферизации декодирования, декодирования и воспроизведения.

Согласно этой модели синхронизации все видеоизображения и отсчеты звука воспроизводятся один раз, если специально не указано обратное, и интервал между кадрами и скорость воспроизведения звука одинаковы в кодере и в декодере. Кодирование системного потока включает информацию о синхронизации.

Условный доступ

Шифрование и кодирование для условного доступа к программам, включенными в программный или транспортный поток, поддерживается на системном уровне. Механизм условного доступа не специфицируется стандартом. Потоки спроектированы так, чтобы было возможно создать систему с условным доступом, несколько синтаксических элементов потока обеспечивают поддержку таких систем.

Общие операции

Общие операции включают координацию восстановления данных с цифровых носителей или из каналов связи, настройку часов (системы синхронизации) и обслуживание буферов. Эти задачи тесно связаны. Если скорость получения данных управляема, то эта скорость может быть настроена таким образом, чтобы буфер декодера никогда не переполнялся и не недозаполнялся; но если эта скорость не регулируется, то декодеры элементарного потока должны засинхронизироваться с источником данных, чтобы избежать переполнения или недозаполнения.

Программные потоки состоят из пакетов, заголовки которых облегчают эту задачу. Заголовки пакетов определяют необходимые времена, когда каждый байт должен попасть на вход декодера, и этот график служит указанием для коррекции часов и обслуживания буфера. Декодер не должен точно придерживаться графика, но необходимо компенсировать отклонения от него.

Аналогично транспортные потоки состоят из транспортных пакетов, заголовки которых несут информацию о временном графике декодирования.

Дополнительная общая операция — это способность декодера устанавливать, какие ресурсы требуются для декодирования транспортного или программного потока. Первый пакет программного потока передает параметры, помогающие декодеру в этой задаче. Например, передается информация о максимальной скорости потока данных и наибольшем количестве одновременно передаваемых видеоканалов. Транспортный поток также содержит такую информацию.

Программный и транспортный потоки содержат информацию, которая определяет характеристики и связи между элементарными потоками, составляющими каждую программу. Эта информация включает язык звуковых каналов и связь между потоками видео при многоуровневом кодировании видеинформации.

Демультиплексирование

Потоковые операции — это демультиплексирование и синхронное воспроизведение множества элементарных потоков.

При кодировании программные потоки формируются мультиплексированием элементарных потоков, транспортные потоки формируются мультиплексированием элементарных потоков, программных потоков или содержания других транспортных потоков.

Элементарные потоки могут включать конфиденциальные, зарезервированные или добавочные потоки в дополнение к звуковым и видеопотокам MPEG-2.

Потоки во времени разделены на пакеты, и пакеты выстроены в последовательности. Пакет содержит кодированные байты одного, и только одного, элементарного потока.

В программном потоке пакеты как фиксированной, так и переменной длины допустимы. Для транспортного потока длина пакета равна 188 байтов.

Длина пакета ПЭП-не определяется стандартом, но обычно выбирается достаточно большой.

При декодировании требуется демультиплексирование для воссоздания элементарных потоков из мультиплексированного программного потока. Это возможно благодаря наличию следующей информации:

- идентификационного кода потока в заголовке пакета программного потока,
- идентификационного кода пакета,
- идентификационной таблицы транспортного потока.

Синхронизация

Синхронизация нескольких элементарных потоков производится с помощью временных меток в программном и транспортном потоках. Временные метки обычно выражены в единицах 90 кГц, но ссылки на системные часы (SCR), ссылки на программные часы (PCR) и ссылки на часы элементарного потока имеют уточнение в единицах 27 МГц.

Информация о синхронизации содержится в нескольких полях данных. Эта информация кодируется как дискретизированное значение системного времени. Значение частоты системных часов f (в Гц) должно удовлетворять ограничению:

$$(27\,000\,000 - 1350) \leq f \leq (27\,000\,000 + 1350).$$

Скорость изменения частоты системных часов $\leq 75 \cdot 10^{-3}$ Гц/сек.

Декодирование N элементарных потоков синхронизируется с помощью подстройки декодирования всех потоков к основной временной базе, а не с помощью подстройки одного потока к другому. Основной временной базой могут быть часы одного из N декодеров, часы канала, устройства цифрового хранения, или какие-либо дополнительные часы.

В транспортном потоке, который содержит несколько программ, каждая программа имеет свою собственную временную базу, и временные базы различных программ в таком потоке могут быть различными.

Так как временные метки для воспроизведения (PTS) используются при декодировании отдельных элементарных потоков, они присутствуют на уровне паке-

тов как в программном, так и в транспортном потоках. Кодер сохраняет временные метки в момент захвата, декодер использует эти метки для синхронизации при воспроизведении.

Синхронизация декодера с источником данных достигается с помощью использования системных часов SCR в программном потоке и эквивалентных им программных часов PCR в транспортном потоке. SCR и PCR — временные метки, кодирующие временную синхронизацию битового потока на той же временной базе, которая используется для звуковых и видео-PTS этой же программы.

Приложения

Потоки, определенные в стандарте, спроектированы так, чтобы они были наиболее удобны для множества приложений. Разработчики приложений должны просто выбрать наиболее подходящий поток.

Например, имеется множество исследований и разработок, которые пытаются посыпать в реальном масштабе времени потоки информации (видео, звуковой и др.) по компьютерным сетям общего назначения. В этой среде необходимо четкое разделение слоев. ПЭП-потоки должны быть инкапсулированы в соответствующий транспортный протокол.

Для приложений, работающих в условиях отсутствия ошибок, программный поток более удобен, чем транспортный. Мультимедиаприложения на CD-ROM попадают в эту категорию.

Структура и параметры кодирования программного потока

Слой кодирования программного потока позволяет создавать единый поток из одного или более элементарных потоков. Данные каждого элементарного потока мультиплексируются и кодируются вместе с информацией, которая позволяет воспроизводить элементарные потоки синхронно.

Программный поток MPEG-2 состоит из одного или более мультиплексированных элементарных потоков. Каждый элементарный поток состоит из единиц доступа, которые являются кодированным представлением единиц воспроизведения. Единица воспроизведения для элементарного потока видео — изображение. Соответствующая единица доступа включает все кодированные данные для изображения. Единица доступа, содержащая первое из группы изображение, также включает всю информацию, предшествующую этой группе.

Например, единица доступа, содержащая первый после заголовка последовательности кадр, включает в себя сам заголовок последовательности. Код конца последовательности включается в единицу доступа, содержащую последний кадр.

Данные элементарных потоков сохраняются в ПЭП-пакетах. ПЭП-пакет состоит из заголовка, за которым следуют данные. Заголовок ПЭП-пакета начинается со стартового кода, который также идентифицирует, к какому потоку принадлежит пакет. Заголовок пакета может содержать метки времени декодирования и/или воспроизведения, которые относятся к первой единице доступа, входящей в пакет.

Данные пакета содержат переменное количество байтов одного элементарного потока.

ПЭП-пакеты программного потока организованы в пачки. Пачка начинается с заголовка, за которым может следовать один или несколько пакетов (пачка может состоять из одного заголовка).

Заголовок пачки может содержать информацию о синхронизации и битовой скорости.

Программный поток начинается с системного заголовка, который может повторяться. Системный заголовок несет информацию о системных параметрах потока.

Структура и параметры кодирования транспортного потока

Слой кодирования транспортного потока позволяет одну или более групп одного или более элементарных потоков объединять в единый поток. Группа элементарных потоков с общей временной базой называется программой. Данные всех элементарных потоков мультиплексируются и кодируются вместе с информацией, которая позволяет синхронно воспроизводить элементарные потоки программы.

Транспортный поток MPEG-2 состоит из одной или более программ, каждая из которых содержит один или несколько мультиплексированных элементарных потоков.

Каждый элементарный поток состоит из единиц доступа, которые являются кодированным представлением единиц воспроизведения (например, кадра изображения). Соответствующая единица доступа содержит все кодированные данные для изображения.

Единица воспроизведения элементарного потока звука — набор дискретных отсчетов, соответствующий отсчетам звукового «кадра».

Данные элементарных потоков сохраняются в ПЭП-пакетах. ПЭП-пакет состоит из заголовка, за которым следуют данные. ПЭП-пакеты сохраняются в транспортных пакетах. Начало каждого ПЭП-пакета совпадает с началом полезной информации в транспортном пакете.

Заголовок ПЭП-пакета начинается со стартового кода, который также идентифицирует, к какому потоку принадлежит пакет. Заголовок пакета может содержать метки времени декодирования и/или воспроизведения, которые относятся к первой единице доступа, входящей в пакет. Заголовок ПЭП-пакета также содержит ряд флагов, устанавливающих значения ряда полей. Данные пакета содержат переменное количество байтов одного элементарного потока.

Транспортный пакет начинается с 4-байтного префикса, который содержит идентификатор пакета и с помощью таблицы специальной информации программы определяет содержание ПЭП-пакета, передаваемого в транспортном потоке.

Транспортный поток содержит сегмент разметки программ, карту потока программ и таблицу, которая находится в специальных транспортных пакетах. Эта карта и таблица содержат необходимую и достаточную информацию для демультиплексирования и воспроизведения программ. Таблица специальной информации программ определяет, наряду с другой информацией, какие идентификаторы пакетов, а следовательно, и какие элементарные потоки связаны с каждой из программ.

14.4.6. Видеопоток стандарта MPEG-2

Поток видеоданных MPEG представляет собой структурированный битовый поток кодов постоянной и переменной длины, синтаксис которого определяется набором стандартов MPEG [5.39–5.37, 5.44–5.47].

В целом поток видеоданных MPEG-2 представляется последовательностью структурных элементов, общая организация потока показана на рис. 14.15. Синтаксис кодированного битового потока видеоданных содержит шесть уровней.

Уровень **видеопоследовательности** (video sequence layer), состоящий из заголовка видеопоследовательности, его расширения, одной или более групп изображений и завершающего кода конца видеопоследовательности.

Если заголовок видеопоследовательности повторяется, то он должен содержать одинаковую информацию за исключением матриц квантования (как внутрикадрового, так и межкадрового). Это необходимо для обеспечения произвольного доступа.

Уровень **группы кадров** (group of pictures layer), состоящий из заголовка группы кадров и одного или нескольких кадров. Группа кадров — это серия, состоящая из одного или более последовательных изображений. В кодированном потоке первое кодированное изображение — это обычно I-кадр.

Порядок следования кадров в группе соответствует порядку поступления кадров в декодер. Если кадры расположить в порядке воспроизведения, то последний кадр группы — обычно I-кадр или P-кадр. Первый кадр группы — это или I-кадр, или первый B-кадр из серии B-кадров, за которой сразу следует первый I-кадр.

Например, предположим, что на входе кодера имеется последовательность кадров (знак | означает границу группы):

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
I	V	B	V	R	V	V	R	P	V	B	V	I	V	B	R	V	V	R	P	V	V	R	V	

В этом случае на входе декодера должны поступать кадры в такой последовательности:

1	4	2	3	7	5	6	10	8	9	13	11	12	16	14	15	19	17	18	22	20	21	25	23	24
I	P	B	V	R	P	V	B	V	I	B	V	P	R	V	B	P	R	V	B	P	V	B	V	

В группе должен содержаться один I-кадр или более. Если приложению необходим произвольный доступ, обратное воспроизведение и т. д., то группы изображений должны быть относительно короткими. Группы могут начинаться в момент смены сюжета и в других местах, где неэффективна компенсация движения.

Уровень **кадра** (pictures layer), состоящий из заголовка кадра, расширения заголовка кадра и данных кадра, разбитых на слайсы.

Стандарт включает обработку изображений как с прогрессивной, так и с че-рессстрочной разверткой. Пары полей могут кодироваться как раздельно, так и вместе как единый кадр. Возможно динамическое переключение кодирования полей и кадров. При прогрессивной развертке возможно только кодирование кадров.

В режиме кодирования полей каждое поле кодируется как отдельное изображение (полукадр). Полукадры передаются в том же порядке, в каком они должны воспроизводиться.

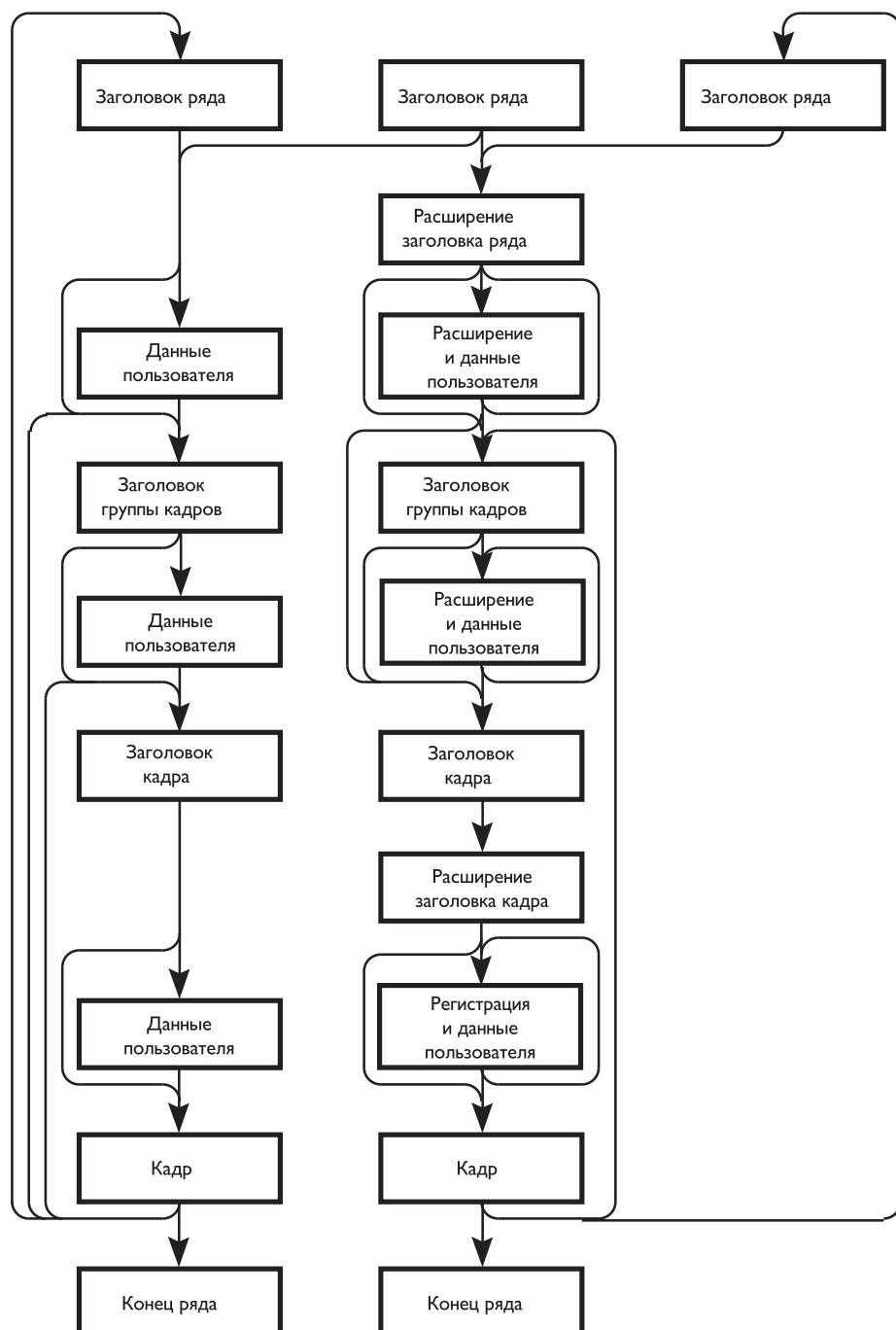


Рис. 14.15. Общая организация битового потока, формируемого в соответствии со стандартом MPEG-2



Два полукадра, составляющие один кадр, всегда следуют один за другим в битовом потоке. Если первый из них является Р- или В-кадром, то и второй полукадр также является Р- или В-кадром соответственно. Если же первый полукадр является I-кадром, то второй полукадр может быть I-кадром или Р-кадром.

При кодировании кадрами чересстрочных изображений каждый кадр набирается из двух полукадров (через строку) и кодируется как единое изображение.

Уровень слайса (slice layer), состоящий из заголовка слайса и макроблоков

Весь кадр подразделяется на области размером 16×16 пикселов (макроблоки). Макроблоки объединяются в слайсы. Слайс состоит из последовательности макроблоков, причем порядок следования макроблоков — слева направо. Первый и последний макроблоки слайса не должны быть пропущенными (отсутствовать в потоке данных кадра). Они должны находиться в одном и том же горизонтальном ряду макроблоков. Каждый слайс должен содержать по крайней мере один макроблок. Слайсы не должны перекрываться, но, вообще говоря, могут иметь интервалы между собой. Стандарт не специфицирует, какие действия декодер должен выполнять для восстановления информации, содержащейся в пропущенных местах в промежутках между слайсами.

Уровень макроблоков (macroblock layer)

Макроблок содержит область компонент яркости и пространственно соответствующие ей области компонент цветности. Макроблок может относиться или к источнику и декодированным данным или к соответствующим кодированным элементам данных. Пропущенный макроблок — это макроблок, для которого не передается никакой информации.

Уровень блоков (block layer)

Термин «блок» может относиться или к источнику и восстановленным данным, или к соответствующим коэффициентам ДКП. Если термин «блок» относится к источнику или к восстановленным данным, то он представляет собой прямоугольную область яркостных или цветостных компонент с одинаковым количеством строк и столбцов (8×8). Если блок обозначает коэффициенты ДКП, то это должно быть 64 коэффициента.

Изображение формируется из трех прямоугольных матриц, содержащих восемьбитовые величины: матрица яркости Y_D и две матрицы цветности — C_R и C_B .

Если в стандарте MPEG-1 матрицы Y_D , C_R и C_B связаны с первичными значениями красного, зеленого и синего согласно рекомендации ITU-R BT.709 [5.61], то в стандарте MPEG-2 могут быть использованы и другие соотношения между Y_D , C_R , C_B и R, G, B (ITU-R BT.624 [5.62], SMPTE 170M [5.63], SMPTE 240M [5.64] и др.).

Стандарт предусматривает три формата передачи цветности.

Формат $4 : 2 : 0$ — матрицы C_R и C_B в два раза меньше как по горизонтали, так и по вертикали, чем матрица Y_D . Матрица Y_D должна иметь четное число строк и столбцов (если изображение передается полями, то количество строк в

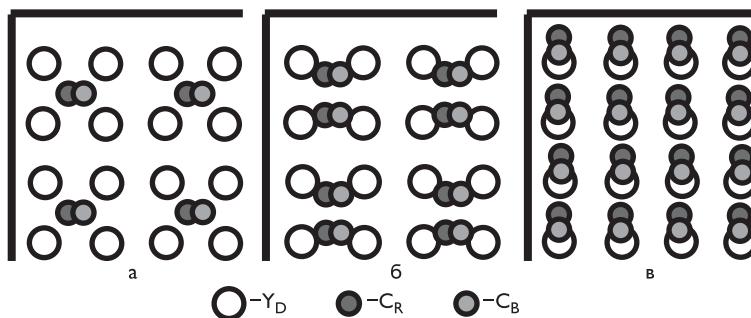


Рис. 14.16. Расположение отсчетов в матрицах Y_D , C_R , C_B в форматах цветности $4:2:0$ (а), $4:2:2$ (б) и $4:4:4$ (в)

изображении должно быть кратно четырем). Каждой паре отсчетов цветности C_R и C_B соответствует матрица из четырех отсчетов яркости (два по вертикали, два по горизонтали, рис. 14.16а).

Формат $4:2:2$ — матрицы C_R и C_B в два раза меньше матрицы Y_D по горизонтали и равны по вертикали. Матрица Y_D должна иметь четное число строк и столбцов (если изображение передается полями, то количество строк в изображении должно быть кратно двум). Каждой паре отсчетов яркости по горизонтали соответствует по одному отсчету из матриц цветности (рис. 14.16б).

Формат $4:4:4$ — матрицы C_R и C_B равны по размеру матрице Y_D . Если изображение передается полями (с чересстрочной разверткой), то количество строк в изображении должно быть кратно двум. Каждому отсчету яркости соответствует по одному отсчету из матриц цветности (рис. 14.16в).

Отметим ряд возможностей, заложенных в потоке данных стандарта MPEG-2. В заголовке видеопоследовательности, в частности, определяются следующие параметры:

- ширина и высота изображения;
- скорость воспроизведения кадров;
- размер буфера, необходимого для декодирования битового потока;
- переопределение матриц квантования (при необходимости).

Заголовок кадра содержит, в частности:

- `temporal_reference` — 10-битовое целое число, оно ставится в соответствие каждому кадру и увеличивается на единицу с каждым кадром; для первого кадра в хронологическом порядке `temporal_reference` равен нулю; если кадр кодируется как два поля, то `temporal_reference` одинаково для обоих полей;
- `picture_coding_type` — определяет, является ли кадр I-кадром (внутрикадровое кодирование), P-кадром (предсказанным на основе предыдущего I- или P-кадра) или B-кадром (кодированным с двунаправленным предсказанием).

Порядок следования блоков в макроблоке различен для разных цветовых форматов.

В формате $4:2:0$ макроблок состоит из шести блоков. Он содержит четыре блока яркости Y_D , и два блока цветности C_R и C_B в порядке, указанном на рис. 14.17а.

В формате $4:2:2$ макроблок состоит из восьми блоков. Он содержит четыре Y_D , два C_R и два C_B блока в порядке, указанном на рис. 14.17б.

В формате $4:4:4$ макроблок состоит из двенадцати блоков. Он содержит четыре Y_D , четыре C_R и четыре C_B блока в порядке, указанном на рис. 14.17в.

Внутренняя организация макроблоков различна при кодировании кадров построчно или по полям, как показано на рис. 14.18. При кодировании полей блоки яркости группируются по полям: верхние — из первого поля, а нижние — из второго. Блоки цветности располагаются в порядке следования кадров для обоих типов кодирования (для кодирования $4:2:2$ и $4:4:4$ данные цветности также могут расщепляться по полям).

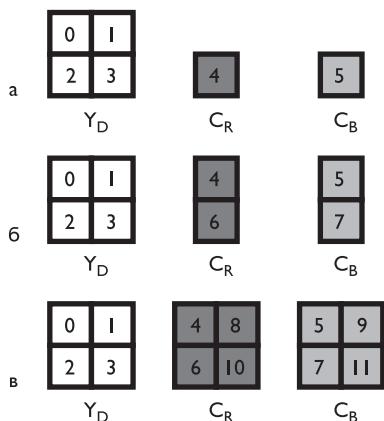


Рис. 14.17. Порядок следования блоков в макроблоках при формате $4:2:0$ (а), при формате $4:2:2$ (б) и при формате $4:4:4$ (в)

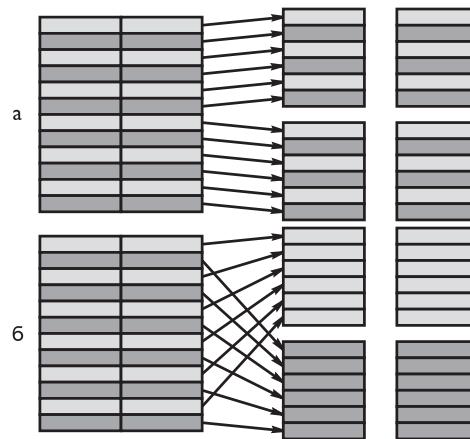


Рис. 14.18. Формирование макроблоков при построчном кодировании кадра (а) и при независимом кодировании по полям (б)

В зависимости от типа кадра используются различные типы макроблоков. В I-кадрах все макроблоки внутрикодированные. В P-кадрах макроблоки могут быть пропущенными, внутрикодированными, кодированными с предсказанием вперед. В B-кадрах кроме типов макроблоков, встречающихся в P-кадрах, передаются макроблоки, кодированные с двунаправленным предсказанием и кодированные с предсказанием назад.

Макроблоки 16×16 разбиваются на блоки 8×8 . Стандарт использует блочное дискретное косинусное преобразование. После преобразования осуществляется поэлементное квантование коэффициентов ДКП. В формате $4:2:0$ по умолчанию используется две матрицы квантования: для внутрикодированных макроблоков и для всех остальных (рис. 14.19).

В иных форматах используется до четырех матриц — для яркости и для цветности матрицы могут быть различными. Матрицы имеют значения по умолчанию, если не загружены иные.

8	16	19	22	26	27	29	34
16	16	22	24	27	29	34	37
19	22	26	27	29	34	34	38
22	22	26	27	29	34	37	40
22	26	27	29	32	35	40	48
26	27	29	32	35	40	48	58
26	27	29	34	38	46	56	69
27	29	35	38	46	56	69	83

a

16	16	16	16	16	16	16	16
16	16	16	16	16	16	16	16
16	16	16	16	16	16	16	16
16	16	16	16	16	16	16	16
16	16	16	16	16	16	16	16
16	16	16	16	16	16	16	16
16	16	16	16	16	16	16	16
16	16	16	16	16	16	16	16

6

Рис. 14.19. Матрицы квантования по умолчанию для I-блоков (а) и для P- и B-блоков (б)

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

a

0	4	6	20	22	36	38	52
1	5	7	21	23	37	39	53
2	8	19	24	34	40	50	54
3	9	18	25	35	41	51	55
10	17	26	30	42	46	56	60
11	16	27	31	43	47	57	61
12	15	28	32	44	48	58	62
13	14	29	33	45	49	59	63

6

Рис. 14.20. Таблицы переупорядочивания коэффициентов ДКП блоков: а — стандартная таблица, б — альтернативная таблица

После квантования блок преобразуется в линейный массив с помощью стандартной или альтернативной таблицы Z-упорядочивания, которые приведены на рис. 14.20.

Коэффициенты ДКП кодируются кодами переменной длины. DC-коэффициент (коэффициент, располагающийся в верхнем левом углу блока) кодируется иначе, чем остальные коэффициенты. В MPEG-2, в отличие от MPEG-1, возможен выбор альтернативной таблицы кодов.

Используется гибридный метод кодирования — длина кода коэффициента определяется предшествующим ему кодом переменной длины с уникальным префиксом, а этот код интерпретируется по одной из стандартных таблиц.

В стандарте заложена возможность использования нескольких типов предсказания и компенсации движения как раздельно, так и в различных комбинациях.

В то время как в MPEG-1 вводится только один тип предсказания движения для макроблока в целом, в стандарте MPEG-2 предусмотрены четыре варианта:

- кадровая компенсация движения (frame motion compensation): вычисляется один вектор движения (в Р-кадре) или два векторы движения (в В-кадре) для макроблока в целом;
- компенсация движения по полям (field motion compensation): вычисляются векторы движения как для верхнего, так и для нижнего полей макроблока, причем для каждого из полей выбирается свое поле сравнения в базовом кадре;
- компенсация движения типа 16×8 с двумя векторами движения (16×8 motion compensation): вычисляются векторы движения для каждой из двух половин (верхней и нижней) макроблока (используется только для полей);
- двухстадийная компенсация движения с уточнением вектора движения (dual prime motion compensation): процедура уточнения вектора движения только для Р-макроблока, в которой вектор движения находится для первого поля, предсказание ведется от двух полей базового макроблока, результаты усредняются, и для второго поля вместо полного вектора движения передается малая поправка к вектору движения для первого поля.

Для внутриблочно кодированных макроблоков предсказание не формируется, однако и с этими блоками возможна передача скрытых векторов движения (concealment motion vectors), которые могут быть использованы для маскирования нарушений изображения при появлении ошибки при передаче потока. Конкретный способ использования скрытых векторов движения стандартом не предусмотрен.

В Р- и В-кадрах предсказание производится различным образом. Пропущенный макроблок в Р-кадрах при восстановлении копируется из опорного кадра. Предсказанием макроблока в Р-кадре, для которого передается информация о векторах движения (macroblock_motion_forward=1), является область 16×16 из опорного кадра, смещенная по вектору движения от положения предсказываемого макроблока. В том случае, если компоненты векторов движения нецелые (стандарт определяет векторы с точностью до $1/2$ пикселя), производится вычисление промежуточных точек линейной интерполяцией.

В В-кадрах пропущенные макроблоки имеют такие же тип, метод предсказания и вектора движения, что и последний не пропущенный, поэтому стандарт не допускает пропуск макроблоков в В-кадрах, следующих сразу за INTRA-макроблоком. Предсказание вперед или назад для В-кадров осуществляется аналогично предсказанию вперед в Р-кадрах. Для макроблоков с векторами движения и вперед, и назад предсказание формируется как среднее между предсказанием вперед и предсказанием назад.

Векторы движения кодируются дифференциально по отношению к ранее переданному вектору движения с использованием кодов переменной длины.

Возможная максимальная длина вектора движения может меняться от кадру к кадру для того, чтобы удовлетворить требованиям различных приложений. Спецификация не определяет способа вычисления векторов движения. Стандарт MPEG-1 использует два вектора движения, в MPEG-2 используется до шести векторов на макроблок.

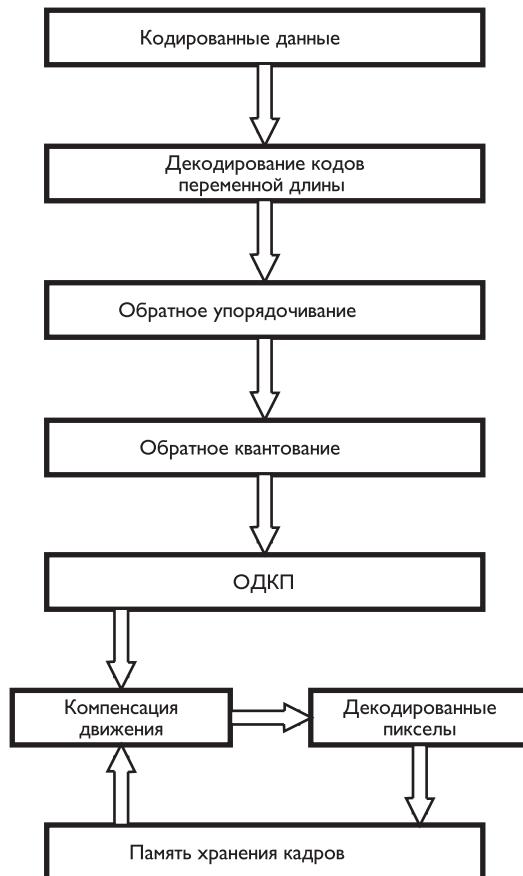


Рис. 14.21. Структура декодирования видеопотока MPEG-2

Декодирование видеопотока MPEG-2

Стандарт MPEG специфицирует требования к так называемому совместимому стандартному декодеру. За исключением обратного косинусного преобразования (ОДКП), для которого стандарт задает лишь требования к точности преобразования, но не к его конкретной реализации, процесс декодирования определен так, чтобы все декодеры давали численно идентичные результаты.

Упрощенная структура декодирования представлена на рис. 14.21.

Процесс восстановления изображения из кодированного потока состоит из следующих основных этапов: восстановление блоков и макроБЛОКОВ, формирование предсказания и суммирование с восстановленными «остатками», несколько контрольных процедур, ограничивающих получаемые значения.

При восстановлении блоков производится достраивание строки восстановленных коэффициентов косинусного преобразования до длины 64 нулевыми значениями, обратное переупорядочивание строки из 64 коэффициентов в матрицу размера 8×8 .



Полученная на предыдущем шаге матрица подвергается процедуре обратного квантования. Затем к матрице применяется процедура обратного дискретного косинусного преобразования. Окончательное восстановление макроблоков заключается в суммировании «остатков», полученных при декодировании блоков, с предсказанием. В случае внутрикодированных макроблоков предсказание не производится.

Кадры, поступающие на декодер, находятся не в хронологическом порядке, а в порядке, необходимом декодеру для последовательного восстановления кадров. Например, для восстановления В-кадра с двунаправленным предсказанием декодеру необходима информация как о предыдущем, так и о последующем кадре, и поэтому последующий опорный кадр должен быть получен декодером раньше. Процедура переупорядочивания стандартом не определяется. В памяти постоянно сохраняется до трех кадров: один — текущий и два для формирования предсказания.

Каждая группа кадров состоит из нескольких кадров. Она начинается и заканчивается опорными кадрами, между которыми находятся В-кадры (В-кадры хронологически могут находиться и перед первым опорным кадром). В группе первым на декодер поступает опорный кадр, открывающий группу. При декодировании в памяти сохраняются два последних опорных кадра, которые используются для предсказания В-кадров.

14.4.7. Структура битового потока видеопоследовательности стандарта MPEG-2

Уровень видеопоследовательности (video sequence layer) содержит заголовок и расширение заголовка видеопоследовательности. Синтаксис видеопотока MPEG-2 обязательно включает расширение заголовка видеопоследовательности, в противном случае видеопоток должен соответствовать стандарту MPEG-1.

Заголовок видеопоследовательности имеет следующую структуру:

заголовок_ видеопоследовательности()	Количество битов
{	
код_заголовка_ видеопоследовательности	32
размер_по_горизонтали	12
размер_по_вертикали	12
код_отношения_размеров	4
код_частоты_кадров	4
битовый_поток	18
маркер	1
код_видеобуфера	10
флаг_ограничения_параметров	1
загружаемая_матрица_I	1
if (загружаемая_матрица_I)	
матрица_I [64] 8 · 64	
загружаемая_матрица_RB	1
if (загружаемая_матрица_RB)	
матрица_RB [64] 8 · 64	
следующий_стартовый_код()	
}	

Ниже в скобках даются соответствующие английские наименования элементов видеопотока, как они приведены в исходном тексте стандарта.

В заголовок видеопоследовательности входят следующие переменные:

Код_заголовка_видеопоследовательности (sequence_header_code) — константа длиной в 32 бита, равная 000001B3 в шестнадцатиричном формате, открывающая видеопоток.

Размер_по_горизонтали (horizontal_size_value) — 12 младших битов горизонтального размера изображения.

Размер_по_вертикали (vertical_size_value) — 12 младших битов вертикального размера изображения.

Размер по горизонтали (horizontal size) и размер по вертикали (vertical size) представляют собой 14-битовые величины, из которых 12 младших битов передаются в заголовке видеопоследовательности, а 2 старших бита — в переменных **расширение_размера_по_горизонтали (horizontal_size_extension)** и **расширение_размера_по_вертикали (vertical_size_extension)** в расширении видеопоследовательности.

Код_отношения_размеров (aspect_ratio_information) — 4 бита, в которых содержится информация об отношении вертикального и горизонтального размера пикселя (SAR — sample aspect ratio).

Отношение может быть равно единице или же вычисляться по следующей формуле:

$SAR = DAR \times \frac{\text{размер_по_горизонтали}}{\text{размер_по_вертикали}}$ — если в потоке отсутствует элемент «расширение воспроизведения видеопоследовательности» (см. ниже), или:

$SAR = DAR \times \frac{\text{горизонтальный_размер_воспроизведения}}{\text{вертикальный_размер_воспроизведения}}$, если в потоке присутствует «расширение воспроизведения видеопоследовательности».

Здесь DAR — отношение вертикального и горизонтального размеров дисплея (display aspect ratio), определяемое по табл. 14.3.

Код_частоты_кадров (frame_rate_code) — 4 бита, в которых содержится информация о частоте кадров, получаемая из табл. 14.4.

Если в таблице присутствует код, для которого непосредственно указано значение частоты кадров (frame_rate), то используется непосредственно это значение и переменные **расширение_частоты_кадров_n (frame_rate_extension_n)** и **расширение_частоты_кадров_d (frame_rate_extension_d)** должны быть равны 0, в противном случае частота кадров вычисляется по формуле:

$$\begin{aligned} \text{частота_кадров} = & \text{частота_кадров} \times \\ & \times \frac{(\text{расширение_частоты_кадров_n} + 1)}{(\text{расширение_частоты_кадров_d} + 1)}, \end{aligned}$$

где **расширение_частоты_кадров_n** и **расширение_частоты_кадров_d** передаются в элементе видеопотока «расширение заголовка видеопоследовательности».

Полученное таким образом значение frame_rate обозначает обратную величину времени между двумя соседними кадрами или обратную величину удвоенного времени между соседними полями.

Таблица 14.3. Коды отношения размеров по вертикали и по горизонтали

Код_отношения_размеров	SAR	DAR
0000	запрещено	запрещено
0001	1.0	—
0010	—	3 : 4
0011	—	9 : 16
0100	—	1 : 2.21
0101	—	зарезервировано
...	—	...
1111	—	зарезервировано

Таблица 14.4. Коды частоты кадров

Код_частоты_кадров	Частота кадров
0000	запрещено
0001	24000/1001 (23,976)
0010	24
0011	25
0100	30000/1001 (29,97)
0101	30
0110	50
0111	60000/1001 (59,94)
1000	60
...	зарезервировано
1111	зарезервировано

Битовый_поток (bit_rate_value) — 18 младших битов кода величины битового потока.

Код битового потока (bit_rate) представляет собой 30-битовый код, из которых 18 младших битов передаются в заголовке видеопоследовательности, а 12 старших битов — в расширении видеопоследовательности в переменной **расширение_битового_потока (bit_rate_extension)**. Если величина из элемента «заголовок кадра» (см. ниже) **задержка_буфера**=FFFF в шестнадцатиричном формате, то битовый поток — переменный, а **битовый_поток** обозначает величину, используемую в транспортном потоке, в противном случае **битовый_поток** задает величину битового потока в единицах 400 битов/сек.

- **Маркер (marker_bit)** — константа длиной в 1 бит, равная единице и пред назначенная для избежания эмуляции стартовых кодов.
- **Код_видеобуфера (vbv_buffer_size_value)** — 10 битов. Размер контрольного видеобуфера (vbv_buffer_size) представляет собой 18-битовую величину, из которых 10 младших битов передаются в заголовке, а 8 старших битов — в расширении видеопоследовательности в переменной **расширение_видеобуфера (vbv_buffer_size_extension)**. Величина контрольного видеобуфера определяется по формуле:

$$B = 16 \times 1024 \times \text{контрольный_видеобуфер} \text{ [битов].}$$

- **Флаг_ограничения_параметров (constrained_parameters_flag)** — 1 бит, используемый только в MPEG-1 и всегда равный нулю в MPEG-2.

Он устанавливается равным 1 в MPEG-1, если размер по горизонтали меньше или равен 768, размер по вертикали меньше или равен 576, частота кадров меньше или равна 30 кадров/сек, размер векторов движения ограничен диапазоном $(-64\dots+63,5)$ (при точности нахождения векторов движения до полпикселя), битовый поток не превышает 1 856 000 битов/сек, контрольный видеобуфер меньше или равен 327 680 битов.

- **Загружаемая_матрица_I** (`load_intra_quantizer_matrix`) — 1 бит, указывающий, нужно ли использовать стандартную матрицу квантования для I-кадров, приведенную ниже в разделе о расширении матриц квантования, или следует загружать другую матрицу. Если бит **загружаемая_матрица_I** = 1, то за ним следуют 64 байта, задающих матрицу квантования для I-кадров.
- **Загружаемая_матрица_PB** (`load_non_intra_quantizer_matrix`) — 1 бит, указывающий, нужно ли использовать стандартную матрицу квантования для P- и B-кадров, приведенную в разделе о расширении матриц квантования, или следует загружать другую матрицу. Если бит **загружаемая_матрица_PB** = 1, то за ним следуют 64 байта, задающих матрицу квантования для P- и B-кадров.

Расширение заголовка видеопоследовательности имеет следующую структуру:

<u>заголовок_видеопоследовательности()</u>	<u>Количество битов</u>
{	
стартовый_код_расширения	32
код_идентификатора_расширения	4
код_профиля_уровня	8
прогрессивная_развертка	1
код_формата_цветности	2
расширение_размера_по_горизонтали	2
расширение_размера_по_вертикали	2
расширение_битового_потока	12
маркер	1
расширение_видеобуфера	8
малая_задержка	1
расширение_частоты_кадров_n	2
расширение_частоты_кадров_d	5
следующий_стартовый_код()	
}	

В него входят следующие переменные.

- **Стартовый_код_расширения** (`extension_start_code`) — константа длиной в 32 бита, равная 000001B5 в шестнадцатиричном формате, открывающая любое расширение видеопотока по сравнению с MPEG-1.
- **Код_идентификатора_расширения** (`extension_start_code_identifier`) — константа длиной в 4 бита, равная 0001 в двоичном формате, обозначающая расширение заголовка видеопоследовательности.

- **Код_профиля_уровня (profile_and_level_indication)** — 8 битов, в которых содержится информация о профиле и уровне видеопотока. Старший бит в настоящем стандарте устанавливается равным нулю (значение 1 зарезервировано для будущих расширений стандарта); следующие три бита указывают профиль и заключительные четыре — уровень потока согласно табл. 14.5 и 14.6.
- **Прогрессивная_развертка (progressive_sequence)** — 1 бит.

Если **прогрессивная_развертка** = 1, то в последовательности могут встречаться только кадры с прогрессивной разверткой; в противном случае в ней могут встречаться как кадры с прогрессивной или с чересстрочной разверткой, так и поля.

Таблица 14.5. Коды профиля

Код	Профиль
110 … 111	зарезервировано
101	простой (simple)
100	основной (main)
011	масштабируемый по отношению сигнал/шум (SNR scalable)
010	пространственно масштабируемый (spatially scalable)
001	профессиональный 422 (professional, 422P)
000	зарезервировано

Таблица 14.6. Коды уровня

Код	Уровень
1011 … 1111	зарезервировано
1010	низкий (low)
1001	зарезервировано
1000	основной (main)
0111	зарезервировано
0110	высокий-1440 (high 1440)
0101	зарезервировано
0100	высокий (high)
0000 … 0011	зарезервировано

Код_формата_цветности (chroma_format) — 2 бита, определяющих значение формата цветности согласно табл. 14.7.

Таблица 14.7. Коды формата цветности

Код_формата_цветности	Формат цветности
00	зарезервировано
01	4 : 2 : 0
10	4 : 2 : 2
11	4 : 4 : 4

- **Расширение_размера_по_горизонтали (horizontal_size_extension)** — 2 старших бита размера по горизонтали.
- **Расширение_размера_по_вертикали (vertical_size_extension)** — 2 старших бита размера по вертикали.

- **Расширение_битового_потока (bit_rate_extension)** — 12 старших битов кода битового потока.
- **Расширение_видеобуфера (vbv_buffer_size_extension)** — 8 старших битов кода размера контрольного видеобуфера.
- **Малая_задержка (low_delay)** — 1 бит. Если **малая_задержка** = 1, то в последовательности не могут встречаться В-кадры и отпадает необходимость в переупорядочивании кадров в декодере и в описании контрольного видеобуфера, допускается опустошение видеобуфера (VBF buffer underflow).
- **Расширение_частоты_кадров_n (frame_rate_extension_n)** — 2 бита.
- **Расширение_частоты_кадров_d (frame_rate_extension_d)** — 5 битов (см. **код_частоты_кадров**).

Расширение воспроизведения видеопоследовательности

Информация в данном разделе не влияет на процесс декодирования, но может использоваться устройствами воспроизведения. Расширение воспроизведения видеопоследовательности имеет следующую структуру:

заголовок_видеопоследовательности()	Количество битов
{	
код_идентификатора_расширения	4
код_формата_видео	3
описание_цветности	1
if (описание_цветности){	
первичные_цвета	8
код_характеристики_передачи	8
код_матрицы_преобразования	8
}	
горизонтальный_размер_воспроизведения	14
маркер	1
вертикальный_размер_воспроизведения	14
следующий_стартовый_код()	
}	

В него входят следующие переменные.

- **Код_идентификатора_расширения (extension_start_code_identifier)** — константа длиной в 4 бита, равная 0010 в двоичном формате, обозначающая расширение воспроизведения видеопоследовательности.
- **Код_формата_видео (video_format)** — 3 бита, обозначающих формат видеоданных исходной (незакодированной) видеопоследовательности согласно табл. 14.8.
- **Описание_цветности (colour_description)** — 1 бит. Если **описание_цветности** = 1, то вслед за ним в битовом потоке указываются характеристики параметров кодирования и воспроизведения цветности.

Таблица 14.8. Коды формата видео

Код формата видео	Формат видео
000	компоненты (независимая передача сигналов яркости и цветности)
001	PAL
010	NTSC
011	SECAM
100	MAC
101	неспецифицированный
110	зарезервировано
111	зарезервировано

- **Первичные_цвета (colour_primaries)** — 8 битов, обозначающих компоненты цветности исходного изображения согласно табл. 14.9. По умолчанию значение **первичные_цвета** полагается равным 1.
- **Код_характеристики_передачи (transfer_characteristics)** — 8 битов, обозначающих оптоэлектронные характеристики передачи исходного изображения согласно табл. 14.10. По умолчанию значение **код_характеристики_передачи** полагается равным 1.

Таблица 14.9. Коды компонент цветности

Первичные_цвета	Компоненты цветности		
0000 0000	запрещено		
0000 0001	Рекомендация ITU-R 709 (1990)		
	Первичные	<i>x</i>	<i>y</i>
	зеленый	0,300	0,600
	синий	0,150	0,060
	красный	0,640	0,330
	белый D65	0,3127	0,3290
0000 0010	неспецифицированный видеоформат, неизвестные характеристики		
0000 0011	зарезервировано		
0000 0100	Рекомендация ITU-R 624-4 System M		
	Первичные	<i>x</i>	<i>y</i>
	зеленый	0,21	0,71
	синий	0,14	0,08
	красный	0,67	0,33
	белый D65	0,310	0,316
0000 0110	Рекомендация ITU-R 624-4 System B, G		
	Первичные	<i>x</i>	<i>y</i>
	зеленый	0,29	0,60
	синий	0,15	0,06
	красный	0,64	0,33
	белый D65	0,313	0,329
0000 0110	SMPTE 170M		
	Первичные	<i>x</i>	<i>y</i>
	зеленый	0,310	0,595
	синий	0,155	0,070
	красный	0,630	0,340
	белый D65	0,3127	0,3290

Таблица 14.9 (окончание)

Первичные цвета	Компоненты цветности		
0000 0111	SMPTE 240M (1987)		
	Первичные зеленый	x 0,310	y 0,595
	синий	0,155	0,070
	красный	0,630	0,340
	белый D65	0,3127	0,3291
0000 1000... 1111 1111	зарезервировано		

Таблица 14.10. Коды характеристики передачи

Код_характеристики_передачи	Характеристики передачи
0000 0000	запрещено
0000 0001	Рекомендация ITU-R 709 (1990) $V = 1,099L_c^{0,45} - 0,099$, для $1 \geq L_c \geq 0,018$ $V = 4,5L_c$, для $0,018 > L_c \geq 0$
0000 0010	неспецифицированный видеоформат, неизвестные характеристики
0000 0011	зарезервировано
0000 0100	Рекомендация ITU-R 624-4 System M, гамма дисплея полагается равной 2,2
0000 0110	Рекомендация ITU-R 624-4 System B, G, гамма дисплея полагается равной 2,8
0000 0110	SMPTE 170M $V = 1,099L_c^{0,45} - 0,099$, для $1 \geq L_c \geq 0,018$ $V = 4,5L_c$, для $0,018 > L_c \geq 0$
0000 0111	SMPTE 240M (1987) $V = 1,1115L_c^{0,45} - 0,1115$, для $L_c \geq 0,0228$ $V = 4,0L_c$, для $0,0228 > L_c$
0000 1000	линейная характеристика, $V = L_c$
0000 1001... 1111 1111	зарезервировано

- **Код_матрицы_преобразования (matrix_coefficients)** — 8 битов, обозначающих информацию о преобразовании сигналов цветности RGB в яркостную и цветоразностные компоненты. В табл. 14.11 E'_Y обозначает аналоговый сигнал яркости в пределах от 0 до 1, E'_{B-Y} и E'_{R-Y} — аналоговые цветоразностные сигналы в пределах от -0,5 до 0,5; они преобразуются в Y_D , C_R , C_B по следующим формулам:

$$Y_D = (219E'_Y) + 16;$$

$$C_B = (224E'_{B-Y}) + 128;$$

$$C_R = (224E'_{R-Y}) + 128.$$

По умолчанию значение **код_матрицы_преобразования** полагается равным 1.

- Горизонтальный_размер_воспроизведения (display_horizontal_size)
- Вертикальный_размер_воспроизведения (display_vertical_size) – по 14 битов, обозначающих предполагаемые размеры по горизонтали и по вертикали дисплея, на котором будет воспроизводиться изображение. Он может быть меньше, чем размер всего изображения, и тогда будет воспроизводиться только часть изображения, но может быть и больше, и в этом случае все изображение будет воспроизводиться в окне на дисплее.

Таблица 14.11. Коды матрицы преобразования цветности

Код_матрицы_преобразования	Формулы преобразования
0000 0000	запрещено
0000 0001	Рекомендация ITU-R 709 (1990) $E'_Y = 0,7154E'_G + 0,0721E'_B + 0,2125E'_R$ $E'_{B-Y} = -0,386E'_G + 0,500E'_B - 0,115E'_R$ $E'_{R-Y} = -0,454E'_G - 0,046E'_B + 0,500E'_R$
0000 0010	неспецифицированный видеоформат, неизвестные характеристики
0000 0011	зарезервировано
0000 0100	FCC $E'_Y = 0,59E'_G + 0,11E'_B + 0,30E'_R$ $E'_{B-Y} = -0,331E'_G + 0,500E'_B - 0,169E'_R$ $E'_{R-Y} = -0,421E'_G - 0,079E'_B + 0,500E'_R$
0000 0110	Рекомендация ITU-R 624-4 System B, G $E'_Y = 0,587E'_G + 0,114E'_B + 0,299E'_R$ $E'_{B-Y} = -0,331E'_G + 0,500E'_B - 0,169E'_R$ $E'_{R-Y} = -0,419E'_G - 0,081E'_B + 0,500E'_R$
0000 0110	SMPTE 170M $E'_Y = 0,587E'_G + 0,114E'_B + 0,299E'_R$ $E'_{B-Y} = -0,331E'_G + 0,500E'_B - 0,169E'_R$ $E'_{R-Y} = -0,419E'_G - 0,081E'_B + 0,500E'_R$
0000 0111	SMPTE 240M (1987) $E'_Y = 0,701E'_G + 0,087E'_B + 0,212E'_R$ $E'_{B-Y} = -0,384E'_G + 0,500E'_B - 0,116E'_R$ $E'_{R-Y} = -0,445E'_G - 0,055E'_B + 0,500E'_R$
0000 1000... 1111 1111	зарезервировано

Заголовок группы кадров имеет следующую структуру:

заголовок_группы_кадров()	Количество битов
{	
стартовый_код_группы	32
временной_код	25
замкнутая_группа	1

```

нарушение_связности      1
следующий_стартовый_код()
}

```

В него входят следующие переменные:

- **Стартовый_код_группы (group_start_code)** — константа длиной 32 бита, равная 000001B8 в 16-ричном формате, обозначающая начало группы кадров.
- **Временной_код (time_code)** — 25 битов, включающих в себя информацию о временных характеристиках и контрольных кодах видеоданных (табл. 14.12) согласно стандарту Международной электротехнической комиссии (IEC).

Таблица 14.12. Временные коды

Временной_код	Диапазон значений	Количество битов
флаг_пропущенного_кадра	0–1	1
код_часа	0–23	5
код_минут	0–59	6
маркер	1	1
код_секунд	0–59	6
временной_код_кадра	0–59	6

- **Замкнутая_группа (closed_gop)** — 1 бит, указывающий на возможность корректного декодирования данной группы кадров независимо от остальных. Если он равен 1, то группа начинается или с I-кадра, или с B-кадров, использующих только предсказание назад с I-кадром в качестве опорного.
- **Нарушение_связности (broken_link)** — 1 бит, который должен быть установлен в 0 при кодировании. Данный бит может быть установлен в 1 в процессе монтажа или редактирования кодированной последовательности, если группа начинается с B-кадров, использующих предсказание вперед с опорным кадром из предыдущей группы, который был удален из потока в процессе редактирования. Таким образом, декодер не будет воспроизводить эти B-кадры, так как они не могут быть корректно декодированы.

Заголовок кадра имеет следующую структуру:

заголовок_кадра()	Количество битов
{	
стартовый_код_кадра	32
временная_метка	10
код_типа_кадра	3
задержка_буфера	16
if (код_типа_кадра==2 код_типа_кадра==3) {	
целочисленный_вектор_движения_вперед	1
код_вектора_движения_вперед	3
}	

заголовок_кадра()	Количество битов
if (код_типа_кадра==3) {	
целочисленный_вектор_движения_назад	1
код_вектора_движения_назад	3
}	
while (следующие_биты() == 1) {	
дополнительный_бит_кадра /* со значением 1 */	1
дополнительные_данные_кадра	3
}	
дополнительный_бит_кадра /* со значением 0 */	1
следующий_стартовый_код()	
}	

В него входят следующие переменные:

- **Стартовый_код_кадра (picture_start_code)** — константа длиной 32 бита, равная 00000100 в 16-ричном формате, обозначающая начало кадра.
- **Временная_метка (temporal_reference)** — 10 битов, номер кадра в группе кадров в порядке воспроизведения. Первому кадру в каждой группе присваивается значение 0, второму — 1 и т. д., по модулю 1024. Оба поля кадра при передаче прогрессивной развертки должны иметь одно и то же значение переменной **временная_метка**.
- **Код_типа_кадра (picture_coding_type)** — 3 бита, обозначающих тип кадра согласно табл. 14.13.

Таблица 14.13. Коды типа кадра

Код_типа_кадра	Тип кадра
000	запрещено
001	I-кадр
010	P-кадр
011	B-кадр
100	запрещено в MPEG-2 (D-кадр в MPEG-1)
101...111	зарезервировано

- **Задержка_буфера (vbv_delay)** — 16 битов, обозначающих или переменный битовый поток (VBR — variable bitrate), тогда **задержка_буфера** = = FFFF в 16-ричном формате, или информацию о начальной задержке заполнения контрольного видеобуфера. Перед началом воспроизведения буфер декодера должен заполниться во избежание переполнения или опустошения. **Задержка_буфера** обозначает время, требуемое для заполнения контрольного видеобуфера от первоначально пустого состояния до того уровня, при котором текущий кадр будет удален из буфера.

Значение **задержка_буфера** измеряется в отсчетах 90-килогерцовых системных часов, в течение которых контрольный видеобуфер должен заполниться после получения последнего бита **стартовый_код_кадра**, и таким образом может быть вычислено по следующей формуле:

$$\text{задержка_буфера}_n = 90000 \times B_n / R,$$

где $n > 0$ — номер кадра, который будет удаляться из буфера; B_n — число битов в буфере, т. е. все биты, начиная с первого, следующего непосредственно за данными заголовков и поле **стартовый_код_кадра** и заканчивая последним битом данных кадра n ; R — точное значение битового потока, а не округленное, передаваемое в заголовке видеопоследовательности.

Следующие четыре параметра использовались в стандарте MPEG-1 и сохранены для обеспечения совместимости потока.

- **Целочисленный_вектор_движения_вперед (full_pel_forward_vector)** и **целочисленный_вектор_движения_назад (full_pel_backward_vector)** — по 1 биту. Если они равны 1, то в MPEG-1 служат признаками передачи векторов движения с точностью до целого пикселя, в противном случае вектора движения передаются с точностью до полпикселя. Эти два параметра должны быть равны 0 в MPEG-2.
- **Код_вектора_движения_вперед (forward_f_code)** и **код_вектора_движения_назад (backward_f_code)** — по 3 бита, в MPEG-1 служат для передачи диапазона векторов движения. Эти коды должны быть равны 111 в двоичном формате в MPEG-2.
- **Дополнительный_бит_кадра (extra_bit_picture)** — 1 бит. Если он равен 0, то за ним не следует никаких дополнительных данных; в противном случае за ним могут следовать дополнительные данные кадра (**extra_information_picture**), которые могут использоваться в будущих расширениях стандарта.

Расширение заголовка кадра имеет следующую структуру:

расширение_заголовка_кадра()	Количество битов
{	
стартовый_код_расширения	32
код_идентификатора_расширения	4
f_код[0][0]	4
f_код[0][1]	4
f_код[1][0]	4
f_код[1][1]	4
точность_нулевого_коэффициента_I	2
структурата_изображения	2
верхнее_поле_первым	1
покадровое_предсказание_и_ДКП	1
скрытые_векторы_движения	1
код_шкалы_квантования	1
формат_таблицы_кодов_I	1
альтернативное_переупорядочение	1
повтор_первого_поля	1
цветность_420	1
кадр_с_прогрессивной_разверткой	1
композитный_сигнал	1
if (композитный_сигнал) {	
v_ось	1

расширение_заголовка_кадра()	Количество битов
последовательность_полей	3
поднесущая	1
амплитуда_вспышки	7
фаза_поднесущей	8
}	
следующий_стартовый_код()	
}	

В него входят следующие переменные.

- **Стартовый_код_расширения** (*extension_start_code*) — константа длиной 32 бита, равная 000001B5 в шестнадцатиричном формате.
- **Код_идентификатора_расширения** (*extension_start_code_identifier*) — константа длиной 4 бита, равная 1000 в двоичном формате, обозначающая расширение заголовка кадра.
- *f_код[0][0]* (*f_code[0][0]*), *f_код[0][1]* (*f_code[0][1]*), *f_код[1][0]* (*f_code[1][0]*), *f_код[1][1]* (*f_code[1][1]*) — 4 элемента по 4 бита (от 0001 до 1001 в двоичном формате, значение 0000 запрещено), обозначающих максимальный диапазон векторов движения вперед по горизонтали, вперед по вертикали, назад по горизонтали, назад по вертикали соответственно. Если данный элемент потока не используется (например, в I-кадрах не используются все 4 элемента, а в P-кадрах — последние два), то значение его должно быть установлено 1111 в двоичном формате.
- **Точность_нулевого_коэффициента_I** (*intra_dc_precision*) — 2 бита, обозначающих точность передачи нулевого коэффициента ДКП согласно табл. 14.14.

Таблица 14.14. Коды точности передачи нулевого коэффициента ДКП

Точность_нулевого_коэффициента_I	Количество битов
00	8
01	9
10	10
11	11

- **Структура_изображения** (*picture_structure*) — 2 бита, означающих, какова структура данного кадра или поля согласно табл. 14.15.
- **Верхнее_поле_первым** (*top_field_first*) — 1 бит, смысл которого зависит от значений полей **структур_изображения, прогрессивная_развертка и повтор_первого_поля**.

Если **прогрессивная_развертка** = 0, то данный бит указывает, какое из двух полей для данного восстановленного кадра идет первым, верхнее или нижнее. Под верхним полем подразумевается первое поле под нижним — второе. Для поля данный бит должен быть равен 0. Значение данного бита в кадре рассматривается в пояснении параметра **повтор_первого_поля** (*repeat_first_field*).

Таблица 14.15. Коды структуры кадра или поля

Структура изображения	Структура данного кадра или поля
00	зарезервировано
01	верхнее поле
10	нижнее поле
11	кадр

- **Покадровое_предсказание_и_ДКП (frame_pred_frame_dct)** – 1 бит, обозначающий, какой тип ДКП и компенсации движения используется. Если он равен 1 (это должно быть так, если **кадр_с_прогрессивной_разверткой=1**), то используется предсказание только относительно кадра. Если передается поле, он должен быть равен 0.
- **Скрытое_вектора_движения (concealment_motion_vectors)** – 1 бит, обозначающий, передаются ли скрытые векторы движения вместе с I-блоками данного кадра или поля.
- **Код_шкалы_квантования (q_scale_type)** – 1 бит, обозначающий, какая шкала квантования используется при кодировании данного кадра или поля.
- **Формат_таблицы_кодов_I (intra_vlc_format)** – 1 бит, обозначающий, какая таблица кодов переменной длины используется при кодировании I-блоков данного кадра или поля.
- **Альтернативное_переупорядочение (alternate_scan)** – 1 бит, обозначающий, какая из двух возможных матриц используется при переупорядочивании коэффициентов ДКП для данного кадра или поля.
- **Повтор_первого_поля (repeat_first_field)** – 1 бит, который применяется только при передаче кадра. При передаче полей он должен быть равен 0 и не влияет на процесс декодирования. Таким образом, если **прогрессивная_развертка = 0** и **кадр_с_прогрессивной_разверткой = 0**, он должен быть равен 0, и на выходе декодера будут два поля.

Если **прогрессивная_развертка = 0** и **кадр_с_прогрессивной_разверткой = 1**, то когда данный бит равен 0, на выходе декодера будут два поля для данного восстановленного кадра – первое поле (верхнее или нижнее, в зависимости от значения **верхнее_поле_первым**), за которым следует второе; если же он равен 1, то на выходе декодера будут три поля: первое поле (верхнее или нижнее, в зависимости от значения **верхнее_поле_первым**), за которым следует второе, и затем повторенное первое.

Если **прогрессивная_развертка=1**, то когда данный бит равен 0, на выходе декодера будет один кадр; если же он равен 1, то на выходе декодера будут два или три кадра, в зависимости от значения **верхнее_поле_первым**.

- **Цветность_420 (chroma_420_type)** – 1 бит, который должен быть равен значению **кадр_с_прогрессивной_разверткой** для формата цветности 4 : 2 : 0. В противном случае он должен быть равен 0 и не влияет на процесс декодирования.
- **Кадр_с_прогрессивной_разверткой (progressive_frame)** – 1 бит, обозначающий являются ли два поля данного кадра независимыми, отстоящими друг от друга на интервал времени между полями (тогда он



должен быть равен 0 и **повтор_первого_поля** должен быть равен 0), или они принадлежат одному и тому же кадру (тогда он должен быть равен 1, поле **структура_изображения** должно быть равно 11 (кадр) и поле **покадровое_предсказание_и_ДКП** должно быть равно 1). Этот параметр используется также в синтаксисе потока с пространственной масштабируемостью. Он применяется в базовом уровне потока и влияет на формирование предсказания для расширенного уровня.

- **Композитный_сигнал (composite_display_flag)** — 1 бит, который указывает, присутствуют ли в заголовке следующие биты, которые используются, если исходное изображение перед кодированием представляло собой композитный сигнал. В этом случае он должен быть равен 1, в противном случае он должен быть равен 0.
- **v_ось (v_axis)** — 1 бит, который используется, если исходное изображение перед кодированием представляло собой ТВ-сигнал системы PAL. Он должен быть равен 1 при положительном знаке и 0 — в других случаях.
- **Последовательность_полей (field_sequence)** — 3 бита, обозначающих номер поля в последовательности восьми полей в системе PAL или четырех полей в системе NTSC согласно табл. 14.16.
- **Поднесущая (sub_carrier)** — 1 бит, который должен быть равен 0, если взаимосвязь между частотами поднесущей и строчной правильна. В противном случае он должен быть равен 1.
- **Амплитуда_вспышки (burst_amplitude)** — 7 битов, определяющих амплитуду вспышки для систем PAL или NTSC. Амплитуда вспышки поднесущей квантуется согласно Рекомендации ITU-R 601, с опусканием старшего бита, поскольку при оценке амплитуды вспышки он всегда постоянен.
- **Фаза_поднесущей (sub_carrier_phase)** — 7 битов, обозначающих фазу опорной (reference) поднесущей в данных синхронизации поля по отношению к началу поля, согласно Рекомендации ITU-R 470, как указано в табл. 14.17.

Таблица 14.16. Коды номера поля

Последовательность_поля	Кадр	Поле
000	1	1
001	1	2
010	2	3
011	2	4
100	3	5
101	3	6
110	4	7
111	4	8

Таблица 14.17. Коды фазы опорной поднесущей

Фаза_поднесущей	Фаза
0	$([360^\circ / 256] \cdot 0)$
1	$([360^\circ / 256] \cdot 1)$
...	...
255	$([360^\circ / 256] \cdot 255)$

Расширение матрицы квантования

В потоке MPEG-2 используются две матрицы квантования коэффициентов ДКП при формате цветности 4 : 2 : 0 и четыре — при формате цветности 4 : 2 : 2 и 4 : 4 : 4. При раскодировании заголовка видеопоследовательности все они устанавливаются согласно их значениям по умолчанию. Синтаксис потока предусматривает возможность загрузки матриц квантования, определяемых пользователем, либо в заголовке видеопоследовательности, либо в расширении матрицы квантования.

Расширение имеет следующую структуру:

расширение_матрицы_квантования	Количество битов
{	
код_идентификатора_расширения	4
загружаемая_матрица_I	1
if (загружаемая_матрица_I)	
матрица_I [64] 8 · 64	
загружаемая_матрица_PB	1
if (загружаемая_матрица_PB)	
матрица_PB [64] 8 · 64	
загружаемая_матрица_цветности_I	1
if (загружаемая_матрица_цветности_I)	
матрица_цветности_I [64] 8 · 64	
загружаемая_матрица_цветности_PB	1
if (загружаемая_матрица_цветности_PB)	
матрица_цветности_PB [64] 8 · 64	
следующий_стартовый_код()	
}	

В него входят следующие переменные.

- **Код_идентификатора_расширения** (`extension_start_code_identifier`) — константа длиной 4 бита, равная 0011 в двоичном формате, обозначающая расширение матрицы квантования.
- **Загружаемая_матрица_I** (`load_intra_quantizer_matrix`) — 1 бит. Если он равен 1, то за ним следуют 64 байта данных пользовательской матрицы квантования для I-блоков как яркости, так и цветности. Первый байт матрицы квантования всегда должен быть равен 8, так как он используется для кодирования нулевого коэффициента ДКП.
- **Загружаемая_матрица_PB** (`load_non_intra_quantizer_matrix`) — 1 бит. Если он равен 1, то за ним следуют 64 байта данных пользовательской матрицы квантования для P- и B-блоков как яркости, так и цветности.
- **Загружаемая_матрица_цветности_I** (`load_chroma_intra_quantizer_matrix`) — 1 бит. Если он равен 1, то за ним следуют 64 байта данных пользовательской матрицы квантования для I-блоков только цветности в форматах 4 : 2 : 2 и 4 : 4 : 4. Для формата цветности 4 : 2 : 0 он должен быть равен 0. Первый байт матрицы квантования должен быть равен 8.
- **Загружаемая_матрица_цветности_PB** (`load_chroma_non_intra_quantizer_matrix`) — 1 бит. Если он равен 1, то за ним следуют 64 байта данных пользовательской матрицы квантования для P- и B-блоков цветности.

Расширение воспроизведения кадра

Информация в данном разделе также не влияет на процесс декодирования, но может использоваться устройствами воспроизведения для смещения воспроизводимого фрагмента изображения, задаваемого в расширении воспроизведения видеопоследовательности, от кадра к кадру. Расширение воспроизведения кадра не может присутствовать в потоке, если отсутствует расширение воспроизведения видеопоследовательности.

Раздел имеет следующую структуру:

расширение воспроизведения кадра	Количество битов
{	
код_идентификатора_расширения	4
for (<i>i</i> = 0; <i>i</i> < смещение_центра_изображения;	
<i>i</i> ++) {	
смещение_центра_изображения_по_гори-	
зонтали	16
маркер	1
смещение_центра_изображения_по_верти-	
кали	16
маркер	1
}	
следующий_стартовый_код()	
}	

В него входят следующие переменные.

- Код_идентификатора_расширения (*extension_start_code_identifier*) – константа длиной 4 бита, равная 0111 в двоичном формате, обозначающая расширение воспроизведения кадра.
- Смещение_центра_изображения_по_горизонтали (*frame_centre_horizontal_offset*) и Смещение_центра_изображения_по_вертикали (*frame_centre_vertical_offset*) – по 16 битов, задающих соответственно горизонтальное и вертикальное смещение воспроизводимого фрагмента изображения в единицах 1/16 отсчета вправо и вниз от центра фрагмента.

Параметр смещение_центра_изображения (*number_of_frame_center_offsets*) равен 1, если прогрессивная_развертка=1 или структура_изображения=«поле», в противном случае он равен 3, если повтор_первого_поля=1, или 2.

Слайсы, макроБлоки и блоки

В некоторых сочетаниях профиля и уровня структура разбиения на слайсы задается жестко, и не допускается пропусков между ними.

В слайсе кодируются данные кадра следующим образом:

данные_кадра()	Количество битов
{	
do {	
слайс()	
} while (следующие_биты() == стартовый_код_слайса)	
следующий_стартовый_код()	
}	
слайс()	Количество битов
{	
стартовый_код_слайса	32
if (размер_по_вертикали > 2800)	
расширение_вертикального_положения_слайса	3
if («признак масштабируемого потока присутствует в потоке»)	
if (масштабируемость == «разделение данных»)	
признак_приоритета	7
код_уровня_квантования	5
if (следующие_биты() == «1») {	
флаг_I_слайса	1
I_слайс	1
зарезервированные_биты	7
while (следующие_биты() == 1) {	
дополнительный_бит_слайса	
/* со значением 1 */	1
дополнительные_данные_слайса	8
}	
}	
дополнительный_бит_слайса /* со значением 0 */	1
do {	
макроблок()	
} while (следующие_биты() != «000 0000 0000 0000 0000 0000»)	
следующий_стартовый_код()	
}	
макроблок()	Количество битов
{	
while (следующие_биты() == «0000 0001 000»)	
escape_код_макроблока	11
код_приращения_адреса_макроблока	1-11
типы_макроблока()	
if (макроблок_с_изменением_уровня_квантования)	

```

    код_уровня_квантования      5
    if (макроблок_с_движением_вперед ||
        (макроблок_I && скрытые_векторы_
        векторы_движения(0))
    if (макроблок_с_движением_назад)
        векторы_движения(1)
    if (макроблок_I && скрытые_векторы_
        движения))
        маркер      1
    if (макроблок_кодируется)
        кодируемые_блоки()
    for (i = 0; i < число_блоков; i++) {
        блок(i)
    }

```

типы_макроблока()	Количество битов
{	
тип_макроблока	1-9
инкремент_адреса_макроблока	1-11
if ((флаг_пространственно_временного_веса == 1) &&	
(код_флага_пространственно_временного_веса != «00»)) {	
код_пространственно_временного_веса	2
}	
if (макроблок_с_движением_вперед	
макроблок_с_движением_назад) {	
if (структура_изображения == «кадр») {	
if (покадровое_предсказание_и_ДКП == 0)	
движение_по_кадрам	2
} else {	
движение_по_полям	2
}	
if (декодировать_тип_ДКП) {	
тип_ДКП	1
}	

векторы_движения()	Количество битов
{	
if (счетчик_векторов_движения == 1) {	
if ((формат_векторов_движения == «поле») && (признак_dmv != 1))	
формат_векторов_движения[0][s]	1
движение_вертикального_поля(0,s)	
} else {	
селектор_вертикального_поля[0][s]	1
вектор_движения(0,s)	
селектор_вертикального_поля[1][s]	1
вектор_движения(1,s)	
}	
}	

вектор_движения(r,s)	Количество битов
{	
код_движения[r][s][0]	1-11
if ((f_код[s][0] != 1) && (код_движения[r][s][0] != 0))	
остаток_движения[r][s][0]	1-8
if (признак_dmv == 1)	
вектор_движения_dm[0]	1-2
код_движения[r][s][1]	1-11
if ((f_код[s][0] != 1) && (код_движения[r][s][1] != 0))	
остаток_движения[r][s][1]	1-8
if (признак_dmv == 1)	
вектор_движения_dm[1]	1-2
}	

кодируемые_блоки()	Количество битов
{	
кодируемые_блоки420	3-9
if (формат_цветности == 4 : 2 : 2)	
кодируемые_блоки1	2
if (формат_цветности == 4 : 4 : 4)	
кодируемые_блоки2	6
}	

блок()	Количество битов
{	
if (код_блока[i]) {	
if (макроблок_I) {	
if (i < 4) {	

```

размер_нулевого_коэффициента_яркости      2-9
if (размер_нулевого_коэффициента_
    яркости != 0)
    разностный_сигнал
} else {
    размер_нулевого_коэффициента_цветности 2-10
    if (размер_нулевого_коэффициента_
        _цветности != 0)
        разностный_сигнал
    }
} else {
    нулевой коэффициент ДКП
}
while (следующие_биты() != КОНЕЦ_БЛОКА)
    последующие коэффициенты ДКП
КОНЕЦ_БЛОКА
}
}

```

В *уровень слайсов* входят следующие переменные.

- **Стартовый_код_слайса** (*slice_start_code*) — 32 бита, из которых первые 24 бита — это константа, равная 000001 в 16-ричном формате, обозначающая начало слайса, а последние 8 битов — **вертикальное_положение_слайса** (*slice_vertical_position*) — обозначают положение слайса по вертикали, выраженное в макроблоках, в пределах от 1 до 175. Таким образом, первый слайс имеет адрес 01. Слайс не может выходить за пределы одного ряда макроблоков, но в одном ряду может располагаться несколько слайсов. Если число строк в изображении превышает 2800, то используется следующий параметр —
- **Расширение_вертикального_положения_слайса** (*slice_vertical_position_extension*), 3 бита. Параметр **вертикальное_положение_слайса** не может быть больше 128, и положение слайса — параметр **номер_ряда** (*mb_row*) вычисляется по следующей формуле:


```

if (размер_по_вертикали > 2800)
    номер_ряда = (расширение_вертикального_положения_
        слайса << 7) + вертикальное_положение_слайса - 1;
else
    номер_ряда = вертикальное_положение_слайса - 1.

```
- **Признак_приоритета** (*priority_breakpoint*) — 7 битов, обозначающих место разделения данных в потоке с использованием разделения данных. В базовом потоке этот параметр должен быть равен 0.
- **Код_уровня_квантования** (*quantizer_scale_code*) — 5 битов, обозначающих коэффициент квантования из основной или альтернативной таблицы коэффициентов квантования, согласно параметру **код_шкалы_квантования**, который используется до тех пор, пока не встретится новый **код_уровня_квантования** на уровне слайса или макроблока.

- **Флаг_I_слайса (intra_slice_flag)** — 1 бит. Если он равен 1, то в потоке присутствуют следующие два параметра: **I_слайс (intra_slice)** и **зарезервированные_биты (reserved_bits)**. Этот параметр может быть установлен 0 без ущерба для декодирования.
- **I_слайс (intra_slice)** — 1 бит. Если он равен 1, то все макроблоки слайса должны быть I-блоками. Этот параметр может быть опущен из потока без ущерба для декодирования, он предназначен для реализации быстрого просмотра в цифровых системах хранения данных.
- **Зарезервированные_биты (reserved_bits)** — 7 битов, которые должны быть равны 0.
- **Дополнительный_бит_слайса (extra_bit_slice)** — 1 бит. Если он равен 0, то за ним не следует никаких дополнительных данных; в противном случае за ним следуют
- **Дополнительные_данные_слайса (extra_information_slice)**, которые могут использоваться в будущих расширениях стандарта.

На **уровне макроблоков** передаются все данные макроблоков. Макроблок содержит область из четырех блоков яркости размером 8×8 пикселов и пространственно соответствующих ей областей компонент цветности. Термин «макроблок» может относиться или к источнику и декодированным данным, или к соответствующим кодированным элементам данных. В соответствии с форматом цветности макроблоков содержит 6 ($4 : 2 : 0$), 8 ($4 : 2 : 2$) или 12 ($4 : 4 : 4$) блоков.

Макроблоки могут быть разного типа в зависимости от типа кадра. В I-кадрах все макроблоки внутрикодированные. В P-кадрах макроблоки могут быть пропущенными, внутрикодированными, кодированными с предсказанием вперед. В B-кадрах кроме типов макроблоков, встречающихся в P-кадрах, передаются макроблоки, кодированные с двунаправленным предсказанием и кодированные с предсказанием назад. Пропущенный макроблок — это макроблок, для которого не передается никакой информации. В стандарте MPEG-2 определены следующие типы макроблоков в I-, P-, и B-кадрах, показанные в табл. 14.18–4.20.

Таблица 14.18. Типы макроблоков и их коды для I-кадра

Тип	Код	Тип	Код
<i>intra-d</i>	1	<i>intra-q</i>	01

Таблица 14.19. Типы макроблоков и их коды для P-кадра

Тип	Код	Тип	Код
<i>pred-mc</i>	1	<i>pred-mcq</i>	0001 0
<i>pred-c</i>	01	<i>pred-cq</i>	0000 1
<i>pred-m</i>	001	<i>intra-q</i>	0000 01
<i>intra-d</i>	0001 1	<i>skipped</i>	нет кода

Аббревиатуры типов макроблока расшифровываются следующим образом:

- *intra* — I-макроблок (устанавливает значение флага **макроблок_I = 1**);
- *d* — используется коэффициент квантования по умолчанию (устанавливает значение флага **макроблок_c_изменением_уровня_квантования = 0**);
- *q* — изменяемый коэффициент квантования (устанавливает значение флага **макроблок_c_изменением_уровня_квантования = 1**);

Таблица 14.20. Типы макроблоков и их коды для В-кадра

Тип	Код	Тип	Код
<i>pred-i</i>	10	<i>intra-d</i>	0001 1
<i>pred-ic</i>	11	<i>pred-icq</i>	0001 0
<i>pred-b</i>	010	<i>pred-bcq</i>	0000 10
<i>pred-bc</i>	011	<i>pred-fcq</i>	0000 11
<i>pred-f</i>	0010	<i>intra-q</i>	0000 01
<i>pred-fc</i>	0011	<i>skipped</i>	нет кода

- *pred* — предсказание движения (не I-блок) (устанавливает значение флага **макроблок_I=0**);
- *t* — компенсация движения с передачей вектора движения (устанавливает значение флага **макроблок_c_движением_вперед (macroblock_motion_forward) = 1**);
- *c* — кодирование и передача как минимум одного блока в макроблоке (устанавливает значение флага **макроблок_кодируется = 1**);
- *i* — блок с двунаправленным предсказанием движения (interpolated) (устанавливает значение флагов **макроблок_c_движением_вперед=1** и **макроблок_c_движением_назад=1**);
- *f* — предсказание движения вперед (forward) (устанавливает значение флага **макроблок_c_движением_вперед = 1**);
- *b* — предсказание движения назад (backward) (устанавливает значение флага **макроблок_c_движением_назад = 1**);
- *skipped* — пропущенный макроблок (для которого не передается никакой код).

Пропущенные макроблоки в В-кадрах имеют тот же вектор движения и тот же тип макроблока, что и предыдущий (базовый) макроблок, в то время как в Р-кадрах вектор движения пропущенного макроблока равен нулю.

В элемент видеопотока **макроблок()** входят следующие переменные.

- **Escape_код_макроблока (macroblock_escape)** — константа из 11 битов со значением «00000 0001000», которая используется, если адрес текущего макроблока превышает адрес предыдущего макроблока более чем на 33. Если, например, было пропущено более 66 макроблоков, то адрес макроблока кодируется с помощью двух кодов **escape_код_макроблока** и кода **инкремент_адреса_макроблока**. Элемент видеопотока MPEG-1 **заполнение_макроблока (macroblock_stuffing)** не используется в MPEG-2.
- **Инкремент_адреса_макроблока (macroblock_address_increment)** — код переменной длины, показывающий разность между адресом текущего макроблока и адресом предыдущего непропущенного макроблока, параметр **инкремент_адреса_макроблока** не может задавать приращение адреса больше 33, которое кодируется с помощью кода **escape_код_макроблока**. Первый макроблок в кадре имеет адрес 0.

Для пропущенных макроблоков не передается никакого кода, и они обозначаются соответствующим приращением адреса макроблока. Первый и последний макроблоки слайса не могут быть пропущенными. В В-кадрах пропущенный макроблок не может следовать сразу за I-макроблоком.

В элемент видеопотока **типы_макроблока()** (`macroblock_modes()`) входят нижеследующие переменные.

- **Тип_макроблока (macroblock_type)** — код переменной длины, показывающий тип текущего макроблока.
- **Макроблок_c_изменением_уровня_квантования (macroblock_quant)**,
- **Макроблок_c_движением_вперед (macroblock_motion_forward)**,
- **Макроблок_c_движением_назад (macroblock_motion_backward)**,
- **Макроблок_кодируется (macroblock_pattern)**,
- **Макроблок_I (macroblock_intra)** — однобитовые флаги признаков макроблока, извлекаемые из **типа_макроблока**, как описано выше, и влияющие на процесс декодирования.
- **Флаг_пространственно_временного_веса (spatial_temporal_weight_code_flag)** — 1 бит, флаг признаков макроблока, извлекаемый из **типа_макроблока** и используемый в потоках с пространственной масштабируемостью.
- **Движение_по_кадрам (frame_motion_type)** — 2 бита, обозначающих предсказание движения в макроблоке. Если **покадровое_предсказание_и_ДКП=1** или **скрытые_вектора_движения=1**, то данный элемент опускается из потока, и предсказание движения осуществляется по умолчанию по кадру (табл. 14.21).

Таблица 14.21. Коды параметра **движение_по_кадрам**

Код	Класс_пространственно_временного_веса	Тип предсказания	Счетчик_векторов_движения	Формат_векторов_движения	Признак_dmv
00		зарезервировано			
01	0, 1	по полям	2	поле	0
01	2, 3	по полям	1	поле	0
10	0, 1, 2, 3	по кадрам	1	кадр	0
11	0, 2, 3	dual-prime	1	поле	1

Движение_по_полям (field_motion_type) — 2 бита, обозначающих предсказание движения в макроблоке. Для I-макроблока если **скрытые_вектора_движения=1**, то данный элемент опускается из потока, и предсказание движения осуществляется по умолчанию по полям (табл. 14.22).

Таблица 14.22. Коды параметра **движение_по_полям**

Код	Класс_пространственно_временного_веса	Тип предсказания	Счетчик_векторов_движения	Формат_векторов_движения	Признак_dmv
00		зарезервировано			
01	0, 1	по полям	1	поле	0
10	0, 1	16 × 8	2	поле	0
11	0	dual-prime	1	поле	1

Декодировать_тип_ДКП (decode_dct_type) – 1 бит, определяющий наличие следующего элемента тип_ДКП (dct_type) в потоке и значение которого зависит от нескольких элементов потока:

```

if ( (структура_изображения == «кадр» ) &&
    (покадровое_предсказание_и_ДКП == 0) &&
    (макроблок_I || макроблок_кодируется) )
    декодировать_тип_ДКП = 1;
else
    декодировать_тип_ДКП = 0.

```

- Тип_ДКП (dct_type) – 1 бит, указывающий, кодирован ли макроблок с ДКП по кадру или полю. Если тип_ДКП=1, то макроблок кодирован по полю. Тип_ДКП включен в битовый поток, только если декодированный тип_ДКП не равен нулю. Если декодировать_тип_ДКП=0, то тип_ДКП (используемый в дальнейшем процессе декодирования) должен определяться согласно табл. 14.23.

Таблица 14.23. Значение тип_ДКП, если тип_ДКП не включен в битовый поток

Условие	Тип_ДКП
структура_изображения == «поле»	не используется, потому что в поле не может быть различия кадр/поле
покадровое_предсказание_и_ДКП == 1	0 («кадр»)
!(макроблок_I макроблок_кодируется)	не используется — макроблок некодированный
макроблок пропущен	не используется — макроблок некодированный

В элемент видеопотока вектора_движения(s) (motion_vectors(s)) входят следующие переменные.

- Счетчик_векторов_движения (motion_vector_count), формат_векторов_движения (mv_format), признак_dmv (dmv) получаются из движение_по_полям или движение_по_кадрам, как показано в таблицах 14.21 и 14.22.
- Селектор_вертикального_поля[r][s] (motion_vertical_field_select [r][s]) – 1 бит, который указывает, какое из полей должно использоваться для формирования предсказания. Если он равен 0, то должно использоваться верхнее поле, если 1 – то нижнее.

В элемент видеопотока вектор_движения(r,s) (motion_vector(r,s)) входят следующие переменные.

- Код_движения[r][s] (motion_code[r][s]),
- Остаток_движения[r][s] (motion_residual[r][s]),
- Вектор_движения_dm (dmvector[t]).

Они передаются в виде кодов переменной длины, по которым восстанавливаются векторы движения.

В элемент видеопотока кодируемые_блоки() (coded_block_pattern()) входят следующие переменные.

- **Кодируемые_блоки420 (coded_block_pattern_420)** — код переменной длины, по которому определяются блоки, подлежащие декодированию. Поскольку некоторые блоки имеют только нулевые коэффициенты, их не надо кодировать, и соответствующий бит в параметре **cbp** устанавливается в ноль. Например, если кодируются первый и пятый блоки, **cbp**=100010 в двоичном формате. Таким образом, если **код_блока[i] (pattern_code[i])** равен 1, где *i* может меняться от 0 до (**количество_блоков** - 1), то коды коэффициентов ДКП для данного блока с номером *i* содержатся в данном макроблоке. Число **количество_блоков**, которое определяет число блоков в макроблоке, определяется из формата цветности и равно 6 для формата 4 : 2 : 0, 8 — для 4 : 2 : 2 и 12 — для 4 : 4 : 4.
- **Кодируемые_блоки1 (coded_block_pattern_1)** и
- **Кодируемые_блоки2 (coded_block_pattern_2)** — 2 и 6 битов соответственно, которые используются для форматов цветности 4 : 2 : 2 и 4 : 4 : 4.

14.5. MPEG-4. Обобщенное кодирование аудиовизуальных объектов

Стандарт MPEG-4 [5.44–5.47] можно рассматривать как совершенствование стандарта MPEG-2 по двум направлениям: обеспечение широкого набора инструментов при работе с оцифрованным видеоматериалом, что расширило область применения стандарта, и реализация более эффективных алгоритмов компрессии, особенно в части обработки динамических изображений, принципы которых изложены в разделе MPEG-4, Часть 2 — Видео [5.45].

MPEG-4 определяет стандартизированные технологические элементы, позволяющие интегрировать идеологию создания, распространения и доступа к содержанию в цифровом телевидении, интерактивных графических приложениях и интерактивных мультимедийных службах [5.65–5.67]. Этот стандарт обеспечивает:

- представление единиц звукового, визуального или аудиовизуального контента, так называемых медийных объектов; эти медийные объекты могут иметь естественное или искусственное происхождение, т. е. они могут быть записаны на камеру или магнитофон или сгенерированы на компьютере;
- описание композиции этих объектов для создания составных медийных объектов, которые образуют аудиовизуальную сцену;
- мультиплексирование и синхронизацию данных, связанных с медийными объектами, для передачи по сети, обеспечивающей уровень качества услуг (QOS), соответствующий природе медийных объектов;
- взаимодействие с аудиовизуальной сценой, созданной на приемном конце.

14.5.1. Кодированное представление медийных объектов

Аудиовизуальная сцена MPEG-4 состоит из нескольких медийных объектов, организованных иерархически. Примитивные медийные объекты — это неподвижные изображения (например, фон), видеообъекты (например, разговаривающий человек), звуковые объекты (например, голос этого человека, фоновая музыка).

MPEG-4 стандартизирует ряд таких примитивных медийных объектов, которые могут представлять как естественные, так и искусственные типы контента, двух- или трехмерные. Кроме вышеперечисленного MPEG-4 определяет кодированное представление таких объектов, как текст и графика, искусственные говорящие головы и связанный с ними текст, используемый для речи и анимации головы, синтетический звук.

Медийный объект в кодированной форме состоит из элементов описания, которые позволяют работать с объектом в аудиовизуальной сцене, и из соответствующих потоковых данных (при необходимости). Важно отметить, что в кодированной форме каждый медийный объект может быть представлен независимо от окружения или фона.

Кодированное представление медийных объектов настолько эффективно, насколько это возможно с учетом заданной функциональности. Примеры такой функциональности — надежная защита от ошибок, простота извлечения и редактирования объекта, масштабируемое представление объекта, ограничение скорости потока данных.

MPEG-4 предполагает стандартизованный путь описания сцен, позволяющий:

- группировать примитивные медиаобъекты для образования составных медиаобъектов;
- перемещать примитивные и составные медиаобъекты в заданной координатной системе;
- применять преобразования геометрического или акустического вида медиаобъекта;
- использовать потоки данных, чтобы видоизменять атрибуты медиаобъектов (например, звук, движущуюся текстуру, принадлежащую объекту, параметры анимации, управляющие синтетическим лицом);
- интерактивно изменять точку присутствия пользователя на сцене (его точку наблюдения и прослушивания).

Описание сцены строится во многих отношениях так же, как и в языке моделирования виртуальной реальности VRML (Virtual Reality Modeling language). На рис. 14.22 приведена в качестве примера сцена, реализуемая с применением стандарта MPEG-4 [5.65].

Информация о медиаобъектах преобразуется в один или несколько элементарных потоков. Эти потоки соответствующим образом идентифицируются, что позволяет иерархически обрабатывать кодированные данные.

Синхронизация элементарных потоков осуществляется за счет временных меток блоков данных, что независимо от типа среды передачи позволяет идентифицировать тип модуля доступа (например, видео- или аудио-кадры, команды

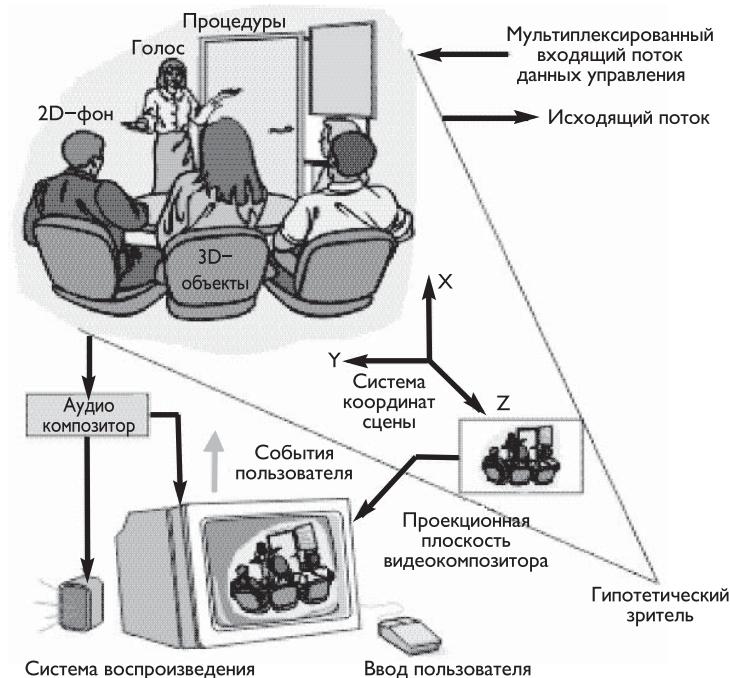


Рис. 14.22. Пример сцены, реализуемой стандартом MPEG-4

описания сцены) в элементарных потоках, восстанавливать временную базу медиаобъекта или описания сцены, и осуществлять их синхронизацию.

Синхронизированная доставка потока данных отправителя получателю специфицирована в терминах слоя синхронизации и доставки, которые содержат двухслойный мультиплексор (рис. 14.23) [5.65].

Первый слой мультиплексирования управляется согласно спецификации DMIF (Delivery Multimedia Integration Framework). Это мультиплексирование может быть реализовано определенным в MPEG мультиплексором FlexMux, который позволяет группировать элементарные потоки ES (Elementary Streams) с низкой избыточностью. Мультиплексирование на этом уровне может использоваться, например, для группирования ES с подобными требованиями по качеству передачи, чтобы уменьшить число сетевых соединений или значения задержек.

Слой «TransMux» (Transport Multiplexing) на рис. 14.23 моделирует уровень транспортных услуг, удовлетворяющих заданным требованиям.

MPEG-4 специфицирует только интерфейс этого слоя, в то время как остальные требования к пакетам данных будут определяться транспортным протоколом. Любой существующий стек транспортных протоколов, например, RTP/UDP/IP, AAL5/ATM, или Транспортный поток MPEG-2 поверх подходящего канального уровня может стать частным случаем TransMux. Выбор оставлен за конечным пользователем или сервис-провайдером, и позволяет использовать MPEG-4 с широким спектром операционного окружения.

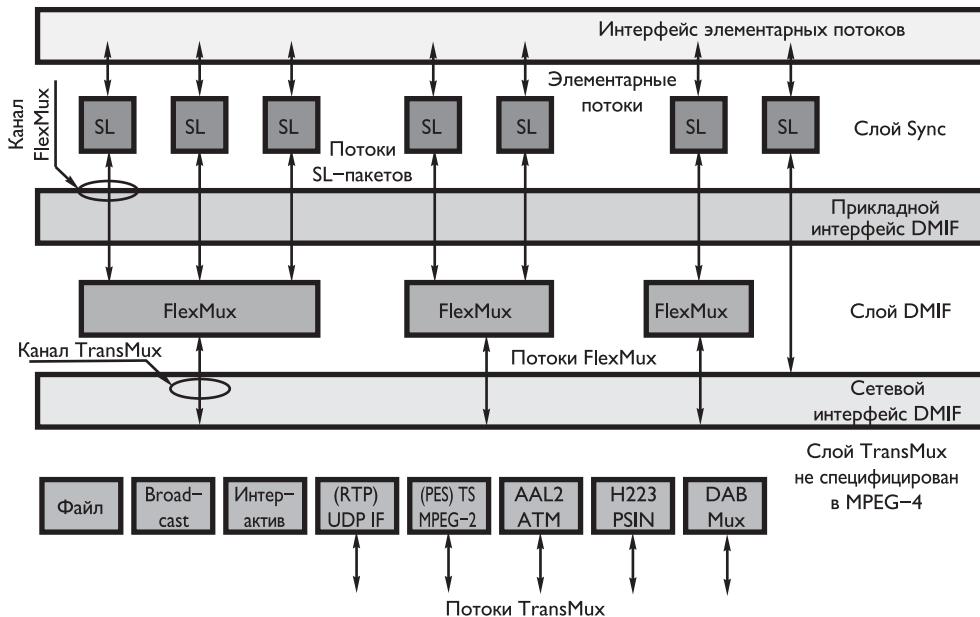


Рис. 14.23. Модель системного слоя MPEG-4

Использование мультиплексора FlexMux является опциональным, и этот слой может быть пустым, если нижележащий слой TransMux предоставляет все необходимые функции. Слой синхронизации, однако, присутствует всегда.

Потоки данных (Elementary Streams, ES), создаваемые в результате кодирования различных видов информации, могут быть переданы или запомнены независимо. Они должны быть объединены так, чтобы на принимающей стороне возникла реальная мультимедийная презентация.

Системные части MPEG-4 определяют взаимодействие между аудио и видео-компонентами, которые образуют сцену.

14.5.2. Аудиообъекты

MPEG-4/Аудио обеспечивает реализацию широкого перечня приложений — от речевой информации до высококачественного многоканального звука, и от естественных до синтетических звуков. В частности, он поддерживает высокоеффективную презентацию аудиообъектов, содержащих:

- кодированную *речь* при скоростях обмена от 2 до 24 кбита/с;
- *синтезированную речь* при масштабируемой скорости в диапазоне от 200 битов/с до 1,2 кбита/с; кодировщики позволяют использовать текст или текст с интонационными параметрами (вариация тона, длительность фонемы и т. д.) в качестве входных данных для генерации синтетической речи;
- общую кодировку звуковых потоков от низких скоростей с относительно низким качеством до высококачественных: рабочий диапазон начинается от 6 кбитов/с (при полосе звукового сигнала ниже 4 кГц) и распространяет-

ся до широковещательного качества передачи звукового сигнала для моно-, стерео- и многоканальных приложений;

- поддержку *синтезированного звука* с применением декодера структурированного звука (Structured Audio Decoder), который позволяет использовать управление музыкальными инструментами с привлечением специального языка описания;
- реализацию *синтетического звука* с ограниченной сложностью с использованием специального структурируемого аудиодекодера.

Дополнительной функциональностью является возможность управления скоростью обмена и масштабируемость в отношении потоков данных, полосы пропускания, вероятности ошибок, сложности. Это дает новые возможности, такие как:

- *изменение временного масштаба* без изменения шага при выполнении процесса декодирования, что может быть использовано, например, для реализации функции «быстрая перемотка вперед» (поиск в базе данных) или для адаптации длины аудиопоследовательности до заданного значения и т. д.;
- *изменение шага* без изменения временного масштаба в процессе кодирования или декодирования, что создает, например, возможность для изменения голоса или для приложений типа караоке;
- *изменение скорости передачи* допускает анализ потока данных с разбивкой на субпотоки меньшей скорости, которые могут быть декодированы в осмысленный сигнал; анализ потока данных может осуществляться при передаче или в декодере;
- *масштабируемость полосы пропускания* как частный случай масштабируемости скорости передачи данных, когда часть потока данных, представляющая часть частотного спектра, может быть отброшена при передаче или декодировании;
- *масштабируемость сложности кодировщика* позволяет кодировщикам различной сложности генерировать корректные и осмысленные потоки данных;
- *масштабируемость сложности декодера* дает возможность заданную скорость потока данных дешифровать посредством декодеров с различным уровнем сложности;
- *реализация аудиоэффектов*, которые создают возможность обрабатывать декодированные аудиосигналы с сохранением полной временной точности с целью достижения эффектов смешения, реверберации, создания объемного звучания и т. д.

В MPEG-4 определены **восемь аудиопрофилей**, список и характерные черты которых приведены в табл. 14.24.

14.5.3. Видеообъекты

Стандарт MPEG-4 предусматривает гибридное кодирование естественных изображений (состоящих из пикселов), видео и искусственных сцен (созданных на компьютере). Это делает возможным, например, виртуальное присутствие участников видеоконференции. Стандарт включает инструменты и алгоритмы, обеспечивающие кодирование естественных неподвижных изображений и видеопоследовательностей, а также инструментарий для сжатия геометрических параметров

Таблица 14.24. Профили MPEG-4/Аудио

№	Аудио профили	Специфические особенности
1.	<i>Разговорный профиль</i>	Использует параметрическое кодирование голоса HVXC (Harmonic Vector Excitation Coding), рассчитанное на низкие скорости передачи, либо кодер CELP (Code Excited Linear Prediction) — узкополосный кодер голоса с линейным предсказанием, либо интерфейс текст-голос.
2.	<i>Профиль синтеза</i>	Обеспечивает синтез аудио, использующий SAOL (Structured Audio Orchestra Language), волновые таблицы и интерфейс текст-голос для генерации звука и речи при очень низких скоростях передачи.
3.	<i>Масштабируемый профиль</i>	Удобен для масштабируемого кодирования речи и музыки для таких сетей, как Интернет и NADIB (Narrowband Audio DIgital Broadcasting). Диапазон скоростей передачи лежит в пределах от 6 до 24 кбита/с при ширине полосы 3,5 и 9 кГц.
4.	<i>Главный профиль</i>	Является расширенным набором всех других профилей, содержит средства для синтетического и естественного аудио.
5.	<i>Профиль высококачественного аудио</i>	Содержит кодировщик голоса CELP и простой кодировщик AAC (Advanced Audio Coding). Масштабируемое кодирование может быть выполнено с помощью AAC масштабируемого объектного типа. Опционно может использоваться синтаксис потока ER (Error Resilient), устойчивый к ошибкам.
6.	<i>Профиль аудио с малой задержкой (Low Delay Audio)</i>	Содержит HVXC и CELP кодировщики голоса (опционно использующие синтаксис ER), AAC-кодеры с низкой задержкой и интерфейс текст-голос TTSI (Text to Speech Interface).
7.	<i>Профиль натурального аудио</i>	Содержит все средства кодирования натурального аудио, доступные в MPEG-4.
8.	<i>Профиль межсетевого мобильного аудио (Mobile Audio Internetworking)</i>	Содержит AAC масштабируемые объектные типы с малой задержкой, включая TwinVQ (Transform-domain weighted interleaved Vector Quantization) и BSAC (Bit Sliced Arithmetic Coding). Этот профиль предназначен для расширения телекоммуникационных приложений за счет не-MPEG алгоритмов кодирования речи с возможностями высококачественного аудиокодирования.

искусственных двумерной и трехмерной графики (например, сжатие параметров сетки, синтетического текста).

Стандарт MPEG-4 Видео поддерживает: типичные *скорости битового потока* от 5 кбитов/с до более чем 1 Гбит/сек; прогрессивную и чересстрочную развертку видео; *типичные разрешения* от sub-QCIF (128×96) до «студийного» (4000×4000).

Для всех битовых скоростей алгоритм компрессии очень эффективен. Это включает компактное кодирование текстур с качеством, которое регулируется в пределах от «приемлемого» для очень высокой степени сжатия до «почти без потерь». Обеспечивается эффективное сжатие текстур для отображения на двумерных и трехмерных решетках. Также обеспечивается произвольный доступ к видеинформации для обеспечения таких функций для сохраненного видео, как приостановка, быстрая перемотка вперед и назад.

Стандарт предоставляет возможность контекстно-ориентированного кодирования изображений и видео, что позволяет раздельно декодировать и реконструи-

ировать произвольные формы видеообъектов. Расширенная манипуляция содержанием видеопоследовательностей позволяет использовать такие возможности, как деформация синтетических и естественных объектов, текстур, изображений в восстановленном контенте видео.

Стандарт поддерживает различные виды масштабируемости текстур, изображений и видео:

- *масштабируемость сложности на кодере* позволяет кодерам разной производительности создавать правильные битовые потоки для заданных текстур, изображений и видео;
- *масштабируемость сложности на декодере* позволяет декодировать потоки, содержащие текстуры, изображения и видео, декодерами разной степени сложности (качество восстановленных объектов, вообще говоря, связано со сложностью используемого декодера — менее мощные декодеры декодируют только часть битового потока);
- *пространственная масштабируемость* позволяет декодировать часть полного битового потока, генерируемого кодером, для восстановления и показа текстур, изображений и видеообъектов при уменьшенном пространственном разрешении (поддерживается до 11 уровней пространственной масштабируемости);
- *временная масштабируемость* позволяет декодеру декодировать часть полного битового потока, генерируемого кодером, для восстановления и показа видео при уменьшенном временном разрешении (поддерживается до 3 уровней временной масштабируемости);
- *масштабируемость качества* позволяет разбирать поток на несколько уровней с разной битовой скоростью так, что комбинация подмножества уровней может быть декодирована в осмысленный сигнал; разбор битового потока может происходить как при передаче, так и на декодере, а восстановленное качество, вообще говоря, связано с количеством слоев, использованных для декодирования и восстановления.

Стандарт предоставляет возможность *кодирования формы объектов и их прозрачности* (альфа-канал).

Кодирование формы позволяет описывать объекты произвольной формы. Приложения, которые используют двоичные побитовые карты изображения, служат для презентаций баз данных изображений, интерактивных игр, наблюдения, и анимации. Предлагаются эффективные методы кодирования двоичных форм.

Двоичная альфа-маска определяет, принадлежит или нет пиксел объекту. Она может быть включена или выключена. Альфа-канал определяет прозрачность объекта, которая не обязательно является однородной. Многоуровневые альфа-карты часто используются для затенения различных слоев последовательности изображений. Другими приложениями, которые используют при работе с изображениями ассоциированные двоичные альфа-маски, являются презентации баз данных изображений, интерактивные игры, наблюдения и анимация. Предлагаются методики, которые позволяют эффективно кодировать двоичные и альфа-плоскости с серой шкалой изображения. Маска с серой шкалой предоставляет возможность точно определить прозрачность каждого пикселя.

Стандарт обеспечивает *высокую надежность* при работе с каналами передачи или средствами хранения, которые могут вносить ошибки. Это включает в себя операции алгоритмов сжатия данных в среде, подверженной сбоям при низких скоростях передачи (меньше 64 кбита/с).

Часть стандарта, связанная с *анимацией лица*, позволяет посыпать параметры, которые помогают специфицировать и анимировать синтезированные лица. Эти модели не являются сами частью стандарта MPEG-4, стандартизирован только ряд параметров: определение и кодирование анимационных параметров лица (модельно независимое); позиции характерных деталей и их ориентация для определения сеток при анимации лица; визуальные конфигурации губ, соответствующие фонемам речи; определение и кодирование параметров описания лица (для калибровки модели); трехмерные позиции характерных признаков (деталей); трехмерные калибровочные сетки для анимации головы; текстурная карта лица; персональные характеристики; кодирование лицевой текстуры.

Стандарт обеспечивает эффективное кодирование двумерных и трехмерных сеток. *Видеобъект* в сцене — это объект, к которому пользователю разрешен доступ (поиск, просмотр) и обработка (удаление и вставка). Экземпляры видеообъектов в заданное время называются *плоскостями видеообъекта* (VOP). В процессе кодирования генерируется закодированное представление VOP, а также информация о композиции, которую требуется отобразить. Кроме того, пользователь может взаимодействовать с процессом композиции в декодере и изменять его по мере необходимости.

Полный синтаксис позволяет кодировать как прямоугольные видеообъекты в сцене, так и имеющие произвольную форму. Кроме того, синтаксис поддерживает как немасштабируемое, так и масштабируемое кодирование. Таким образом, становится возможным работать как с обычной, так и с объектно-ориентированной масштабируемостью. Масштабируемый синтаксис позволяет восстановить полезные видеоданные из фрагментов общего битового потока. Это достигается путем структурирования общего битового потока на два или более уровня, начиная с автономного нижнего слоя и добавляя ряд уровней расширения. Для кодирования нижнего слоя может быть использован немасштабируемый синтаксис, или, в случае кодирования изображений, даже синтаксис различных стандартов кодирования видео.

Для обеспечения возможности распознавания индивидуальных объектов необходимо получить представление их формы в закодированном виде. Естественный видеообъект состоит из последовательности двумерных представлений (в различные моменты времени), именуемых *плоскостями видеообъекта* (VOP). Для эффективного кодирования VOP используется устранение как временной, так и пространственной избыточности. Таким образом, представление VOP в закодированном виде включает представление их формы, движения и структуры.

Для достижения сильного сжатия используется несколько методов. Сначала алгоритм запускает компенсацию движения на основе блоков для уменьшения временной избыточности. Компенсация движения используется как для каузального предсказания текущей VOP по предыдущей VOP, так и для некаузального, интерполяционного предсказания по прошедшим и последующим VOP. Векторы движения определяются для каждой области VOP (макроблока) размером 16×16 отсчетов или 8×8 отсчетов (в компоненте яркости) в зависимости от

требований. Затем ошибка предсказания сжимается при помощи дискретного косинусного преобразования (ДКП) для устранения пространственной корреляции до квантования, в необратимом процессе которого менее важная информация отбрасывается. В заключение сведения о форме, векторы движения и квантованные данные ДКП кодируются с использованием кодов переменной длины.

Вследствие противоречащих друг другу требований произвольного доступа и высокоэффективного сжатия определено три основных типа *плоскостей видеообъекта* (VOP). VOP с внутрикадровым кодированием (I-VOP) кодируются независимо от других изображений. Они обеспечивают точки доступа к закодированной последовательности, в которых может начаться декодирование, но кодируются с умеренным сжатием. VOP с применением кодирования с предсказанием (P-VOP) кодируются более эффективно при использовании предсказания с компенсацией движения по предшествующим VOP, закодированным при помощи внутрикадрового кодирования или кодирования с предсказанием, и обычно используются в качестве опорных для дальнейшего предсказания. VOP, закодированные с двусторонним предсказанием (B-VOP), обеспечивают наивысшую степень сжатия, но требуют для компенсации движения как предшествующих, так и последующих опорных VOP. VOP, закодированные с двусторонним предсказанием, никогда не используются в качестве опорных для предсказания (за исключением тех случаев, когда возникающая в результате VOP используется в качестве опорной для уровня с масштабируемым расширением). Сформированная из трех типов VOP последовательность является очень гибкой структурой. Выбор оставлен за кодером, он будет зависеть от требований приложения.

В сценах естественного видео VOP генерируются путем сегментации сцены в соответствии с определенным семантическим значением. Таким образом, для этих сцен сведения о форме являются двоичными (двоичная форма). Сведения о форме также называются альфа-каналом. Двоичный альфа-канал кодируется на основе макроблоков при помощи кодера, использующего данные контекста, компенсацию движения и арифметическое кодирование.

Для кодирования формы VOP сначала создается ограничительный прямоугольник, он расширяется до множества из блоков размером 16×16 с расширенными обнуленными альфа-отсчетами. Затем инициируется кодирование формы на основе блоков 16×16 ; эти блоки также называются двоичными альфа-блоками.

Блоки 16×16 (макроблоки) выбраны в качестве единицы компенсации движения в результате компромисса между необходимостью увеличения эффективности кодирования за счет учета движения объектов и необходимостью передачи дополнительного объема данных об этом движении. В зависимости от того, насколько позволяет дополнительный объем данных, каждый макроблок может быть в дальнейшем разбит на блоки 8×8 для оценки и компенсации движения.

В MPEG-4 определено **множество видеопрофилей**, список и характерные черты которых приведены в табл. 14.25.

Профили MPEG-4/Видео определяют инструменты кодирования, а параметры формируемых битовых потоков задаются так называемыми **уровнями**. Каждый уровень устанавливает ограничения на максимальную производительность, требуемую для декодирования битового потока в стандарте MPEG-4. Например, мультимедийный терминал с маломощным процессором и ограниченной памятью сможет обеспечить работу кодека простого профиля с уровнем L0. Уровень



определяет ограничения на объем буфера памяти, размер видеокадра и частоту следования кадров, скорость обработки потока, число видеообъектов. Высокие уровни соответствующего профиля должны справляться с обработкой информации более низкого уровня.

Таблица 14.25. Профили MPEG-4/Видео

№	Видео профили	Специфические особенности
1.	<i>Простой</i> (<i>Simple</i>)	Обеспечивает эффективное, устойчивое к ошибкам кодирование прямоугольных видеообъектов, подходящих для приложений, используемых в мобильных сетях.
2.	<i>Простой продвинутый</i> (<i>Advanced Simple</i>)	Содержит только прямоугольные объекты, но имеет несколько дополнительных средств, которые делают его более эффективным: В-кадры, компенсация движения деталей до 1/4 пикселя и компенсация общего перемещения изображения (глобальная компенсация движения, GMC).
3.	<i>Простой масштабируемый</i> (<i>Simple Scalable</i>)	Обеспечивает поддержку кодирования временных и пространственных масштабируемых объектов в простом профиле, полезен для приложений, обеспечивающих услуги на более чем одном уровне качества, связанных с ограничениями скорости передачи данных или ресурсами декодера, такими как использование Интернете и программное декодирование.
4.	<i>Простой продвинутый реального времени</i> (<i>Advanced Real-Time Simple</i>)	Реализует продвинутый метод кодирования прямоугольных видеообъектов, устойчивый к ошибкам, использующий обратный канал и улучшенную стабильность временного разрешения при минимальной задержке буферизации. Он удобен для кодирования в случае приложений реального времени, таких как видеотелефон, телеконференции и удаленное наблюдение.
5.	<i>Простой студийный</i> (<i>Simple Studio</i>)	Применяется в приложениях студийного редактирования, обеспечивая очень высокое качество кодирования. Он работает только с I-кадрами, но действительно поддерживает произвольные формы и большое число alpha-каналов. Возможная скорость передачи достигает 2 Гбита/с.
6.	<i>Основной</i> (<i>Core</i>)	Обеспечивает поддержку кодировки масштабируемых по времени объектов произвольной формы в простой визуальный профиль, полезен для приложений, осуществляющих относительно простую интерактивность (Интернет, мультимедиа).
7.	<i>Основной продвинутый</i> (<i>Advanced Core</i>)	Комбинирует возможность декодирования видеообъектов произвольной формы (как в основном визуальном профиле) с возможностью декодирования масштабируемых статических объектов произвольной формы (как в продвинутом масштабируемом профиле текстур). Он удобен для различных мультимедийных приложений, таких как интерактивная передача потоков мультимедиа через Интернет.
8.	<i>Основной масштабируемый</i> (<i>Core Scalable</i>)	Реализует поддержку кодирования объектов произвольной формы с пространственным и временным масштабированием в дополнение к основному профилю. Главной особенностью этого профиля является масштабируемость (пространственная, временная и сигнал/шум) для областей и объектов, представляющих интерес. Он полезен для таких приложений, как Интернет, мобильные сети и видеовещание.
9.	<i>Основной студийный</i> (<i>Core Studio</i>)	Дополняет простой студийный вариант Р-кадрами, делая его более эффективным, но требующим более сложной реализации.

Таблица 14.25 (продолжение)

10.	<i>Главный (Main)</i>	Дополняет кодирование чересстрочных, полупрозрачных и виртуальных объектов в основном профиле. Он полезен для интерактивного широковещательного обмена (с высоким качеством) и для DVD-приложений.
11.	<i>Улучшенной эффективности кодирования (Advanced Coding Efficiency)</i>	Реализует устойчивый к ошибкам продвинутый метод кодирования прямоугольных видеообъектов, использующий обратный канал и улучшенную стабильность временного разрешения при минимальной задержке буферизации. Он удобен для кодирования в случае приложений реального времени, таких как видеотелефон, телеконференции и удаленное наблюдение.
12.	<i>N-битовый (N-bit)</i>	Добавляет поддержку кодирования видеообъектов, имеющих пиксельную глубину в диапазоне от 4 до 12 битов, в главный профиль. Он удобен для использования в приложениях видеонаблюдения.
13.	<i>Масштабируемый профиль мелкой гранулярности (Fine Granular Scalability)</i>	Допускает большое число масштабных уровней (до 8), так что качество доставки можно легко адаптировать к условиям передачи и декодирования. Он может использоваться с простым или продвинутым простым профилем в качестве базового уровня.
14.	<i>Масштабируемых текстур (Scalable Texture)</i>	Предоставляет пространственное масштабируемое кодирование статических объектов изображений (текстур), полезное для приложений, где нужны уровни масштабируемости, такие как установление соответствия между текстурой и объектами игр, а также для работы с цифровыми фотокамерами высокого разрешения.
15.	<i>Продвинутый масштабируемый текстур (Advanced Scalable Texture)</i>	Поддерживает декодирование текстур произвольной формы и статических изображений, включая масштабируемое кодирование формы, мозаичное заполнение и повышенную устойчивость к ошибкам. Полезен для приложений, требующих быстрого произвольного доступа, а также нескольких уровней масштабируемости и кодирования статических объектов произвольной формы. Примерами таких приложений могут служить просмотр статических изображений в Интернете, а также считывание через Интернет изображений, полученных из цифровых фотоаппаратов с высоким разрешением.
16.	<i>Основной анимации двумерных текстур (Basic Animated 2D-Texture)</i>	Предоставляет пространственную масштабируемость, масштабируемость сигнал/шум и анимацию, базирующуюся на сетках для статических объектов изображений (текстур), а также простую анимацию объектов лица.
17.	<i>Простой анимации лиц (Simple Face Animation)</i>	Предоставляет простые средства анимации модели лица, удобные для таких приложений, как аудио/видеопрезентации для лиц с ухудшенным слухом.

Таблица 14.25 (окончание)

18.	<i>Простой анимации лица и тела (Simple Facial and Body Animation)</i>	Является надстройкой профиля простой анимации лица с добавлением анимации тела.
19.	<i>Гибридный (Hybrid)</i>	Комбинирует возможность декодировать масштабируемые объекты натурального видеопроизвольной формы (как в главном визуальном профиле) с возможностью декодировать несколько синтетических и гибридных объектов, включая анимационные статические объекты изображения. Он удобен для различных сложных мультимедиа приложений.

В табл. 14.26 приводятся некоторые параметры уровней ряда профилей обработки динамических изображений.

Таблица 14.26. Уровни профилей кодирования/декодирования динамических изображений

№	Профиль	Уровень	Разрешение	Максимальная скорость	Максимальное количество объектов
1.	<i>Простой</i>	L0	QCIF (176×144)	64 кбита/с	1
		L1	QCIF	64 кбита/с	4
		L2	CIF (352×288)	128 кбитов/с	4
		L3	CIF	384 кбита/с	4
2.	<i>Простой продвинутый</i>	L0	QCIF	128 кбитов/с	1
		L1	QCIF	128 кбитов/с	4
		L2	CIF	384 кбита/с	4
		L3	CIF	768 кбитов/с	4
		L4	352×576	3 Мбита/с	4
		L5	720×576	8 Мбитов/с	4
3.	<i>Простой масштабируемый</i>	L1	CIF	128 кбитов/с	4
		L2	CIF	256 кбитов/с	4
4.	<i>Простой продвинутый реального времени</i>	L1	QCIF	64 кбита/с	4
		L2	CIF	128 кбитов/с	4
		L3	CIF	384 кбита/с	4
		L4	CIF	2 Мбита/с	16

Таблица 14.26 (окончание)

5.	<i>Простой студийный</i>	L1	ITU-R 601:422 ITU-R 601:444	180 Мбитов/с	1
		L2	ITU-R 601:422 ITU-R 601:444	600 Мбитов/с	1
		L3	ITU-R 709.601:444 ITU-R 709.601:422	900 Мбитов/с	1
		L4	ITU-R 709.601:444	1800 Мбитов/с	1
6.	<i>Основной</i>	L1	QCIF	384 кбита/с	4
		L2	CIF	2 Мбита/с	16
7.	<i>Основной продвинутый</i>	L1	QCIF	384 кбита/с	4
		L2	CIF	2 Мбита/с	16
8.	<i>Основной масштабируемый</i>	L1	CIF	768 кбитов/с	4
		L2	CIF	1,5 Мбитов/с	8
		L3	ITU-R 601	4 Мбита/с	16
9.	<i>Основной студийный</i>	L1	ITU-R 601:422 ITU-R 601:444	90 Мбитов/с	4
		L2	ITU-R 709.601:422 ITU-R 601:444	300 Мбитов/с	4
		L3	ITU-R 709.601:444 ITU-R 709.601:422	450 Мбитов/с	8
		L4	ITU-R 709.601:444 ITU-R 709.601:444 2K×2K×30P:444	900 Мбитов/с	16
10.	<i>Главный</i>	L2	CIF	2 Мбита/с	16
		L3	ITU-R 601	15 Мбитов/с	32
		L4	1920×1088	38,4 Мбитов/с	32
11.	<i>Улучшенной эффективности кодирования</i>	L1	CIF	384 кбита/с	4
		L2	CIF	2 Мбита/с	16
		L3	ITU-R 601	15 Мбитов/с	32
		L4	1920×1088	38,4 Мбитов/с	32
12.	<i>N-битовый</i>	L2	CIF	2 Мбита/с	16
13.	<i>Масштабируемый профиль мелкой гранулярности</i>	L0	QCIF	128 кбитов/с	4
		L1	QCIF	128 кбитов/с	4
		L2	CIF	384 кбита/с	4
		L3	CIF	768 кбитов/с	4
		L4	352×576	3 Мбита/с	4
		L5	720×576	8 Мбитов/с	4

Таким образом, стандарт MPEG-4 определяет принципы работы с контентом, цифровым представлением медиаданных, для трех областей: собственно интерактивного мультимедиа (включая продукты, распространяемые на оптических дисках и через компьютерные сети), графических приложений (синтетического контента) и цифрового телевидения. Фактически данный формат задает пра-

вила организации объектно-ориентированной среды. Он имеет дело не просто с потоками и массивами медиаданных, а с медиаобъектами (ключевое понятие стандарта).

Стандарт MPEG-4 предоставляет пользователям гибкие средства работы с мультимедийным контентом. Помимо работы с аудио и видео формат позволяет работать с естественными и синтезированными компьютером двумерными и трехмерными объектами, производить привязку их взаимного расположения и синхронизацию друг относительно друга, а также указывать их интерактивное взаимодействие с пользователем. Кроме того, формат обеспечивает доступ к мультимедийной информации через каналы различной пропускной способности.

Кардинальное нововведение при компрессии видео в MPEG-4 заключается в следующем. В отличие от предыдущих форматов, которые делили изображение на прямоугольники, при обработке изображений кодек оперирует объектами с произвольной формой. К примеру, человек, двигающийся по комнате, будет воспринят как отдельный объект, перемещающийся относительно неподвижного объекта — заднего плана. Естественно, алгоритмы поиска и обработки подобных объектов требуют гораздо больше вычислительных ресурсов, чем при применении стандартов MPEG-1 или MPEG-2. Но с учетом быстродействия современных компьютеров последнее обстоятельство нельзя рассматривать сегодня как крупное препятствие на пути широкого распространения формата MPEG4.

14.6. H.264/MPEG-4 часть 10 AVC — улучшенное видеокодирование

В 2001 г. VCEG (Группа экспертов по видеокодированию МСЭ-Т, SG16 Q.6) и MPEG (Экспертная группа по движущимся изображениям ISO/IEC, JTC 1/SC 29/WG 11) сформировали совместную видеогруппу (JVT) с полномочиями завершения проекта нового стандарта видеокодирования, который был утвержден как H.264/AVC в 2003 г. [5.48, 5.49].

Область действия стандарта проиллюстрирована на рис. 14.24, который показывает типичный цикл кодирования/декодирования (исключая передачу или хранение видеосигнала).

В H.264/AVC, как и во всех предшествующих стандартах видеокодирования, стандартизуется только декодер путем накладывания ограничений на битовый поток и синтаксис и определения процесса декодирования синтаксических элементов таким образом, чтобы любой декодер, отвечающий стандарту, создавал одинаковый выходной поток, если ему на вход подавать кодированный поток, отвечающий ограничениям стандарта. Это ограничение области действия стандарта позволяет давать максимальную свободу для оптимизации конкретных реализаций для конкретных приложений (соблюдая баланс между качеством компрессии, стоимостью, временем разработки и т. д.). Однако этот подход не дает никаких гарантий качества кодирования/декодирования, так как позволяет применять даже грубую методику кодирования, если только она соответствует ограничениям стандарта.

Целью реализации проекта H.264/AVC было создание стандарта, способного обеспечить хорошее качество изображения при существенно более низких скоро-

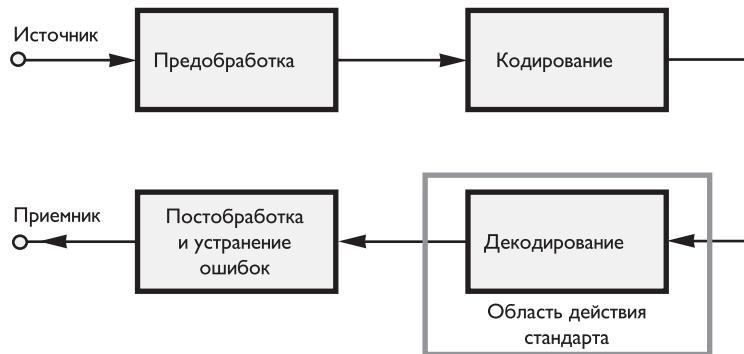


Рис. 14.24. Цикл кодирования/декодирования изображений

стях, чем предыдущие стандарты (например, в два и более раз меньшую скорость передачи чем при кодировании MPEG-2, H.263 или MPEG-4 часть 2) [5.67–5.72]. Дополнительной целью было обеспечение достаточной гибкости для того, чтобы стандарт мог применяться для широкого множества приложений в различных сетях и системах, включая низкие и высокие скорости передачи, низкие и высокие разрешения видео, вещание, хранение на DVD, передача по пакетным сетям по протоколу

Стандарт видеокодирования H.264/AVC предназначен для технических решений, включающих по крайней мере следующие области применения:

- кабельное, спутниковое, наземное вещание, вещание с помощью кабельных и DSL-модемов;
- хранение на оптических и магнитных носителях, DVD и т. д.;
- диалоговые службы, работающие в различных сетях (например, ISDN, Ethernet, LAN, DSL, беспроводные и мобильные сети, а также любые сочетания этих сетей);
- видео по запросу или службы потокового мультимедиа в различных сетях;
- MMS (службы мультимедийных сообщений) в различных сетях и т. д.

Более того, новые приложения могут развертываться в существующих и перспективных сетях.

Для того чтобы удовлетворить потребности в гибкости и настраиваемости для множества приложений и разнообразия сетей, H.264/AVC включает два уровня: уровень видеокодирования (Video Coding Layer, VCL), предназначенный для эффективного представления видеоконтента, и уровень сетевой абстракции (Network Abstraction Layer, NAL), который форматирует представление видео VCL и обеспечивает заголовочную информацию в виде, подходящем для передачи на различных транспортных уровнях или средствах хранения. На рис. 14.25 приведена структура видеокодера H.264/AVC.

14.6.1. Профили и уровни стандарта H.264/AVC

Стандарт включает множество профилей, ориентированных на конкретные классы приложений [5.67, 5.69]. Список профилей и их характерные черты приведены в табл. 14.27.

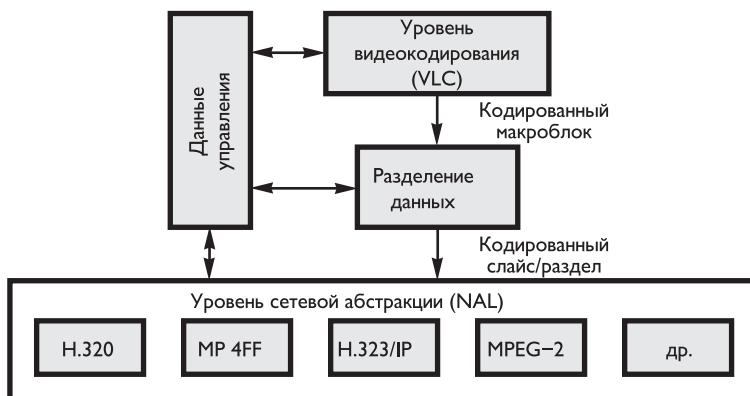


Рис. 14.25. Структура видеокодера H.264/AVC

Таблица 14.27. Профили H.264/AVC

№	Видео профили	Специфические особенности
1.	<i>Ограниченный базовый профиль</i> (Constrained Baseline Profile – CBP)	Широко используется в первую очередь для недорогих приложений: в видеоконференциях и системах мобильного приема.
2.	<i>Базовый профиль</i> (Baseline Profile – BP)	Применяется для дешевых приложений, когда требуется дополнительная защита от ошибок. Этот профиль редко используется в видеоконференцсвязи и мобильных приложениях.
3.	<i>Главный профиль</i> (Main Profile – MP)	Первоначально рассматривался как основной профиль для передачи и хранения данных, использование этого профиля было практически сведено к нулю после разработки High-профилей для тех же применений.
4.	<i>Расширенный профиль</i> (Extended Profile – XP)	Предназначен для организации потокового видео, этот профиль использует относительно большие возможности сжатия и некоторые дополнительные приемы для обеспечения устойчивости к потере данных и коммутации потоков.
5.	<i>Высокий профиль</i> (High Profile – HiP)	Основной профиль для передачи и хранения приложений на дисках, особенно для приложений высокой четкости (например, HD DVD и Blu-ray).
6.	<i>Высокий профиль 10</i> (High 10 Profile – Hi10P)	Добавляется поддержка до 10 битов на каждую компоненту пикселя изображения.
7.	<i>Высокий профиль 4 : 2 : 2</i> (High 4 : 2 : 2 Profile – Hi422P)	Ориентирован на профессиональные приложения, при использовании видео с чересстрочной разверткой. Этот профиль основан на применении HiP, добавляя поддержку режима 4 : 2 : 2.
8.	<i>Высокий интеллектуальный профиль 4 : 4 : 4</i> (High 4 : 4 : 4 Predictive Profile – Hi44PP)	Этот профиль основывается на профиле 4 : 2 : 2 с поддержкой режима 4 : 4 : 4 и использованием до 14 битов на каждую компоненту пикселя, а также реализацией эффективного кодирования без потерь и кодирования фотографии в виде трех плоскостей отдельных цветов.

Таблица 14.27 (окончание)

9.	<i>Высокий стерео-профиль (High Stereo Profile)</i>	Этот профиль предназначается для стереоскопического трехмерного видео и комбинирует инструменты <i>High</i> -профилей с возможностями предсказания при кодировании компонент видеинформации.
Профили внутрикадрового кодирования		
10.	<i>Высокий 10 Intra-профиль (High 10 Intra Profile)</i>	10 битов на пиксель при использовании только внутрикадрового кодирования.
11.	<i>Высокий 4 : 2 : 2 Intra-профиль (High 4 : 2 : 2 Intra Profile)</i>	Поддержка режима 4 : 2 : 2 при использовании только внутрикадрового кодирования.
12.	<i>Высокий 4 : 4 : 4 Intra-профиль (High 4 : 4 : 4 Intra Profile)</i>	Поддержка режима 4 : 4 : 4 при использовании только внутрикадрового кодирования.
13.	<i>CAVLC 4 : 4 : 4 Intra-профиль (CAVLC 4 : 4 : 4 Intra-Profile)</i>	Применяется энтропийное кодирование <i>CAVLC</i> при использовании только внутрикадрового кодирования, т. е. не поддерживается <i>CABAC</i> .
Дополнительные масштабируемые профили		
14.	<i>Масштабируемый базовый профиль (Scalable Baseline Profile)</i>	В первую очередь ориентирован на видеоконференции, мобильные приложения и наблюдение, этот профиль основывается на версии профиля <i>BP</i> .
15.	<i>Масштабируемый высокий профиль (Scalable High Profile)</i>	В первую очередь ориентирован на приложения потокового вещания, этот профиль основывается на <i>High</i> -профиле.
16.	<i>Масштабируемый высокий Intra профиль (Scalable High Intra Profile)</i>	В первую очередь ориентирован на приложения создания программ, этот профиль ограничивает возможности масштабируемого <i>High</i> -профиля использованием только внутрикадрового кодирования.

Как и в MPEG-4/Видео, при обработке динамических изображений по стандарту H.264/AVC уровни профилей задаются в виде ограничения параметров формируемых битовых потоков.

Таблица 14.28. Уровни профилей стандарта H.264/AVC и их максимально возможные параметры

Уровни	Макс. к-во макроблоков		Максимальный цифровой поток (VCL)				Примеры разрешения/макс. частота кадров (память макс. числа кадров)
	В сек.	В кадре	BP, XP, MP (кбиты/с)	HiP (кбиты/с)	Hi10P (кбитов/с)	Hi422P, Hi444PP (кбиты/с)	
1.	1485	99	64	80	192	256	128 × 96/30,9 (8) 176×144/15,0 (4)

Таблица 14.28 (окончание)

1b.	1485	99	128	160	384	512	$128 \times 96/30,9$ (8) $176 \times 144/15,0$ (4)
1.1.	3000	396	192	240	576	768	$176 \times 144/30,3$ (9) $320 \times 240/10,0$ (3) $352 \times 288/7,5$ (2)
1.2.	6000	396	384	480	1152	1536	$320 \times 240/20,0$ (7) $352 \times 288/15,2$ (6)
1.3.	11880	396	768	960	2304	3072	$320 \times 240/36,0$ (7) $352 \times 288/30,0$ (6)
2.	11880	396	2000	2500	6000	8000	$320 \times 240/36,0$ (7) $352 \times 288/30,0$ (6)
2.1.	19800	792	4000	5000	12000	16000	$352 \times 480/30,0$ (7) $352 \times 576/25,0$ (6)
2.2.	20250	1620	4000	5000	12000	16000	$352 \times 480/30,7$ (10) $352 \times 576/25,6$ (7) $720 \times 480/15,0$ (6) $720 \times 576/12,5$ (5)
3.	40500	1620	10000	12500	30000	40000	$352 \times 480/61,4$ (12) $352 \times 576/51,1$ (10) $720 \times 480/30,0$ (6) $720 \times 576/25,0$ (5)
3.1	108000	3600	14000	17500	42000	56000	$720 \times 480/80,0$ (13) $720 \times 576/66,7$ (11) $1280 \times 720/30,0$ (5)
3.2.	216000	5120	20000	25000	60000	80000	$1280 \times 720/60,0$ (5) $1280 \times 1024/42,2$ (4)
4.	245760	8192	20000	25000	60000	80000	$1280 \times 720/68,3$ (9) $1920 \times 1080/30,1$ (4) $2048 \times 1024/30,0$ (4)
4.1.	245760	8192	50000	62500	150000	200000	$1280 \times 720/68,3$ (9) $1920 \times 1080/30,1$ (4) $2048 \times 1024/30,0$ (4)
4.2.	522240	8704	50000	62500	150000	200000	$1920 \times 1080/64,0$ (4) $2048 \times 1080/60,0$ (4)
5.	589824	22080	135000	168750	405000	540000	$1920 \times 1080/72,3$ (13) $2048 \times 1024/72,0$ (13) $2048 \times 1080/67,8$ (12) $2560 \times 1920/30,7$ (5) $3680 \times 1536/26,7$ (5)
5.1.	983040	36864	240000	300000	720000	960000	$1920 \times 1080/120,5$ (16) $4096 \times 2048/30,0$ (5) $4096 \times 2304/26,7$ (5)

14.6.2. Эффективность сжатия в стандарте H.264/AVC

Среди функциональных возможностей H.264/AVC, которые позволяют поднять эффективность кодирования по отношению к предшествующим стандартам кодирования видео, следует подчеркнуть следующие усовершенствования и новые возможности предсказания содержания кодируемого изображения [5.67, 5.68, 5.71, 5.72].

- *Компенсация движения с использованием переменных размеров блока, включая малые размеры блока.* Стандарт поддерживает достаточно большую гибкость в выборе размеров и форм блоков при компенсации движения по сравнению со всеми предшествовавшими стандартами, причем минимальный размер блока составляет 4×4 пикселя и позволяет точно выделять области движения.
- *Компенсация движения с точностью до четверти пикселя.* Большинство предшествовавших стандартов поддерживали компенсацию движения с точностью не более полпикселя. Стандарт повышает точность компенсации движения до четверти пикселя (что особенно актуально для медленного движения), как и в расширенном профиле MPEG-4, но в стандарте H.264/AVC уменьшена сложность интерполяции. Цветность, как правило, хранится с разрешением, уменьшенным вдвое по вертикали и горизонтали, поэтому компенсация движения для компонента цветности использует точность в одну восьмую пикселя цветности.
- *Векторы движения, выводящие за границы изображения.* В MPEG-2 и предшествовавших ему стандартах векторов движения могли указывать только на пиксели, находящиеся в границах декодированного опорного изображения. Методика экстраполяции за границы изображения, появившаяся как опция в H.263, включена также в стандарт H.264/AVC.
- *Компенсация движения с несколькими опорными изображениями.* Изображения, кодированные с предсказанием (P), в MPEG-2 и предшествовавших ему стандартах использовали только одно предыдущее изображение для предсказания значений в новом изображении. В стандарте H.264/AVC расширяется методика увеличенного выбора опорного изображения, появившаяся в приложении к стандарту H.263; это поднимает эффективность кодирования, так как позволяет кодеру выбирать для компенсации движения между большим количеством изображений, декодированных и сохраненных на декодере. Такое же расширение возможностей выбора опорного изображения применяется и для двунаправленного предсказания (B-кадры). В MPEG-2 для двунаправленного предсказания должны использоваться строго определенные изображения — предшествующее и последующее в порядке воспроизведения внутрикодированные (I) или кодированные с предсказанием (P) изображения. В данном стандарте предусматривается возможность использования до 32 ссылок на другие кадры. В большинстве сцен данная функция обеспечивает не очень большое улучшение в качестве и не даёт заметного понижения цифрового потока. Однако для некоторых сцен, например с часто повторяющимися участками, возвратно-поступательным движением и т. п. подобный подход при сохранении качества позволяет очень сильно снизить затраты цифрового потока.
- *Независимость порядка воспроизведения изображений и порядка опорных изображений.* В предшествующих стандартах устанавливалась жесткая зависимость между порядком следования изображений для использования при компенсации движения и порядком следования изображений при воспроизведении. В стандарте H.264/AVC эти ограничения в значительной мере устранены, что позволяет кодеру выбирать порядок изображений для компенсации движения и для воспроизведения с высокой степенью гибко-



сти, которая ограничена только объемом памяти, который гарантирует возможность декодирования. Устранение ограничения также позволяет в ряде случаев исключить дополнительную задержку, ранее связанную с двунаправленным предсказанием.

- *Независимость методов обработки изображений и возможности их использования для предсказания движения.* В предшествующих стандартах изображения, закодированные с использованием некоторых методов (например, двунаправленного предсказания), не могли использоваться в качестве опорных для предсказания движения других изображений видеопоследовательности. Устранив это ограничение, стандарт H.264/AVC обеспечивает кодеру большую гибкость и, во многих случаях, возможность использовать для предсказания движения изображение, более близкое по содержанию к кодируемому.
- *Взвешенное предсказание.* Эта новая возможность стандарта позволяет взвешивать и сдвигать сигнал после компенсации движения на величины, указанные кодером. Такая методика может чрезвычайно сильно поднять эффективность кодирования для сцен с изменением освещенности, а также гибко использоваться для других целей.
- *Улучшенная обработка «пропущенных» (skipped) блоков и блоков с «прямым» (direct) предсказанием движения.* В предшествующих стандартах «пропущенные» области изображения, кодируемого с предсказанием, не могли содержать движения, что отрицательно сказывалось на эффективности кодирования, когда кодируемое видео содержало движение сцены в целом. Новый стандарт предполагает возможное наличие движения в «пропущенных» областях. Для областей с двунаправленным предсказанием новый стандарт также предусматривает расширенный метод компенсации движения, известный под названием «прямая» компенсация движения, который позволяет улучшить «прямое» предсказание, введенное в приложении к H.263 и MPEG-4.
- *Направленное пространственное предсказание для внутрикадрового кодирования.* Новая методика экстраполяции краев ранее декодированных частей текущего изображения применяется в областях изображений, кодируемых с помощью внутрикадровой методики (т. е. без ссылок на содержание других изображений). Эта методика повышает качество сигнала, используемого для предсказания, а также позволяет использовать для предсказания соседние области, которые были закодированы не с помощью внутрикадровой методики (это не допускалось в стандартах H.263 и MPEG-4).
- *Гибкие функции чересстрочного сжатия* (поддерживается не во всех проффилях). Адаптивное к изображению кодирование полей, позволяющее кодировать каждый кадр как кадр или как пару полей (полукадров) – в зависимости от наличия/отсутствия движения. Адаптивное к макроблокам кодирование полей, позволяющее независимо кодировать каждую вертикальную пару макроблоков (блок 16×32) как прогрессивные или чересстрочные. Позволяет использовать макроблоки 16×16 в режиме разбиения на поля.
- *Логарифмическое управление длиной шага квантования* для упрощения распределения цифрового потока кодером и упрощенного вычисления об-

ратной величины квантования. Частотно-оптимизированные матрицы масштабирования квантования, выбираемые кодером для оптимизации квантования на основе человеческих особенностей восприятия (поддерживается не во всех профилях).

- *Деблокинговая фильтрация в цикле кодирования.* Кодирование видео, основанное на обработке блоков, приводит к искажениям, называемым «блокинг-эффектом». Причиной может служить как предсказание, так и кодирование остатков предсказания. Применение адаптивного деблокингового фильтра — хорошо известный метод повышения качества восстановленного видео, и при правильном проектировании может повысить как субъективное, так и объективное качество. Построенный на основе концепции приложения к стандарту H.263, деблокинговый фильтр в стандарте H.264/AVC внесен в петлю обратной связи предсказания и компенсации движения. Таким образом, повышение качества изображения может быть использовано в межкадровом предсказании, что, в свою очередь, улучшает возможность предсказания движения других изображений.

В дополнение к улучшенным методам предсказания, другие части стандарта были также расширены с целью повышения эффективности кодирования. Стандарт предусматривает ряд дополнительных функций обработки динамических изображений.

- *Преобразование блоков небольшого размера.* Все предшествующие стандарты кодирования видео использовали преобразование блоков размером 8×8 , в то время как новый стандарт основан главным образом на преобразовании блоков 4×4 . Это позволяет кодеру представлять сигнал более локально-адаптивным образом, что уменьшает искажения, известные под названием «окантовки» или «звон» (ringing). Меньший размер блоков также частично оправдан тем, что использование описанных выше методов позволяет лучше предсказывать содержание видео, но при этом требуется обеспечить области преобразования с границами, соответствующими меньшим областям предсказания.
- *Иерархическое преобразование блоков.* Хотя в большинстве случаев использование малых размеров блоков преобразования (4×4) визуально предпочтительнее, есть некоторые сигналы, содержащие существенную корреляцию, для которых выгоднее использовать более длительные базисные функции. Новый стандарт позволяет делать это двумя способами: с помощью иерархического преобразования (для увеличения эффективного размера блоков) для низкочастотной цветностной информации (8×8); и путем предоставления кодеру возможности выбрать специальный тип внутрикадрового кодирования, позволяющий расширить размер преобразования яркости для низкочастотной информации до размера блока 16×16 .
- *Преобразование с использованием 16-битной арифметики.* Все предшествующие стандарты требовали от кодеров и декодеров более сложных вычислений при преобразовании. В то время как ранее требовались, как правило, вычисления с использованием 32-битной арифметики, новый стандарт использует только 16-битную точность вычислений.
- *Точное обратное преобразование.* В предшествующих стандартах видеокодирования преобразование, используемое для представления видео (дис-

крайнее косинусное преобразование, ДКП), оговаривалось только в рамках допустимых ошибок из-за практической нереализуемости точного обратного преобразования. В итоге каждая реализация декодера могла давать слегка различное декодированное видео, обуславливающее несовпадение представления декодированного видео на кодере и декодере, что приводит к снижению эффективного качества видео. Новый стандарт, следуя пути, намеченному в одном из приложений к H.263, первым достиг точного совпадения декодированного видео на всех декодерах.

- *Арифметическое энтропийное кодирование.* В стандарт включен передовой метод энтропийного кодирования, известный как арифметическое кодирование. Хотя арифметическое кодирование включалось как необязательная возможность в стандарте H.263, новый стандарт использует более эффективную методику – контекстно-адаптивное двоичное арифметическое кодирование CABAC (Context-adaptive binary arithmetic coding). Оно поддерживается только в Главном профиле и выше. Предусмотрено также использование контекстно-адаптивного кодирования кодами переменной длины – CAVLC (Context-adaptive variable-length coding). CAVLC обладает меньшей сложностью, чем CABAC, но тем не менее сложнее и эффективнее, чем алгоритмы, применяемые для тех же целей в более ранних технологиях сжатия видео (как правило, это алгоритм Хаффмана).

Устойчивость к ошибкам и потерям данных и гибкость работы на множестве сетевых структур в новом стандарте определяется целым рядом новых методик, к которым можно отнести ниже следующие.

- *Структура набора параметров.* Набор параметров обеспечивает устойчивую и эффективную передачу заголовочной информации. Так как потеря нескольких ключевых битов информации (таких как заголовок последовательности или заголовок изображения) может оказать сильное отрицательное воздействие на процесс декодирования в предыдущих стандартах, эта ключевая информация была отделена, и обрабатывается более гибким специализированным способом в новом стандарте.
- *Синтаксическая структура блока Уровня сетевой абстракции (NAL unit).* Каждая синтаксическая структура в новом стандарте помещается в логический пакет данных, называемый блоком NAL. Синтаксическая структура блоков NAL предусматривает достаточно большую гибкость для передачи видеоконтента способом, наиболее подходящим для каждого специфического вида сетей.
- *Гибкий размер слайса.* В отличие от жесткой слайсовой структуры MPEG-2 (которая уменьшает эффективность кодирования, увеличивая объем заголовочной информации и уменьшая эффективность предсказания), размеры слайсов в новом стандарте задаются очень гибко (как это было ранее в стандарте MPEG-1).
- *Гибкое упорядочивание макроблоков.* В H.264/AVC была разработана новая возможность деления изображения на области, называемые группами слайсов, причем каждый слайс становится независимо декодируемым подмножеством группы слайсов. При эффективном использовании гибкое упорядочивание макроблоков может существенно повысить устойчивость

к потере данных путем обработки пространственных зависимостей между областями, которые кодируются в каждом слайсе. Гибкое упорядочивание макроблоков может также использоваться для множества других целей.

- *Произвольное упорядочивание слайсов.* Так как каждый слайс кодированного изображения может быть декодирован независимо от других слайсов изображения (при определенных ограничениях кодирования), новый стандарт позволяет посыпать и получать слайсы в произвольном порядке друг относительно друга. Эта возможность, впервые появившаяся в необязательном приложении к стандарту H.263, может снизить задержку в приложениях реального времени, особенно при использовании на сетях, имеющих режим работы «доставка вне очереди» (например, в IP-сетях).
- *Избыточные изображения.* Для увеличения устойчивости к потере данных новый стандарт содержит возможность посылки кодером избыточного представления областей изображений, позволяя воспроизвести области изображений (обычно с некоторой потерей качества), для представления которых данные были потеряны в процессе передачи.
- *Разбиение данных.* Так как некоторая кодированная информация для представления каждой области (например, вектора движения и другая информация предсказания) имеет большую значимость для представления видео-контента, чем остальная информация, новый стандарт позволяет разделить синтаксис каждого слайса на части (до трех частей) для передачи в зависимости от категории синтаксических элементов. Эта часть стандарта построена на основе, заложенной в MPEG-4 и необязательном приложении к H.263. В новом стандарте методика упрощена благодаря использованию единого синтаксиса.
- *SP/SI синхронизация/переключение изображений.* H.264/AVC вводит новые типы изображений, которые делают возможной точную синхронизацию процесса декодирования некоторых декодеров с непрерывным видеопотоком, производимым другими декодерами, не приводя к потере эффективности декодеров, обусловленной посылкой внутрикодированного (I) изображения. Это свойство может позволить переключение работы декодера между представлениями видеоконтента с разными скоростями потока, восстановление после потерь или ошибок данных, а также использование специальных режимов, таких как быстрая перемотка вперед или назад и др.

14.6.3. Уровень сетевой абстракции

Уровень сетевой абстракции (NAL) создан для обеспечения «дружественности к сетям», позволяя простую и эффективную настройку к использованию уровня видеокодирования (VCL) в широком спектре систем.

Уровень сетевой абстракции облегчает возможность переносить данные уровня видеокодирования H.264/AVC на транспортном уровне, таком как:

- RTP/IP для любых типов проводных и беспроводных Интернет-сервисов реального времени (диалоговых или потоковых);
- файловые форматы, например ISO MP4, для хранения и MMS;
- H.32X для проводных и беспроводных диалоговых сервисов;
- систем MPEG-2 для вещательных услуг и т. д.



Полная настройка видеоконтента для нужд каждого конкретного приложения находится вне области стандартизации H.264/AVC, но дизайн уровня сетевой абстракции предвосхищает множество таких настроек.

Кодированные видеоданные организованы в элементы NAL, каждый из которых представляет собой пакет, содержащий целое количество байтов. Первый байт каждого элемента NAL – заголовочный байт, который содержит информацию о типе данных в этом элементе, остальные байты содержат сами данные (данные полезной нагрузки), тип которых указан в заголовке.

Данные полезной нагрузки в элементе NAL по необходимости перемешаны с байтами, препятствующими эмуляции стартовых кодов.

Определение структуры элемента NAL указывает общий формат для использования как в пакетных, так и в потоковых транспортных системах; последовательность элементов NAL, генерируемая кодером, называется потоком элементов NAL.

Некоторые системы (например, H.320 и MPEG-2|H.222.0) требуют доставки всего или части потока элементов NAL как упорядоченного потока байтов или битов, в котором положение границ элементов NAL следует идентифицировать среди самих кодированных данных.

Для использования в таких системах новый стандарт определяет формат байтового потока. В этом формате каждому элементу NAL предшествует специфический трехбайтовый код, называемый префиксом стартового кода. Тогда границы элемента NAL могут быть определены с помощью поиска в кодированных данных уникального префикса стартового кода. Использование байтов, препятствующих эмуляции стартовых кодов, гарантирует, что префикс стартового кода однозначно идентифицирует начало нового элемента NAL.

Небольшое количество дополнительных данных (один байт на видеоизображение) также добавляется для того, чтобы позволить декодерам, работающим в системах с потоками битов без выравнивания по границе байта, восстанавливать необходимое выравнивание из данных в потоке. Дополнительные данные также могут вставляться в поток байтового формата, если это необходимо для увеличения количества посылаемых данных и для достижения более быстрого восстановления выравнивания байтов.

В других системах (например, IP/RTP) кодированные данные переносятся пакетами, которые собираются системным транспортным протоколом, и идентификация границ элементов NAL в пакетах может быть произведена без использования префиксов стартовых кодов. В таких системах включение этих префиксов снижало бы пропускную способность данных. Вместо этого элементы NAL могут переноситься в пакетах данных без стартовых префиксов.

Элементы NAL классифицируются на VLC (уровень видеокодирования) и не VLC. Первые содержат данные, представляющие величины отсчетов видеоизображений, вторые – любую связанную дополнительную информацию, такую как наборы параметров (важные заголовочные данные, которые применяются к большому количеству VLC элементам NAL) и дополнительная расширенная информация (временная информация и другие дополнительные данные, которые могут расширять используемость декодируемого видеосигнала, но не являются необходимыми для декодирования величин отсчетов в видеоизображениях).

Наборы параметров содержат информацию, которая должна редко изменяться, и позволяют декодировать большое количество VLC-элементов NAL. Есть два типа наборов параметров:

- наборы параметров последовательности, которые применяются к ряду последовательно закодированных видеоизображений, называемых кодированная видеопоследовательность, и
- наборы параметров изображений, которые применяются при декодировании одного или более отдельных изображений в кодированной видеопоследовательности.

Механизм наборов параметров последовательностей и изображений разделяет передачу редко меняющейся информации и передачу кодированного представления значений отсчетов в видеоизображениях.

Множество элементов NAL в определенной форме называется единицей (элементом) доступа. Декодирование каждой единицы доступа дает одно декодированное изображение.

Кодированная видеопоследовательность состоит из набора единиц доступа, которые последовательно идут в потоке элементов NAL и используют только один набор параметров последовательности. Каждая кодированная видеопоследовательность может быть декодирована независимо от других видеопоследовательностей при наличии необходимого набора параметров, который может быть передан «в полосе» или «вне полосы». В начале кодированной видеопоследовательности располагается единица доступа, называемая *мгновенное обновление декодирования* (instantaneous decoding refresh, IDR). Эта единица доступа содержит внутрикодированное (intra) изображение – кодированное изображение, которое может быть декодировано без декодирования каких-либо предыдущих кадров в потоке элементов NAL, и присутствие этой единицы доступа означает, что ни одно последующее изображение не потребует для декодирования ссылки на изображения, предшествующие внутрикодированному, которое содержится в этой единице доступа.

14.6.4. Уровень видеокодирования

Хотя в стандарте H.264/AVC нет явного определения кодирующего и декодирующего устройств, однако кодер и декодер должны включать функциональные элементы, приведенные на рис. 14.26 и рис. 14.27.

Кодер включает две ветви обработки видеопотока:

- канал прямого кодирования (светлые блоки);
- канал реконструкции видеоизображения (темные блоки).

На вход кодера поступает кадр Fn.

Как и во всех предыдущих стандартах видеокодирования, начиная с H.261, построение уровня видеокодирования следует так называемой модели блочного гибридного видеокодирования, когда каждое закодированное изображение представляется участками блочной формы, состоящими из связанных отсчетов яркости и цветности и называемых *макроблоками* размером 16×16 пикселов в исходном изображении. Базовый алгоритм кодирования источника является гибридным, включая межкадровое предсказание, использующее времененную статисти-

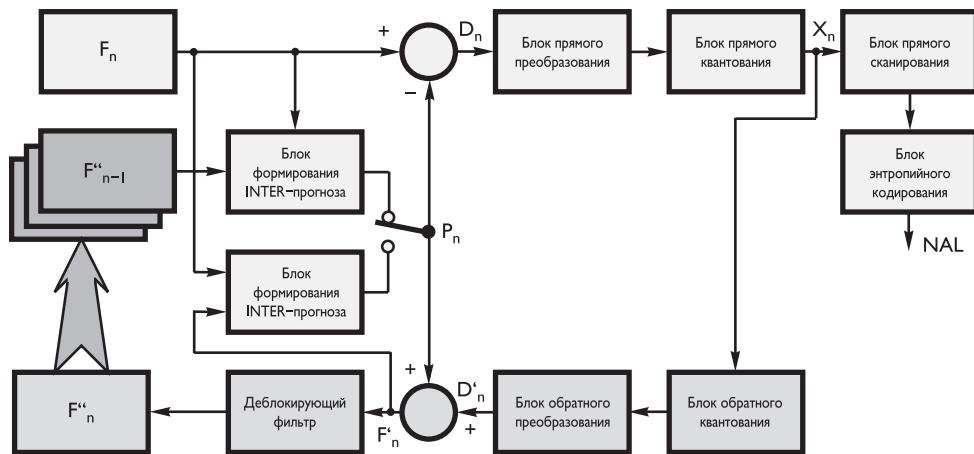


Рис. 14.26. Функциональная схема кодера H.264/AVC

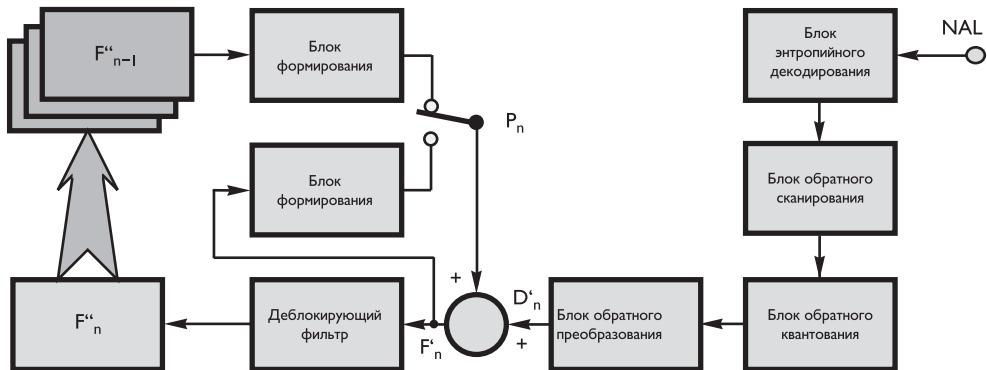


Рис. 14.27. Функциональная схема декодера H.264/AVC

ческую зависимость, и кодирование с преобразованием остатков предсказания, использующее пространственные статистические связи. На уровне видеокодирования нет ни одного элемента кодирования, который обеспечивал бы большую часть существенного увеличения эффективности кодирования по отношению к предшествующим стандартам видеокодирования. Введено множество небольших улучшений, которые вместе приводят к существенному росту эффективности кодирования.

Каждый макроблок может быть обработан в двух режимах: INTRA или INTER. В любом режиме прогноз макроблока P_n формируется на основе восстановленного кадра.

В режиме INTRA прогноз формируется из выборок текущего кадра n , предварительно закодированных в кодере (до деблокирующего фильтра) и восстановленных в декодере (F'_n).

В режиме INTER прогноз формируется с учетом изменений, которые произошли в текущем кадре по сравнению с одним или несколькими предыдущими (или последующими). Кадры, служащие для прогноза, должны быть предва-

рительно закодированы и восстановлены. В блоках формирования INTRA- и INTER-прогноза происходит выбор наиболее подходящего способа в зависимости от типа кодируемого кадра. Задача выбора решается разработчиком видеокодека самостоятельно (критерии выбора стандартом не обозначены).

Полученный прогноз P_n вычитается из текущего макроблока. В результате вычисляется макроблок остаточных коэффициентов D_n . Этот макроблок поступает в преобразователь, где происходит квазиортогональное частотное преобразование остаточных коэффициентов. Таким образом, значительно уменьшается вычислительная сложность алгоритма, но в процессе сжатия появляются дополнительные искажения. Частотные коэффициенты квантуются (масштабируются), что позволяет произвести сжатие видеоданных с потерями. Полученный набор преобразованных и квантованных коэффициентов X_n служит исходным для обратного канала реконструкции данных. В дальнейшем коэффициенты переупорядочиваются в блоке прямого сканирования. В результате коэффициенты выстраиваются в линейный массив в порядке возрастания. Затем происходит энтропийное кодирование упорядоченного массива коэффициентов, что обеспечивает сжатие данных без потерь. Закодированные коэффициенты вместе с дополнительной информацией, требуемой для правильного декодирования макроблока (режима прогноза, коэффициент квантования и т. д.), составляют сжатый битовый поток данных (bitstream) уровня сетевой абстракции (NAL). Этот поток может передаваться по каналу связи либо быть записан на любой носитель для хранения.

В канале реконструкции вначале происходит обратное квантование, затем обратное частотное преобразование. В итоге получаются восстановленные разностные коэффициенты D'_n . Они суммируются с прогнозом P_n , и это позволяет получить восстановленный кадр F'_n . Необходимо отметить, что восстановленный кадр не является идентичными исходному. В него внесены искажения, обусловленные квантованием и квазиортогональным частотным преобразованием. Именно такой кадр будет получен на приемной стороне, и поэтому именно его необходимо использовать для формирования прогноза в режиме INTRA.

При обработке кадра по макроблокам возникают специфические искажения (блочность), проявляющиеся в резких перепадах значений коэффициентов на границе между макроблоками. Для их уменьшения используется деблокирующий фильтр. Полученный после него восстановленный кадр служит в качестве опорного для формирования прогноза в режиме INTER. Отметим, что невозможно использовать деблокирующий фильтр в режиме INTRA, так как на момент формирования прогноза в этом режиме обработанной является только часть макроблоков и полной информации о границе между макроблоками нет.

Декодер получает сжатый цифровой поток уровня сетевой абстракции NAL. Декодер осуществляет обратные процедуры по отношению к кодеру. Стоит только отметить, что в блоках формирования INTRA- и INTER-прогнозов нет элементов, отвечающих за анализ кадра. Информация о конкретном режиме и способе его реализации извлекается из цифрового потока.

Кодированная видеопоследовательность в H.264/AVC состоит из последовательности *кодированных изображений*. Кодированное изображение может представлять или целый кадр, или одно *поле*, как это было в случае видео MPEG-2.

Обычно видеокадр можно считать состоящим из двух построчно перемешанных полей — верхнего и нижнего. Верхнее поле состоит из строк с четными номерами $0, 2, \dots, H/2 - 1$, где H — количество строк в кадре. Нижнее поле содержит строки с нечетными номерами (начиная со второй строки кадра). Если два поля кадра были захвачены в разные моменты времени, кадр называется кадром с чересстрочной разверткой, в противном случае он называется кадром с прогрессивной разверткой.

Зрительная система человека воспринимает сцену в терминах яркостной и цветовой информации раздельно, причем чувствительность к деталям в яркости существенно выше, чем в цветности. Система передачи видео может быть спроектирована с использованием этой особенности (так и делается как в обычном аналоговом ТВ, так и в цифровом). В H.264/AVC, как и в предшествующих стандартах, это осуществляется путем использования цветового пространства YCrCb и уменьшения пространственного разрешения цветовой информации Cr и Cb в два раза по вертикали и по горизонтали (модель цветности 4 : 2 : 0), точность представления отсчетов яркости и цветности — 8 битов. Стандартом предусмотрена возможность использования моделей цветности 4 : 2 : 2 и 4 : 4 : 4, а также точности представления отсчетов до 12 битов.

Изображение делится на макроБЛОКИ фиксированного размера, которые охватывают прямоугольные области изображения 16×16 отсчетов яркостной компоненты и 8×8 отсчетов каждой из двух цветостных компонент (для формата цветности 4 : 2 : 0). Это деление на макроБЛОКИ было принято во всех предшествующих стандартах видеокодирования, начиная с H.261. МакроБЛОКИ являются основными строительными блоками стандарта, для которых определяется процесс декодирования.

Слайсы представляют собой последовательности макроБЛОКОВ, которые обрабатываются в порядке сканирования раstra, если не используется *гибкое упорядочивание макроБЛОКОВ* (flexible macroblock ordering, FMO). Изображение может быть разделено на один или несколько слайсов. Изображение в H.264/AVC, таким образом, является набором из одного или более слайсов. Слайсы самодостаточны в том смысле, что если имеются активные наборы параметров последовательности и изображения, то синтаксические элементы слайса могут быть извлечены из битового потока и значения отсчетов в области изображения, которое представляет слайс, могут быть корректно декодированы без использования данных других слайсов при условии, что используемые опорные изображения идентичны на кодере и декодере. Некоторая информация из других слайсов может понадобиться для деблокинговой фильтрации по границам слайса.

FMO изменяет способ деления изображения на слайсы и макроБЛОКИ путем использования концепции *слайсовых групп*. Каждая слайсовая группа — это множество макроБЛОКОВ, определенное с помощью *таблицы распределения макроБЛОКОВ по слайсовым группам*, которая определена содержанием набора параметров изображения и некоторой информацией из заголовка слайса. Таблица распределения макроБЛОКОВ по слайсовым группам состоит из идентификационного номера слайсовой группы для каждого макроБЛОКА изображения, указывая, какой слайсовой группе принадлежит соответствующий макроБЛОК. Каждая слайсовая группа может подразделяться на один или несколько слайсов таким образом, что слайс является последовательностью макроБЛОКОВ в составе

этой слайсовой группы, которые обрабатываются в порядке сканирования растра. (Случай, когда FMO не используется, можно рассматривать как простой специальный случай FMO, когда все изображение состоит из одной слайсовой группы.)

При использовании FMO изображение можно разбить на множество групп макроблоков, таких как перемежающиеся слайсы, рассредоточенные макроблоки, одна или более слайсовых групп «переднего плана» и «остальные» слайсовые группы (таким образом, например, выделяются области интереса), распределение макроблоков по слайсовым группам в шахматном порядке (этот случай может быть полезным в приложениях видеоконференцсвязи, слайсовые группы передаются в разных пакетах, которые могут теряться).

Независимо от использования FMO, каждый слайс может кодироваться с использованием следующих типов:

- I-слайс — слайс, все макроблоки которого закодированы с использованием внутреннего (intra) предсказания;
- P-слайс — помимо типов кодирования I-слайса, некоторые макроблоки могут быть закодированы с использованием одностороннего предсказания;
- B-слайс — помимо типов кодирования P-слайса, некоторые макроблоки могут быть закодированы с использованием двунаправленного предсказания.

Описанные выше три типа кодирования очень близки к типам, использующимся в предшествующих стандартах видеокодирования, за исключением использования опорных кадров. Следующие два типа кодирования для слайсов являются новыми:

- SP-слайс — так называемый переключаемый P-слайс, который кодируется таким образом, что становится возможным эффективное переключение между различными заранее кодированными изображениями;
- SI-слайс — так называемый переключаемый I-слайс, который делает возможным точное совпадение макроблоков в SP-слайсе для произвольного доступа и восстановления ошибок.

Все отсчеты яркости и цветности макроблока предсказываются либо в пространстве, либо во времени, результирующий остаток предсказания кодируется с использованием преобразования. Для кодирования с преобразованием каждая цветовая компонента сигнала остатка предсказания делится на меньшие блоки 4×4 . Каждый блок подвергается целочисленному преобразованию, коэффициенты преобразования квантуются и кодируются с помощью энтропийного кодирования.

Внутрикадровое предсказание

Каждый макроблок может передаваться в одном из нескольких режимов кодирования в зависимости от типа кодирования слайса. Во всех типах кодирования слайса поддерживаются типы внутрикадрового кодирования, обозначаемые Intra_4 \times 4 или Intra_16 \times 16, а также предсказание цветности и I-PCM-предсказание.

Режим Intra_4 \times 4 основан на предсказании каждого блока яркости 4 \times 4 отдельно и подходит для кодирования частей изображения с существенными

деталями. Режим Intra_16 × 16, с другой стороны, осуществляет предсказание блока яркости 16 × 16 целиком и более подходит для очень гладких областей изображения. В дополнение к этим двум типам предсказания яркости вводится отдельный режим предсказания цветности. Как альтернатива режимам кодирования Intra_4 × 4 и Intra_16 × 16, режим кодирования I-PCM позволяет кодеру просто обходить процессы предсказания и кодирования с преобразованием и вместо этого прямо посыпать значения кодируемых отсчетов.

В отличие от некоторых предшествующих стандартов видеокодирования (а именно, H.263 и MPEG-4), в которых внутрикадровое предсказание проводилось в пространстве преобразования, внутрикадровое предсказание в H.264/AVC всегда проводится в пространственной области с помощью ссылок на соседние отсчеты ранее закодированных блоков, которые находятся слева и/или выше предсказываемого блока.

При использование режима Intra_4 × 4 каждый блок 4 × 4 предсказывается по пространственно соседним отсчетам, как показано на рис. 14.28. 16 отсчетов блока 4 × 4, помеченные буквами a–p, предсказываются с использованием ранее декодированных отсчетов соседних блоков, отмеченных буквами A–H по горизонтали и I–L по вертикали.

Для каждого блока 4 × 4 используется один из 9 режимов предсказания. В дополнение к «DC»-предсказанию (одно значение используется для предсказания всего блока 4 × 4), имеются 8 режимов «направленного» предсказания, как показано на рис. 14.29. Эти режимы полезны для предсказания направленных структур в изображении, таких как границы, расположенные под различными углами.

Режим 0. Верхние отсчеты A, B, C, D экстраполируются вертикально.

Режим 1. Левые отсчеты I, J, K, L экстраполируются горизонтально.

Режим 2. Все отсчеты в прогнозе Р являются средним из выборок A...D и I...L (DC-предсказания).

Режим 3. Отсчеты интерполируются под углом в 45° в направлении между нижним левым и верхним правым отсчетами.

Режим 4. Отсчеты интерполируются под углом 45° вниз и направо.

Режим 5. Интерполяция отсчетов производится под углом приблизительно 26,6° влево к вертикальному направлению (отношение ширины к высоте равно 1/2).

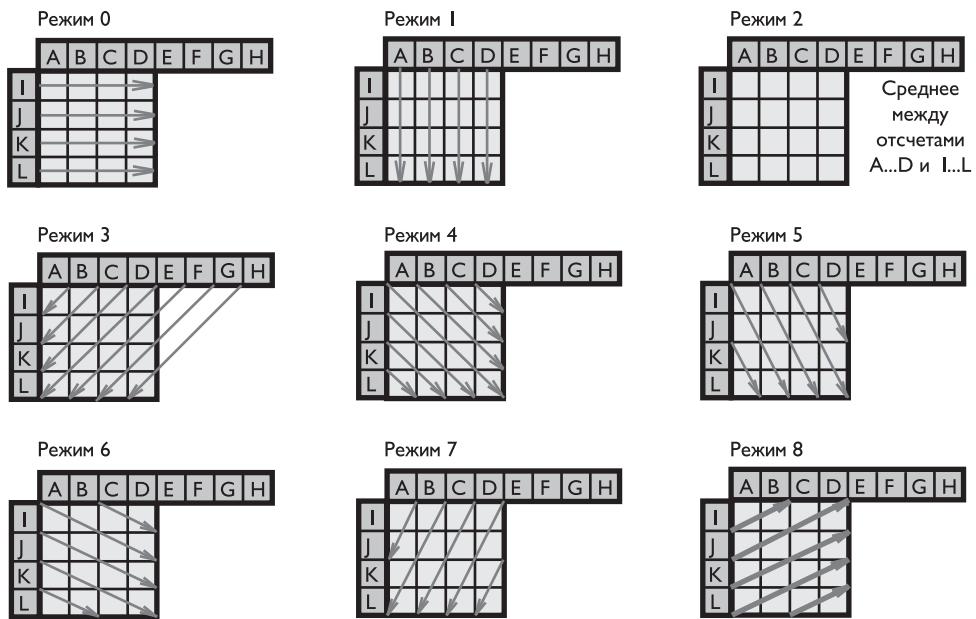
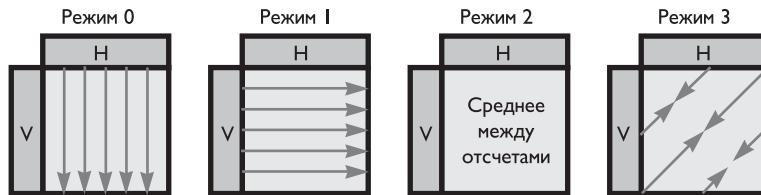
Режим 6. Интерполяция отсчетов производится под углом приблизительно 26,6° вниз по отношению к горизонтальному направлению.

Режим 7. Интерполяция отсчетов производится под углом приблизительно 26,6° вправо от вертикального направления.

Режим 8. Интерполяция отсчетов производится под углом приблизительно 26,6° выше горизонтального направления.

	A	B	C	D	E	F	G	H
I	a	b	c	d				
J	e	f	g	h				
K	i	j	k	l				
L	m	n	o	p				

Рис. 14.28. Маркировка отсчетов для блока прогноза размером 4 × 4

Рис. 14.29. Режимы предсказания Intra 4×4 Рис. 14.30. Режим формирования прогноза INTRA-блоков размером 16×16

Стрелки на рис. 14.29 указывают направление предсказания в каждом режиме. В режимах 3 — 8 отсчеты прогноза формируются из средневзвешенных выборок A...L. Например, если выбран режим 4, то отсчет прогноза d (рис. 14.28) высчитывается по формуле:

$$d = \text{round} \left(\frac{B}{4} + \frac{C}{2} + \frac{D}{4} \right). \quad (14.8)$$

При использовании режима Intra 16×16 предсказывается целиком яркостная компонента макроблока. Поддерживаются 4 режима предсказания. Режим 0 (вертикальное предсказание), режим 1 (горизонтальное предсказание) и режим 2 (DC-предсказание) определены аналогично соответствующим режимам предсказания Intra 4×4 за исключением того, что вместо 4 соседей на каждой стороне для предсказания блока 4×4 , используются 16 соседних отсчетов на каждой стороне для предсказания блока 16×16 . Четвертый режим предсказания — так называемое плоскостное (plane) предсказание. На рис. 14.30 приведены диаграммы прогноза блоков яркости 16×16 .

Режим 0. Экстраполяция верхних выборок H.

Режим 1. Экстраполяция левых выборок V.

Режим 2. Среднее между верхними H и левыми выборками V.

Режим 3. Линейное сглаживание между верхними H и левыми выборками V.

Отсчеты цветности макроблока предсказываются с помощью методики, аналогичной предсказанию яркости в режиме Intra_16 × 16, так как цветность обычно является гладкой функцией на достаточно больших областях. Используется также четыре режима формирования прогноза. Необходимо отметить, если любой из яркостных блоков макроблока кодируется в Intra-режиме, то и оба цветоразностных блока также должны быть закодированы в Intra-режиме.

Внутрикадровое предсказание (и все остальные формы предсказания) не используется поперек границ слайса для того, чтобы сохранить слайсы независимыми друг от друга.

Межкадровое предсказание в P-слайсах

В дополнение к типам макроблоков с внутрикадровым кодированием определены различные типы P-макроблоков как типы кодирования с компенсацией движения (предсказательные). Каждый тип P-макроблока соответствует специальному делению макроблока на блоки, используемые для предсказания с компенсацией движения. Синтаксис поддерживает деление на блоки яркости размером 16 × 16, 16 × 8, 8 × 16 и 8 × 8 отсчетов.

Если выбрано деление на блоки 8 × 8 отсчетов, то передается дополнительный синтаксический элемент для каждого блока 8 × 8. Этот синтаксический элемент описывает, производится ли дальнейшее деление блока 8 × 8 на блоки 8 × 4, 4 × 8 или 4 × 4 отсчета яркости и соответствующие отсчеты цветности. На рис. 14.31 приводятся варианты таких разбиений.

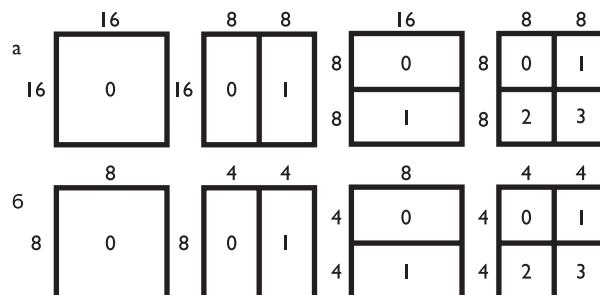


Рис. 14.31. Разбиение макроблоков 16 × 16 – а) и блоков 8 × 8 – б)

На рис. 14.32 приведен пример разбиения остаточного кадра (до процедуры компенсации движения). Кодер выбирает «лучший» размер блока для каждой части кадра, т. е. такой размер, который минимизирует кодирование остаточных коэффициентов и векторов движения. В областях кадра, где изменения незначительны (остаточные коэффициенты кажутся серыми), выбирается размер блока 16 × 16; в областях с большими изменениями (остаточные коэффициенты кажутся черными или белыми) выбираются меньшие размеры блоков.

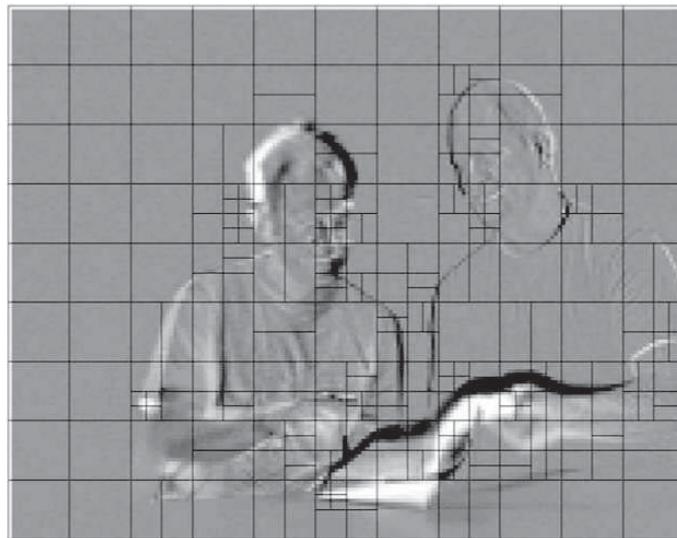


Рис. 14.32. Пример оптимального разбиения кадра изображения

Каждый блок цветности разбивается таким же образом, как яркостный. Однако вертикальные и горизонтальные размеры такого блока в два раза меньше (для модели цветности $4 : 2 : 0$), т. е. если, например, блок яркости имеет размер 8×16 , то соответствующий ему блок цветности имеет размер 4×8 , а яркости блоку 4×4 соответствует блок цветности 2×2 . Горизонтальные и вертикальные составляющие каждого вектора движения (одного на блок) делятся на два, когда применяются к блокам цветности.

Сигнал предсказания для каждого кодированного с предсказанием блока яркости $M \times N$ получается путем сдвига области соответствующего опорного изображения, который определяется вектором движения (параллельного переноса) и индексом опорного изображения. Таким образом, если макроблок кодируется с использованием 4 блоков 8×8 и каждый блок далее поделен на 4 блока 4×4 , то для Р-макроблока могут передаваться максимум 16 векторов движения.

Точность компенсации движения — четверть расстояния между отсчетами яркости. Когда вектор движения указывает на целочисленное положение отсчета, сигнал предсказания состоит из соответствующих отсчетов опорного изображения, в противном случае соответствующие отсчеты получаются с помощью интерполяции значений на нецелых позициях. Величины предсказания в полупиксельных позициях получаются с помощью применения одномерного 6-точечного КИХ-фильтра с коэффициентами $1/32, -5/32, 5/8, 5/8, -5/32, 1/32$. Рис. 14.33 поясняет введение в блок яркости полупиксельных выборок.

Например, полуниксельная выборка b рассчитывается из 6 горизонтальных целочисленных выборок E, F, G, H, I и J по формуле:

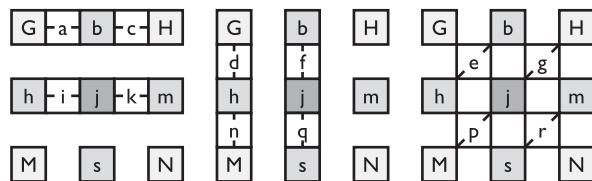
$$b = \text{round} \left[\frac{1}{32} (E - 5F + 20G + 20H - 5I + J) \right]. \quad (14.9)$$

Точно так же выборка h интерполируется на основе выборок A, C, G, M, R и T . После формирования полупиксельных выборок, смежных с целочисленными отсчетами, производится вычисление остальных полупиксельных выборок (например, выборки в позиции j) с интерполированием 6 горизонтальных или вертикальных полупиксельных выборок, полученных на первом шаге. Например, выборка j формируется на основе выборок cc, dd, h, m, ee и ff .

Величины предсказания в четвертьпиксельных позициях получаются с помощью усреднения значений в целочисленных и полуピксельных позициях.). Они формируются на основе двух горизонтальных или вертикальных смежных полу- или целочисленных выборок (например, a, c, i, k и d, f, n, q на рис. 14.34). Например:

$$a = \text{round} \left[\frac{1}{2}(G + b) \right], \quad e = \text{round} \left[\frac{1}{2}(h + b) \right]. \quad (14.10)$$

Рис. 14.34. Введение в блок яркости четвертьпиксельных выборок



Значения для предсказания компонент цветности всегда реализуются с помощью билинейной интерполяции. Так как сетка дискретизации цветности для формата кадра YCrCb 4 : 2 : 0 имеет меньшее разрешение по сравнению с решеткой дискретизации яркости, смещение цветности должно иметь точность восьмой части отсчета.

Интерполирование выборок в $1/8$ -пиксельных интервалах производится на основе целочисленных выборок в каждом цветоразностном компоненте. В этом случае используется линейная интерполяция для формирования цветоразностных выборок с $1/8$ -пиксельной точностью (рис. 14.35).

Для точки a аппроксимация производится на основе целочисленных пикселов A, B, C и D в соответствии с соотношением:

$$a = \text{round} \left\{ \frac{1}{64} [(8 - d_x)(8 - d_y)A + d_x(8 - d_y)B + (8 - d_x)d_yC + d_xd_yD] \right\}. \quad (14.11)$$

Более точное предсказание движения с использованием целопиксельного, полуликсельного и четвертьпиксельного предсказания сигналов яркости и дополнительно 1/8-пиксельное предсказание сигнала цветности представляет собой одно из главных улучшений по отношению к предшествующим стандартам по нескольким причинам. Наиболее очевидная причина — более точное представление движения. Другая причина — большая гибкость фильтрации предсказания. Целопиксельное, полуликсельное и четвертьпиксельное предсказание представляет различные степени низкочастотной фильтрации, которые автоматически выбираются в процессе поиска движения. В этом отношении шеститочечный фильтр оказывается лучшим выбором среди необходимых вариантов фильтрации, и имеет способность сохранять высокочастотные компоненты.

Синтаксис допускает использование так называемых векторов движения за границами изображения, то есть векторов движения, которые указывают за границы изображения. В этом случае опорное изображение экстраполируется за границы путем повторения граничных отсчетов до интерполяции.

Компоненты векторов движения кодируются дифференциально с помощью медианного или направленного (directional) предсказания относительно соседних блоков. Предсказание компонент векторов движения (или любые другие формы предсказания) не осуществляется через границы слайсов.

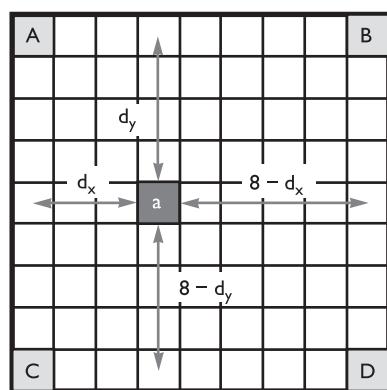
Синтаксис поддерживает предсказание с компенсацией движения по нескольким опорным изображениям. То есть более чем одно ранее закодированное изображение может быть использовано в качестве опорного для предсказания с компенсацией движения.

Рис. 14.35. Введение в блок цветности 1/8-пиксельных выборок

В дополнение к описанным выше режимам макроблоков с компенсацией движения P-макроблок может быть закодирован в так называемом режиме P_Skip (пропущенный P). Для этого типа кодирования не передается ни квантованный сигнал ошибки предсказания, ни вектор движения, ни индекс опорного изображения. Вектор движения, используемый для восстановления P_Skip макроблока, аналогичен предсказателю вектора движения для блока 16 × 16. Полезный эффект такого определения типа кодирования P_Skip состоит в том, что большие области без изменений или с постоянным вектором движения, таким как при медленном панорамировании, могут быть представлены очень малым количеством битов.

Межслайдовое предсказание в B-слайсах

По сравнению с предшествующими стандартами видеокодирования, в H.264/AVC концепция B-слайсов обобщена. Например, другие изображения могут ссылаться на изображения, содержащие B-слайсы при предсказании с компенсацией движения. Таким образом, существенная разница между B- и P-слайсами состоит



в том, что B-слайсы кодируются так, что некоторые макроблоки или блоки могут использовать взвешенное среднее двух различных значений предсказания с компенсацией движения для построения сигнала предсказания.

В B-слайсах поддерживается четыре типа межкадрового предсказания: список 0, список 1, двунаправленное предсказание и прямое предсказание. В режиме двунаправленного предсказания сигнал предсказания формируется взвешенным усреднением сигналов предсказания с компенсацией движения из списка 0 и списка 1. Режим прямого предсказания выводится из ранее переданного синтаксического элемента и может быть предсказанием из списка 0, списка 1 или двунаправленным.

B-слайсы используют такое же разбиение макроблоков, как и P-слайсы. Наряду с $P_{16 \times 16}$, $P_{16 \times 8}$, $P_{8 \times 16}$, $P_{8 \times 8}$ типами внутрикадрового кодирования имеются двунаправленное предсказание и так называемое прямое предсказание. Для каждого блока 16×16 , 16×8 , 8×16 и 8×8 метод предсказания (список 0, список 1, двунаправленное) может быть выбран независимо. Блок 8×8 B-макроблока также может быть закодирован в прямом (direct) режиме. Если сигнал ошибки предсказания не передается в режиме прямого (direct) кодирования макроблока, то этот режим также называется B_Skip (пропущенный B), и макроблок может быть закодирован очень эффективно аналогично режиму P_Skip в P-слайсах. Кодирование векторов движения в B-слайсах похоже на кодирование векторов движения в P-слайсах с соответствующими изменениями, так как соседние блоки могут быть закодированы с использованием различных режимов предсказания.

Предсказания векторов движения

Векторы движения соседних блоков зачастую оказываются коррелированными, поэтому вектор-предсказание MVp формируется на основе ранее вычисленных векторов движения и передается только вектор разности MVD между истинным значением вектора и предсказанным значением.

На рис. 14.36 варианты оценки вектора-предсказания для блока E по векторам движения соседних блоков одинакового и разных размеров.

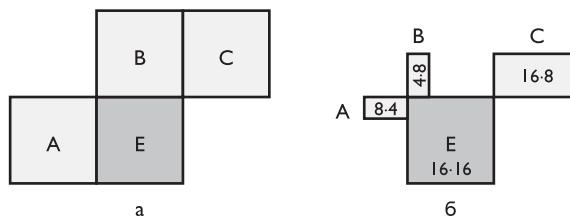


Рис. 14.36. Текущие и соседние блоки одинакового размера — а) и разного размера — б)

Вектор-прогноз определяется следующим образом:

- Для блоков всех размеров, исключая размеры 16×8 и 8×16 , вектор MVp устанавливается равным медиане векторов движения A, B и C. Этот алгоритм используется также для пропущенных макроблоков размером 16×16 .
- Для блоков размером 16×8 вектор MVp для верхнего блока размером 16×8 прогнозируется по вектору блока B, а для нижнего блока размером 16×8 — по вектору блока A.

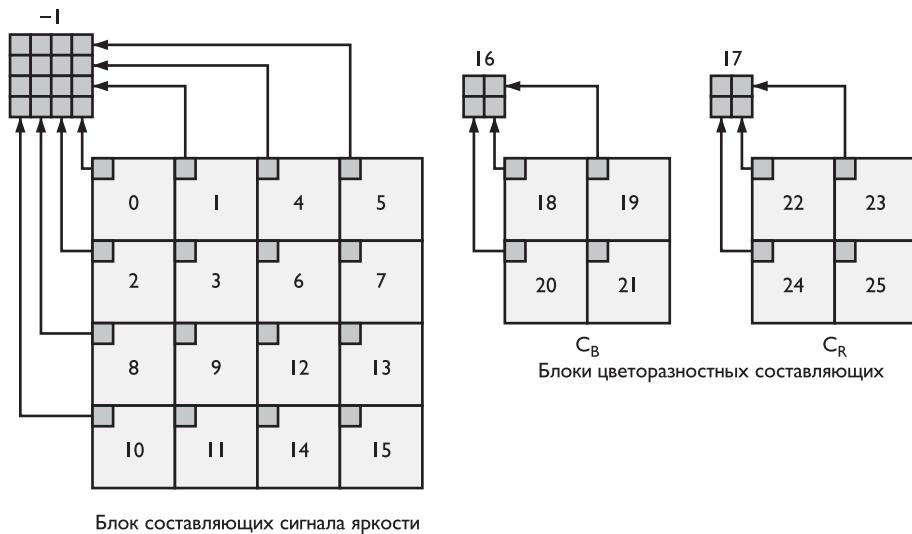


Рис. 14.37. Порядок сканирования составляющих блоков макроблока

- Для блоков размером 8×16 вектор MVp для левого блока размером 8×16 прогнозируется по вектору блока A, а для правого блока размером 8×16 – по вектору блока C.

14.6.5. Преобразование, масштабирование и квантование

Аналогично предшествующим стандартам видеокодирования H.264/AVC использует кодирование с преобразованием остатков предсказания.

Стандарт определяет три вида преобразования в зависимости от типа обрабатываемых блоков:

- преобразование блока размером 4×4 нулевых (DC) яркостных коэффициентов, полученных из яркостных макроблоков размером 16×16 , сформированных в режиме Intra;
- преобразование блока размером 2×2 нулевых (DC) цветоразностных коэффициентов, полученных из цветоразностных блоков любого типа;
- преобразование для всех остальных разностных блоков размером 4×4 .

Данные в пределах макроблока передаются в порядке, показанном на рис. 14.37.

Если макроблок размером 16×16 обрабатывается в режиме Intra, то вначале передается блок DC-коэффициентов размером 4×4 . Он маркируется значением «-1». Затем передаются остальные (0–15) яркостные блоки в порядке, показанном на этом рисунке (с обнуленными DC-коэффициентами). Блоки 16 и 17 содержат массивы размером 2×2 DC-коэффициентов, полученных из цветоразностных C_b и C_r составляющих соответственно. Наконец, передаются оставшиеся цветоразностные блоки 18 и 25 (с обнуленными DC-коэффициентами).

В стандарте H.264/AVC преобразование применяется к блокам 4×4 , и вместо дискретного косинусного преобразования (ДКП) используется сепарабельное целочисленное преобразование, близкое по свойствам к ДКП.

Задаваемое стандартом преобразование входного массива \vec{X} размером 4×4 имеет следующий вид:

$$\vec{Y} = (C \vec{X} C^T) \otimes E = \left\{ \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{bmatrix} [X] \times \begin{bmatrix} 1 & 2 & 1 & 1 \\ 1 & 1 & -1 & -2 \\ 1 & -1 & -1 & 2 \\ 1 & -2 & 1 & -1 \end{bmatrix} \right\} \otimes \begin{bmatrix} a^2 & \frac{ab}{2} & a^2 & \frac{ab}{2} \\ \frac{ab}{2} & \frac{b^2}{4} & \frac{ab}{2} & \frac{b^2}{4} \\ a^2 & \frac{ab}{2} & a^2 & \frac{ab}{2} \\ \frac{ab}{2} & \frac{b^2}{4} & \frac{ab}{2} & \frac{b^2}{4} \end{bmatrix},$$

$$a = \frac{1}{2}, \quad b = \sqrt{\frac{2}{5}}. \quad (14.12)$$

Это преобразование обеспечивает почти такую же степень сжатия, что и ДКП, но обладает целым рядом преимуществ:

- «ядро» преобразования $C \vec{X} C^T$ может быть выполнено в целочисленной арифметике при помощи только операций суммирования, вычитания и сдвига (умножение на 2);
- динамический диапазон результата действий данного преобразования такой, что везде может быть использована 16-разрядная арифметика без риска выхода за пределы диапазона ± 255 ;
- масштабирование при помощи матрицы E требует только одного умножения для каждого коэффициента, которое может быть «включено» в процесс квантования.

Обратное преобразование определяется следующим соотношением:

$$\vec{X} = \hat{C}^T (\vec{Y} \otimes \hat{E}) \hat{C} = \left[\begin{array}{cccc} 1 & 1 & 1 & \frac{1}{2} \\ 1 & \frac{1}{2} & -1 & -1 \\ 1 & -\frac{1}{2} & -1 & 1 \\ 1 & -1 & 1 & -\frac{1}{2} \end{array} \right] \times \times \left\{ [Y] \otimes \begin{bmatrix} a^2 & ab & a^2 & ab \\ ab & b^2 & ab & b^2 \\ a^2 & ab & a^2 & ab \\ ab & b^2 & ab & b^2 \end{bmatrix} \right\} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & \frac{1}{2} & -\frac{1}{2} & -1 \\ 1 & -1 & -1 & 1 \\ \frac{1}{2} & -1 & 1 & -\frac{1}{2} \end{bmatrix}. \quad (14.13)$$

Здесь матрица \vec{Y} вначале масштабируется матрицей \hat{E} путем поэлементного умножения. Операции матрицами \hat{C} и \hat{C}^T могут быть выполнены без потери точности, потому что коэффициенты матрицы \vec{Y} вначале подвергаются масштабированию.

Прямое и обратное преобразования являются ортогональными, т. е.

$$\mathbf{T}^{-1}(\mathbf{T}(\mathbf{X})) = \mathbf{X}.$$

Основной процесс кодирования с преобразованием очень похож на процесс в предшествующих стандартах. На стороне кодера процесс включает прямое пре-

образование, z-упорядочивание, масштабирование и округление (процесс квантования), за которыми следует энтропийное кодирование. На стороне декодера осуществляются обратные операции за исключением округления.

Уже было отмечено, что режимы предсказания $\text{Intra } 16 \times 16$ и режимы внутрикадрового предсказания цветности предназначены для кодирования гладких областей. По этой причине DC-коэффициенты подвергаются второму преобразованию, что приводит к тому, что преобразуются коэффициенты всего макроблока. Дополнительное преобразование 2×2 также применяется к DC-коэффициентам четырех блоков 4×4 каждой компоненты цветности.

Есть несколько причин использования меньшего размера преобразования:

- Одно из основных улучшений нового стандарта — это улучшенный процесс предсказания как для внутрикадрового, так и для межкадрового кодирования. Следовательно, остаточный сигнал имеет меньшую пространственную корреляцию. Это в целом означает, что преобразование меньше дает в плане декорреляции. Это также означает, что преобразование 4×4 также эффективно в плане устранения статистической корреляции, как и большие преобразования.
- При близких объективных возможностях компрессии, меньшее преобразование 4×4 имеет визуальное преимущество, приводящее к меньшим шумам возле границ (известным как «эффект комаров», mosquito noise, или «окантовки», ringing).
- Меньшее преобразование обладает меньшей вычислительной сложностью. Так как процесс преобразования в стандарте H.264/AVC включает только сложения и сдвиги, он определен таким образом, что устраняется несовпадение между кодированием и декодированием (это было проблемой в более ранних стандартах, использовавших ДКП 8×8).

Квантование

Стандарт H.264/AVC предполагает использование скалярного квантования, реализующего требования:

- исключить деление и арифметику с плавающей точкой;
- включить в процесс квантования масштабирование при помощи матриц E и \hat{E} .

Входной блок \vec{X} сначала преобразуется с использованием соотношения $\vec{W} = C \vec{X} C^T$ (см. формулу 14.11), и блок сформированных промежуточных коэффициентов W_{ij} квантуется и масштабируется при помощи одной операции:

$$Z_{ij} = \text{round} \left(W_{ij} \frac{PF}{Q_{\text{step}}} \right), \quad (14.14)$$

где обозначенные PF коэффициенты масштабирования $a^2 = \frac{1}{4}$, $\frac{ab}{2} = \frac{\sqrt{10}}{20}$, $\frac{b^2}{4} = \frac{1}{10}$ включены в прямое квантование и используемые в соответствии с табл. 14.29, а Q_{step} — шаг квантования.

Общее количество значений шага квантования Q_{step} равно 52, они передаются как параметр QP , приведенный в табл. 14.30. Значение Q_{step} удваивается

для каждого приращения QP на 6; Q_{step} увеличивается на 12,5% для каждого единичного приращения параметра QP . Широкий диапазон значений шага квантования позволяет кодирующему устройству точно и гибко управлять обменом между количеством передаваемых битов и качеством закодированных динамических изображений.

Таблица 14.29. Позиции коэффициентов квантования

Позиции в блоке \vec{X}	Значение коэффициента PF
(0,0), (2,0), (0,2) и (2,2)	a^2
(1,1), (1,3), (3,1) и (3,3)	$b^2/4$
В остальных случаях	$ab/2$

Таблица 14.30. Размер шага квантования в кодеке H.264/AVC

QP	0	1	2	3	4	5	6	7	8	9	10	11	12	...
Q_{step}	0,625	0,6875	0,8125	0,875	1	1,125	1,25	1,375	1,625	1,75	2	2,25	2,5	
QP	...	18	...	24	...	30	...	36	...	42	...	48	...	51
Q_{step}		5		10		20		40		80		160		224

Для исключения операции деления в соотношении (14.14) стандарт предусматривает преобразование:

$$\frac{PF}{Q_{\text{step}}} = \frac{MF}{2^{qbits}}, \quad qbits = 15 + \text{floor}\left(\frac{QP}{6}\right). \quad (14.15)$$

Здесь $\text{floor}(v)$ — целое числа v . При этом операция квантования

$$Z_{ij} = \text{round}\left(W_{ij} \frac{MF}{2^{qbits}}\right) \quad (14.16)$$

преобразуется так, что коэффициенты W_{ij} умножаются на коэффициент-фактор MF , а затем производится сдвиг вправо на величину $qbits$. В целочисленной арифметике (14.16) может быть выполнено следующим образом:

$$Z_{ij} = (W_{ij}MF + f) >> qbits, \quad (14.17)$$

где знак $>>$ определяет логический сдвиг вправо, а $f = 2^{qbits}/3$ для Intra-блоков и $f = 2^{qbits}/6$ для Inter-блоков. Величины MF для первых шести значений QP приведены в табл. 14.31.

Таблица 14.31. Коэффициент-фактор MF

QP	Позиции (0,0), (2,0), (0,2) и (2,2)	Позиции (1,1), (1,3), (3,1) и (3,3)	В остальных случаях
0	13107	5243	8066
1	11916	4660	7490
2	10082	4194	6554
3	9362	3647	5825
4	8192	3355	5243
5	7282	2893	4559

Для значений $QP > 5$ коэффициенты умножения MF будут повторяться в соответствии с табл. 14.31, а делитель 2^{qbits} будет увеличиваться в два раза для каждого приращения QP на 6. Например, $qbits = 16$ для $6 \leq QP \leq 11$, $qbits = 17$

для $6 \leq QP \leq 17$, и т. д. Следует сказать, что изменение шага квантования приблизительно на 12% также в первом приближении означает уменьшение битовой скорости примерно на 12%.

Преобразование и квантование яркостных DC-коэффициентов блока размером 4×4

Это преобразование осуществляется только для моды Intra. Если макроблок размером 16×16 обрабатывается в Intra-режиме, то каждый остаточный блок размером 4×4 сначала преобразуется «ядром» $C \vec{X} C^T$, описанным выше. Полученные яркостные DC-коэффициенты каждого блока размером 4×4 выделяются в отдельный блок и обрабатываются в соответствии с преобразованием Адамара:

$$\vec{Y}_D = \frac{1}{2} \left(\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} [W_D] \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} \right). \quad (14.18)$$

Матрица $[W_D]$ представляет собой блок DC-коэффициентов размером 4×4 , а матрица \vec{Y}_D — блок коэффициентов после преобразования и округления. После этого коэффициенты $Y_D(i, j)$ квантуются по формуле:

$$|Z_D(i, j)| = (|Y_D(i, j)| \cdot MF(0, 0) + 2f) \gg (qbits + 1), \\ \text{sign} [(Z_D(i, j))] = \text{sign} [(Y_D(i, j))], \quad (14.19)$$

где MF — множитель с координатами $(0, 0)$ из табл. 14.31, f и $qbits$ определяются как обычно (см. соотношения (14.14) и (14.16)).

Преобразование и квантование DC-коэффициентов блоков цветности размером 2×2

Каждая составляющая цветности в макроблоке состоит из четырех блоков размером 4×4 . DC-коэффициенты каждого такого блока группируются в блок размером 2×2 (W_D), а затем полученный блок преобразуется следующим образом:

$$\vec{Y}_D = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} [W_D] \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}. \quad (14.20)$$

Квантование такого блока $[Y_D]$ выполняется в соответствии с соотношением:

$$|Z_D(i, j)| = [|Y_D(i, j)| \cdot MF(0, 0) + 2f] \gg (qbits + 1), \\ \text{sign} [Z_D(i, j)] = \text{sign} [Y_D(i, j)], \quad (14.21)$$

где коэффициенты $MF(0, 0)$ определены в табл. 14.31, f и $qbits$ определяются как было приведено выше.

Обратное квантование

Масштабирующие множители PF' обратного преобразования из матрицы \hat{E} соотношения (14.12) приведены в табл. 14.32.

Эти масштабирующие множители встроены в операцию обратного квантования вместе с постоянным множителем масштаба 64 во избежание ошибок округ

ления:

$$W'_{ij} = Z_{ij} \cdot Q_{\text{step}} P F' \cdot 64. \quad (14.22)$$

Коэффициенты W'_{ij} преобразуются базовым обратным преобразованием $\hat{C}^T \hat{W}' \hat{C}$. Значения полученных коэффициентов делятся на 64, чтобы компенсировать предварительное масштабирование (это осуществляется только операциями суммирования и сдвига вправо)

Таблица 14.32. Позиции коэффициентов обратного квантования

Позиции в блоке \hat{W}_D	Значение коэффициента $P F'$
(0,0), (0,2), (2,0) и (2,2)	a^2
(1,1), (1,3), (3,1) и (3,3)	b^2
В остальных случаях	ab

Стандарт не специфицирует значения Q_{step} и $P F'$. Вместо этого определяется параметр $V = (Q_{\text{step}} \times P F' \times 64)$ для $0 \leq QP \leq 5$, и операция масштабирования принимает вид:

$$W'_{ij} = Z_{ij} \cdot V_{ij} \cdot 2^{\text{floor}(QP/6)}. \quad (14.23)$$

Величины коэффициентов V приведены в табл. 14.33.

Таблица 14.33. Коэффициенты V

QP	Позиции (0,0), (2,0), (0,2) и (2,2)	Позиции (1,1), (1,3), (3,1) и (3,3)	В остальных случаях
0	10	16	13
1	11	18	14
2	13	20	16
3	14	23	18
4	16	25	20
5	18	29	23

Коэффициент $2^{\text{floor}(QP/6)}$ в соотношении (14.20) увеличивает результат в два раза при каждом приращении величины QP на 6.

Обратное преобразование и квантование DC-коэффициентов блока яркости размером 4×4

В декодере применяется обратное преобразование Адамара (отметим, что порядок умножения не изменен):

$$\hat{W}_D = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} [Z_D] \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix}. \quad (14.24)$$

Обратное квантование выполняется следующим образом:

$$\begin{aligned} W'_D(i, j) &= W_D(i, j) \cdot V(0, 0) \cdot 2^{\text{floor}(QP/6)-2} \text{ при } QP \geq 12, \\ W'_D(i, j) &= \left[W_D(i, j) \cdot V(0, 0) + 2^{1-\text{floor}(QP/6)} \right] \gg [2 - \text{floor}(QP/6)] \text{ при } QP < 12. \end{aligned} \quad (14.25)$$

Коэффициенты V определяются табл. 14.33. Полученная после процедуры обратного квантования матрица $[W'_D]$ является блоком DC-коэффициентов, которые

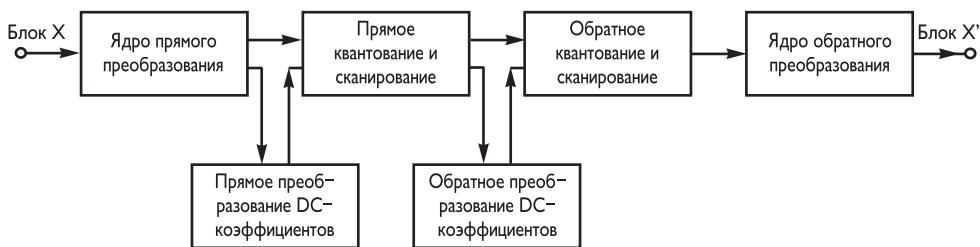


Рис. 14.38. Процесс прямого и обратного преобразования и квантования

вставляются в соответствующие блоки, и выполняется «ядро» обратного преобразования $\hat{C}^T \hat{W}'_D \hat{C}$.

В макроблоке 16×16 в mode Intra большая часть энергии концентрируется в DC коэффициентах каждого блока 4×4 , которые зачастую сильно коррелированы. После дополнительного преобразования энергия концентрируется в малом наборе существенных коэффициентов.

Обратное преобразование и квантование DC-коэффициентов блока цветности размером 2×2

Обратное преобразование осуществляется по формуле:

$$\hat{W}'_D = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} [Z_D] \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}. \quad (14.26)$$

Обратное квантование выполняется следующим образом:

$$W'_D(i, j) = W_D(i, j) \cdot V(0, 0) \cdot 2^{\text{floor}(QP/6)-1} \text{ при } QP \geq 6,$$

$$W'_D(i, j) = \left[W_D(i, j) \cdot V(0, 0) + 2^{1-\text{floor}(QP/6)} \right] \gg 1 \text{ при } QP < 6.$$

Коэффициенты, полученные после обратного квантования, размещаются в соответствующих цветоразностных блоках размером 4×4 , которые затем обрабатываются преобразованием $\hat{C}^T \hat{W}'_D \hat{C}$. Как и в случае с кодированием яркостных DC-коэффициентов в Intra-режиме, применяется дополнительное преобразование, помогающее декоррелировать блок размером 2×2 DC-коэффициентов компонент цветности и, следовательно, увеличивающее степень сжатия.

Квантованные коэффициенты преобразования блока обычно переупорядочиваются зигзагообразным сканированием и передаются с помощью методов энтропийного кодирования. 2×2 DC коэффициенты цветностных компонент упорядочиваются в порядке сканирования раstra. Все операции обратных преобразований в H.264/AVC могут быть выполнены с помощью только сложений и сдвигов 16-битных целых величин. Аналогично, только 16-битный доступ к памяти требуется для хорошей реализации процессов прямого преобразования и квантования в кодере.

На рис. 14.38 приведена структура полного процесса прямого и обратного преобразования и квантования от момента поступления на вход обрабатываемого блока X до получения на выходе восстановленного блока X' .

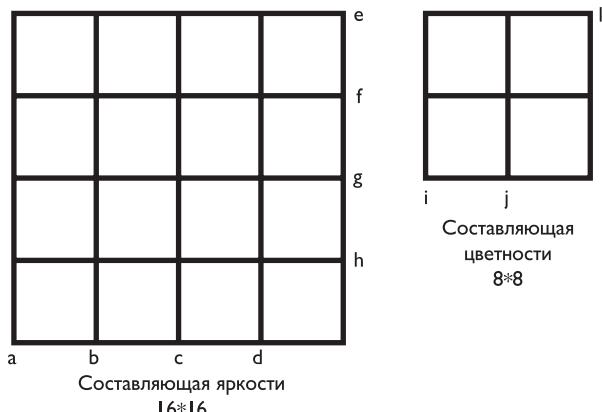


Рис. 14.39. Порядок фильтрации границ макроблока

14.6.6. Деблокирующая фильтрация

Деблокирующий фильтр применяется после обратного преобразования в кодере (при восстановлении и сохранении макроблока для формирования прогноза следующего кадра) и в декодере (при восстановлении и отображении макроблока).

Фильтр используется для сглаживания блочных артефактов. При этом:

- границы блока сглаживаются, улучшая зрительное восприятие декодированных изображений (особенно при больших коэффициентах сжатия);
- отфильтрованный макроблок используется в режиме компенсации движения деталей изображения в кодере, что значительно уменьшает величину остаточных коэффициентов.

Необходимо отметить, что в режиме Intra макроблоки фильтруются, но для формирования прогноза не используются, кроме этого границы кадра не фильтруются в любом режиме.

Фильтрация применяется к вертикальным или горизонтальным границам блоков размером 4×4 в следующем порядке:

- фильтрация четырех вертикальных границ яркостной компоненты (в порядке a, b, c, d на рис. 14.39);
- фильтрация четырех горизонтальных границ яркостной компоненты (в порядке e, f, g, h на рис. 14.39);
- фильтрация двух вертикальных границ каждой цветоразностной компоненты (i, j);
- фильтрация двух горизонтальных границ каждой цветоразностной компоненты (k, l).

Каждая операция фильтрации использует до четырех пикселов с обеих сторон границы (рис. 14.40). В зависимости от величины текущего шага квантования, режима кодирования соседних блоков и градиента выборок изображения поперек границы используются четыре варианта фильтрации.

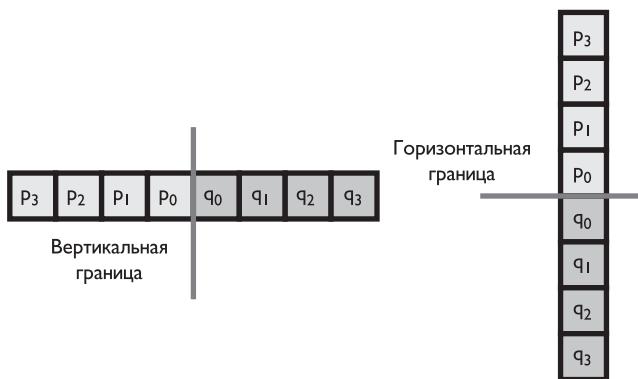


Рис. 14.40. Использование смежных пикселов при деблокинговой фильтрации

Параметры деблокирующего фильтра зависят от типа границ между блоками:

- при кодировании блоков P и/или Q границей является граница между блоками яркости 16×16 или блоками цветности 8×8 ;
- при кодировании блоков P и/или Q границей не является граница между блоками яркости 16×16 или блоками цветности 8×8 ;
- ни блок P, ни блок Q не кодируются непосредственно, но используются кодированные коэффициенты;
- ни блок P, ни блок Q не кодируются, но имеют различные ссылки;
- во всех прочих случаях деблокинговая фильтрация не используется.

На рис. 14.41 показан пример применения деблокирующей фильтрации [5.70]. Деблокирующий фильтр увеличивает эффективность сжатия незначительно: скорость выходного битового потока уменьшается не более чем на 1–3%, а пиковое отношение сигнал/шум (PSNR) увеличивается примерно на 1%. Однако субъективное качество восстановленной видеопоследовательности с деблокирующей фильтрацией значительно выше. Увеличение эффективности кодирования, обеспечиваемое деблокирующим фильтром, зависит от содержания последовательности и требуемой выходной скорости закодированного потока.

14.6.7. Энтропийное кодирование

В стандарте H.264/AVC определено несколько алгоритмов кодирования, которые применяются к различным элементам передаваемой информации. Так, элементы синтаксиса, относящиеся к кодированию кадра в целом, кодируются кодами фиксированной или переменной длины, а элементы, относящиеся непосредственно к кодированию остаточных коэффициентов, кодируются кодом переменной длины, или арифметическим кодом.

Использование кодов переменной длины

В стандарте определены два вида кодирования с переменной длиной кодового слова:



6

Рис. 14.41. Пример применения деблокирующего фильтра: а) — восстановленный кадр без фильтрации; б) — восстановленный кадр с деблокирующей фильтрацией

- экспоненциальное кодирование кодами Голомба (ExpGolomb coding);
- кодирование переменной длины с адаптацией на основе контекста (CAVLC – context-adaptive variable length coding).

Экспоненциальные коды переменной длины с регулярной структурой используются для передачи служебных элементов синтаксиса: информации о коэффициенте квантования, режиме кодирования, данных о векторах движения и т. д.

Кодовое слово в данном случае имеет следующую структуру:

$$[M][1][INFO], \quad (14.27)$$

где *INFO* — информационное поле, состоящее из *M* битов, *M* — последовательность нулей «0», также состоящая из *M* битов.

Кодовое слово создается в соответствии с формулами:

$$M = \log 2(code_num + 1), \quad INFO = code_num + 1 - 2M, \quad (14.28)$$

где *code_num* — кодируемое значение. Декодирование этой информации происходит по следующему алгоритму:

- считать *M* начальных нулей, заканчивающихся 1;
- считать *M*-разрядное поля *INFO*;
- определить кодируемое значение по формуле:

$$code_num = 2M + INFO - 1.$$

Следует отметить, что для кодируемого значения «0», поля *INFO* и *M* имеют нулевую длину.

Кодируемое значение не является непосредственно тем параметром, который получен в результате обработки кадра. Стандартом определены три способа замены параметра *v*, полученного в результате обработки кадра, кодируемым

значением $code_num$, которое является входным для алгоритма экспоненциального кодирования:

- $ue(\nu)$ — прямое соответствие без знака, $code_num = \nu$;
- $se(\nu)$ — знаковое соответствие —

$$\begin{cases} code_num = 2|v|, & v < 0; \\ code_num = 2|v| - 1, & v > 0; \end{cases} \quad (14.29)$$

- $me(\nu)$ — символьное соответствие; параметр ν соответствует $code_num$ согласно таблице, определенной в стандарте.

Каждый способ (ue , se и me) разработан так, чтобы формировать короткие кодовые слова — для часто встречающихся значений параметров и более длинные кодовые слова для менее частых.

В схеме с адаптацией на основе контекста CAVLC таблицы кодов переменной длины (VLC) для различных синтаксических элементов переключаются в зависимости от уже переданных синтаксических элементов. Так как таблицы VLC созданы так, чтобы они подходили для соответствующих условных статистик, производительность энтропийного кодирования повышена по сравнению со схемой, использующей одну таблицу VLC.

Метод CAVLC обладает нижеуказанными особенностями.

После прогнозирования, преобразования и квантования блоки обычно содержат множество нулей. CAVLC использует кодирование run-level, при котором вместо последовательности нулей передается одно число, определяющее их количество.

Самые большие ненулевые коэффициенты после зигзагообразного сканирования часто завершаются последовательностью ± 1 . CAVLC позволяет компактно представить последовательность замыкающих высоких частот, равных ± 1 .

Число ненулевых коэффициентов в соседних блоках коррелировано. Поэтому число ненулевых коэффициентов кодируется на основе таблицы поиска, которая выбирается из нескольких, в зависимости от числа ненулевых коэффициентов в соседних блоках.

Величина ненулевых коэффициентов обычно выше в начале переупорядоченного массива (около DC-коэффициента) и ниже в конце. CAVLC использует эти свойства, обеспечивая для кодирования уровня остаточного коэффициента несколько таблиц для кодирования, причем выбор той или иной таблицы происходит в зависимости от уровня предыдущего закодированного коэффициента.

Код CAVLC включает следующие поля:

- $coeff_token$ — определяет общее количество ненулевых коэффициентов в кодируемом блоке остаточных коэффициентов ($TotalCoeffs$), а также количество коэффициентов в конце блока, равных единице ($T1s$);
- $T1$ — знак единичного коэффициента в конце блока (таких полей может быть ≤ 3);
- $Level$ — значение ненулевых коэффициентов в блоке (количество полей определяется количеством ненулевых значений коэффициентов кодируемого блока);
- $TotalZeros$ — количество нулевых коэффициентов на промежутке от начала блока до последнего ненулевого коэффициента;
- run_before — количество нулей перед ненулевым коэффициентом.

Пусть, например, блок 4×4 остаточных коэффициентов определен табл. 14.34.

В результате переупорядочивания коэффициентов формируется одномерный массив 0,3,0,1,-1,-1,0,1,0,0,0,0,0,0,0,0.

Этот массив подвергается кодированию, и в результате получается битовый поток 000010001110010111101101.

Значения полей этого массива приведены в табл. 14.35. Фактически в конце блока имеется четыре единичных коэффициента, но только три из них могут быть закодированы своим знаком. Кроме этого, количество нулей перед первым ненулевым коэффициентом не передается, так как может быть вычислено на приемной стороне из уже полученной информации.

Эффективность энтропийного кодирования может быть еще улучшена при использовании контекстно-адаптирующегося двоичного арифметического кодирования (context-adaptive binary arithmetic coding, CABAC). С одной стороны, арифметическое кодирование позволяет использовать нецелое количество битов для каждого символа алфавита, что чрезвычайно выгодно для символов, вероятность которых больше 0,5. С другой стороны, использование адаптивных кодов позволяет приспосабливаться к нестационарной статистике символов. Другое важное свойство CABAC — моделирование контекста. Статистика уже закодированных синтаксических элементов используется для оценки условных вероятностей. Эти условные вероятности используются для переключения нескольких моделей оценки вероятности. В H.264/AVC ядро арифметического кодирования и связанные с ним оценки вероятностей определены как методы низкой сложности без умножений, которые используют только сдвиги и поиск по таблицам. По сравнению с CA VLC CABAC обычно обеспечивает снижение битового потока на 5–15%. Наибольший выигрыш обычно достигается при кодировании чересстрочных телевизионных сигналов.

Таблица 14.34. Пример массива остаточных коэффициентов

0	3	-1	0
0	-1	1	0
1	0	0	0
0	0	0	0

Таблица 14.35. Пример кодирования остаточных коэффициентов кодом CAVLC

Поле кода	Значение	Код
<i>coeff_token</i>	$TotalCoeffs = 5, T1s = 3$	0000100
$T1$ знак (4)	+	0
$T1$ знак (3)	-	1
$T1$ знак (2)	-	1
<i>Level</i> (1)	+1	1
<i>Level</i> (0)	+3	00100
<i>TotalZeros</i>	3	111
<i>run_before</i> (4)	1	10
<i>run_before</i> (3)	0	1
<i>run_before</i> (2)	0	1
<i>run_before</i> (1)	1	01
<i>run_before</i> (0)	1	-

Алгоритм арифметического кодирования САВАС обеспечивает хорошую степень сжатия вследствие следующих причин:

- выбор вероятностной модели для каждого элемента синтаксиса;
- адаптивная оценка вероятности, основанная на локальной статистике;
- использование арифметического кодирования.

Кодирование символа данных включает нижеследующие стадии.

- Бинаризация исходного параметра, подобная процессу преобразования символа данных в код переменной длины.
- Выбор контекстной модели для каждого бита (или «дискрета») бинаризированного символа. Контекстная модель является моделью вероятности для одного или более дискретов. Эта модель может быть выбрана из ряда доступных моделей в зависимости от статистики ранее закодированных символов.
- Непосредственно само арифметическое кодирование. Кодер шифрует каждый дискрет согласно выбранной вероятностной модели.
- Обновление вероятностной модели. Выбранная модель контекста обновляется на основании фактически закодированных данных (например, если был закодирован дискрет, равный «1», тогда частота повторения «1» увеличивается).

Контекстные модели и схемы бинаризации каждого элемента синтаксиса определены в стандарте. При этом возможно использование до 267 контекстных моделей для различных элементов синтаксиса. Некоторые модели имеют различное использование в зависимости от типа кодируемого значения. Первоначальный выбор контекстной модели производится в начале кодирования каждой независимой секции видеозображения (обычно это кадр) в зависимости от параметра квантования QP .

В целом САВАС обеспечивает гораздо более высокую эффективность кодирования по сравнению с VLC за счет большей вычислительной сложности.

Таким образом, стандарт видеокодирования H.264/AVC представляет множество улучшений стандартной методики видеокодирования, как в плане улучшения эффективности кодирования, так и в плане гибкости для эффективного использования в широком множестве типов сетей и областей приложений. Его методики уровня кодирования видео основаны на концепциях условного блочного гибридного видеокодирования с компенсацией движения, но включают некоторые существенные отличия (улучшения) по сравнению с предшествующими стандартами.

При совместном использовании эти отличия новой методики кодирования обеспечивают уменьшение битового потока примерно на 50% при сохранении качества воспроизведения по сравнению с предшествующими стандартами (особенно для приложений со сравнительно большими задержками, которые позволяют использовать обратное временное предсказание).

14.7. Некоторые проблемы реализации видеокодирующих систем

Иерархия битового потока видеокодирующих систем существующих стандартов в целом приведена в разделах 14.1–14.6. В настоящем разделе рассматриваются некоторые специфические особенности использования эффективных алгоритмов цифрового кодирования динамических изображений, к которым следует отнести:

- методику построения промежуточных кадров видеопоследовательности;
- методику регулировки потока при сжатии динамических видеоизображений;
- анализ смены сюжета;
- оценку быстродействия кодирования изображений;
- возможности кодирования стереоскопических видеоизображений в рамках совместимости с действующими стандартами.

14.7.1. Методика построения промежуточных кадров видеопоследовательности

В целом ряде прикладных задач возникает необходимость формирования промежуточных кадров видеопоследовательности. Например, в системах видеоконференцсвязи по каналам с невысокой пропускной способностью представляется предпочтительным (с точки зрения качества декодируемого изображения) снизить кадровую скорость кодируемого видео, а на приемной стороне для увеличения плавности движений на воспроизведимом изображении формировать и показывать пропущенные на кодирующей стороне кадры. Или при использовании режима так называемой «100 Гц-развертки» формировать не только полные кадры из полей (25 Гц → 50 Гц), но и промежуточные кадры между полями (50 Гц → 100 Гц).

Одним из достаточно эффективных методов формирования промежуточных кадров является метод, использующий анализ движения деталей изображения [5.73]. Используемая методика анализа движения блоков изображения очень близка к методикам, применяемым в стандартных алгоритмах видеокомпрессии. Методика предполагает несколько стадий обработки изображений.

На первой стадии производится предварительное построение интерполируемого кадра. Кадр заполняется данными по формуле, учитывающей его временное положение относительно предшествующего и последующего (если он должен находиться не посередине), — проводится усреднение опорных кадров без учета движения деталей:

$$P_i = \text{round} \left[\frac{P_{i-1} \cdot (t_{i+1} - t_i) + P_{i+1} \cdot (t_i - t_{i-1})}{(t_{i+1} - t_{i-1})} \right], \quad (14.30)$$

где P — значение пикселя (яркость, цветность), t — время кадра, индекс i относится к результирующему кадру, $i - 1$ — к предшествующему, а $i + 1$ — к последующему по времени кадру.

На следующей стадии один из опорных кадров разбивается на блоки (обычно фиксированного размера, например, 8×8 или 16×16 пикселов). Блоки могут

быть как неперекрывающимися, так и частично перекрывааться. Этот кадр будет называться текущим. Для каждого блока текущего кадра осуществляется поиск наиболее близкого участка такого же размера в другом опорном кадре. В качестве меры близости блоков может использоваться, например, средняя абсолютная или среднеквадратичная разность пикселов блоков. Хорошие результаты показало использование в качестве меры близости разности взвешенного среднего нескольких старших коэффициентов, полученных после применения к блокам ортогонального преобразования (например, косинусного или Адамара). При поиске векторов движения следует также учитывать согласованность получаемого поля векторов, так как согласованное поле более соответствует реальным векторам движения деталей в изображении. Эксперимент показал, что для получения хорошего качества интерполируемого кадра следует использовать векторы движения с точностью до половины или даже до четверти пикселя.

После того как были найдены векторы движения, производится вычисление интерполированных блоков и запись их в промежуточный кадр. В случае, когда пиксел интерполируемого кадра предсказывается более одного раза, выбирается наилучшее предсказание (например, в смысле меры близости блоков при анализе движения).

При вычислении интерполяции блока учитывается временное положение интерполируемого кадра так же, как и в случае предварительного построения кадра. Кроме того, положение блоков в опорных кадрах сдвигается на величину менее пикселя таким образом, чтобы интерполируемые пиксели попадали на сетку дискретизации формируемого кадра.

Тем не менее после всех этих процедур на формируемом кадре могут оставаться зоны, заполненные не интерполяцией, а усреднением (на первом этапе предварительного построения кадра). Эти зоны, как правило, соответствуют областям, которые были закрыты на одном из опорных кадров движущимися деталями, или областям, содержащим новые объекты. Для более корректного заполнения таких зон следует применять экстраполяцию по двум предшествующим или двум последующим кадрам видеопоследовательности.

Также некоторое улучшение при формировании промежуточного кадра дает повторный анализ движения в обратном направлении (текущий и опорный кадры меняются местами).

Примером практического применения описанного алгоритма является преобразование видеопоследовательности с частотой кадров 50 Гц в видеопоследовательность с частотой кадров 60 Гц.

Построение кадров осуществляется либо с помощью копирования кадров исходной последовательности (если временное положение кадра результирующей последовательности близко к временному положению кадра в исходной последовательности), либо с помощью интерполяции по соседним (по временному положению) кадрам исходной последовательности.

Если кадры в исходной последовательности расположены регулярно, то результирующая последовательность может включать три или пять построенных кадров (рис. 14.42а, б).

На практике достаточно проводить интерполяцию 3 кадров, так как кадры во временном положении 16,7 и 83,3 мс незначительно отличаются от кадров во временном положении 20 и 80 мс в исходной последовательности, так что

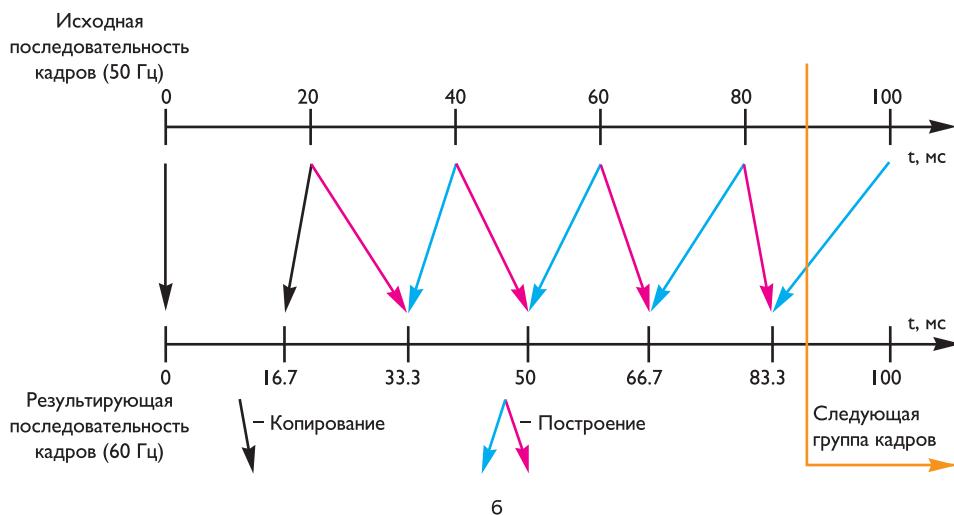
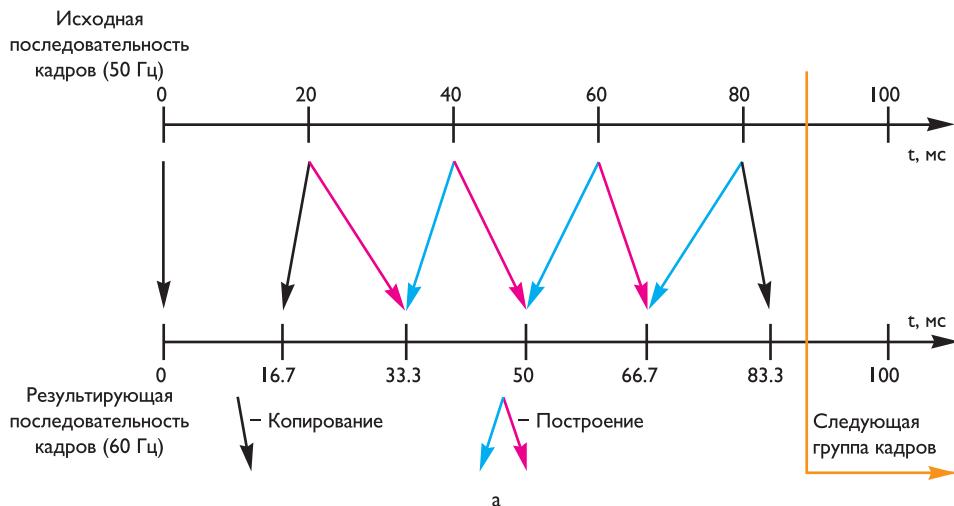


Рис. 14.42. Построение трех кадров — а) и пяти кадров — б) при преобразовании частоты кадров из 50 Гц в 60 Гц

неравномерность движения не будет заметна, в то же время снижается требуемая вычислительная емкость и отсутствуют возможные при интерполяции артефакты.

Если входные кадры расположены по времени неравномерно, то временное положение кадров результирующей последовательности не изменяется, но могут измениться номера кадров исходной последовательности, по которым осуществляется построение новых кадров.

На рис. 14.43 приведены 5 кадров видеопоследовательности с частотой кадров 50 Гц, а на рис. 14.44 — 6 соответствующих кадров этой видеопоследовательности после преобразования к частоте 60 Гц.



Рис. 14.43. Кадры видеопоследовательности с частотой кадров 50 Гц

Такое преобразование оказывается необходимым при воспроизведении изображений телевидения высокой четкости на больших экранах, поскольку при частоте кадров 50 Гц оказывается заметной модуляция яркости экрана.

14.7.2. Методика регулировки потока при сжатии динамических изображений

При сжатии динамических изображений, как правило, необходимо регулировать скорость выходного потока данных в соответствии с пропускной способностью

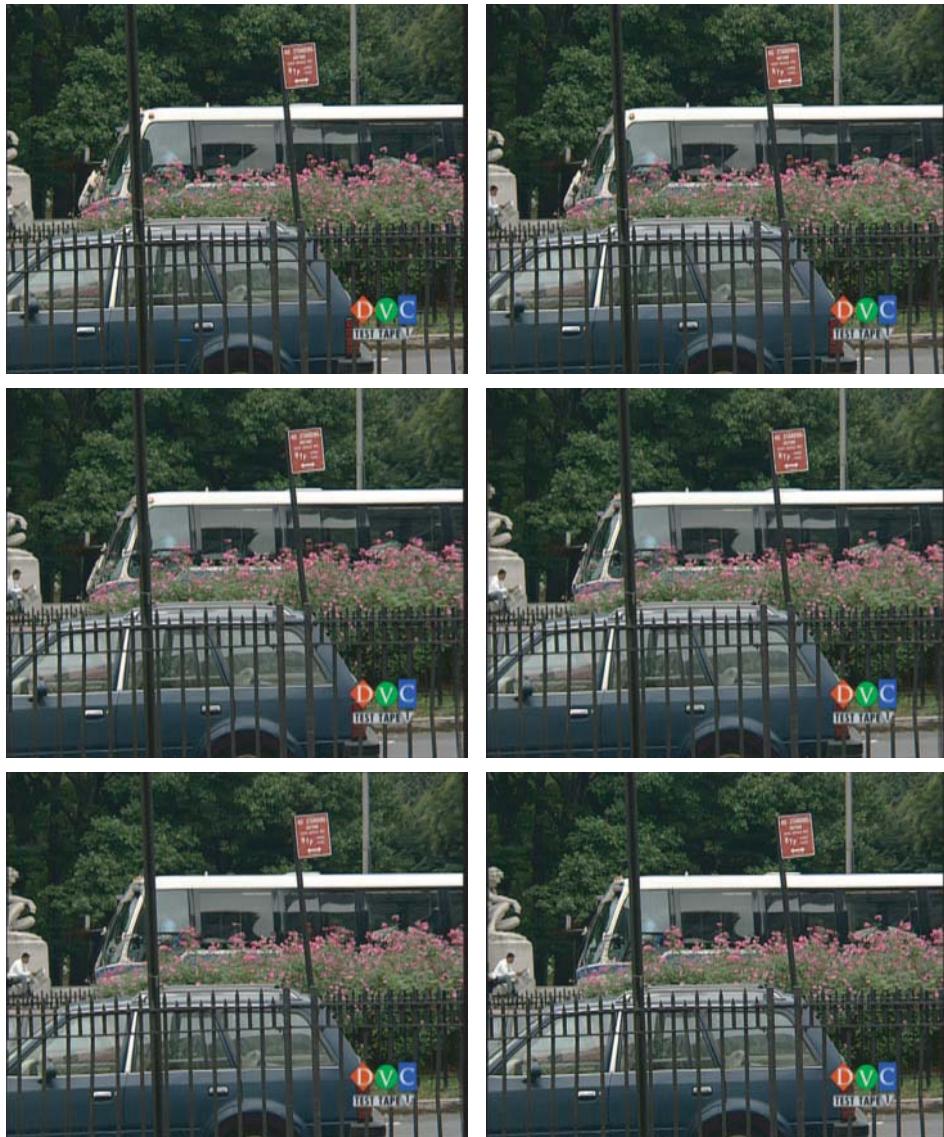


Рис. 14.44. Кадры видеопоследовательности, преобразованной в 60 Гц

канала связи, по которому будет передаваться компрессированный видеопоток. На практике часто возникает задача доставить закодированные данные через канал с ограниченной пропускной способностью с минимальной задержкой при сохранении максимально возможного качества восстановленного изображения [5.74].

При разработке эффективного метода регулирования скорости выходного потока данных необходимо рассмотреть особенности кодирования различных типов кадров (с внутрикадровым кодированием — I, кодированные с предсказанием — P,

кодированные с двунаправленным предсказанием — В), встречающихся во многих распространенных стандартах кодирования. Чтобы обеспечить наиболее полное использование пропускной способности канала связи с минимальной задержкой, необходимо подобрать достаточно точную модель взаимосвязи параметров кодирования и скорости выходного потока.

Управление потоком сводится к принятию решений о параметрах квантования видеообъектов (кадров, макроблоков). Описываемая методика основана на так называемой R-D-модели (Rate-Distortion — модель связи скорости выходного потока и степени сжатия) [5.75–5.78]. В этой модели учитываются характеристики изображения, влияющие на выбор параметров квантования. Так как указанные выше стандарты видеокомпрессии используют одинаковый базовый алгоритм кодирования (внутрикадровое косинусное преобразование блоков и квантование, межкадровое предсказание макроблоков), то данная схема регулирования потока применима к каждому из них.

Прежде чем приступить к описанию R-D-модели, необходимо ввести некоторую меру «сложности» видеообъекта (в данном случае — макроблока). Часто используются такие меры, как среднеквадратичный разброс яркостей (*CKP*), средний абсолютный разброс яркостей (*CAP*), максимальный разброс яркостей (*MP*):

$$\begin{aligned} CKP &= \frac{1}{255} \sum_{i=1}^{16} \sum_{j=1}^{16} [P(i, j) - \mu]^2; \\ CAP &= \frac{1}{255} \sum_{i=1}^{16} \sum_{j=1}^{16} |P(i, j) - \mu|; \\ MP &= \max_{(i=1 \dots 16, j=1 \dots 16)} |P(i, j)|, \end{aligned} \quad (14.31)$$

где $P(i, j)$ — значение яркости в точке (i, j) , $\mu = \frac{1}{255} \sum_{i=1}^{16} \sum_{j=1}^{16} P(i, j)$ — средняя яркость макроблока. На практике из соображений минимизации объема вычислений часто применяется мера *CAP*.

Для макроблоков, кодируемых с предсказанием, значение в каждой точке берется после компенсации движения, а значение средней яркости можно считать равным нулю. Для кадра *CAP* рассчитывается как сумма *CAP* всех макроблоков.

В R-D-модели предполагается, что количество битов T , которые будут потрачены на кодирование коэффициентов ДКП, зависит от коэффициента квантования видеообъекта Q следующим образом:

$$T = CAP \cdot \left(\frac{x_1}{Q} + \frac{x_2}{Q^2} \right) = R - H, \quad (14.32)$$

где x_1 и x_2 — параметры модели, R — количество битов, приходящихся на видеообъект; H — количество битов, приходящихся на заголовки (в том числе векторы движения).

При использовании единого коэффициента квантования для кадра параметры системы x_1 и x_2 можно уточнять после того, как кадр будет закодирован. Для уточнения используется n закодированных кадров.

Пусть матрицы M_1 и M_2 соответственно равны:

$$M_1 = \begin{bmatrix} 1 & 1/Q_1 \\ 1 & 1/Q_2 \\ \dots & \dots \\ 1 & 1/Q_n \end{bmatrix}, \quad M_2 = \begin{bmatrix} Q_1 \cdot T_1 \\ Q_2 \cdot T_2 \\ \dots \\ Q_n \cdot T_n \end{bmatrix}. \quad (14.33)$$

Тогда параметры модели могут быть определены уравнением:

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = (M_1^T \cdot M_1)^{-1} \cdot M_1^T \cdot M_2,$$

где T_i , Q_i ($i = 1, \dots, n$) — количество битов, потраченных на кодирование коэффициентов ДКП и коэффициенты квантования закодированных кадров соответственно.

Регулирование потока с использованием постоянного коэффициента квантования в кадре состоит из следующих стадий:

- вычисление количества битов, выделяемых на кадр и на коэффициенты ДКП;
- подсчет коэффициента квантования;
- кодирование кадра;
- обновление R-D-модели.

Первая стадия методики отличается для разных типов кадров, так как различен механизм подсчета количества битов, выделяемых на коэффициенты ДКП, для I-, P- и B-кадров. Хотя данная модель не привязана к сложности кадра (сложность в несколько раз отличается для различных типов кадров), однако предполагается, что последовательно идущие кадры имеют схожую сложность, примерно одинаковое количество битов на заголовки (не коэффициенты ДКП) и т. д. В связи с этим строится R-D-модель для каждого типа кадров.

Рассмотрим подробнее первую и четвертую стадии.

Вычисление количества битов, выделяемых на кадр и на коэффициенты ДКП

Количество битов, выделяемых на кадр и на его коэффициенты ДКП, определяется в том числе типом кадра. Например, размер I-кадра должен в несколько раз превышать размер P-кадра. Пусть кодирование видео осуществляется при следующих условиях: максимальное расстояние (в кадрах) между двумя последовательными I-кадрами равно N_I ; I-кадры в среднем занимают в K_I раз больше битов, чем P-кадры; временное расстояние между двумя последовательными I-кадрами (без учета потенциальной смены сюжета) равно τ_I ; ПСК — пропускная способность канала связи. Тогда перед началом кодирования группы кадров, начинающейся I-кадром и продолжающейся до следующего I-кадра, необходимо рассчитать количество битов, приходящихся на I-кадр:

$$T_I = K_I \cdot \tau_I \cdot \text{ПСК} / (N_I + K_I).$$

Допустим, закодированный I-кадр занял T_I битов. Тогда количество битов $T_{\text{ост}}$, оставшихся на N_P P-кадров группы ($N_P N_I - 1$), равно

$$T_{\text{ост}} = N_I \cdot \text{ПСК} - T_I.$$

Таким образом, среднее количество битов, приходящихся на Р-кадр группы равно

$$T_P = T_{\text{ост}}/N_P.$$

Так как количество битов, реально потраченных на Р-кадр, отличается от расчетного значения, то в целях обеспечения более равномерного распределения битов на кадры используется виртуальный буфер. Этот буфер также служит для сглаживания неравномерности потока кодированных данных, которая обусловлена использованием различных типов кадров, различной сложностью кадров и пр.

Пусть B_S — размер виртуального буфера (часто его размер задается как $N_I \cdot \text{ПСК}/4$), а B — его заполненность (перед началом кодирования буфер считается заполненным наполовину — $BB_S/2$). Тогда перед кодированием очередного Р-кадра предполагаемое количество битов можно рассчитать как

$$T_P = T_{\text{ост}}/N_P \cdot [B + 2 \cdot (B_S - B)]/[2 \cdot B + (B_S - B)].$$

Таким образом, если буфер заполнен больше чем наполовину, количество выделяемых битов уменьшается, в противном случае — увеличивается. После того как Р-кадр закодирован, следует уточнить количество битов для следующего Р-кадра и обновить заполненность виртуального буфера.

$$T_{\text{ост}} = T_{\text{ост}} - T_P,$$

$$N_P = N_P - 1,$$

$$B = B + T_P - B_K,$$

где B_K — количество битов, уходящих из виртуального буфера в канал за время между двумя последовательными кадрами.

Чтобы избежать сильной деградации качества Р-кадров или переполнения виртуального буфера в случае, когда реальные размеры Р-кадров получаются больше расчетных, можно пропускать Р-кадры. Однако следует учитывать, что пропуск кадров приводит к ухудшению предсказания и к большей сложности следующего кадра. Количество пропускаемых Р-кадров N_{Π} рассчитывается следующим образом:

```

 $N_{\Pi} = 0;$ 
 $\text{while}(B > 0,8 \cdot B_S) \{$ 
 $N_{\Pi} = N_{\Pi} + 1;$ 
 $N_P = N_P - 1;$ 
 $B = B - B_K;$ 
 $\}$ 

```

Также следует отметить те случаи, когда группа кадров при кодировании не укладывается в отведенный ей поток, например при смене сюжета. В этом случае количество перерасходованных битов можно учесть при кодировании следующей группы кадров.

Обновление R-D модели

Обновление модели можно разбить на два этапа: определение количества кадров n , результаты кодирования которых будут учтены при уточнении параметров модели, и уточнение параметров модели.

Описываемый ниже механизм позволяет предотвратить (уменьшить) негативное влияние изменения сложности кадров (смены медленного сюжета более быст-

рым и наоборот). Например, если сложность текущего кадра значительно больше, чем у предыдущего, то количество кадров, принимаемых во внимание, (размер окна статистики) следует уменьшить. Таким образом, кодер может принимать решение, основываясь, например, на разнице в сложности текущего и предыдущего кадров.

Допустим, что закодировано N_3 кадров группы, тогда $W = \min(N_3, 20)$, т. е. максимальный размер окна статистики равен 20.

Пусть сложность текущего закодированного кадра равна $CAP_{тек}$, а предыдущего — $CAP_{пред}$, тогда

$$\begin{aligned} n &= W \cdot CAP_{пред}/CAP_{тек}, \text{ если } CAP_{тек} > CAP_{пред}, \\ n &= W \cdot CAP_{тек}/CAP_{пред}, \text{ если } CAP_{тек} < CAP_{пред}. \end{aligned}$$

Уточнение параметров модели x_1 и x_2 осуществляется в три этапа. Сначала рассчитываются новые значения x_1 и x_2 (см. вышеупомянутые формулы). Затем с помощью этих новых параметров проводится оценка количества битов для всех кадров окна статистики, и из него исключаются те кадры, для которых оценка количества битов отличается от реального размера закодированного кадра больше чем на среднеквадратичное отклонение оценки от реального размера в окне статистики. На третьем этапе повторяется расчет новых значений x_1 и x_2 в уточненном окне статистики.

Описанная выше схема регулирования скорости выходного потока была реализована в программных кодерах, отвечающих стандартам H.261, H.263, MPEG-2, MPEG-4. Результаты работы этой методики рассмотрены на примере работы кодера MPEG-4 и отображены на графиках зависимости количества битов в виртуальном буфере (заполненность буфера) от номера кадра видеопоследовательности. После кодирования очередного кадра количество битов в буфере увеличивается, а скорость уменьшения количества битов в буфере постоянна и равна заданной пропускной способности канала. При идеальной работе методики регулирования скорости потока график должен иметь пилюобразный вид, с большими пиками на I-кадрах и плавным уменьшением до нуля к концу группы кадров.

На рис. 14.45–14.48 представлены зависимости количества битов в буфере при кодировании тестовых последовательностей Carphone и Foreman с различными параметрами.

На приведенных графиках количество битов в виртуальном буфере колеблется около нулевого значения и почти всегда положительно, т. е. модель регулировки потока работает корректно. Однако во множестве задач требуется более точное регулирование скорости потока. Кроме того, единый коэффициент квантования на кадр не обеспечивает наилучшего качества восстановленного изображения при заданной пропускной способности канала. Решением в данном случае может быть введение регулировки потока на уровне макроблоков. Описанная выше модель применима к различным видеообъектам, не только к кадрам целиком, но и к макроблокам.

Допустим, что известно расчетное количество битов, приходящихся на коэффициенты ДКП кадра (см. выше). Тогда перед кодированием каждого макроблока расчетное количество битов $T_{M\ i}$, приходящихся на него, можно получить следующим образом:

$$T_{M\ i} = CAP_i / CAP_{ост\ i} \cdot T_{ост\ i},$$

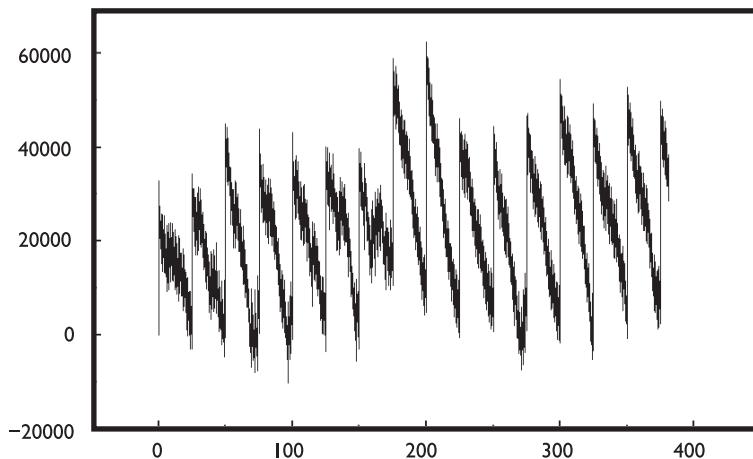


Рис. 14.45. Заполненность виртуального буфера (количество битов в буфере) для последовательности Sarnphone формата QCIF (кадры от 0 до 400), ПСК = 300 кбитов/сек, $N_I = 25$ кадров, $K_I = 5$

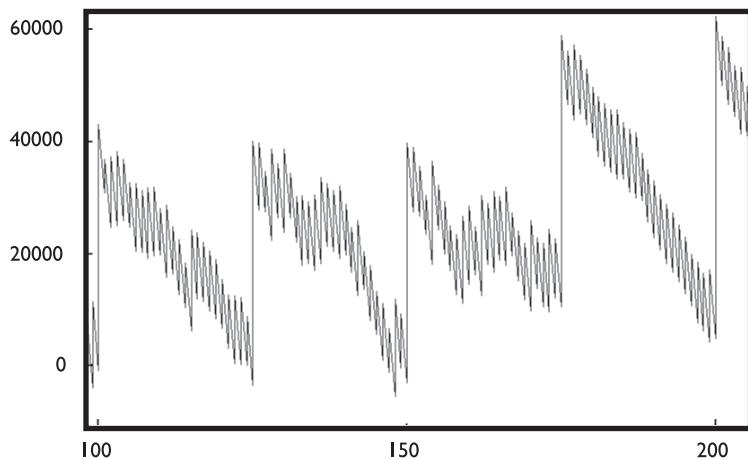


Рис. 14.46. Заполненность виртуального буфера для последовательности Sarnphone формата QCIF (кадры от 100 до 200), ПСК = 300 кбитов/сек, $N_I = 25$ кадров, $K_I = 5$

где i — номер макроБЛОКА, CAP_i — его сложность, $CAP_{ост\ i}$ — сложность незакодированной части кадра, $T_{ост\ i}$ — количество битов, оставшееся для незакодированной части кадра.

При кодировании первого макроБЛОКА в кадре $CAP_{ост\ i}$ совпадает со сложностью кадра. При кодировании i -го макроБЛОКА она уменьшается на CAP_i . Аналогично, $T_{ост\ i}$ перед началом кодирования кадра равно количеству битов, отводимых на весь кадр, после кодирования i -го макроБЛОКА $T_{ост\ i}$ уменьшается на $T_M\ i$. Следует также отметить, что рассчитанное значение коэффициента квантования для макроБЛОКА иногда может или должно ограничиваться. Напри-

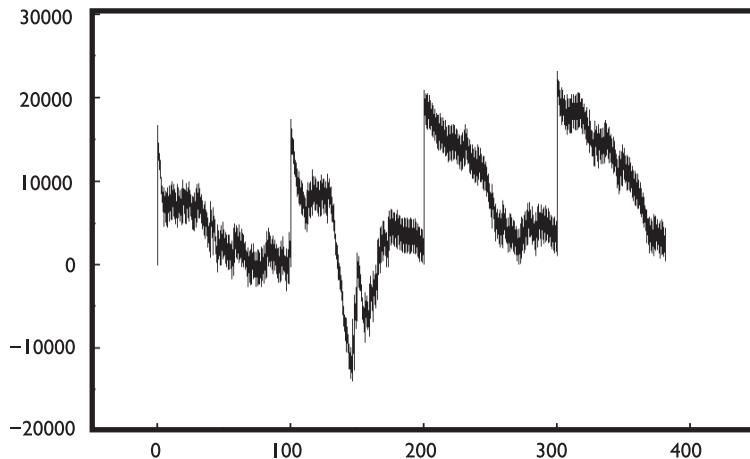


Рис. 14.47. Заполненность виртуального буфера для последовательности Carphone формата QCIF (кадры от 0 до 400), ПСК = 100 кбитов/сек, $N_I = 100$ кадров, $K_I = 5$

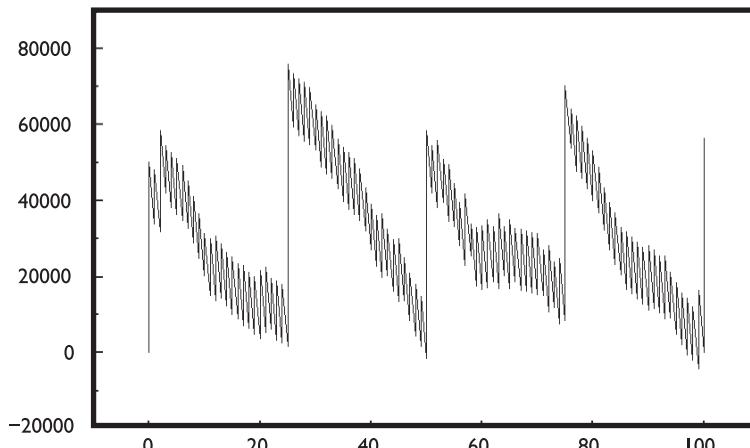


Рис. 14.48. Заполненность виртуального буфера для последовательности Foreman формата CIF (кадры от 0 до 400), ПСК = 400 кбитов/сек, $N_I = 25$ кадров, $K_I = 5$

мер, в стандартах MPEG-4 и H.263 при определенных условиях коэффициенты квантования соседних макроблоков в кадре не могут отличаться более чем на 2. Ограничение значений коэффициента квантования может потребоваться также по ряду других причин, например для ограничения деградации качества части изображения из-за сильного квантования. В связи с этим часто применяется ограничение на изменение коэффициента квантования в пределах $\pm 25\%$ относительно некоторого значения для кадра.

Преимущества помакроблочкой регулировки потока можно проиллюстрировать, если отобразить результаты на тех же графиках, что и для регулировки потока с одним коэффициентом квантования на кадр. На приведенных рис. 14.49–

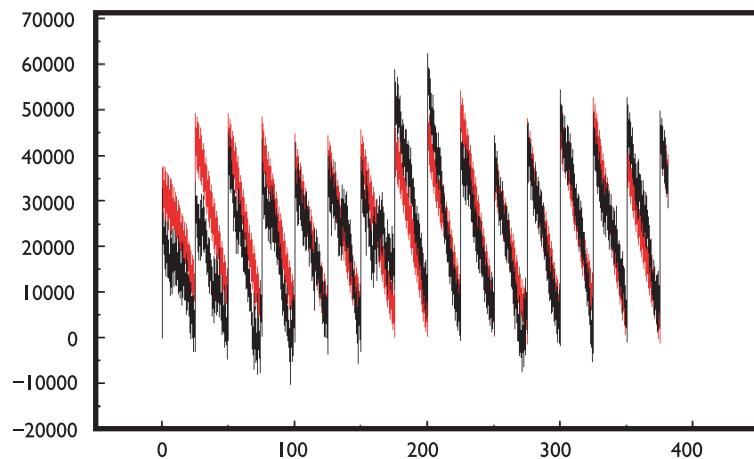


Рис. 14.49. Заполненность виртуального буфера для последовательности Carphone формата QCIF (кадры от 0 до 400), ПСК = 300 кбитов/сек, $N_I = 25$ кадров, $K_I = 5$

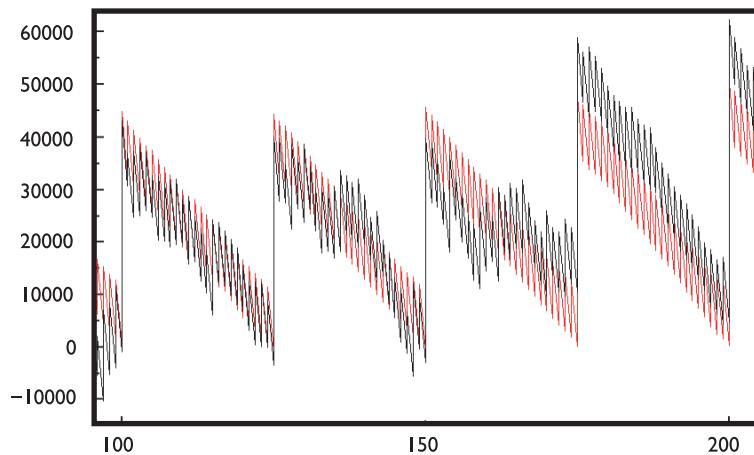


Рис. 14.50. Заполненность виртуального буфера для последовательности Carphone формата QCIF (кадры от 100 до 200), ПСК = 300 кбитов/сек, $N_I = 25$ кадров, $K_I = 5$

14.52 красными линиями обозначены результаты помакроблочного регулирования скорости потока, черными — покадрового.

Из приведенных графиков видно, что при помакроблочном регулировании скорости потока кодирование лучше согласуется с используемой моделью, т. е. расчетные и реальные размеры кадров более согласованы.

Существенной дополнительной возможностью уменьшения цифрового потока закодированных динамических изображений является использование переменного разрешения по кадру.

В некоторых специфических приложениях требуется обеспечить более четкое изображение в заданной части кадра. Например, при стыковке кораблей в космо-

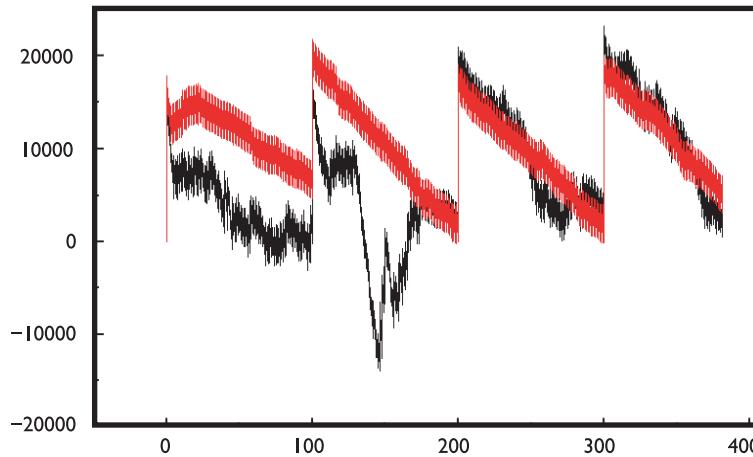


Рис. 14.51. Заполненность виртуального буфера для последовательности Carphone формата QCIF (кадры от 0 до 400), ПСК = 100 кбитов/сек, $N_I = 100$ кадров, $K_I = 5$

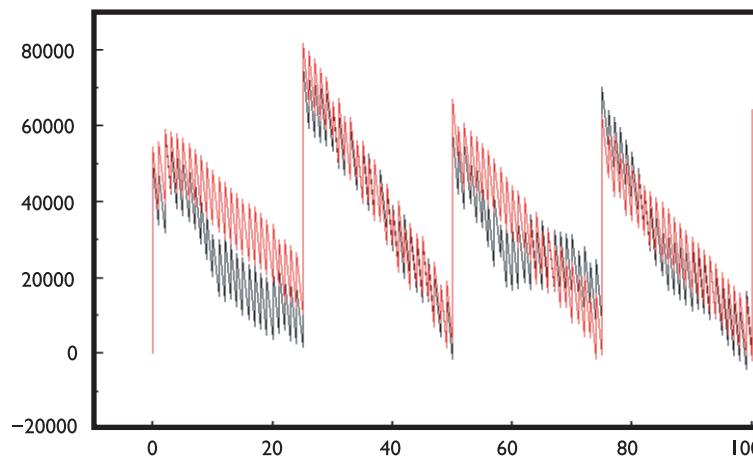


Рис. 14.52. Заполненность виртуального буфера для последовательности Foreman формата CIF (кадры от 0 до 400), ПСК = 400 кбитов/сек, $N_I = 25$ кадров, $K_I = 5$

се требуется наблюдать за всей картиной в целом, но зону стыковки необходимо показывать с повышенной четкостью. Обеспечивать повышенную четкость по всему кадру невыгодно с точки зрения требуемой для этого пропускной способности канала связи.

Эта задача решается одним из двух методов. В первом случае используется описанный выше метод помакроблочной регулировки выходного потока, но расчеты количества битов, коэффициентов квантования, параметров R-D модели и пр. осуществляются раздельно для макроблоков части кадра с повышенной четкостью и для остальных макроблоков. Соотношение между скоростями потоков данных, выделенных для более четкой части кадра T_q и остальной части



Рис. 14.53. Кодирование с переменным разрешением по кадру

кадра $T_{\text{нч}}$ ($T_{\text{ч}} + T_{\text{нч}}T$ — суммарная заданная скорость потока), определяет степень повышения четкости. Во втором методе используется масштабированный систаксис потока кодирования MPEG-2 или MPEG-4, причем в базовом потоке изображение может кодироваться с меньшим разрешением, а в дополнительном потоке (используется пространственное масштабирование) кодируется заданная область изображения с повышенным разрешением. На рис. 14.53 приведены восстановленные изображения из кодированных последовательностей, в выделенной черной квадратной рамкой части которых обеспечена повышенная четкость.

14.7.3. Анализ смены сюжета

Смена сюжета в видеопоследовательности — это резкое изменение содержания сцены в соседних кадрах. Смена сюжета происходит при переключении источника видео (камеры), при соединении фрагментов видео (редактирование) и т. д.

При кодировании видеопоследовательностей и использовании предсказания по соседним кадрам следует учитывать смену сюжета по ряду причин:

- предсказание на основе кадров другого сюжета неэффективно (возможны только случайные совпадения), но на предсказание тратятся вычислительные ресурсы;
- учет смены сюжета позволяет повысить качество декодируемого видео при регулировании скорости выходного потока кодирования;
- использование опорных кадров при смене сюжета повышает качество и удобство работы при последующем редактировании закодированного видеоматериала.

Как известно, любая видеопоследовательность представляется в виде сменяющих друг друга сюжетов, связанных, например с переключением камеры, ведущей трансляцию, с монтажом или с другими эффектами. Очевидно, что выделение отдельного сюжета, внутри которого сохраняется высокая временная корреляция кадров, позволило бы увеличить эффективность предсказания движения, поднять степень сжатия и избежать искажений изображения, возникающих при нераспознавании смены сюжета. Другим приложением, где требуется распознавание смены сюжета, является создание баз видеоданных и организация видеосерверов, в которых необходимо как проводить каталогизацию, так и обеспечивать быстрый доступ к отдельному сюжету или кадру.

Распознавание движения камеры

Если съемка текущего фрагмента проводится одной и той же камерой, то при движении камеры возникают характерные особенности «движения» изображения, вызванные именно камерой. Разработаны методы, позволяющие классифицировать различные виды операций камеры, которые можно разбить на следующие основные группы:

- отсутствие движения, стационарная камера;
- движение камеры в горизонтальном направлении (panning), которое может заключаться как в повороте камеры относительно вертикальной оси, так и в поступательном движении камеры как целого по горизонтали;
- движение камеры в вертикальном направлении (tilting), поворот камеры относительно горизонтальной оси или поступательное движение по вертикали;
- «наезд» камеры (zooming), т. е. изменение фокуса или поступательное движение камеры по направлению к объекту или от него.

Могут быть также комбинации указанных движений. Если для кадра в целом удается выделить изменения, связанные с тем или иным движением камеры, то задача обработки движения облегчается.

Методы распознавания движения, связанного с работой камеры, излагаются в работах Маэды [5.76], Шерфауи и Бертена [5.77]. В данных методах проводится качественный анализ динамики последовательности кадров в заданном временном интервале, и критерием детектирования режима работы камеры является наличие объектов, движущихся по заранее рассчитанным траекториям, характерным для того или иного движения камеры. При этом в методе Маэды учитываются только прямолинейные траектории, что может приводить к неоправданно частой смене режимов, в то время как Шерфауи и Бертен используют аппроксимацию траекторий ломаной линией и достигают более адекватного отображения движения камеры.

Распознавание смены сюжета, не связанного с движением камеры

В ныне существующих методах распознавания смены сюжета для каждого кадра из анализируемой видеопоследовательности определяется индекс согласования кадров (IM, image match), т. е. некоторое число, интегральный показатель соответствия соседних кадров, динамика которого служит критерием смены сюжета. Очевидно, что смена сюжета может происходить как мгновенно, при этом индекс согласования кадров меняется скачкообразно, так и постепенно, в результате специальных эффектов, таких как затухание (fading), размывание (dissolving), редакторское наложение при монтаже и т. д. Детектирование постепенной смены сюжета является гораздо более сложной задачей, для которой еще не разработано надежных методов. Критерием эффективности того или иного метода является надежность распознавания смены сюжета и вычислительные затраты на определение динамики индекса согласования кадров.

В настоящее время применяются два основных подхода к распознаванию: первый, основанный на анализе статистических характеристик яркостных и цветностных составляющих кадра, и второй, основанный на анализе векторов движе-

ния отдельных областей кадра. Простейшим методом из первой группы является сравнение сумм абсолютных значений разностей только яркостных компонентов или всех трех компонентов настоящего и предыдущего кадров при переходе к последующему кадру. Более продвинутые методы сравнивают гистограммы отдельных компонентов кадров. В работе Нагасаки и Танаки [5.78] указывается, что хорошие результаты дает χ^2 -тест гистограмм цветности между кадрами. Полагается, что при смене сюжета меняются цветовые характеристики изображения, и разности между пиками на гистограммах цветности будут очень заметны. Более надежные результаты дает тот же метод применительно к кадрам, разбитым на отдельные блоки, когда сравнение проводится для всех блоков, и результаты суммируются (см. раздел 12.4).

Методика расчета по сравнению гистограмм следующая. В первую очередь подсчитываются суммы абсолютных разностей S_Y , S_U и S_V между значениями столбцов гистограммы для данного и предшествующего кадров, и затем они нормируются на размерность гистограммы H :

$$f = (S_Y + S_U + S_V)^2 / H.$$

Из экспериментов находятся пороговые значения фактора f , отвечающие смене сюжета, и проблема заключается в том, чтобы установить адекватные значения порогов для различных типов изображений.

Уточнением данного метода является анализ дисперсии разности между кадрами $|\Delta\sigma^2|$. Как было обнаружено, смена сюжета часто сопровождается резким увеличением дисперсии. Однако к этому же эффекту приводит наличие сцен с быстрым движением.

Другая группа методов начала развиваться в последнее время в связи с появлением MPEG-кодированных видеоизображений, для которых необходимо производить поиск смены сюжета в уже сжатом потоке видеоданных, например, для составления библиотек и каталогизации видеоданных. Вследствие того, что полное восстановление изображений с последующим применением методов, развитых для исходных несжатых изображений, является очень трудоемкой и отнимающей много ресурсов операцией, были предложены способы распознавания смены сюжета, основывающиеся на анализе данных из MPEG-потока, не требующих полного восстановления, а именно статистических характеристик значения нулевого коэффициента косинусного преобразования, которое соотносится со средним значением яркости в блоке, и значений векторов движения для блоков.

Полагается, что для В-кадра, за которым следует смена сюжета, большинство векторов движения блоков будут направлены вперед, и отношение числа векторов движения, направленных вперед, к числу направленных назад будет иметь ярко выраженный максимум.

Хотя указанные методы в том виде, как они изложены в [5.77], относятся к уже готовому потоку данных MPEG, они могут быть также применены и на стадии кодирования, так как при MPEG-кодировании находятся величины, которые используются в этих методах: коэффициенты косинусного преобразования и векторы движения для блоков. Представляется оптимальным использовать эти методы в сочетании с мониторингом фактора f и дисперсии $|\Delta\sigma^2|$, так как последние не требуют больших вычислительных затрат на их нахождение.

Анализ смены сюжета можно проводить по результатам блочного анализа движения — если предсказывается небольшая часть блоков, то произошла смена сюжета. Этот алгоритм легко встраивается в процедуру кодирования, уже содержащую блочный анализ движения, но имеет некоторые недостатки. При этом алгоритме за смену сюжета будет также приниматься быстрое панорамирование, масштабирование, изменение освещенности и пр. Кроме того, на анализ движения (впоследствии ненужный из-за смены сюжета) расходуются вычислительные ресурсы.

В результате проведения исследований был разработан и предложен быстрый, эффективный и вычислительно простой метод поиска смены сюжета. Он основан на анализе изменения гистограммы распределения яркости в соседних кадрах [5.79–5.81].

Метод состоит из нескольких шагов.

- Изображение разбивается на несколько областей R_i , $i = 0 \dots (N - 1)$. Области могут быть как перекрывающимися, так и неперекрывающимися, они могут в совокупности охватывать либо кадр целиком, либо большую часть кадра.
- Для каждой области j кадра i подсчитывается гистограмма распределения яркостей пикселов $H_{i,j} = \{h_{i,j,k}\}$, $j = 0, \dots, (N - 1)$, $k = 0, \dots, (256/S - 1)$. Шаг вычисления гистограммы S выбирается больше единицы, например $S = 8$. Яркости пикселов распределены в диапазоне от 0 до 255, $h_{i,j,k}$ — процентная доля пикселов области R_j кадра i , имеющих яркость в диапазоне $[k \cdot S, (k + 1) \cdot S]$.
- Для каждой области подсчитывается сумма квадратов разностей значений гистограмм текущего и предшествующего кадров: $D_{i,j} = \sum_k (h_{i,j,k} - h_{i-1,j,k})^2$.
- Определяется количество областей, в которых $D_{i,j} > T$, где T — пороговое значение, например $T = 500$.
- Если количество областей, в которых порог превышен, не менее половины, то детектируется смена сюжета.

Разработанный метод показал высокую эффективность анализа смены сюжета на изображениях различного разрешения и различного качества. Пороговые значения T определения смены сюжета необходимо изменять в зависимости от разрешения изображения и частоты кадров.

14.7.4. Оценка быстродействия кодирования изображений

При оценке быстродействия кодирования изображений по стандартам семейства MPEG важно получить информацию не только об общей скорости кодирования различных типов кадров (I, P, B), но и о том, как время кодирования кадра распределено по отдельным стадиям кодирования [5.82].

В качестве существенных стадий кодирования можно выделить следующие:

- предварительная обработка кадров (префильтрация и т. д.);
- поиск движения для макроблоков в P-кадрах или B-кадрах (во втором случае эта стадия может делиться на две — поиск движения вперед и назад);

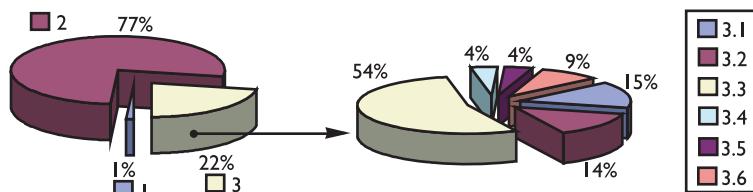


Рис. 14.54. Пример оценки распределения вычислительных ресурсов при MPEG-кодировании: 1 — предварительная обработка — 1%; 2 — поиск движения — 77%; 3 — кодирование — 22%; 3.1 — компенсация движения — 15%; 3.2 — ДКП — 14%; 3.3 — квантование — 54%; 3.4 — обратное ДКП — 4%; 3.5 — восстановление — 4%; 3.6 — формирование выходного потока — 9%

- компенсация движения для макроблоков в Р- и В-кадрах;
- выбор типа макроблока (I- или Р- в Р-кадрах; I-, Р- или В- в В-кадрах) в зависимости от эффективности компенсации движения для данного макроблока;
- дискретное косинусное преобразование блоков изображения;
- подбор коэффициентов квантования макроблоков для регулирования скорости выходного потока;
- квантование коэффициентов ДКП;
- обратное квантование, ОДКП, обратная компенсация движения для восстановления опорных кадров, используемых в дальнейшем для компенсации движения;
- кодирование и формирование выходного битового потока.

При кодировании I-кадров присутствуют стадии 1, 5–9. При кодировании Р-кадров присутствуют стадии 1–9. При кодировании В-кадров присутствуют стадии 1–7, 9.

Вообще говоря, некоторые стадии могут отсутствовать (например, 1, 6) или быть совмещены с другими стадиями в зависимости от выбранной стратегии кодирования, алгоритма регулирования скорости выходного потока и т. д.

Сложность оценки быстродействия различных стадий кодирования может быть связана с тем, что эти стадии могут выполняться не последовательно над целыми кадрами, а группироваться и выполняться над слайсами или отдельными макроблоками. В этом случае времена отдельных действий надо суммировать в процессе обработки кадра.

Оценка скорости кодирования по стадиям особенно важна при построении высокоэффективного программного кодера, так как позволяет определить, какие стадии кодирования нуждаются в оптимизации. В зависимости от используемой архитектуры вычислительной системы может оказаться эффективным объединение некоторых стадий кодирования в одном блоке или разделение их на несколько отдельных вычислительных блоков.

На рис. 14.54 приведен пример оценки распределения вычислительных ресурсов при MPEG-кодировании динамических изображений.

14.7.5. Возможности кодирования стереоскопических изображений в рамках совместимости со стандартами MPEG

На современном этапе развития телевидения и видеовещания стереоскопическое представление видеонформации превращается из отдаленной перспективы развития техники в реальную услугу. Кроме того, быстрое развитие вычислительной техники, резкое повышение ее производительности позволяют проводить исследования и реализацию ресурсоемких (с точки зрения математических вычислений) систем без использования уникальных компьютеров или специализированных вычислителей.

Использование формата потока закодированных стереоскопических видеоданных в виде, совместимом с общепринятыми международными стандартами MPEG-2, MPEG-4, позволит расширить базу пользователей и использовать для просмотра видеопрограмм не только новые устройства, дающие стереоскопическую картину, но и старые приемники, работающие по стандартам MPEG-2, MPEG-4, которые дадут возможность просмотра программ (без стереоэффекта).

Были проведены теоретические и экспериментальные исследования компрессии стереоскопического видеосигнала, изучались пороги сохранения стереоэффекта при различных разрешениях видео и скоростях выходного потока и был выбран формат выходного потока, обеспечивающий совместимость со стандартами MPEG.

Результаты исследований показали эффективность кодирования последовательности стереоскопических изображений по следующим причинам [5.83]:

- методика позволяет обеспечить совместимость вещаемого стереоскопического сигнала с ныне существующим оборудованием;
- при таком подходе достигается значительная экономия пропускной способности канала связи при передаче стерео за счет высокой степени схожести между любыми двумя изображениями стереопары и использования этого факта при осуществлении компрессии последовательности стереоскопических видеокадров.

Стереоскопические пары изображений должны быть представлены таким образом, чтобы каждым глазом видеть только то изображение, которое для этого глаза предназначено, и не видеть изображение, снятое другой камерой [5.84].

Плоское изображение, снятое одной камерой, оба глаза видят одинаково: здесь смещение близких предметов относительно дальних обнаружить невозможно, так как изображение снято с одной точки зрения. Только стереоскопические изображения способны воспроизвести объективную действительность во всех трех измерениях.

Из-за различия в точках зрения для каждого глаза существует небольшое горизонтальное смещение, называемое рассогласованием, между одними и теми же точками предметов, изображенными на сетчатой оболочке левого и правого глаза. В стереоскопическом зрительном восприятии объекты, на которых глаза сфокусированы и настроены, имеют нулевое рассогласование и расположены в центре сетчатой оболочки обоих глаз, в то время как объекты спереди и сзади имеют отрицательное и положительное рассогласование соответственно. Это рас-

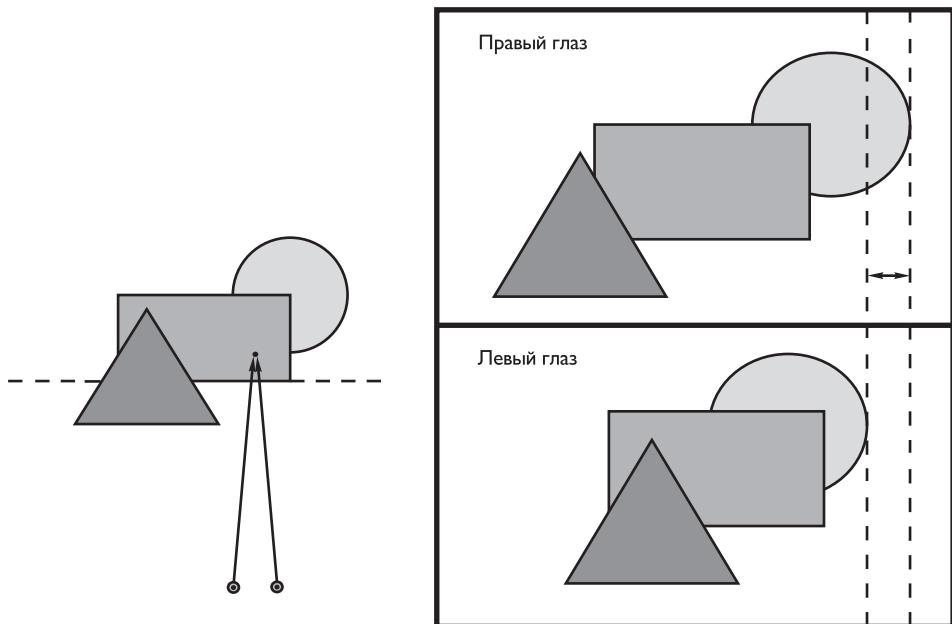


Рис. 14.55. Различие в восприятии одной сцены разными глазами в зависимости от глубины расположения объектов

согласование и интерпретируется мозгом как различие в глубине расположения предметов относительно наблюдателя (рис. 14.55).

Для достижения способности воспринимать глубину расположения объектов, используя записанные изображения, стереоскопическая камера должна состоять из двух камер, которые бы снимали с двух различных горизонтально сдвинутых относительно друг друга точек зрения. В большинстве случаев дистанция между двумя оптическими осями стереоскопической камеры того же порядка, что и расстояние между глазами человека (от 6 до 8 сантиметров). В более простых моделях оптические оси располагаются параллельно.

Наблюдаемый объект проецируется в различные позиции на плоскости изображения, поскольку он рассматривается под разными углами. Различие проецируемых позиций может быть легко вычислено из подобия треугольников (рис. 14.56):

$$d = rd - ld = f \cdot (b/2 + x)/z - f \cdot ((b/2 - x)/z) = 2 \cdot (f \cdot (x/z)),$$

где d — различие в позициях проекции точки для разных точек наблюдения; rd и ld — смещение позиции проекции точки относительно правой и левой оптической оси соответственно; f — расстояние между фокальной плоскостью и плоскостью изображения, которое для отдаленных объектов сопоставимо с фокусным расстоянием камеры; x — удаленность точки от центральной оптической оси; b — расстояние между двумя оптическими осями; z — глубина расположения точки в пространстве.

Таким образом, смещение (рассогласование) объектов в рассматриваемой сцене между левой и правой точками зрения обратно пропорционально глубине расположения.

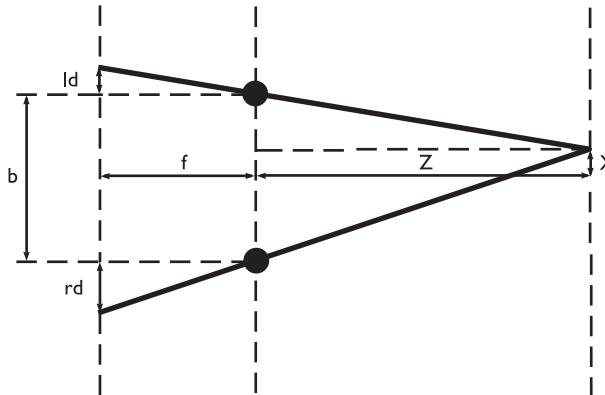


Рис. 14.56. Проекция точки на плоскость изображения при стереоскопической съемке

локации объектов (для объектов в бесконечности стремится к нулю) и обратно пропорционально фокусному расстоянию камеры.

В более сложных моделях камеры берется в расчет сходимость оптических осей камеры под определенным углом.

На данный момент существуют различные техники создания стереоэффекта, используемые для воссоздания иллюзии глубины рассматриваемых объектов. В некоторых из них пары стереоскопических изображений оставляют без изменений, в других подвергают дополнительной обработке перед воспроизведением. Но в основе всех этих подходов лежит принцип формирования незначительно отличных друг от друга изображений для каждого из глаз наблюдателя.

Для реализации кодирования стереоизображений в рамках стандартов MPEG необходимо использовать понятие масштабирования.

Под масштабированием в MPEG-2 (и других стандартах видеокомпрессии) понимается возможность разбиения выходного потока данных на несколько потоков, из которых один несет основную информацию, а остальные предоставляют возможность повышения качества декодированного видеосигнала, полученного с помощью основного потока. Предусмотрено масштабирование пространственное, по качеству (отношению сигнал/шум), по времени и разделение видеопотока на две или более составляющих, передаваемых по разным каналам. При этом реализуется иерархический поток видеоданных. Первый (базовый) слой всегда декодируется независимо от остальных. При этом получается видеопоток с пониженным качеством. Остальные слои могут быть декодированы только вместе с предшествующими им по иерархии.

Пространственное масштабирование позволяет кодировать отдельные слои с различным разрешением. Базовый слой обеспечивает базовое разрешение, а уточняющий слой использует пространственно интерполированный базовый слой для увеличения пространственного разрешения.

Масштабирование по качеству (по отношению сигнал/шум) служит для создания слоев с различным качеством видео. Базовый слой обеспечивает базовое качество, а уточняющий слой улучшает качество видео путем повышения точности квантования коэффициентов ДКП базового слоя.

Временное масштабирование позволяет переход от меньшей кадровой скорости к большей. Базовый слой обеспечивает базовую кадровую скорость, а уточняющий слой использует временное предсказание относительно базового слоя. Базовый и уточняющий слои могут комбинироваться для создания выходного потока с полной кадровой скоростью.

Распределение данных между слоями производится для повышения помехоустойчивости видео. При этом более важная и значимая часть данных размещается в потоке более высокого приоритета, а менее важная часть — в потоке более низкого приоритета.

В стандарте MPEG-2 определен профиль, который позволяет передавать одну и ту же сцену, снятую разными камерами с разных точек наблюдения (MultiView Profile, MVP), используя режим масштабирования по времени. Для кодирования стереоскопического видео последовательность изображений, полученных с левой камеры, кодируется как базовый слой. Уточняющий слой содержит изображения с правой камеры и используется для вставки в качестве дополнительных кадров между кадрами базового слоя, что позволяет при воспроизведении сформировать последовательность изображений с удвоенной кадровой скоростью, если оба слоя оказываются доступными в декодированном потоке.

Данный подход позволяет передавать сигнал стереоскопического вещания в рамках стандарта MPEG-2 и при этом обеспечивает возможность приема данного сигнала как декодерами, способными отображать стерео, так и декодерами, предназначенными для воспроизведения лишь обычного сигнала от одной камеры.

Теоретические и экспериментальные исследования по возможности эффективной компрессии стереоскопических динамических видеоизображений на базе стандарта MPEG-2 проводились на различных последовательностях стереоскопических изображений. В каждой из последовательностей присутствовали как фрагменты, которые можно было бы охарактеризовать как статичные, так и фрагменты, преимущественно динамичные. Статичные фрагменты представлены малоподвижным от кадра к кадру фоном, на котором происходит движение небольших объектов. На динамичных фрагментах изменения от кадра к кадру значительны либо за счет быстрого движения снимающей сцену камеры, либо за счет движения в кадре отдельных крупных объектов или большого числа объектов.

В режиме масштабирования по времени для кодирования базового слоя, содержащего изображения с левой камеры, допустимо использование только предсказания внутри данной последовательности изображений. В то же время, для предсказания в уточняющем слое, содержащем изображения с правой камеры, может применяться как предсказание только внутри слоя, так и предсказание, опирающееся на изображения базового слоя (рис. 14.57).

Таким образом, для устранения межкадровой избыточности в базовом слое применяется устранение избыточности информации за счет поиска векторов движения, которые отражают перемещение фрагментов изображения от кадра к кадру на временной оси. А для уточняющего слоя, кроме устранения избыточности по времени, используется тот факт, что пары стереоскопических изображений являются очень схожими, поскольку снимают одну сцену в один и тот же момент времени. При быстром движении точно вычислить векторы движения при перемещении объектов от кадра к кадру представляется затруднительным из-за выхода объектов за пределы зоны поиска векторов или за границы кадра. Но в

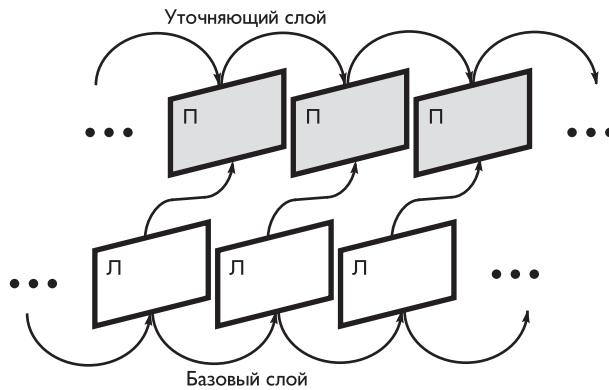


Рис. 14.57. Предсказание в режиме масштабирования по времени

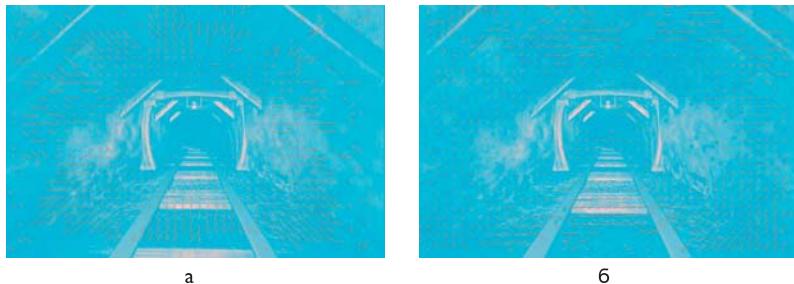


Рис. 14.58. Векторы движения для кадра при предсказании по предшествующему ему по времени кадру (а) и по второму кадру из состава стереопары (б)

в этом случае возможно построить поле различий двух кадров стереопары и использовать его в качестве предсказания для устранения избыточной информации (см. рис. 14.58, 14.59).

Для поиска движения между кадрами последовательности внутри одного слоя и для построения поля различий двух кадров одного стереоскопического изображения нужна мера определения сходства двух участков (макроблоков размером 16×16 пикселов) изображения.

В качестве такой меры могут быть использованы в соответствии с соотношением (14.31) среднеквадратичная разность (CKP), сумма абсолютных различий (CAP) или максимальная разность (MP) яркостей пикселов.

Из соображений минимизации объема вычислений для поиска движения каждого макроблока кадра применяется мера CAP .

Как показал анализ различных тестовых последовательностей, для двух изображений стереопары использование при вычислениях средней яркости макроблока, как описано в формулах (14.31), является принципиально необходимым. Это обусловлено тем, что изображения поступают с двух различных камер, каждая из которых имеет свои индивидуальные аппаратные особенности и по-разному расположены по отношению к источнику освещения (рис. 14.60).

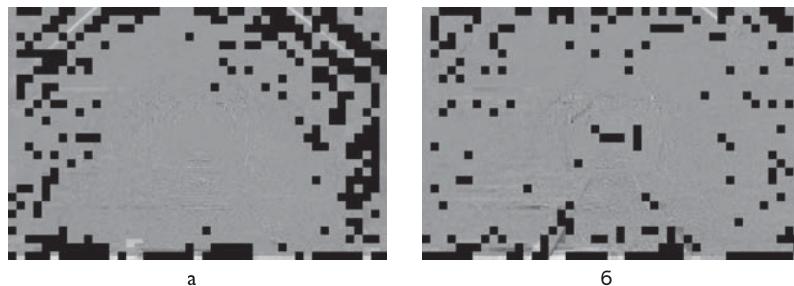


Рис. 14.59. Разностные кадры при предсказании по предшествующему ему по времени кадру (а) и по второму кадру из состава стереопары (б)



Рис. 14.60. Кадры одной стереопары могут отличаться по яркости из-за различного расположения по отношению к источнику освещения и из-за индивидуальных аппаратных особенностей каждой из камер

Исследования подтвердили теоретические предположения о том, что поле векторов между двумя стереоскопическими изображениями направлено в среднем горизонтально и соседние векторы на любом участке кадра имеют высокую степень корреляции. Те области, в которых соседние векторы имеют сильные отличия друг от друга, являются с большой долей вероятности зонами ошибочного предсказания, что объясняется существованием преград, мешающих обнаружению этих зон с одного угла зрения, но не оказывающих никакого негативного влияния на их видимость с другого угла зрения (рис. 14.61, 14.62).

Размер области поиска векторов выбирается как результат компромисса между скоростью работы кодера и качеством предсказания движения. Если использовать в качестве алгоритма поиска векторов метод полного перебора всех возможных значений в области поиска, то при большем размере области поиска становится возможным обнаружить перемещение даже очень быстро двигающихся объектов, но при этом значительно увеличивается вычислительная нагрузка. Эксперименты показывают, что для достижения качественного результата в подавляющем большинстве случаев необходимо иметь область поиска, превосходящую размер макроБЛОКА по вертикали и горизонтали в 4 раза.

Экспериментально подтверждено, что, как и в случае с кодированием последовательности кадров от одной камеры, при кодировании стереоскопических последовательностей изображений, если каждое из изображений стереопары име-

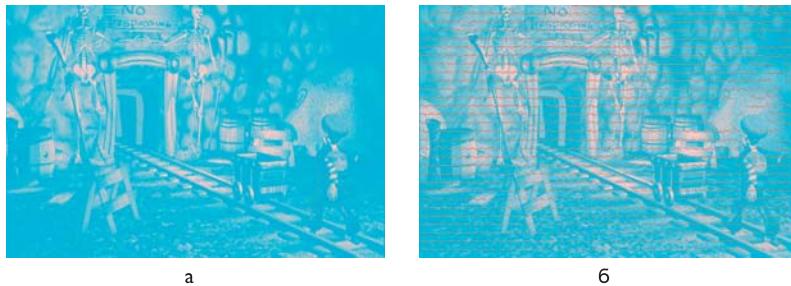


Рис. 14.61. Векторы движения для кадра при предсказании по предшествующему ему по времени кадру (а) и по второму кадру из состава стереопары (б) при использовании только одностороннего предсказания движения

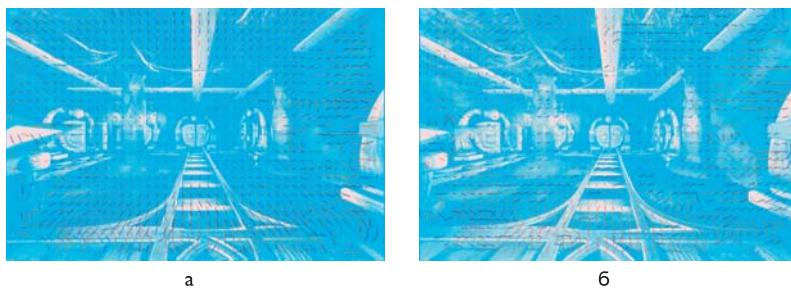


Рис. 14.62. Векторы движения для кадра при предсказании по предшествующему ему по времени кадру (а) и по второму кадру из состава стереопары (б) при использовании двунаправленного предсказания движения

ет чересстрочную развертку, то выполнять поиск векторов предпочтительнее по отдельным полям, поскольку это дает несколько лучшие результаты предсказания.

Опираясь на все вышесказанное, процесс кодирования стереоскопических изображений может быть представлен в виде схемы, изображенной на рис. 14.63.

Здесь левые кадры последовательности стереоскопических изображений поступают в блок поиска векторов движения относительно опорных кадров из числа кадров этой же последовательности и результаты предсказания подвергаются кодированию для формирования битового потока базового слоя, который может быть декодирован любым стандартным декодером.

Пороги сохранения стереоэффекта для различных разрешений кадра и скоростей выходного потока определяются условиями появления заметных искажений и артефактов на декодированных кадрах. Заметные визуальные искажения, особенно несогласованные между левой и правой компонентами стереопары, разрушают стереоэффект.

Для кодирования видеопоследовательностей в стандарте MPEG-2 наиболее часто используемым значением размера группы кадров (последовательный набор кадров, который начинается с кадра, имеющего только внутрикадровое предсказание) является 16. При этом последовательность типов кадров в группе выглядит следующим образом: IBBPBBPBBPBBPBBP. Именно таким образом и выглядит состав группы кадров при кодировании базового слоя (левых кадров сте-

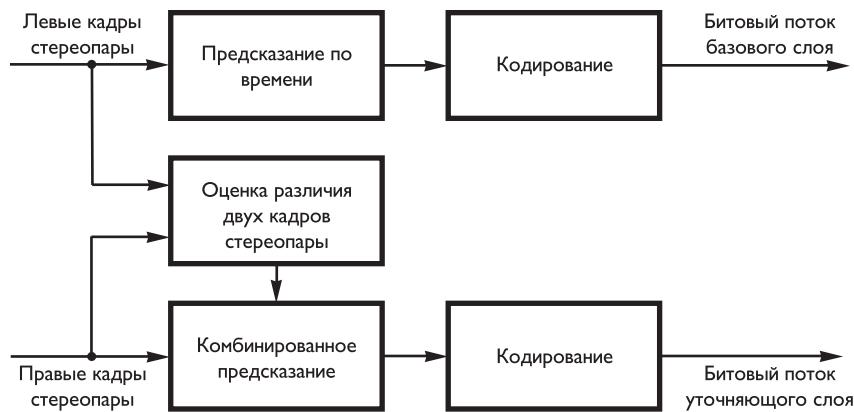


Рис. 14.63. Схема кодирования стереоскопической видеопоследовательности

реопар). Но для кодирования уточняющего слоя типы кадров меняются: РВВВ ВВВВВВВВВВВ. Проведенные ранее исследования показали, что в среднем размер I-кадра превосходит размер B-кадра в 5 раз, а P-кадр больше B-кадра в 2 раза. Это означает, что размер группы кадров из шестнадцати левых изображений стереоскопического сигнала превосходит по количеству выделенных на него битов размер группы кадров из шестнадцати правых изображений в среднем в два раза.

Таким образом, при кодировании стереоскопического видео с разделением на базовый и уточняющий слои и использованием оценки различия между двумя изображениями стереопары формируется битовый поток, всего лишь в полтора раза превосходящий по размеру поток кодирования кадров от одной камеры.

14.8. Визуализация параметров видеокодирования

Сложность и многообразие методов обработки видеинформации, а так же большая гибкость в выборе этих методов, предоставляемая кодеру стандартом H.264/AVC, делают непростой задачу анализа результатов обработки динамических изображений. Поэтому представляется крайне важным создание системы визуализации, которая позволяет облегчить решение задачи анализа путем предоставления детальной информации о параметрах закодированного видео в удобном, визуально легко воспринимаемом виде [5.85]. Полученная информация позволяет выявлять возможные проблемы и корректировать алгоритм кодера (рис. 14.64).

Основная идея заключается в определении параметров закодированного кадра и их визуализации путем наложения во время воспроизведения видео. Графический интерфейс приложения предоставляет пользователю возможность загружать, сохранять и анализировать видео-файлы, а также возможность навигации внутри видеопотока. Пользователь может использовать покадровое воспроизведение для отображения исходных видеоданных и дополнительной информации, которая накладывается на воспроизведение видео.

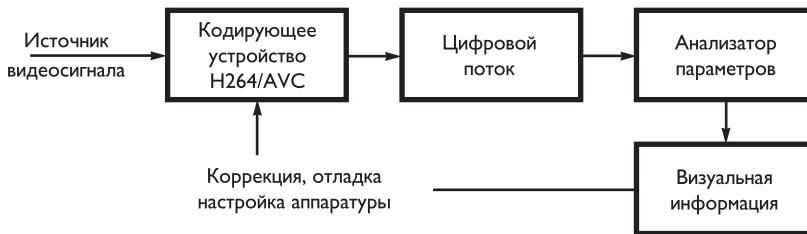


Рис. 14.64. Принцип визуализации алгоритмов обработки видеинформации

Внутрикадровое предсказание

Во всех типах кодирования I-кадров стандартом H.264/AVC поддерживаются типы внутрикадрового кодирования, обозначаемые Intra_16 × 16 или Intra_4 × 4, а также предсказание цветности и I-PCM предсказание (см.раздел 14.5).

Режим Intra_16 × 16 осуществляет предсказание блока яркости 16 × 16 целиком и более подходит для очень гладких областей изображения. Режим Intra_4 × 4 основан на предсказании каждого блока яркости 4 × 4 отдельно и подходит для кодирования частей изображения с существенными деталями. В дополнение к этим двум типам предсказания яркости вводится отдельный режим предсказания цветности. Как альтернатива режимам кодирования Intra_4 × 4 и Intra_16 × 16 режим кодирования I-PCM позволяет кодеру просто обходить процессы предсказания и кодирования с преобразованием и вместо этого прямо посыпать значения кодируемых отсчетов.

При использовании режима Intra_16 × 16 предсказывается целиком яркостная компонента макроБЛОКА. Поддерживаются 4 режима предсказания. Режим 0 (вертикальное предсказание), режим 1 (горизонтальное предсказание) и режим 2 (DC-предсказание) определены аналогично соответствующим режимам предсказания Intra_4 × 4 за исключением того что вместо 4 соседей на каждой стороне для предсказания блока 4 × 4 используются 16 соседних отсчетов на каждой стороне для предсказания блока 16 × 16. Четвертый режим предсказания — так называемое плоскостное (plane) предсказание. Режимы формирования прогноза INTRA-блоков размером 16 × 16 поясняются рис. 14.30.

При использовании режима Intra_4 × 4 каждый блок 4 × 4 предсказывается по пространственно соседним отсчетам. Для каждого блока 4 × 4 используется один из 9 режимов предсказания. В дополнение к DC-предсказанию (одно значение используется для предсказания всего блока 4 × 4), имеются 8 режимов «направленного» предсказания, как показано на рис. 14.29. Эти режимы полезны для предсказания направленных структур в изображении, таких как границы, расположенные под различными углами.

Режим 0 (вертикальное предсказание), режим 1 (горизонтальное предсказание) и режим 2 (DC-предсказание), в режимах 3 и 4 отсчеты интерполируются под углом в 45° в направлении между нижним левым и верхним правым отсчетами и под углом 45° вниз и направо соответственно, в режимах 5 — 8 интерполяция отсчетов производится под углом приблизительно 26,6° соответственно влево к вертикальному направлению, вниз по отношению к горизонтальному направлению, вправо от вертикального направления и выше горизонтального направления.

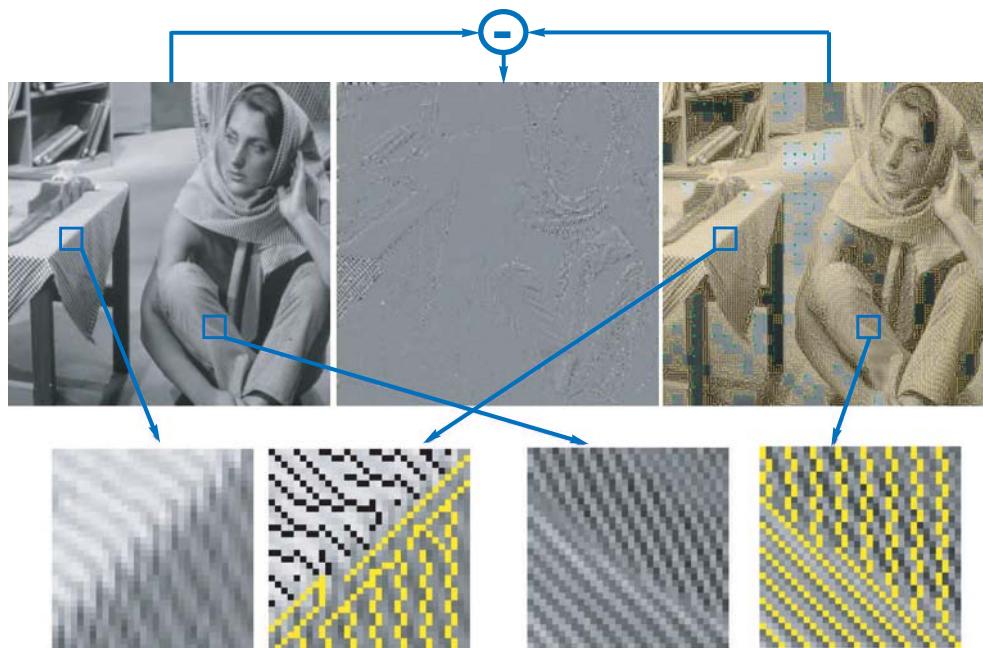


Рис. 14.65. Визуализация внутрикадрового предсказания изображения «Барбара»

Принцип визуализации этих возможностей приведен на рис. 14.65–14.67. Режимы предсказания блоков 16×16 изображаются зеленым цветом, режимы блоков 4×4 — желтым, начало блока 4×4 обозначается красной точкой. Это дает возможность визуально воспринимать картину предсказания во всех деталях и в целом.

На этих рисунках внизу приведены увеличенные фрагменты изображения, на котором можно более детально рассмотреть как предсказываются текстуры различных направлений. К остаточному кадру для увеличения яркости добавляется постоянная составляющая, соответствующая уровню серого. Остаточный кадр иллюстрирует малую энергию остатков и высокую эффективность внутрикадрового предсказания. Кроме того, визуализатор позволяет выводить предсказанный кадр. Данные изображения позволяют сделать вывод об адекватной работе кодера.

Межкадровое предсказание в P-кадрах

В дополнение к типам макроблоков с внутрикадровым кодированием определены различные типы P-макроблоков как типы кодирования с компенсацией движения. Синтаксис поддерживает деление на блоки яркости размером 16×16 , 16×8 , 8×16 и 8×8 отсчетов.

Если выбрано деление на блоки 8×8 отсчетов, то передается дополнительный синтаксический элемент, который определяет, производится ли дальнейшее деление блока 8×8 на блоки 8×4 , 4×8 или 4×4 отсчета яркости и соответствующие отсчеты цветности (рис. 14.31).

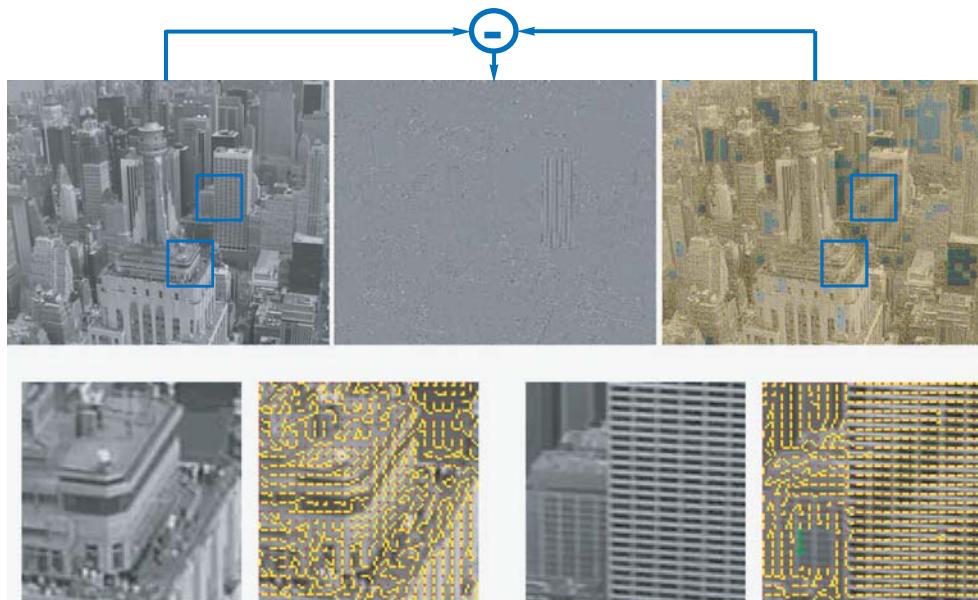


Рис. 14.66. Визуализация внутрикадрового предсказания изображения «Город»

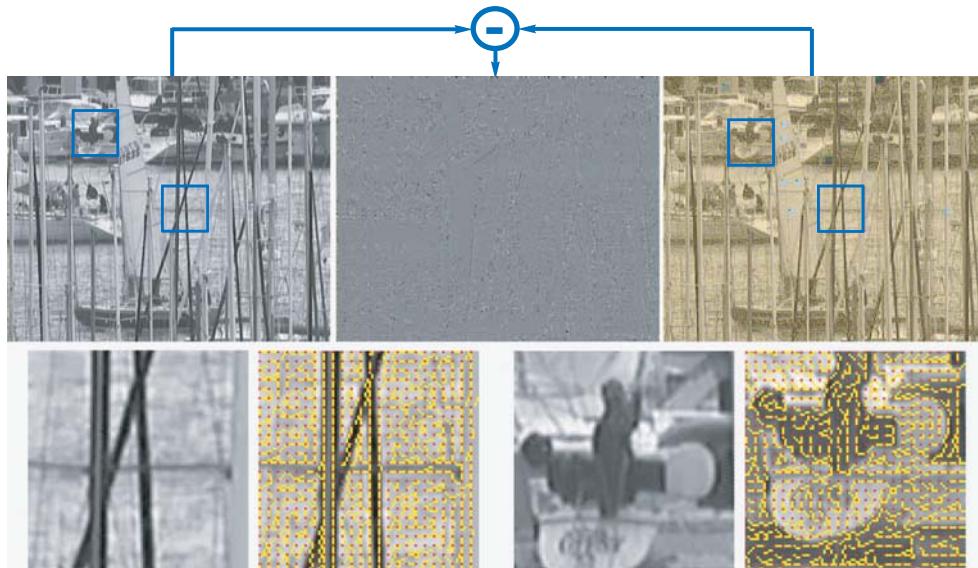


Рис. 14.67. Визуализация внутрикадрового предсказания изображения «Залив»

Кодер выбирает «лучший» размер блока для каждой части кадра, т. е. такой размер, который минимизирует кодирование остаточных коэффициентов и векторов движения. В областях кадра, где изменения незначительны, выбирается

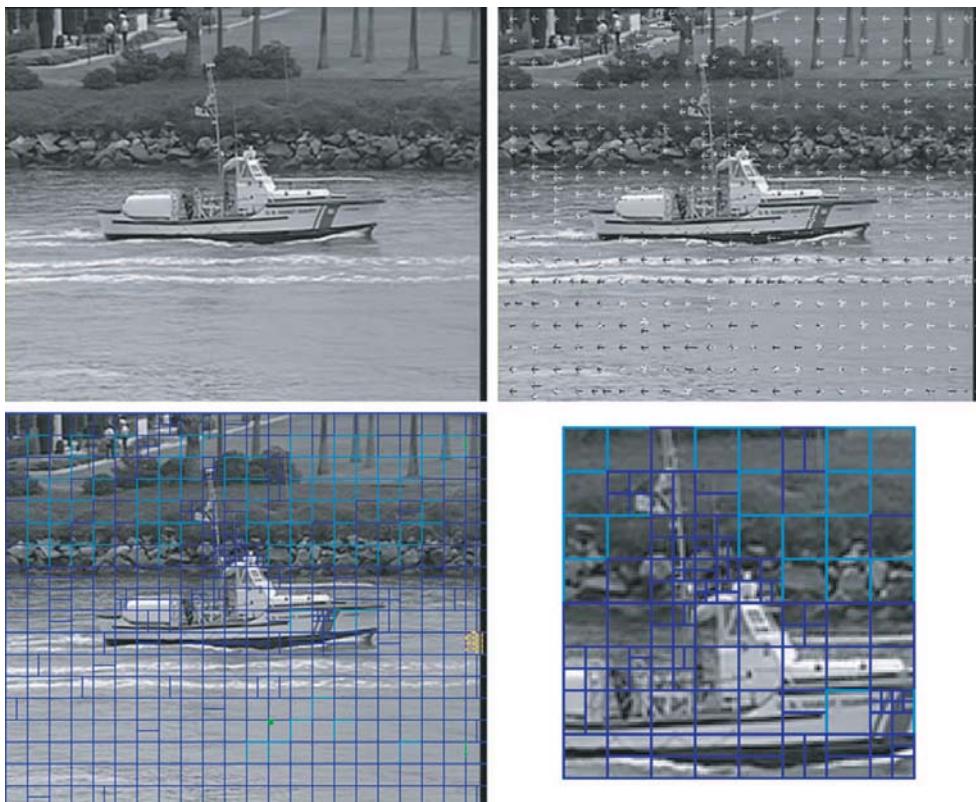


Рис. 14.68. Кадр изображения и его разбиение на блоки различных размеров

размер блока 16×16 ; в областях с большими изменениями выбираются меньшие размеры блоков.

Визуализатор позволяет показывать для каждого кадра размеры блоков предсказания и вектора движения для них. На рис. 14.68 слева направо приведен исходный кадр, исходный кадр с нанесенными векторами движения, кадр с нанесенными размерами блоков и его фрагмент.

Помимо этого визуализатор позволяет показывать для каждого кадра коэффициенты квантования макроблоков, битовый размер макроблоков, а так же программа может выдать статистическую информацию о видеопотоке: среднее число битов на макроблок, средний коэффициент квантования, соотношение различных типов кадров, размер текущего кадра в байтах, число макроблоков различных типов текущего кадра.

Каждый блок цветности разбивается таким же образом, как яркостный. Однако вертикальные и горизонтальные размеры такого блока в два раза меньше.

Точность компенсации движения — четверть расстояния между отсчетами яркости. Величины предсказания в полупиксельных позициях получаются с помощью применения одномерного 6-точечного КИХ-фильтра (рис. 14.33). Величины предсказания в четвертьпиксельных позициях получаются с помощью

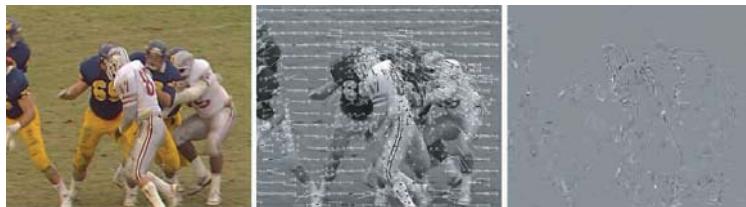


Рис. 14.69. Формирование разностного сигнала в Р-кадрах

усреднения значений в целочисленных и полупиксельных позициях (рис. 14.34). Оценка смещения элементов цветности обеспечивается с точностью до 1/8-пиксельного интервала.

Векторы движения соседних блоков зачастую оказываются коррелированными, поэтому вектор-предсказание формируется на основе ранее вычисленных векторов движения и передается только вектор разности между истинным значением вектора и предсказанным значением. Более точное предсказание движения с использованием целопиксельного, полупиксельного и четвертьпиксельного предсказания сигналов яркости и дополнительно 1/8-пиксельное предсказание сигнала цветности представляет собой одно из главных улучшений по отношению к предшествующим стандартам по нескольким причинам. Наиболее очевидная причина — более точное представление движения. Другая причина — большая гибкость фильтрации предсказания. Целопиксельное, полупиксельное и четвертьпиксельное предсказание представляет различные степени низкочастотной фильтрации, которые автоматически выбираются в процессе поиска движения.

На рис. 14.69 приведен вариант обработки Р-кадра с четвертьпиксельным предсказанием. Здесь слева приведено исходное изображение, в середине — изображена составляющая яркости с векторами движения, а справа — разностная составляющая.

Межкадровое предсказание в В-кадрах

Существенная разница между В- и Р-кадрами состоит в том, что В-кадры кодируются так, что некоторые макроБЛОКИ или блоки могут использовать взвешенное среднее двух различных значений предсказания с компенсацией движения для построения сигнала предсказания.

Кодирование векторов движения в В-кадрах похоже на кодирование векторов движения в Р-кадрах с соответствующими изменениями, так как соседние блоки могут быть закодированы с использованием различных режимов предсказания.

На рис. 14.70 представлено изображение кадра и двух разностных сигналов, сформированных с использованием прямого и двунаправленного предсказания. Для визуального восприятия различия изображений разностные сигналы увеличены в два раза.

Деблокирующая фильтрация

Деблокирующий фильтр применяется после обратного преобразования в кодере (при восстановлении и сохранении макроБЛОКА для формирования прогноза следующего кадра) и в декодере (при восстановлении и отображении макроБЛОКА).



Рис. 14.70. Формирование разностного сигнала при прямом и двунаправленном предсказаниях

Фильтр используется для сглаживания блочных артефактов. При этом:

- границы блока сглаживаются, улучшая зрительное восприятие декодированных изображений (особенно при больших коэффициентах сжатия);
- отфильтрованный макроблок используется в режиме компенсации движения деталей изображения в кодере, что значительно уменьшает величину остаточных коэффициентов.

Фильтрация применяется к вертикальным и горизонтальным границам блоков размером 4×4 (рис. 14.40).

Параметры деблокирующего фильтра зависят от типа границ между блоками. На рис. 14.41 показан пример применения деблокирующей фильтрации. Деблокирующий фильтр увеличивает эффективность сжатия незначительно: скорость выходного битового потока уменьшается не более чем на 1–3%, а PSNR увеличивается примерно на 1%. Однако субъективное качество восстановленной видео-последовательности с деблокирующей фильтрацией значительно выше. Увеличение эффективности кодирования, обеспечиваемое деблокирующими фильтром, зависит от содержания последовательности и требуемой выходной скорости за-кодированного потока.

Особо следует подчеркнуть, что деблокирующая фильтрация весьма эффективна в режимах компенсации движения деталей изображения.

В презентационном диске приведены динамические изображения результатов визуализации алгоритмов обработки видеоинформации. Они также проиллюстрированы на рис. 14.71–14.76.

Рассмотренная система является удобным инструментом визуального анализа и контроля работы алгоритма кодирования, позволяющая разработчику или исследователю выявлять ошибки и неоптимальные решения в работе кодера, сравнивать работу разных кодеров, может использоваться для обучения студентов и специалистов, изучающих современные системы видеокодирования и, в частности, стандарт H.264.



Рис. 14.71. Разбиение последовательности кадров на блоки различных размеров



Рис. 14.72. Динамика векторов движения в блоках различного размера



Рис. 14.73. Векторы преимущественно горизонтального движения объектов

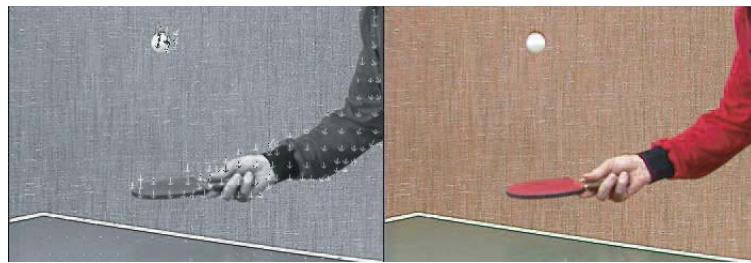


Рис. 14.74. Векторы преимущественно вертикального движения объектов



Рис. 14.75. Динамика движения деталей в эталонной видеопоследовательности «Сад цветов»



Рис. 14.76. Динамика движения деталей в эталонной видеопоследовательности «Американский футбол»

Литература к части V

- 5.1. Цифровая обработка телевизионных и компьютерных изображений. 2-е изд., перераб. и доп. / Под ред. Ю.Б. Зубарева и В.П. Дворковича В.П. М.: НАТ, 1997.
- 5.2. *Ватолин Д., Ратушняк А., Смирнов М., Юкин В.* Методы сжатия данных. Устройство архиваторов, сжатие изображений и видео. М.: ДИАЛОГ-МИФИ, 2003.
- 5.3. Стандарты кодирования мультимедиа и гипермедиа-информации ISO (рабочая группа JTS1/SC29).
http://blog.piclab.ru/wp-content/uploads/ISO_Standarts.pdf
- 5.4. ISO/IEC DIS 10918-1. Information Technology — Digital Compression and Coding of Continuous-tone Still Images: Requirements and Guidelines./ Ed.1, JTS 1/ SC 29, 1994
ITU. CCITT. Recomendation T 61. Terminal Eqipment and Protocols for Telematic Servises. Information Technology — Digital Compession and Coding of Continuou-Tone Still Images – Requirements and Guidelines, 1993.
- 5.5. ISO/IEC DIS 10918-2. Information Technology — Digital Compression and Coding of Continuous-tone Still Images: Compliance Testing./ Ed.1, JTS 1/ SC 29, 1994
Recommendation ITU-T T.83. Series T: Terminals for Telematic Servises. Information Technology - Digital compression and coding of continuous-tone still images: Compliance testing , 1995.
- 5.6. ISO/IEC DIS 10918-3. Information Technology — Digital Compression and Coding of Continuous-tone Still Images: Extensions./ Ed.1, JTS 1/ SC 29, 1994
Recommendation ITU-T T.84. Series T: Terminals for Telematic Servises. Information Technology- Amendment 1: Provisions to allow Registration of new Compression Types and Versions in the SPIFF Header, 1999.
- 5.7. ISO/IEC DIS 10918-4. Information technology — Digital compression and coding of continuous-tone Still images: Registration of JPEG profiles, SPIFF profiles, SPIFF tags, SPIFF colour spaces, APPn markers, SPIFF compression types and Registration Authorities (REGAUT)./ Ed.1, JTS 1/ SC 29, 1999.
Recommendation ITU-T T.86. Series T: Terminals for Telematic Servises. Information technology — Digital compression and coding of continuous-tone still images: Registration of JPEG Profiles, SPIFF Profiles, SPIFF Tags, SPIFF colour Spaces, APPn Markers, SPIFF Compression types and Registration authorities (REGAUT), 1999.
- 5.8. The JPEG Still Picture Compression Standard// Communications of the ACM. V.34. № 4. April, 1991.
- 5.9. *Wallace G.K.* The JPEG Still Picture Compression Standard// Communication of the ACM. V.34. № 4. April, 1991.
- 5.10. *King N. Ngan, Hee C. Koh., Wai C. Wong.* Hybrid Image Coding Scheme Incorporating Human Visual System Characteristics// Optical Engineering. V. 30. № 7. 1991.

- 5.11. МККР. Параметры кодирования сигналов цифрового телевидения для студий: Рекомендация 601.
- 5.12. Сэломон Д. Сжатие данных изображений и звука / Пер. с англ. В.В. Чепыжова. М.: Техносфера, 2004.
- 5.13. Госалес Р., Вудс Р. Цифровая обработка изображений/ Пер. с англ. П.А. Чочиа. М.: Техносфера, 2005.
- 5.14. Dvorkovitch V.P., Nechepaev V.V., Mokhin G.N., Dvorkovitch A.V. Digital Compression of TV Images for the Transmission through the Satellite Channels: International Conference on Satellite Communications. Proceedings. V.1. Moscow. October, 1994.
- 5.15. ISO/IEC 15444-1. Information technology — JPEG 2000 image coding system: Core coding system./ Ed.1, JTS 1/ SC 29, 2004
Recommendation ITU-T T.800. Series T: Information technology — JPEG 2000 image coding system: Core coding system, 2005.
- 5.16. ISO/IEC 15444-2. Information technology — JPEG 2000 image coding system: Extensions./ Ed.1, JTS 1/ SC 29, 2004
Recommendation ITU-T T.801. Series T: Information technology — JPEG 2000 image coding system: Extensions, 2005.
- 5.17. ISO/IEC 15444-3. Information technology — JPEG 2000 image coding system: Motion JPEG 2000./ Ed.1, JTS 1/ SC 29, 2007
Recommendation ITU-T T.802. Series T: Information technology — JPEG 2000 image coding system: Motion JPEG 2000, 2008.
- 5.18. ISO/IEC 15444-4. Information technology — JPEG 2000 image coding system: Conformance testing./ Ed.1, JTS 1/ SC 29, 2004
Recommendation ITU-T T.803. Series T: Information technology — JPEG 2000 image coding system: Conformance testing, 2008.
- 5.19. ISO/IEC 15444-6. Information technology — JPEG 2000 image coding system — Part 6: Compound image file format./ Ed.1, JTS 1/ SC 29, 2003.
- 5.20. ISO/IEC 15444-9. Information technology — JPEG 2000 image coding system: Interactivity tools, APIs and protocols./ Ed.1, JTS 1/ SC 29, 2005
Recommendation ITU-T T.808. Series T: Information technology — JPEG 2000 image coding system: Interactivity tools, APIs and protocols, 2005.
- 5.21. ISO/IEC 15444-12. Information technology — JPEG 2000 image coding system — Part 12: ISO base media file format./ Ed.1, JTS 1/ SC 29, 2008.
- 5.22. ISO/IEC 14495-1. Information technology. Lossless and near-lossless compression of continuous-tone still images. Part 1: Baseline/JTC 1/SC 29, 1999.
- 5.23. ISO/IEC 14495-2. Information technology — Lossless and near-lossless compression of continuous-tone still images. Part 2: Extensions/JTC 1/SC 29, 2003.
- 5.24. Рыбаков Г., Суслов А. JPEG, JPEG2000, JPEG-LS. Сжатие изображений с потерями и без. <http://rain.ifmo.ru/cat/view.php/theory/data-compression/jpeg-2006>
- 5.25. Борисов М. Сжатие изображений: JPEG и JPEG 2000.
http://nnm.ru/blogs/badylai/szhatie_izobrazhenij_jpeg_i_jpeg2000



- 5.26. Дворкович В.П., Дворкович А.В. Расчет банков фильтров дискретного вейвлет-преобразования и анализ их характеристик // ЦОС. 2006. № 2.
- 5.27. Recommendations of the H-Series, CCITT Study Group XV, Report R37, 1990.
- 5.28. ITU-T. Recommendations : H Series : H.261—Video codec for audiovisual services at p x 64 kbit/s, 1993
- 5.29. ISO/IEC 11172-1: Information technology — Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s — Part 1: Systems./ Ed.1, JTS 1/ SC 29, 1993.
- 5.30. ISO/IEC 11172-2: Information technology — Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s — Part 2: Video./ Ed.1, JTS 1/ SC 29, 1993.
- 5.31. ISO/IEC 11172-3. Information Technology — Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1.5 Mbit/s. Part 3: Audio./ Ed.1, JTS 1/ SC 29, 1993.
- 5.32. ISO/IEC 11172-4. Information Technology — Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1.5 Mbit/s. Part 4: Conformance Testing./ Ed.1, JTS 1/ SC 29, 1993.
- 5.33. ISO/IEC 13818-1. Information Technology — Generic Coding of Moving Pictures and Associated Audio Information. Part 1: Systems./ Ed.1, JTS 1/ SC 29, 1994.
- 5.34. ISO/IEC 13818-2. Information Technology — Generic Coding of Moving Pictures and Associated Audio Information. Part 2: Video./ Ed.1, JTS 1/ SC 29, 1994.
- 5.35. ISO/IEC 13818-3. Information Technology — Generic Coding of Moving Pictures and Associated Audio Information. Part 3: Audio./ Ed.1, JTS 1/ SC 29, 1994.
- 5.36. ISO/IEC 13818-4. Information Technology — Generic Coding of Moving Pictures and Associated Audio Information. Part 4: Conformance Testing./ Ed.1, JTS 1/ SC 29, 1995.
- 5.37. ISO/IEC 13818-5. Information Technology — Generic Coding of Moving Pictures and Associated Audio Information. Part 5: Technical Report./ Ed.1, JTS 1/ SC 29, 1995.
- 5.38. ITU-T. Recommendations: H Series: H.262—Information technology — Generic coding of moving pictures and associated audio information: Video, 2000.
- 5.39. Digicipher HDTV System. Submitted by General Instrument Corporation Vi-deocipher Division. San Diego, California. June, 1990.
- 5.40. Advanced Digital Television (ADTV). System Discription. NBC- Philips-Thomson./ David Sarnoff Research Center. Feb.,1991.
- 5.41. 422P@ML Amendment: «ISO/IEC 13818-2 Amendment 2: Information Technology — Generic Coding of Moving Pictures and Associated Audio Information. Video. 1996.
- 5.42. ITU-T. Recommendations: H Series: H.263—Information technology — Video coding for low bit rate communication, 1995.

- 5.43. ITU-T. Recommendations: H Series : H.263v.2—Information technology — Video coding for low bit rate communication (H.263+), 1998.
ITU-T. Recommendations: H Series: H.263v.3—Information technology — Implementors Guide for H.263: «Video coding for low bit rate communication» (H.263++ or H.263 2000), 2000.
- 5.44. ISO/IEC 14496-1:Information technology — Coding of audio-visual objects — Part 1: Systems, AMENDMENT 4: Usage of LASer in MPEG-4 systems and Registration Authority for MPEG-4 descriptors/ JTC1/SC29/WG11, 2004.
- 5.45. ISO/IEC 14496-2: Information technology — MPEG-4/Coding of audio-visual objects — Part 2: Visual, Advanced Simple Profile (ASP)/ JTC1/SC29/WG11, 2004.
- 5.46. ISO/IEC 14496-3: Information technology — Coding of audio-visual objects — Part 3: Audio, AMENDMENT 1: HD-AAC profile and MPEG Surround signaling/ JTC1/SC29/WG11, 2004.
- 5.47. ISO/IEC 14496-4: Information technology — Coding of audio-visual objects — Part 4: Conformance testing, AMENDMENT 37: Additional file format conformance, JTC1/SC29/WG11, 2004.
- 5.48. ISO/IEC 14496-10: Advanced Video Coding: A codec for video signals which is also called AVC and is technically identical to the ITU-T H.264 standard/JTC1/ SC29/WG11, 2005.
- 5.49. ITU-T. Recommendations: H Series: H.264—Information technology — Advanced video coding for generic audiovisual services, 2004.
- 5.50. ITU-T. Recommendations: H Series: H.320-Narrow-band visual telephone systems and terminal equipment, 1990.
- 5.51. *Rao K.R., Hwang J.J.* Techniques and Standards for Image, Video and Audio Coding, Prentice Hall PTR, NJ, 1996.
- 5.52. CCITT SG XV WP/1/Q4 Specialist Group on Coding for Visual Telephony, Description of Ref. Model 6 (RM6). Document 396. Oct., 1988.
- 5.53. *Pang K.K., Tan T.K.* «Optimum loop filter in Hybrid coders»// IEEE Trans CSVT. 1994. V.4, 4. P.158-167.
- 5.54. Prof. Tsuhan Chen. Multimedia Communications: Coding, Systems, and Networking./ H.263, H.263v2, and H.26L./
<http://www.ece.cmu.edu/~ece796/h263.PDF>
- 5.55. MPEG-1/<http://en.wikipedia.org/wiki/MPEG-1>
- 5.56. MPEG: Вопросы и ответы./<http://www.elecard.com/mpeg/faq/index.php>
- 5.57. Семейство форматов MPEG. Часть первая. MPEG-1/
<http://www.videoton.ru/Articles/mpeg1.html>
- 5.58. Семейство форматов MPEG. Часть вторая. MPEG-2/
<http://www.3dnews.ru/multimedia/mpeg2.html>
- 5.59. Short MPEG-2 description/
<http://www.chiariglione.org/mpeg/standards/mpeg-2/mpeg-2.htm>



- 5.60. Rao K.R., Hwang J.J. Techniques and Standards for Image, Video and Audio Coding, Prentice Hall PTR, NJ. 1996.
- 5.61. ITU-R Recommendation BT.709-5 Parameter values for the HDTV standards for production and international programme exchange. 2002.
- 5.62. ITU-R Report BT.624-4 Characteristics of television systems. 1990.
- 5.63. SMPTE 170M — Television — Composite Analog Video Signal — NTSC for Studio Applications. 2004.
- 5.64. SMPTE 240M — Television — 1125-Line High-Definition Production Systems — Signal Parameters. 1999 (2004).
- 5.65. Семенов Ю.А. Стандарт MPEG-4. Ч. 1; Ч. 2. /
<http://directshow.Wonderu.cjm/mpeg-4; mpeg-4-2>
- 5.66. Levels for MPEG-4 Visual Profiles. /
<http://www.m4if.org/resources/profiles/index.php>
- 5.67. Ричардсон Ян. Видеокодирование. H.264 и MPEG-4 — стандарты нового поколения / Пер. с англ. В.В. Чепыжева. М.: Техносфера, 2005.
- 5.68. Гук И. Особенности сжатия видеоданных по рекомендации H.264/MPEG-4, Part 10./«Компоненты и технологии», № 2, 2006,
http://www.kit-e.ru/articles/dsp/2006_2_20.php
- 5.69. H.264/MPEG-4 AVC is a Standart vor Videj Compression. /
http://en.wikipedia.org/wiki/H.264/MPEG-4_AVC
- 5.70. Стандарт сжатия видеонизображения H.264. Новые возможности в области охранного видеонаблюдения
<http://articles.security-bridge.com/articles/10/12198>
- 5.71. Дворкович А.В. Новые технологии видеокодирования в стандарте H.264// 7-я Международная конференция «Цифровая обработка сигналов и ее применение», 16-18 марта 2005. Москва. Доклады, т. 2.
- 5.72. Дворкович А.В. Эффективное кодирование видеонформации в новом стандарте H.264/AVC// Труды НИИР, 2005.
- 5.73. Дворкович А.В., Мингазов И.Д. Методика построения промежуточных кадров видеопоследовательности // 6-я Международная конференция «Цифровая обработка сигналов и ее применение», 31 марта — 2 апреля 2004. Москва. Доклады, т. 2.
- 5.74. Дворкович В.П., Дворкович А.В., Мохин Г.Н., Соколов А.Ю. Компьютерный масштабируемый видеокодек для узкополосных каналов связи// Электросвязь. 1999. № 10.
- 5.75. Lee Hung-Ju, Chiang Tihao, Zhang Ya-Qin. Scalable rate control for MPEG-4 video// IEEE Transactions on circuits and systems for video technology. Vol. 10. № 6. September, 2000.
- 5.76. Maeda J. «Method for Extracting Camera Operations in Order to Describe SuB-Scenes in Video Sequences», in Digital Video Compression on Personal Computers.Algorithms and Technologies / Arturo A. Rodriguez Ed. Proc. SPIE 2187, 1994.

- 5.77. *Cherfaoui M., Bertin C.* «Temporal Segmentation of Videos: a New Approach», in Digital Video Compression: Algorithms and Technologies 1995 / Arturo A. Rodriguez Ed., Proc. SPIE 2419, 1995.
- 5.78. *Nagasaka A. and Tanaka Y.* «Automatic Video Indexing and Full-Video Search for Object Appearances», in Proc. of IFIP 2nd Working Conf. on Video Database Systems / E. Knuth and L.M. Wegner, eds. Elsevier Science Publ., 1992.
- 5.79. *Дворкович А.В., Быстров С.В.* Методика контроля потока при сжатии динамических изображений// 5-я Международная конференция «Цифровая обработка сигналов и ее применение», 12–14 апреля 2003. Москва. Доклады, т. 1.
- 5.80. *Дворкович А.В., Быстров С.В.* Методика регулировки потока при сжатии динамических видеоизображений// ЦОС. 2003. № 2.
- 5.81. *Гулевич А.Е., Дворкович А.В., Коcharян А.Э., Мингазов И.Д.* Реализация многофункционального кодера H.264// Международная конференция «Цифровая обработка сигналов и ее применение», 25–27 марта 2009. Москва. Доклады, т. 2.
- 5.82. *Дворкович А.В.* Особенности построения программного телевизионного кодера MPEG-2// ЦОС. 2002. № 1.
- 5.83. *Дворкович А.В., Мингазов И.Д.* Особенности программной реализации цифрового телевизионного кодера MPEG-2// 5-я Международная конференция «Цифровая обработка сигналов и ее применение», 12–14 апреля 2003. Москва. Доклады, т. 1.
- 5.84. *Schreer Oliver, Kauff Peter, Sikora Thomas.* 3D Videocommunication — Algorithms, concepts and real-time systems in human centred communication. МЕСТО 2005.
- 5.85. *Дворкович В.П., Гулевич А.Е.* Визуализация параметров кодирования видеоинформации// 12-я Международная конференция «Цифровая обработка сигналов и ее применение», март–апрель 2010. Москва. Доклады, т. 1.

ЧАСТЬ VI

ОСНОВНЫЕ МЕТОДЫ КОДИРОВАНИЯ РЕЧЕВОЙ И ЗВУКОВОЙ ИНФОРМАЦИИ

Введение

Процесс цифровой обработки речевой и звуковой информации охватил различные виды приложений и существенно расширил их спектр: кроме широко распространенных систем телефонии, традиционного телерадиовещания и различных разновидностей систем специального назначения появились и повсеместно внедряются самые различные приложения — видеотелефония и конференцсвязь, интернет-вещание и мобильные мультимедийные приложения, телевидение высокой четкости и цифровое кино.

Цифровые системы кодирования речи и звука, их передачи и приема позволяют достичь невиданного доселе уровня качества и предоставляют пользователю массу новых возможностей и новых видов услуг.

В настоящей части пособия рассматриваются специфические особенности преобразования аналоговых и цифровых речевых и звуковых сигналов, форматы их представления при формировании, записи, передаче и воспроизведении. Конкретные виды аппаратной и программной реализации систем цифровой обработки речевой и звуковой информации определяются их исходными характеристиками, особенностями слухового восприятия и требованиями к качеству воспроизведения.

Речевая информация, образующая свойственные используемому языку фонетические комбинации и формирующая те или иные смысловые элементы, по своим физическим параметрам принципиально отличается от звуковой информации, содержащей сочетание голосовых данных с музыкальным сопровождением.

и обладающей различными форматами (от моно- и стерео- до многоканальных систем). Особенности и отличия друг от друга речевой и звуковой информации используются при их цифровой обработке и сжатии.

Известно, что основную информацию о звуковых колебаниях мозг человека получает в области частот примерно до 4 кГц и именно эти частоты «ответственны» за разборчивость и ясность аудиоинформации.

Спектральный состав речи занимает полосу частот примерно от 50 до 7000–10 000 Гц. Но еще в аналоговой телефонии было принято использовать полосу частот 0,3–3,4 кГц, что ухудшило восприятие ряда звуков (например, шипящих, существенная часть энергии которых сосредоточена в верхней части речевого спектра), но практически не отразилось на разборчивости передаваемой информации. В цифровых системах к влиянию ограничения спектра добавляются еще шумы дискретизации, квантования и обработки, дополнительно запутывающие речь. Для совместимости по полосе с распространенными аналоговыми сетями в цифровой телефонии отсчеты аналоговой речи приходится брать согласно теореме Котельникова с частотой 8 кГц. Однако в цифровых системах возможно использовать спектр речи за пределами полосы 0,3–3,4 кГц и тем самым повысить качество. Например, при полосе исходного сигнала 6–7 кГц и тактовой частоте отсчетов около 16 кГц сжатый цифровой сигнал требует для передачи канал с пропускной способностью около 12 кбитов/с.

Что касается разрядности аналого-цифрового преобразования речевой информации, то очевидно, что при очень большой разрядности АЦП в кодере восстановленная речь в декодере будет практически точно соответствовать оригиналу. На практике приходится выбирать разрядность кодека и мириться с некоторой погрешностью квантования. Для всех типов кодеков справедливо правило: чем меньше плотность цифрового потока, тем больше восстановленный сигнал отличается от оригинала.

Идея преобразовывать в цифровой вид не сам речевой сигнал, а его параметры (количество переходов через ноль, спектральные характеристики и др.), чтобы затем по этим параметрам выбирать модель голосового тракта и синтезировать исходный сигнал, лежит в основе синтезирующих кодеков или вокодеров.

Принцип работы гибридных кодеков основан на модели кодирования с использованием линейного предсказания с возбуждением по алгебраической кодовой книге. При этом производится анализ речевого сигнала и выделяются параметры модели (коэффициенты системы линейного предсказания, индексы и коэффициенты усиления в адаптивной и фиксированной кодовых книгах). Далее эти параметры кодируются и передаются в канал. В декодере битовая посылка используется для восстановления параметров сигнала возбуждения и коэффициентов синтезирующей системы. Восстановленный таким образом сигнал обладает вполне высокими характеристиками — восстанавливается тембр речевого сигнала, его динамические характеристики, другими словами, его «узнаваемость» и «распознаваемость».

В практическом приложении при вокодерной обработке возникает задержка воспроизведения речевых сигналов, определяемая временем, затрачиваемым на: процессы кодирования и декодирования; передачу по каналу; мультиплексирование при комбинировании аудиоданных с другими видами данных.

Иные проблемы возникают при кодировании звуковой информации, содержащей дикторскую речь, актерское пение, музыкальное сопровождение и др.

При цифровой обработке звуковой информации, как и при кодировании речи, необходимо обеспечить существенное сжатие информации при минимизации задержки воспроизведения звуковых сигналов. Однако в этом случае исходный сигнал обладает существенно иными параметрами, поскольку слуховой аппарат человека различает частотные составляющие звука приблизительно в пределах от 20–30 Гц до 20 кГц (так называемый слышимый звук). Верхняя граница может несколько отличаться от указанной величины в зависимости от возраста человека, условий воспроизведения информации и др.

Несмотря на то что основную информацию несут составляющие звука с частотами до 4 кГц, более высокие частоты отвечают за субъективное качество его воспроизведения.

Частоты несколько ниже 20–30 Гц (инфразвук) также могут восприниматься человеком, но не ухом, а как вибрации всем телом. Считается, что частоты, превышающие верхний порог в 20 кГц (ультразвук) также могут восприниматься человеком на подсознательном уровне, непосредственно мозгом.

В связи с этим при кодировании звуковой информации в различных кодеках используются частоты дискретизации 44,1 кГц, 48 кГц и даже 96 кГц, а применяемые шкалы линейного квантования составляют 16 битов или 24 бита. При этом цифровой поток продискретизированного сигнала может изменяться от $44,1 \times 16 = 705,6$ кбитов/с до $96 \times 24 = 2304$ кбитов/с. Разумеется, с увеличением исходного цифрового потока качество воспроизводимого звучания будет повышаться, однако зачастую использование значительного цифрового потока нерационально, и современные кодеки нередко динамически изменяют величину цифрового потока в зависимости от сложности и насыщенности участка композиции.

Для существенного сокращения избыточности аудиоинформации применяются различные методы линейной и нелинейной обработки звуковых сигналов, которые приводят к сжатию с потерями, уменьшающими размер кодированной последовательности файлов по сравнению с оригинальным за счет удаления информации, невоспринимаемой человеком. Технология сжатия с потерями недостаток человеческого слуха превращает в преимущество, отбрасывая «ненужную» информацию. Компромисс между малым цифровым потоком и качеством воспроизводимого аудиосигнала достигается путем изменения количества отбрасываемой информации.

Дополнительные проблемы в кодировании аудиоинформации возникают при обработке различных форматов — от стереофонического сигнала до объемного многоканального (5,1, 7,1 и др.) звукового сигнала. Естественно, что в этих случаях для значительного сокращения избыточности многоканальных сигналов используются существенные корреляционные связи между ними.

Цифровая технология обработки звуковой информации находит все большее количество применений — от звукового сопровождения различных видов видеоинформации и радиовещания до компьютерной записи/воспроизведения и музыкальных компакт-дисков. Хотя цифровая технология позволяет точно воспроизвести закодированную звуковую информацию, специалисты-акустики считают, что такому звуку не хватает характерной колоритности, которой аналоговый звук обязан исключительно аналоговой схемотехнике. Правильно настроенное профессиональное оборудование теоретически способно обеспечить более широкий

частотный диапазон звукозаписи, чем стандартные записи, например на музыкальных компакт-дисках. Но цифровая система обработки звуковой информации лучше защищена от шумов и помех, чем аналоговая. С каждым перекопированием аналоговые звукозаписи деградируют по качеству, а в цифровых системах такой проблемы не возникает.

Цифровой звук обладает существенными достоинствами: разработано множество систем кодирования, записи и передачи звуковой информации, которые можно реализовать только на базе цифровой технологии.

ГЛАВА 15

ОСНОВНЫЕ ХАРАКТЕРИСТИКИ И ЦИФРОВОЕ ПРЕДСТАВЛЕНИЕ ЗВУКОВОЙ ИНФОРМАЦИИ

Аналоговое представление звуковых сигналов основано на подобии форм и основных характеристик соответствующих им электрических сигналов [6.1–6.8]. В терминах теории информации количество информации в таком электрическом сигнале в точности равно количеству информации в сигнале исходном, и электрическое представление не содержит избыточности, которая могла бы защитить переносимый сигнал от искажений при хранении, передаче и усилении.

Прежде чем перейти к анализу характеристик и параметров электрических эквивалентов звуковых сигналов, необходимо дать несколько определений, присущих распространению звуковых волн в пространстве.

К наиболее важным понятиям относятся:

- **интерференция** — усиление колебаний звука в одних точках пространства и ослабление колебаний в других точках в результате наложения двух или нескольких звуковых волн;
- **рефракция** — преломление, изменение направления движения звуковой волны от границы раздела с иной средой, поглощение или переход в другую среду;
- **реверберация** — отражения звуковых колебаний в замкнутом пространстве, вызывающие специфический гул, изменяя тембральную окраску, насыщенность, глубину воспринимаемого звука;
- **дифракция** — способность звуковых волн огибать препятствия;
- **эхо** — возникновение сдвинутых во времени и различаемых раздельно повторов кратковременных звуковых колебаний;
- **эффект резонанса** — способность звуковой волны, создаваемой некоторым колеблющимся телом, переносить энергию колебаний другому телу, которое, поглощая эту энергию, само становится источником звука;
- **индекс диффузности** — фактор изотропности и однородности звукового поля;
- **звуковое давление** — характеристика громкости звука, непосредственно воспринимаемого ухом человека;
- **эффект Допплера** — изменение длины звуковых волн при изменении скорости движения слушателя относительно источника звука.

Эти и другие явления используются в различных приложениях, например в акустике, обработке звуковой информации, радиолокации и др. Задачей преобразования звука в электронный вид является повторение всех его волновых характеристик.

15.1. Статистика мгновенных значений и уровней звуковых сигналов

Звуковой сигнал или его электрический эквивалент $u(t)$ (рис. 15.1 и 15.2) обычно считают случайным процессом с распределением мгновенных значений, которое характеризуется некоторой плотностью вероятностей $W(u)$.

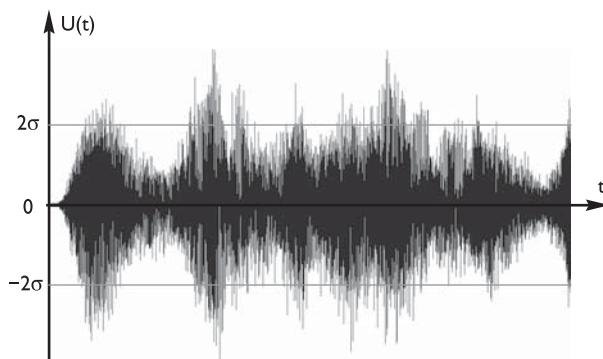


Рис. 15.1. Сигналограмма фрагмента музыкальной аудиозаписи

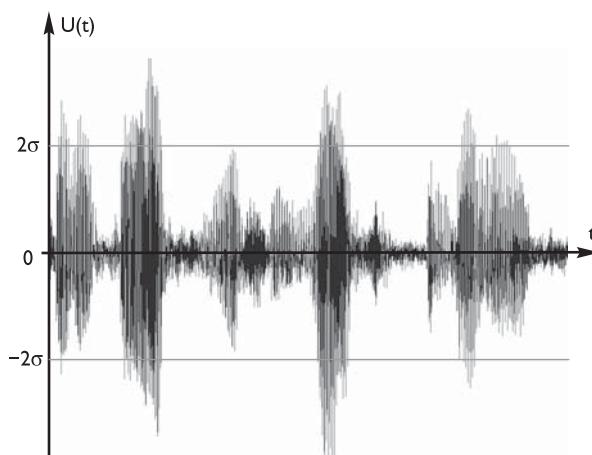


Рис. 15.2. Сигналограмма фрагмента аудиозаписи речи

На рис. 15.3 приведены графики относительных плотностей вероятности мгновенных значений сигналов, соответствующих наиболее распространенных звуковых программ [6.5].

При этом воспроизведение звучания больших симфонических оркестров, хоров, оркестров с хором соответствует нормальному распределению плотности вероятностей:

$$W(u) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{u^2}{2\sigma^2}\right); \quad (15.1)$$

звукование отдельных музыкальных инструментов, легкой музыки — распределению Лапласа:

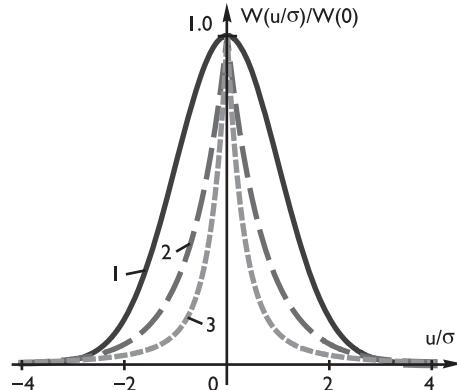
$$W(u) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{|u|\sqrt{2}}{\sigma}\right), \quad (15.2)$$

а речи — усредненному эмпирическому закону:

$$W(u) = \frac{1}{\sigma} \left[\exp \left(-\frac{3,31|u|}{\sigma} \right) + 0,18 \exp \left(-\frac{0,91|u|}{\sigma} \right) \right]. \quad (15.3)$$

В соотношениях (15.1)–(15.3) величина σ соответствует среднеквадратичному значению сигналов.

Рис. 15.3. Относительные плотности вероятностей мгновенных значений звуковых сигналов $W(u/\sigma)/W(0)$: 1 — симфонический оркестр, 2 — легкая музыка, 3 — речь



Уровень электрического эквивалента звукового сигнала обычно характеризуют напряжением, формируемым на выходе квазипикового детектора с малой величиной постоянной времени заряда (порядка 5–10 мс) и значительной величиной τ постоянной времени разряда (1–2 с) [6.9]. Отношение усредненной величины выпрямленных мгновенных значений сигнала $U(t, \tau)$ на выходе квазипикового детектора (или выделяемой на нагрузке мощности $P(t, \tau)$ сигнала) к некоторой условной величине U_0 (или P_0) определяют формулой:

$$N(t, \tau) = 20 \lg \frac{U(t, \tau)}{U_0} = 10 \log \frac{P(t, \tau)}{P_0} \text{ дБ}, \quad (15.4)$$

где U_0 — принятая за начало отсчета среднеквадратическая величина электрического сигнала с эффективным напряжением 0,775 В на нагрузке 600 Ом (выделяемая мощность $P_0 = 1$ мВт); выраженная в децибелях, эта величина определяет значение уровня, равное 0 дБ.

Временная зависимость величины $N(t, \tau)$ — уровнеграмма сигнала $u(t)$ — в качестве примера приведена на рис. 15.4. Она характеризует изменение эффективного напряжения электрического эквивалента звукового сигнала.

Закон распределения уровней $W(N)$ при больших значениях постоянной времени τ практически мало зависит от характера звуковой программы и чаще всего соответствует нормальному распределению.

Интенсивность звука или звуковое давление оценивают либо в Паскалях, либо в децибелах относительно некоторого порога, величина которого принята равной $p_0 = 2 \cdot 10^{-5}$ Па = 20 мкПа и соответствует порогу слышимости здорового молодого человека в диапазоне звуковых частот 1–4 кГц.

Для характеристики уровней звукового давления (SPL — Sound pressure level) используется уравнение:

$$SPL = 20 \lg \frac{P_k}{P_0}, \quad (15.5)$$

где p_0 и p_k — звуковое давление, выраженное, например, в Паскалях.

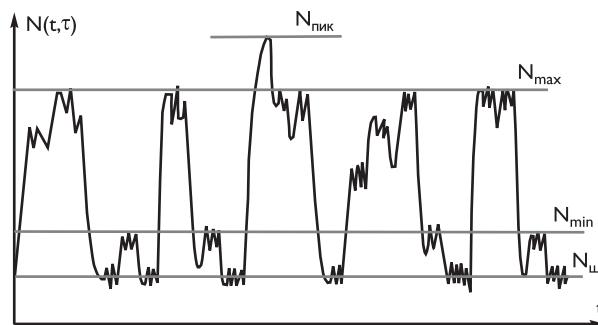


Рис. 15.4. Пример уровнеграммы звукового сигнала: $N_{\max} - N_{\min}$ — диапазон квазимаксимального изменения уровня звукового сигнала; $N_{ш}$ и $N_{\text{пик}}$ — уровень шума и случайный выброс квазипикового уровня в звуковом сигнале соответственно

Приблизительные уровни звукового давления, характерные для различных источников, приведены в табл. 15.1 [6.10–6.12].

Таблица 15.1. Уровни звукового давления различных источников

Источник звука	Звуковое давление, Па	Уровень звукового давления, дБ
Слуховой порог	$2 \cdot 10^{-5}$	0
Шелест листвы, слабого ветра, спокойное дыхание, тиканье наручных часов	$6 \cdot 10^{-5} - 6 \cdot 10^{-4}$	10–30
Шум в помещении, тихий разговор, скрипка пианистико, тихая музыка	$6 \cdot 10^{-4} - 6 \cdot 10^{-3}$	30–50
Разговор средней громкости, звук телевизора, шум в магазине, шум в ресторане	$6 \cdot 10^{-3} - 2 \cdot 10^{-2}$	40–60
Громкий разговор, легковой автомобиль, шумная улица, шум в машинописном бюро	$2 \cdot 10^{-2} - 2 \cdot 10^{-1}$	60–80
Грузовой автомобиль, движение на занятом шоссе, шум в метро во время движения, авторемонтная мастерская	$2 \cdot 10^{-1} - 6 \cdot 10^{-1}$	80–90
Женское и мужское пение	$6 \cdot 10^{-3} - 6 \cdot 10^{-1}$	40–90
Эстрадный и симфонический оркестры	$2 \cdot 10^{-2} - 6$	40–110
Громкая дискотека, пневматический молот, самолет на взлете, кузнецкий цех	2–20	100–120
Реактивный самолет с расстояния 100 м	6–200	110–140
Порог болевого ощущения	60	130

Принцип преобразования акустической энергии в электрическую, обработки электрического эквивалента и его последующего преобразования вновь в акустическую энергию поясняется на модели простейшей звуковой системы, приведенной на рис. 15.5 [6.11]. Источник звукового сигнала формирует акустическую энергию, которая преобразуется в колебания электрического аудиосигнала. После электрической обработки сигнала, включающей, например, усиление, формирование спецэффектов, изменение динамического диапазона и др., аудиосигнал



Рис. 15.5. Модель простейшей звуковой системы

подвергается обратному преобразованию в акустическую энергию и ее воспроизведению.

Уровни электрического эквивалента звуковой системы (N , дБ) пропорциональны уровням звукового давления (SPL , дБ). Так, при уровне $SPL = 120$ дБ звукового давления на входе преобразователя звука в электрический эквивалент максимальный уровень электрического сигнала может достигать величины $N = +25$ дБ. Учитывая линейную зависимость между величинами SPL и N (если в системе не применяется компрессия, эквалайзация, ограничение или отсечка сигналов), соответствие этих величин может быть представлено в виде, изображенном на рис. 15.6. Такое соотношение характерно для вещательных систем любого типа, звукоусиления, звукозаписи.

Требования к динамическому диапазону звуковой системы зависят от ее назначения и области использования. Так, динамический диапазон системы в звуконепроницаемой студии звукозаписи может быть большим, поскольку в таком случае шумы в ней не превышают 10–15 дБ. Звуковая система, предназначенная для усиления симфонической музыки, должна иметь запас динамического диапазона более 20 дБ, так как пиковые значения, соответствующие звукам некоторых инструментов (литавр, скрипок и др.), могут достигать $SPL = -120$ дБ.

В системах, предназначенных только для воспроизведения речи или предупреждающих сигналов, уровень звука можно контролировать и удерживать в очень узком диапазоне.

Речевой сигнал можно рассматривать как последовательность импульсов, разделенных паузами, при которых уровень сигнала ниже некоторого минимального уровня N_{min} (рис. 15.7) [6.5].

Установлено, что среднесекундное число звуковых импульсов или пауз в речи дикторов телевизионного вещания — практически стабильная величина, приблизительно равная 2с^{-1} , а потоки переходов от импульсов к паузам или от пауз к импульсам обладают плотностью вероятности и функцией ее распределения, соответствующими соотношениям:

$$W(t_k) = 2 \exp(-2t_k), \quad F(t_k) = 1 - \exp(-2t_k), \quad (15.6)$$

где t_k — общее обозначение величин t_i и t_{np} . Математическое ожидание, средне-

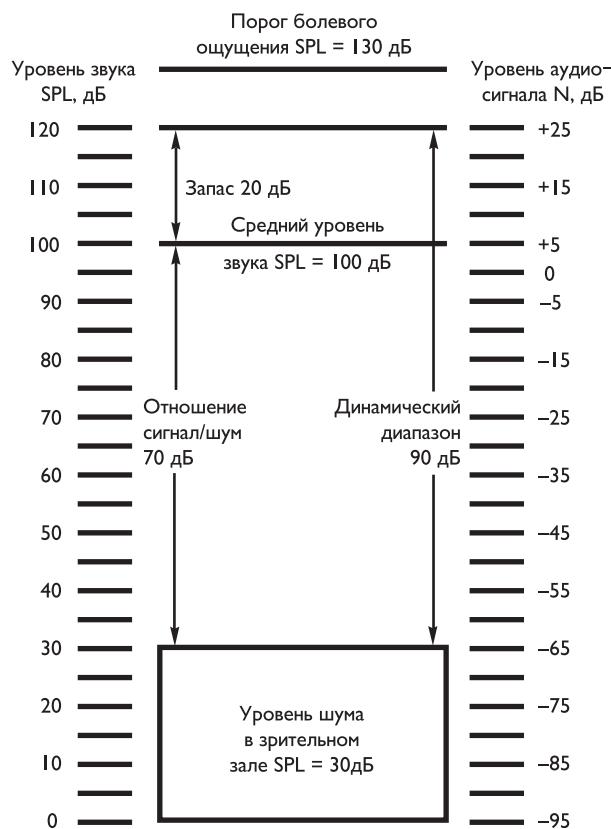


Рис. 15.6. Динамический диапазон звукового давления и его электрического эквивалента

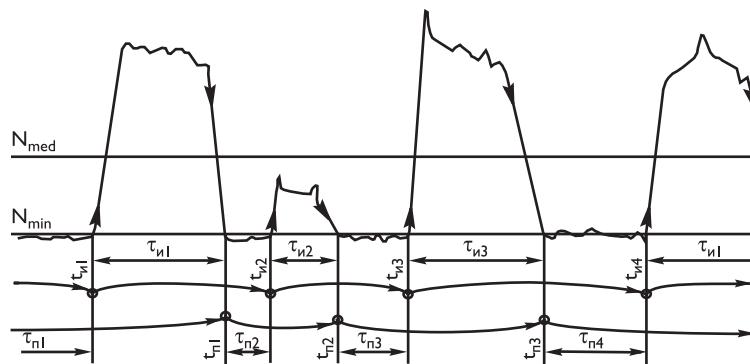


Рис. 15.7. Участок речевого сигнала: $\tau_{i1}, \tau_{i2}, \dots$ — интервалы последовательности импульсов; $\tau_{\text{пп}1}, \tau_{\text{пп}2}, \dots$ — интервалы последовательности пауз; t_{i1}, t_{i2}, \dots — времена переходов от пауз к импульсам; $t_{\text{пп}1}, t_{\text{пп}2}, \dots$ — времена переходов от импульсов к паузам.

квадратическое отклонение и медиана этого распределения соответственно равны

$$m_k = \sigma_k = 0,5 \text{ с}; \quad t_k(0,5) = 0,35 \text{ с}.$$

Считается, что при $(N_{\text{med}} - N_{\text{min}}) \gg \sigma_N$, где N_{med} — средний уровень сигнала, а σ_N — среднеквадратичное отклонение уровней, потоки переходов от импульсов к паузам или от пауз к импульсам почти пуассоновские и вероятность того, что за интервал T наступит хотя бы одно событие (например, пауза), равна $p(T) = 1 - \exp(-\lambda T)$, где $\lambda = 2 \text{ с}^{-1}$.

Отсюда находим, что с вероятностью $p = 0,5$ пауза встречается в отрывках дикторской речи продолжительностью $T = 0,35$ с и вероятностью $p = 0,95$ — в отрывках с длительностью $T = 1,5$ с.

Длительность звуковых импульсов речи $\tau_{\text{и}}$ (в секундах) подчиняется экспоненциальному закону распределения

$$W(\tau_{\text{и}}) = 3,5 \exp(-3,5\tau_{\text{и}}), \quad F(\tau_{\text{и}}) = 1 - \exp(-3,5\tau_{\text{и}}) \quad (15.7)$$

с параметрами $m_{\text{и}} = \sigma_{\text{и}} = 0,286$ с; $\tau_{\text{и}}(0,5) = 0,2$ с.

Длительность пауз $\tau_{\text{пп}}$ (в миллисекундах) приближенно подчиняется степенному закону распределения

$$W(\tau_{\text{пп}}) = 168,5 \cdot \tau_{\text{пп}}^{-2,23}, \quad F(\tau_{\text{пп}}) = 1,137 \cdot \tau_{\text{пп}}^{-1,23}, \quad \tau_{\text{пп}} \geq 54 \text{ мс.} \quad (15.8)$$

Первый момент и медиана этого распределения соответственно равны

$$m_{\text{пп}} = 293 \text{ мс}; \tau_{\text{пп}}(0,5) = 90 \text{ мс},$$

а остальные моменты не существуют.

15.2. Спектральные характеристики звуковых сигналов

Реальные звуковые сигналы практически невозможно описать какой-либо математической функцией или эмпирической зависимостью. По этой причине, как правило, анализируются лишь ограниченные во времени фрагменты звукового сигнала, выделенные некоторой оконной функцией. При этом используется понятие «мгновенный спектр»:

$$F(f, \tau) = \int_{-\tau/2}^{\tau/2} W(t) \cdot U(t) \cdot \exp(-j2\pi ft) dt, \quad (15.9)$$

где $u(t)$ — электрический эквивалент звукового сигнала, $w(t)$ — оконная функция, равная нулю вне пределов заданного интервала, $-\tau/2 \leq t \leq \tau/2$, f — текущая частота.

Часто оценивают спектральную плотность мощности аудиосигнала с применением относительно узкополосного фильтра с полосой пропускания Δf :

$$G(f_0, \tau) = \frac{2}{\Delta f \cdot \tau} \int_{f_0 - \Delta f/2}^{f_0 + \Delta f/2} |F(f, \tau)| df. \quad (15.10)$$

Результат спектрального анализа выражают в децибелах

$$N(f_0, \tau) = 10 \lg G(f_0, \tau) G_0 \text{ дБ,} \quad (15.11)$$

где G_0 — значение, соответствующее нулю шкалы уровней.

Для расчета частотного спектра ограниченного по длительности сигнала, представленного его дискретными значениями, используют дискретное преобразование Фурье (ДПФ) или его разновидность — быстрое преобразование Фурье (БПФ).

В основе ДПФ лежит идея представления анализируемого фрагмента сигнала в виде одного периода некоторого бесконечного периодического сигнала, спектр которого и подвергается анализу.

На рис. 15.8 приведены сигналограмма записи отрывка синтезированного музыкального фрагмента и график его спектра, рассчитанный с использованием ДПФ.

С точки зрения обработки электрического эквивалента звука и его передачи весьма полезной является информация о спектрах разговорных и певческих голосов, музыкальных инструментов, природных и индустриальных шумов и т. д. Сведения о частотных диапазонах некоторых источников звука даны в табл. 15.2 [6.3, 6.5].

Таблица 15.2. Частотные диапазоны некоторых источников звука

Источник звука	Границчная частота, Гц	
	нижняя	верхняя
Мужской голос	100	7000
Женский голос	200	9000
Рояль	100	5000
Скрипка	200	14000
Флейта	250	14000
Тарелки	400	12000
Литавры	65	3000
Шум шагов	100	10000
Аплодисменты	150	15000

Специфической особенностью восприятия звука является разделение его спектра на полосы равной разборчивости (критические полосы), когда сигнал в пределах одной и той же полосы создает весьма близкие слуховые ощущения. В частотном промежутке от 0 до 16 кГц опытным путем определены 24 полосы, оцениваемые в Барках (по фамилии немецкого ученого Баркгаузена Г.Г., 1881–1956). В табл. 15.3 приведены критические полосы и соответствующие им частотные разбиения.

Восприятие громкости чистых тонов различных звуковых частот также существенно различается. На экспериментальной основе построена известная шкала зависимостей громкости звука — так называемые кривые равной громкости (рис. 15.9).

Каждая кривая на этом графике характеризует уровень равной громкости с начальной точкой отсчета на частоте 1000 Гц. Одна из единиц определения уровня громкости «фон» (phon) характеризует относительную субъективную оценку интенсивности звука. Так, линия «10 фон» на этом графике определяет уровни сигнала в дБ на разных частотах, воспринимаемых слуховой системой человека как равные громкости с сигналом частоты 1000 Гц и уровнем 10 дБ.

Области громкости и спектральных областей речи и музыки различны, как показано на диаграмме рис. 15.10. Следует заметить, что приведенные графики определены для случая полной тишины. В случае иных условий, например в шумной комнате, диаграммы окажутся другими.

Важными факторами изменения восприятия звуковой информации являются эффекты частотной и временной маскировки.

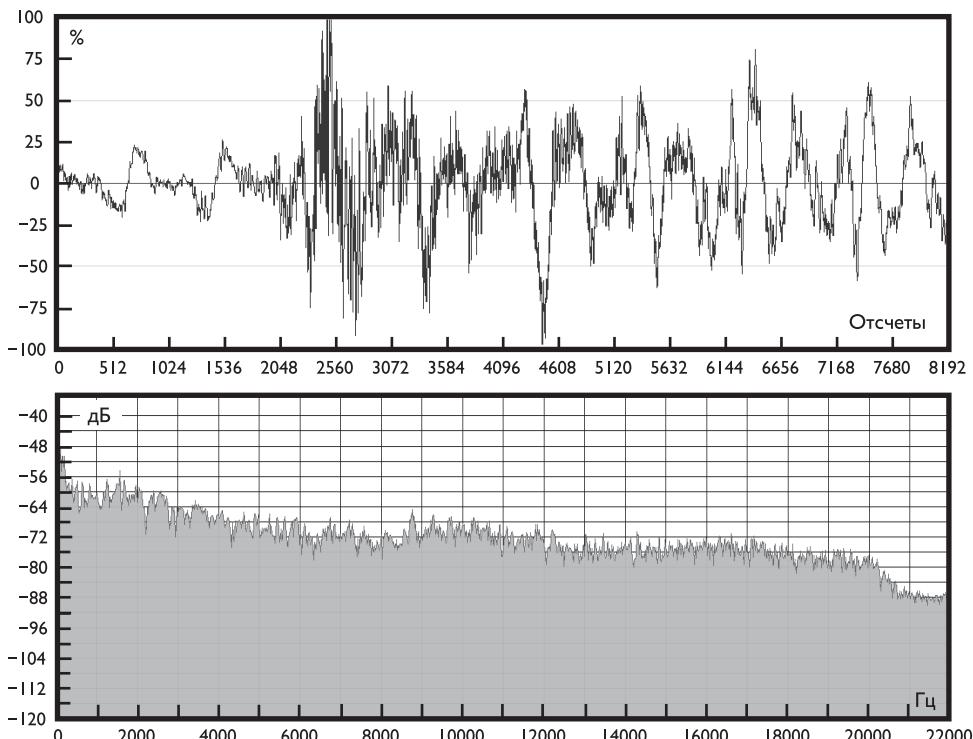


Рис. 15.8. Участок записи звукового сигнала (а) и его спектр (б)

На рис. 15.11 схематично показано, как тон одной частоты f_0 изменяет кривую порога слышимости в тишине и создает маскирующий эффект, при котором тон частоты f_m оказывается замаскированным и не воспринимаемым. Как видно из этого графика, составляющая частоты f_m имеет уровень, превышающий порог слышимости в тишине (штриховая линия) и была бы воспринята при отсутствии тона частоты f_0 . Если в качестве фонового маскирующего звука выступает шумовой сигнал с определенным спектром, то кривая порога слышимости оказывается накрытой маскирующим порогом, величина которого зависит от уровней составляющих спектра шума.

В профессиональной литературе рассматриваются несколько различных по спектру разновидностей шумов [6.2, 6.12].

- Белый шум обладает постоянной спектральной плотностью на всей протяженности спектра.
- Розовый шум, спектральная плотность которого уменьшается на 3 дБ с каждой последующей октавой.
- Оранжевый шум, спектральная плотность которого квазипостоянна и имеет полоски нулевой энергии, рассеянные на всей протяженности спектра. Такие полоски располагаются около частот, соответствующих музыкальным нотам.
- Зеленый шум подобен розовому шуму с усиленной областью в районе 500 Гц.

Таблица 15.3. Критические частотные полосы звуковых сигналов

Номер полосы, Барк	Диапазон критической полосы, Гц	Ширина критической полосы, Гц	Центральная частота критической полосы, Гц
0	0–100	100	50
1	100–200	100	150
2	200–300	100	250
3	300–400	100	350
4	400–510	110	450
5	510–630	120	570
6	630–770	140	700
7	770–920	150	840
8	920–1080	160	1000
9	1080–1270	190	1170
10	1270–1480	210	1370
11	1480–1720	240	1600
12	1720–2000	280	1850
13	2000–2310	320	2150
14	2310–2700	380	2500
15	2700–3150	450	2900
16	3150–3700	550	3400
17	3700–4400	700	4000
18	4400–5300	900	4800
19	5300–6400	1100	5800
20	6400–7700	1300	7000
21	7700–9500	1800	8500
22	9500–12000	2500	10500
23	12000–15500	3500	13500

- Синий шум, спектральная плотность которого увеличивается на 3 дБ с каждой последующей октавой.
- Фиолетовый шум или дифференцированный белый шум, спектральная плотность которого увеличивается на 6 дБ с каждой последующей октавой.
- Серый шум, спектр имеет форму, аналогичную графику психоакустической кривой порога слышимости.
- Коричневый шум, спектральная плотность которого уменьшается на 6 дБ с каждой последующей октавой.
- Тональный шум, в спектре которого имеются слышимые дискретные тоны.
- Черный шум имеет постоянную конечную спектральную плотность за пределами частотного порога слышимости, равного 20 кГц.

Эффект частотной маскировки справедлив для частотных составляющих, присутствующих в спектре сигнала в одно и то же время. Однако за счет инерционности слуха эффект маскировки распространяется и во временной области, как показано на рис. 15.12. В случае, когда маскирующий тон прекращается раньше маскируемого тона, создается постмаскирующий эффект. Возможна и пред-

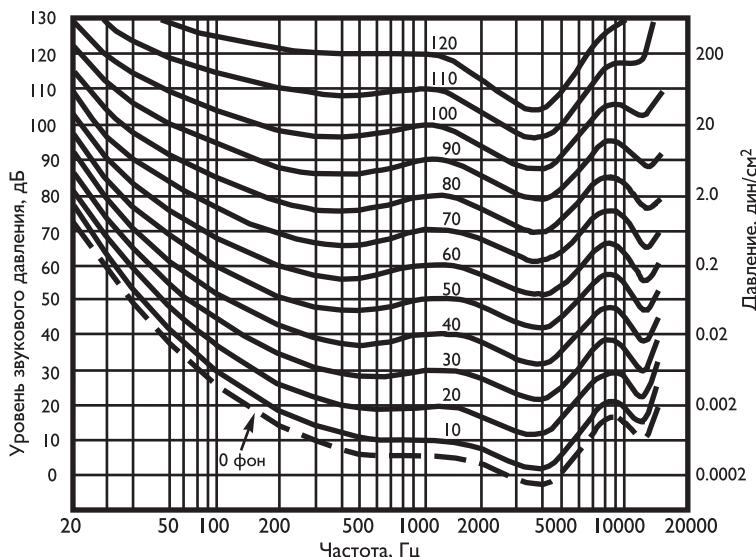


Рис. 15.9. Зависимости равной громкости

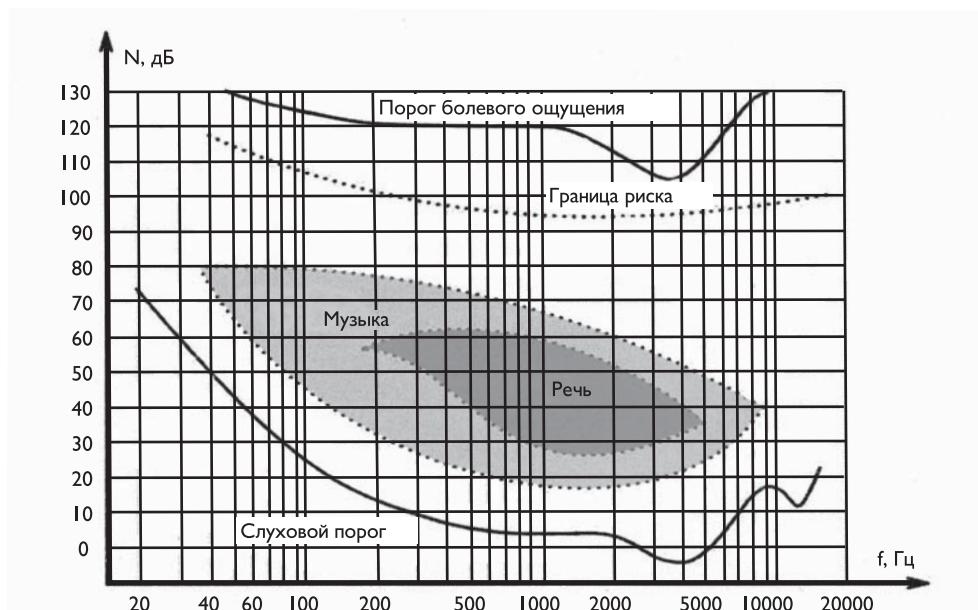


Рис. 15.10. Диаграммы порогов и области слышимости для случая полной тишины

маскировка, когда маскирующий тон появляется несколько позже маскируемого тона. Из диаграммы видно, что интервал предмаскировки существенно меньше интервала постмаскировки.

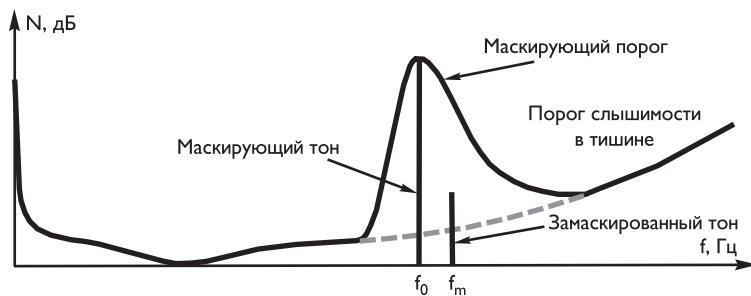


Рис. 15.11. Эффект частотной маскировки

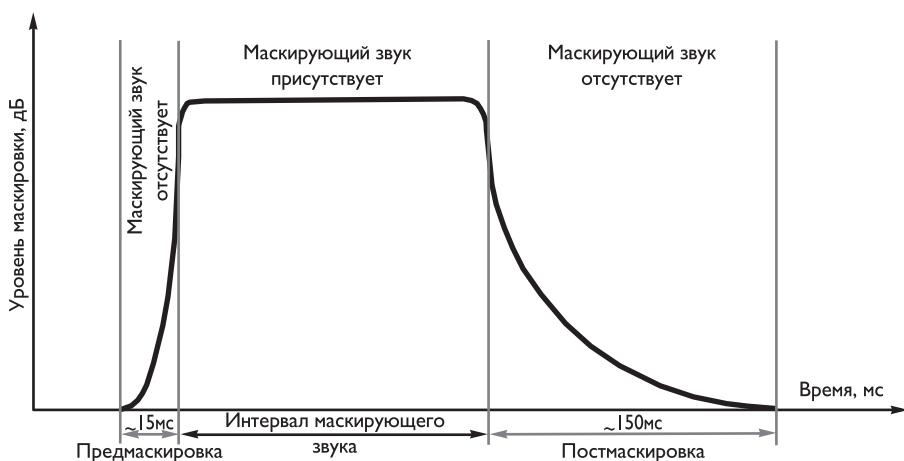


Рис. 15.12. Диаграмма частотно-временной маскировки

Следует заметить, что рассмотренные эффекты маскировки и изменения порогов слышимости широко используются в самых различных технологиях цифрового сжатия аудиоинформации.

15.3. Огибающая и мгновенная частота звуковых сигналов

По форме огибающей и изменению мгновенной частоты звуковых сигналов производится анализ переходных процессов в преобразователях акустической энергии в ее электрический эквивалент при обработке и восстановлении аудиосигналов.

Оценка этих характеристик звуковых сигналов осуществляется с использованием двух сигналов: исходного $u(t)$ и сопряженного с ним по Гильберту $u_G(t)$:

$$u_G(t) = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{u(\tau)}{t - \tau} d\tau; \quad u(t) = -\frac{1}{\pi} \int_{-\infty}^{\infty} \frac{u_G(\tau)}{t - \tau} d\tau. \quad (15.12)$$

Пусть функция $u(t)$ ограничена по спектру частотой $f_{\text{нгр}}$ и определена дискретными отсчетами $u(nT)$, $0 \leq n \leq N - 1$. Положим также, что интервал между отсчетами в соответствии с теоремой Котельникова определяется соотношением $Tt_{n+1} - t_n \leq 1/2 \cdot f_{\text{нгр}}$. Тогда форму сигнала и его спектр можно представить в виде:

$$u(t) = \sum_{n=0}^{N-1} u(nT) \cdot \frac{\sin v(t, n)}{v(t, n)}, \quad F_u(f) = \Psi(f) \cdot \sum_{n=0}^{N-1} u(nT) \cdot \exp[-2\pi j \cdot Tfn], \quad (15.13)$$

где $v(t, n) = \frac{\pi}{T}(t - n \cdot T)$; $|t| < \infty$; $\Psi(f) = \begin{cases} 1, & 0 \leq f \leq f_{\text{гр}}; \\ 1/2, & f = f_{\text{гр}}; \\ 0, & f > f_{\text{гр}}. \end{cases}$

Форма преобразованного по Гильберту сигнала и его спектра в данном случае определяются следующими соотношениями:

$$u_{\Gamma}(t) = \sum_{n=0}^{N-1} u(nT) \cdot \frac{1 - \cos v(t, n)}{v(t, n)}, \quad F_{u_{\Gamma}}(f) = -jF_u(f). \quad (15.14)$$

Огибающая и изменение фазы звукового сигнала рассчитываются по формулам:

$$U_{\text{ор}}(t) = \sqrt{u^2(t) + u_{\Gamma}^2(t)}, \quad \varphi(t) = \text{Arctg} \frac{u_{\Gamma}(t)}{u(t)}, \quad (15.15)$$

а мгновенная частота определяется производной фазы

$$f_{\text{мгн}}(t) = \frac{1}{2\pi} \frac{d\varphi(t)}{dt} = \frac{1}{2\pi} \cdot \frac{u(t) \cdot u'_{\Gamma}(t) - u'(t) \cdot u_{\Gamma}(t)}{u^2(t) + u_{\Gamma}^2(t)}, \quad (15.16)$$

где $u'(t) = \frac{du(t)}{dt}$, $u'_{\Gamma}(t) = \frac{du_{\Gamma}(t)}{dt}$.

Производные исходного и сопряженного по Гильберту сигналов и их спектры определяются следующими формулами:

$$u'(t) = \sum_{n=0}^{N-1} u(nT) \cdot \left[\frac{\cos v(t, n)}{v(t, n)} - \frac{\sin v(t, n)}{v^2(t, n)} \right], \quad F'_u(f) = 2\pi j \cdot fT \cdot F_u(f); \quad (15.17)$$

$$u'_{\Gamma}(t) = \sum_{n=0}^{N-1} u(nT) \cdot \left[\frac{\sin v(t, n)}{v(t, n)} - \frac{1 - \cos v(t, n)}{v^2(t, n)} \right], \quad F'_{u_{\Gamma}}(f) = -2\pi \cdot fT \cdot F_u(f). \quad (15.18)$$

Приведенные в соотношениях (15.13)–(15.18) преобразования сигналов и их спектров можно использовать для расчетов огибающей и мгновенной частоты звукового сигнала с применением быстрых прямого и обратного преобразований Фурье (БПФ и ОБПФ) как показано на рис. 15.13. В данном случае исходно с помощью БПФ формируются спектральные отсчеты $F_u(m)$ дискретного сигнала $u(n)$. Затем спектральные отсчеты производной исходного сигнала, Гильбертовой составляющей и ее производной формируются путем перемножения $F_u(m)$ на соответствующие коэффициенты. Формирование всех необходимых сигналов осуществляется с использованием ОБПФ. Огибающая и мгновенная частота звукового сигнала вычисляются по формулам (15.15) и (15.16).

На некотором интервале T ограниченный по спектру, не имеющий постоянной составляющей сигнал может быть представлен, например, набором косинусо-

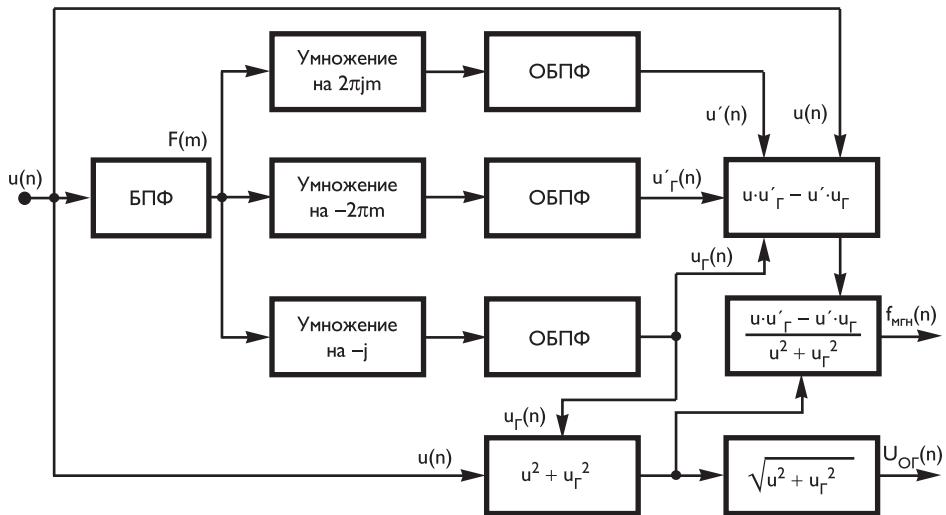


Рис. 15.13. Схема расчета огибающей и мгновенной частоты звукового сигнала с использованием БПФ и ОБПФ

и идеальных составляющих:

$$u(t) = \sum_{n=1}^N a_n \cos \left(2\pi n \frac{t}{T} + \varphi_n \right), \quad (15.19)$$

где граничная частота равна $f_{\text{гр}} = N/T$, а сопряженная по Гильберту составляющая равна

$$u_\Gamma(t) = \sum_{n=1}^N a_n \sin \left(2\pi n \frac{t}{T} + \varphi_n \right). \quad (15.20)$$

На рис. 15.14а в качестве примера приведена диаграмма обозначенного синим цветом сигнала, состоящего из пяти косинусоидальных колебаний различных частот с различными начальными фазами и имитирующего небольшой отрезок речи. На этом же рисунке красным цветом изображена сопряженная по Гильберту составляющая этого сигнала. На рис. 15.14б приведены диаграммы огибающей этого сигнала и изменения мгновенной частоты, рассчитанные по формулам (15.15) и (15.16). Следует заметить, что в случае если уровень огибающей сигнала близок к нулю, то изменение мгновенной частоты приобретает скачок значительной величины, что практически не может опущаться на слух. По этой причине при расчете мгновенной частоты квадрат огибающей ограничивается некоторой величиной в соответствии с соотношением:

$$U_{\text{пог}}^2(t) = \begin{cases} u^2(t) + u_\Gamma^2(t), & \text{если } U_{\text{пог}}^2(t) > v, \\ v, & \text{если } U_{\text{пог}}^2(t) \leq v, \end{cases} \quad (15.21)$$

где v — уровень ограничения. В данном случае $v = 0,02$.

Из рис. 15.14б видно, что большие выбросы мгновенной частоты и ее отрицательные значения возникают в минимумах огибающей звукового сигнала.

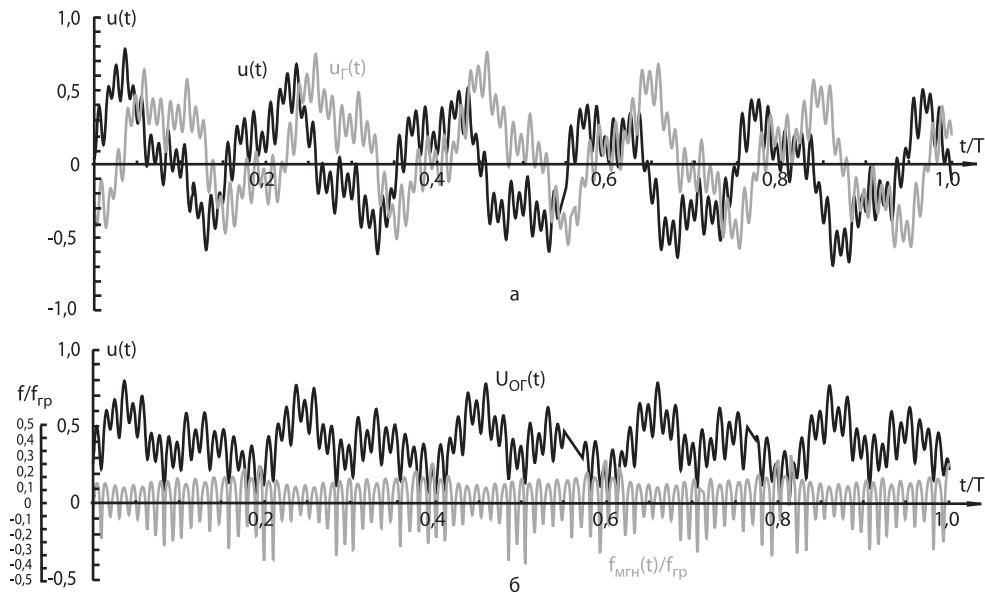


Рис. 15.14. Диаграммы звукового сигнала и его гильбертовой составляющей (а), огибающая и мгновенная частота звукового сигнала (б)

15.4. Пространственное восприятие звуковых сигналов

Локационные способности восприятия звука, так называемый **бинауральный эффект** [6.2, 6.13], объясняются фазовым смещением звуковых волн, неодинаковым уровнем звуковых давлений в ушах, особенностями тембров знакомых источников звуков и их изменений. При неподвижном источнике звука слух способен определить направление движения звуковых волн по горизонтали не точнее 12 градусов, а по вертикали — 17...20 градусов. Бинауральный эффект практически отсутствует на частотах ниже 300 Гц, на частотах от 300 до 1000 Гц становится заметным сдвиг фаз звуковых волн, попадающих в правое и левое ухо. При частотах более 1000 Гц сдвиг фаз становится очень небольшим и поиск направления осуществляется за счет сравнения силы звука, приходящего с разных сторон.

На рис. 15.15а поясняется принцип интегральной локализации восприятия информации от двух источников звука. Два одинаковых источника (1 и 2) расположены на расстоянии $2L_y$ один от другого, а на расстоянии L_x расположен слушатель, уши которого находятся на расстояниях r_1 и r_2 от соответствующих источников.

Если на оба источника излучают звуковую энергию одинаковой мощности, то звук от каждого источника достигнет ушей одновременно, и идентичность звуков не позволит слуху разделить их в пространстве, например, на левый и правый. При этом возникает иллюзия: виртуальный (кажущийся) источник звука как бы находится в середине между источниками звука.

Если уменьшить мощность излучения одного из источников, то это воспринимается как перемещение кажущегося источника в сторону второго источника.

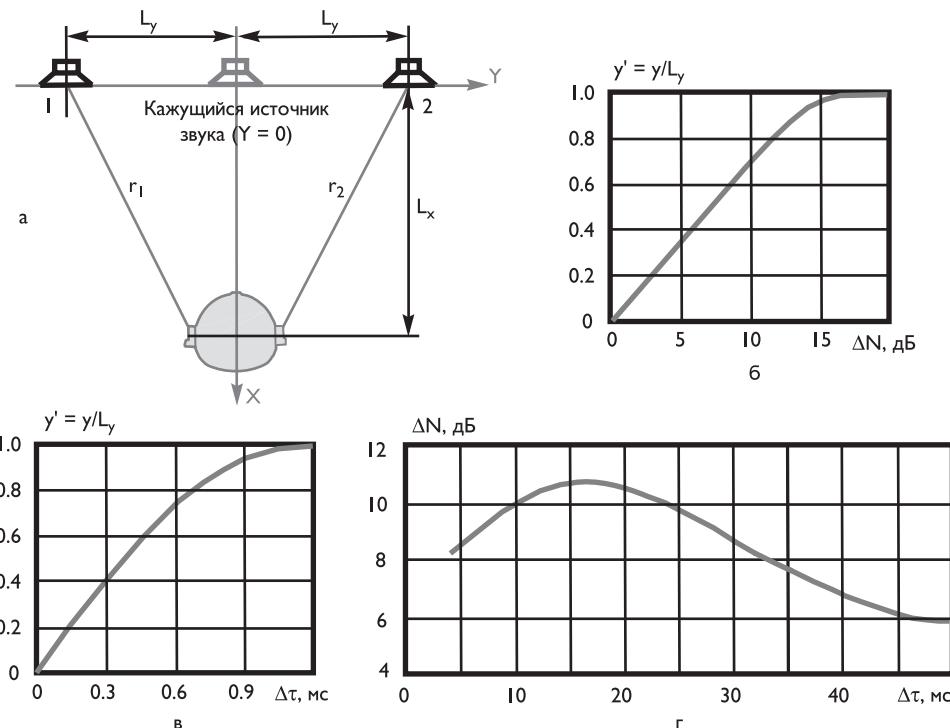


Рис. 15.15. Пространственное восприятие звукового сигнала от двух источников: а) схема создания эффекта интегральной локализации; б) зависимость локализации от разных уровней звуковых сигналов; в) зависимость локализации от временно-го сдвига звуковых сигналов; г) влияние запаздывающего повторения сигнала на локализацию виртуального источника звука

Таким образом, варьируя громкость звучания левого и правого источника, можно вызывать и поддерживать иллюзию перемещения виртуального источника звука. Зависимость относительного перемещения виртуального источника от различия излучаемых мощностей двух источников приведена на рис. 15.15б.

Аналогичная иллюзия перемещения виртуального источника возникает, если создать запаздывание звука от одного из источников. На рис. 15.15в приведена зависимость перемещения виртуального источника от временного сдвига звуковых сигналов. Оба эти эффекта широко используются при цифровой обработке и записи музыки.

При задержках одного из сигналов на время более 50 мс наличие запаздывающего сигнала ощущается как помеха в виде эха. Опережающий сигнал при одинаковом уровне с задержанным в этом случае подавляет (маскирует) последний. Повысив уровень запаздывающего сигнала можно добиться того, что оба источника звука будут восприниматься раздельно даже при запаздывании менее 50 мс.

На рис. 15.15г показано необходимое превышение уровня ($\Delta N, \text{ дБ}$) запаздывающего сигнала в зависимости от временной задержки. При $\tau = 15 \dots 20$ мс уровень задержанного сигнала должен быть повышен на 11 дБ, чтобы оба ис-



точника звука воспринимались раздельно. При $\tau < 50$ мс для этого эффекта достаточно превышение уровня всего на 6 дБ. При $\tau < 5$ мс наблюдается неустойчивый режим: виртуальный источник звука как бы перепрыгивает из одного источника в другой, совпадая то с источником опережающего, то с источником задержанного сигнала.

Для качественного восприятия реального пространственного звучания музыкальных программ использование двухканальной (стереофонической) системы воспроизведения звуковых сигналов не всегда является достаточным. Основная причина этого кроется в том, что стереосигнал, приходящий к слушателю от двух физических источников звука, определяет расположение мнимых источников лишь в той плоскости, в которой расположены реальные физические источники звука. Поэтому в последние десятилетия стали развиваться системы многоканального воспроизведения звука, реализующие так называемый **трансуральный эффект**.

Для воссоздания более или менее реалистичного, действительно объемного звучания прибегают к применению сложных приемов, моделирующих особенности слуховой системы человека, а также физические особенности и эффекты передачи звуковых сигналов в пространстве. Главная проблема заключается в том, чтобы создать такой сигнал, который бы при помощи двух или более источников звука воспринимался слушателем трехмерным [6.14, 6.15].

Для описания 3D-звука применяются функции HRTF (Head Related Transfer Function) [6.16]. Принцип использования HRTF основан на анализе искажений объемного звука при восприятии человеком. В зависимости от расположения источника звук искажается по-разному. По получаемым параметрам звука и степени его искажения, человеческий мозг определяет положение источника. Таким образом, если заранее известна эта функция, а также направление и расстояние до источника, с помощью сложных вычислений можно смоделировать звук, распространяющийся из определенной точки пространства.

Из-за того, что расчет функций HRTF — сложная задача, во многих системах пространственного звука разработчики полагаются на использование данных, полученных экспериментальным путем. Тем не менее основной причиной использования HRTF является желание воспроизвести звук в вертикальной плоскости, наряду с азимутальными звуковыми эффектами. При этом восприятие звуковых сигналов, источники которых расположены в вертикальной плоскости, чрезвычайно чувствительно к особенностям каждого конкретного слушателя.

HRTF представляет собой набор трансформаций, которые претерпевает звуковой сигнал на пути от источника звука к слуховой системе человека. Рассчитанные однажды опытным путем функции HRTF могут быть применены для обработки звуковых сигналов с целью имитации реальных изменений звука на его пути от источника к слушателю. Несмотря на удачность идеи, HRTF имеет, конечно, и свои отрицательные стороны, однако в целом идея использования HRTF является вполне удачной. Использование HRTF в том или ином виде лежит в основе множества современных технологий пространственного звучания.

15.5. Основные принципы цифровых преобразований звуковых сигналов

Преобразование аналогового звукового сигнала путем временной дискретизации и квантования выбранных дискретных значений неизбежно приводит к невозможности его абсолютно точного восстановления.

Если предположить, что аналоговый сигнал строго ограничен по спектру, то в соответствии с теоремой Котельникова он полностью определяется дискретной последовательностью своих мгновенных значений, взятых с частотой выборки $f_{\text{пд}}$, как минимум вдвое превышающей граничную частоту спектра: $f_{\text{пд}} \geq 2f_{\text{пгр}}$.

Квантование дискретных отсчетов сигнала вызывает шумовые искажения, величина которых зависит от разрядности аналого-цифрового преобразователя (АЦП), типа преобразуемого сигнала и формы шкалы квантования.

Следует заметить, что при дискретизации сигнала с частотой $f_{\text{пд}} > 2f_{\text{пгр}}$ спектр шума квантования распространяется вплоть до частоты $f_{\text{пд}}/2$ и при восстановлении сигнала с помощью ФНЧ с граничной частотой, примерно равной $f_{\text{пгр}}$, уровень шума может быть уменьшен. При равномерной дискретизации сигнала с числом уровней квантования 2^N отношение сигнал/шум (динамический диапазон АЦП) оценивают в дБ по формуле:

$$S = 20N \lg 2 + 10 \lg \frac{f_{\text{пд}}}{2f_{\text{пгр}}} + C \cong 6N + 10 \lg \frac{f_{\text{пд}}}{2f_{\text{пгр}}} + C, \quad (15.22)$$

где C — константа, зависящая от формы преобразуемого сигнала.

При преобразовании синусоидального сигнала $C = 1,7$ дБ, для звуковых сигналов константа C изменяется от -15 дБ до 2 дБ.

Принято, что в звуковой аппаратуре максимально допустимый уровень сигнала соответствует 0 дБ. В связи с этим величина $-S$ характеризует уровень шумов квантования и одновременно минимально возможный уровень полезного сигнала. Как следует из соотношения (15.19), при $f_{\text{пд}} = 4f_{\text{пгр}}$ интенсивность шума уменьшится на 3 дБ.

На рис. 15.16а приведена диаграмма уровней сигнала при аналого-цифровом преобразовании. Очевидно, что отсчеты сигнала не должны превышать некоторого заданного максимально допустимого уровня. В противном случае квантователь «обрезает» сигнал, как показано на рис. 15.16б. Это явление называют клиппингом (от англ. clipping). На рис. 15.16б и иллюстрируется также формирование шума квантования. При цифровом преобразовании аналогового сигнала $u(t)$ приведенные на рис. 15.16в различия между его истинными значениями в точках дискретизации и цифровыми отсчетами и определяют шум квантования. Специфическим видом шума квантования является **гранулярный шум** (granular noise), проявляющийся вследствие нестабильности операции округления, когда уровень аудиосигнала незначительно изменяется и располагается приблизительно посередине между двумя ближайшими уровнями квантования j и $j + 1$.

В качестве примера предположим, что на протяжении нескольких дискретных отсчетов сигнал имеет вид $u(t) = j + \frac{1}{2} + 0,1 \cdot \sin \omega t$. Тогда в зависимости от знака синусоидальной составляющей этого сигнала значения дискретных значений

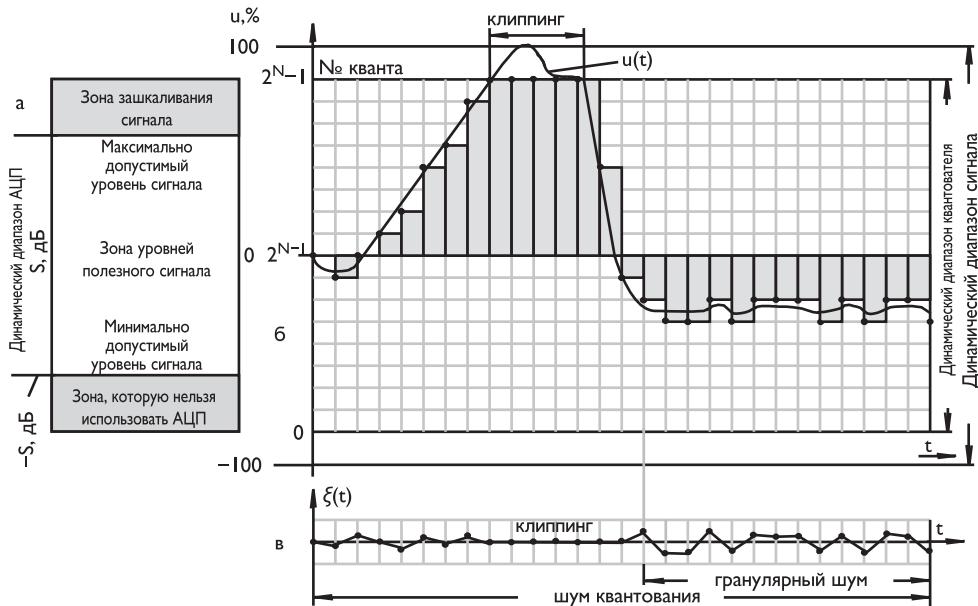


Рис. 15.16. Диаграмма уровней сигнала на входе АЦП (а) и пример возникновения клиппинга (б) и шумов квантования (в)

цифрового сигнала $d(t_n)$, как показано справа на рис. 15.16б, будут равны

$$d(t_n) = \begin{cases} j, & \text{при } \sin \omega t_n \leq 0, \\ j+1, & \text{при } \sin \omega t_n > 0. \end{cases}$$

Для оцифровки речевой информации, ограниченной по спектру до 2–5 кГц, обычно используют 7- или 8-разрядные АЦП при частотах дискретизации от 8 до 10 кГц.

В большинстве современных цифровых систем обработки и кодирования звуковой информации обычно применяют 16-разрядное квантование и стандартные частоты дискретизации 44,1 или 48 кГц, ограничивая частотный диапазон сигнала примерно равным 20 кГц.

В студийной аппаратуре зачастую используют 18-, 20-, 24- и 32-разрядное квантование при частотах дискретизации 56, 96 и 192 кГц, что обеспечивает сохранность высших гармоник звукового сигнала, непосредственно не воспринимаемых слухом, но влияющих на формирование общей звуковой картины. При ограничении спектра сигнала, учитывая что $f_{\text{пд}} > f_{\text{пгр}}$, дополнительно возможно уменьшить уровень шумов квантования.

В табл. 15.4 приведены характеристики цифрового звукового сигнала для различных видов аналого-цифрового преобразования (при расчете отношения сигнала к шуму квантования не учтено влияние константы C , величина которой зависит от формы преобразуемого сигнала).

В этой таблице приведены параметры цифрового потока импульсно-кодовой модуляции (ИКМ) монофонического сигнала. При преобразовании стереофонического или квадрофонического звукового сигнала величина цифрового потока должна быть увеличена в два или четыре раза соответственно.

Таблица 15.4. Параметры цифрового звукового сигнала при использовании различных видов АЦП

Разрядность АЦП N	Частота дискретизации, кГц	Цифровой поток, кбитов/с	Отношение сигнал/шум $6N$, дБ	Отношение сигнал/шум при ограничении спектра сигнала до частоты $f_{\text{пгр}}$
7–8	8–12	56–96	42–48	—
16	44,1	705,6	96	96
16	48	768	96	96,4
18	56	1008	108	109
20	96	1920	120	123,4
24	96	2304	144	147,4
24	192	4608	144	148,4
32	192	6144	192	196,4

Чтобы влияние шумов квантования было не очень большим, необходимо максимально использовать весь динамический диапазон АЦП, поддерживая уровень преобразуемого аналогового сигнала в определенных границах. Этот процесс реализуется путем использования компандеров, обеспечивающих обработку динамично изменяющегося уровня звукового сигнала. Однако этот метод не всегда приводит к положительным результатам, особенно в тех случаях, когда в течение некоторого времени звуковая информация попеременно изменяется от тихой к громкой. В этих случаях прибегают к иным методам уменьшения влияния шума квантования.

Один из них, называемый **дизерингом** (от англ. dithering — дрожание), используется в тех случаях, когда шум существенно зависит от преобразуемого сигнала и проявляется на слух в виде специфической мешающей помехи (что практически всегда имеет место). Принцип дизеринга заключается в добавлении к преобразованному сигналу псевдослучайной последовательности малого по уровню (не более двух дискретных значений) шума, реализуя тем самым практически полную независимость шума от исходного сигнала, что для слуха является более приемлемым.

Другой принцип снижения влияния связанных с формой сигнала шумов квантования, называемый методом **формовки шума** (noise shaping), заключается в преобразовании спектра шума таким образом, чтобы большая часть его энергии располагалась в менее заметных для слуха областях спектра в соответствии с формой кривых равной громкости.

На рис. 15.17 приведена структурная схема реализации дизеринга и формовки шума. Выбор схемы преобразования сигнала зависит от положения ключей D и F (0 — соответствует разомкнутому, а 1 — замкнутому ключу):

- при $D = 0$ и $F = 0$ обеспечивается только квантование сигнала;
- при $D = 1$ и $F = 0$ реализуется квантование с дизерингом;
- при $D = 0$ и $F = 1$ осуществляется квантование с формовкой шума;
- при $D = 1$ и $F = 1$ обработка сигнала осуществляется с дизерингом и формовкой шума.

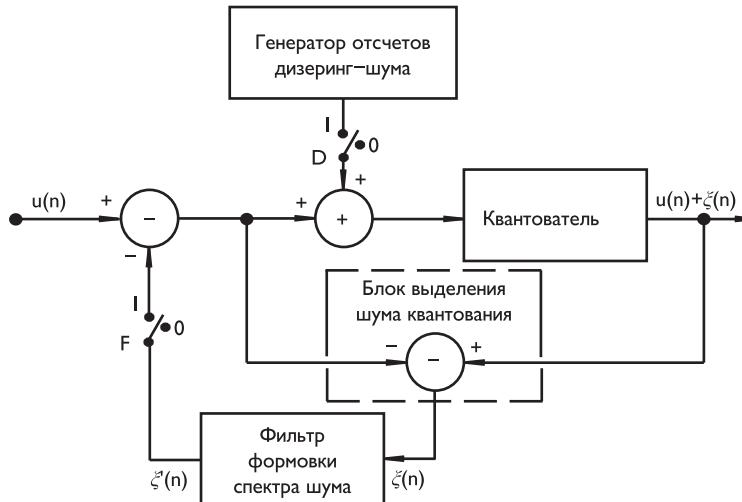


Рис. 15.17. Структурная схема реализации дизеринга и формовки шума

В последнем случае дискретные отсчеты сигнала $u(n)$ подаются через подключенный ко входу вычитатель и далее — на сумматор, на второй вход которого поступают дискретные отсчеты псевдослучайного дизеринг-шума. На выходе этого сумматора дискретные отсчеты сигнала объединены с отсчетами дизеринг-шума, а на выходе квантователя они дополняются отсчетами шумов квантования $\xi(n)$. Второй вычитатель реализует выделение этих шумов квантования и дизеринга $\xi(n)$ путем формирования разности уровней отсчетов выходного сигнала и отсчетов сигнала, выделяемого на выходе первого вычитателя. Изменение формы спектра шумов осуществляется с помощью моделирующего фильтра, шум $\xi'(n)$ с выхода которого через ключ F соединен со вторым входом первого вычитателя. В случае если $F = 0$ или $D = 0$ соответствующие преобразования не производятся.

Применение рассмотренных методов преобразования шумов квантования не всегда рационально, особенно в случаях необходимости дальнейшей обработки аудиосигналов.

Поэтому чаще всего для уменьшения влияния шумов квантования обработку аудиосигналов производят с применением АЦП с большей разрядностью при более высокой частоте дискретизации.

При этом чем выше частота дискретизации, тем большие требования предъявляются к ее стабильности, поскольку несовершенства преобразующей аппаратуры, приводящие к случайным времененным отклонениям дискретизирующих импульсов, вызывают эффект **джиттера** (от англ. jitter — дрожание). Для борьбы с джиттером применяются высокостабильные кварцевые генераторы.

15.6. Аналого-цифровое и цифро-аналоговое преобразование звуковых сигналов

Кодирующее и декодирующее устройства на входе и выходе оперируют с аналоговыми сигналами, преобразование которых в цифровой код и восстановление являются важнейшими этапами обработки информации. Сигналы на выходе декодера даже при отсутствии линейной и нелинейной обработки информации практически никогда не совпадают по форме с сигналами, поступающими на вход кодера. Ниже приводятся наиболее широко используемые варианты преобразования звуковых сигналов.

15.6.1. Импульсно-кодовая модуляция (ИКМ)

ИКМ — наиболее простой способ преобразования сигналов, обычно содержащий многоразрядный АЦП (чаще с линейной шкалой квантования) в кодере и ЦАП в декодере, имеющий такую же разрядность.

Если число N — разрядность АЦП, а частота дискретизации сигнала равна $f_{\text{пд}}$, то формируемый цифровой поток определяется их произведением: $N \cdot f_{\text{пд}}$.

На выходе ЦАП при этом последовательно с частотой дискретизации $f_{\text{пд}}$ выделяются импульсные дискретные значения звукового сигнала. Каждый из этих почти прямоугольных импульсов длительностью $\tau = 1/f_{\text{пд}}$ обладает sinc-спектром $F(f) \equiv \text{sinc } \pi \cdot f / f_{\text{пд}} = \frac{\sin \pi \cdot f / f_{\text{пд}}}{\pi \cdot f / f_{\text{пд}}}$, плавно изменяющимся от 1 на частоте $f = 0$ до $2/\pi$ на частоте $f_{\text{пд}}/2$.

Если частота дискретизации мало отличается от удвоенной величины граничной частоты преобразуемого сигнала $f_{\text{пд}} \approx 2 \cdot f_{\text{пгр}}$, то спектральные составляющие сигнала в области граничной частоты оказываются значительно ослабленными. Устранение этого эффекта возможно либо путем использования ФНЧ с соответствующим подъемом частотной характеристики в области граничной частоты сигнала, либо с применением sinc-предкоррекции дискретных отсчетов АЦП.

Форма частотной характеристики предкорректирующей цепи определяется формулой:

$$F_{\text{пр}}(f) = \frac{(\pi \cdot f / f_{\text{пд}})}{(\sin \pi \cdot f / f_{\text{пд}})} = a_0 + 2 \sum_{i=1}^{\infty} a_i \cos(2i\pi \cdot f / f_{\text{пд}}), \quad (15.23)$$

где $a_0 = \frac{4}{\pi} G$, $a_i = \frac{4}{\pi} \left[G - \sum_{k=0}^{i-1} \frac{(-1)^k}{(2k+1)^2} \right]$, $G = \sum_{k=0}^{\infty} \frac{(-1)^k}{(2k+1)^2} = 0,915965594\dots$ — постоянная Каталана.

Ряд (15.23) достаточно быстро сходится, и по этой причине можно ограничиться конечным числом его членов. С точностью выше 0,1% процесс предварительной коррекции реализуется в соответствии с соотношением:

$$\bar{u}(n) = a_0 \cdot u(n) + \sum_{k=1}^4 a_k \cdot [u(n-k) + u[n+k]], \quad (15.24)$$

где $a_0 = 1,1662$, $a_1 = -0,1070$, $a_2 = 0,0345$, $a_3 = -0,0165$, $a_4 = 0,0095$.

15.6.2. Дифференциальная импульсно-кодовая модуляция (ДИКМ)

Как и при использовании ДИКМ видеосигнала, на вход квантователя последовательно подаются отсчеты не исходного сигнала, а разность между его текущим значением и предсказанной величиной:

$$e(n) = u(n) - \tilde{u}(n). \quad (15.25)$$

При использовании метода линейного предсказателя величина $\tilde{u}(n)$ определяется соотношением:

$$\tilde{u}_n = \sum_{k=1}^K a_k \hat{u}(n-k), \quad (15.26)$$

где K и a_k — порядок и коэффициенты предсказания, $\hat{u}(n-k)$ — отсчеты сигнала, полученные на предыдущих шагах преобразования исходных величин $u(n-k)$.

Формирование величин $\hat{u}(n)$ определяется принципом кодирования информации о разностной величине $e(n)$ и с учетом поправки $Q[e(n)]$:

$$\hat{u}(n) = \tilde{u}(n) + Q[e(n)]. \quad (15.27)$$

Величины коэффициентов предсказания зависят от корреляционных свойств преобразуемой аудиоинформации и подбираются опытным путем. Например, при оцифровке речевой информации может быть использован предсказатель пятого порядка с коэффициентами [6.2]:

$$a_1 = 0,86; a_2 = 0,64; a_3 = 0,40; a_4 = 0,26; a_5 = 0,20.$$

Наиболее простым способом кодирования с предсказанием является **дельта-модуляция (ДМ)**, реализуемая с помощью однобитного квантователя. Ошибка предсказания $e(n)$ и поправка $Q[e(n)]$ определяются соотношениями:

$$e(n) = u(n) - \hat{u}(n-1); \quad Q[e(n)] = \begin{cases} 1, & \text{при } e(n) \geq 0; \\ 0, & \text{при } e(n) < 0. \end{cases} \quad (15.28)$$

Принцип формирования сигнала на выходе декодирующего устройства заключается в следующем:

- если величина $Q[e(n)] = 1$, то значение отчета выходного сигнала вычисляется по формуле $\hat{u}(n) = \hat{u}(n-1) + \Delta$, где Δ — некоторое дискретное приращение сигнала;
- при $Q[e(n)] = 0$ результат преобразования отсчета имеет вид $\hat{u}(n) = \hat{u}(n-1) - \Delta$.

На рис. 15.18 приведен пример обработки сигнала при дельта-модуляции.

Применение такого вида преобразования приводит к двум видам искажений — **перегрузке крутизны** (или наклонной перегрузке), связанной с неспособностью кодирующего устройства реагировать на быстрые изменения величин отсчетов аудиосигнала, и появлению **гранулярного шума**. В связи с этим использование дельта-модуляции эффективно при высокой корреляции соседних отсчетов сигнала и применяется при повышенных частотах дискретизации звуковых сигналов.

Более эффективно использование **адаптивной дельта-модуляции (АДМ)**, при которой в зависимости от характера поступающих на вход кодирующего

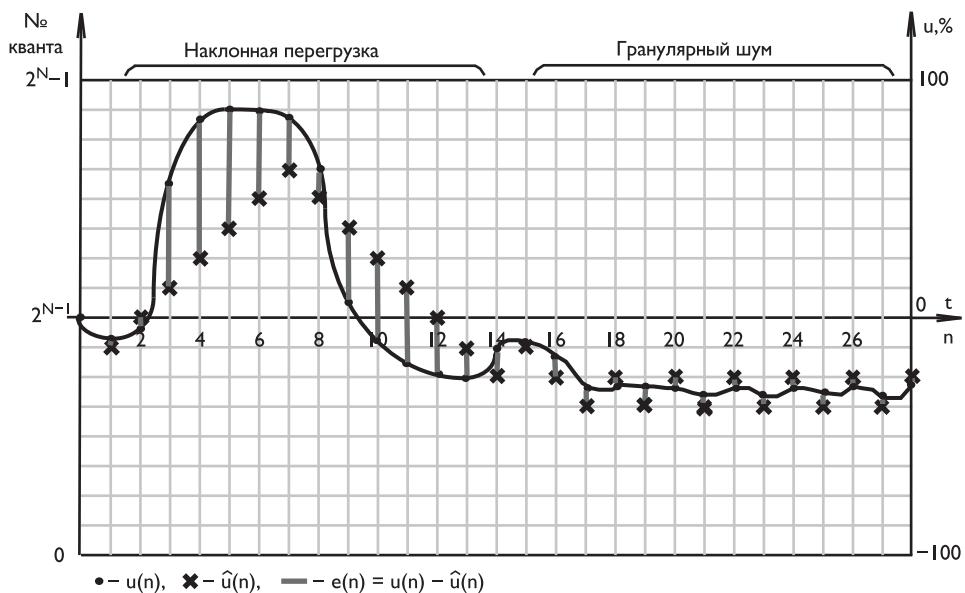


Рис. 15.18. Обработка сигналов при дельта-модуляции

устройства отсчетов сигнала изменяется шаг квантования. Наиболее простой способ изменения шага квантования поясняется следующим примером.

На начальном этапе преобразования в качестве исходного шага квантования принимаются значение поправки $Q[e(0)] = 1$ и величина шага квантования $\Delta(0) = \Delta$.

Последующее изменение шага квантования осуществляется, например, в соответствии с формулой:

$$\Delta(n) = \begin{cases} \Delta(n-1) + \Delta/2, & \text{если } Q[e(n)] = Q[e(n-1)]; \\ \Delta(n-1) - \Delta/2, & \text{если } Q[e(n)] \neq Q[e(n-1)]. \end{cases} \quad (15.29)$$

На рис. 15.19 приводится пример преобразования сигнала с применением АДМ, иллюстрирующий более эффективное кодирование быстро изменяющегося сигнала и частично компенсирующий нежелательный эффект перегрузки крутизны. Гранулярный шум также компенсируется за счет автоматического уменьшения шага квантования на каждом новом этапе преобразования.

Еще большая эффективность кодирования аудиоинформации может быть получена при квантователе, использующем предсказание более высокого порядка и реализуемом **адаптивной относительной импульсно-кодовой модуляции АОИКМ** [6.17].

Шаг квантования $\Delta(n)$ зависит от результата кодирования на предыдущем шаге: $\Delta(n) = M \cdot \Delta(n-1)$, а функция квантования $Q[e(n)]$ зависит от величины $\Delta(n)$ и передается на декодер тремя битами.

Значение коэффициента M , увеличивающего или уменьшающего шаг квантования, определяется следующим соотношением:

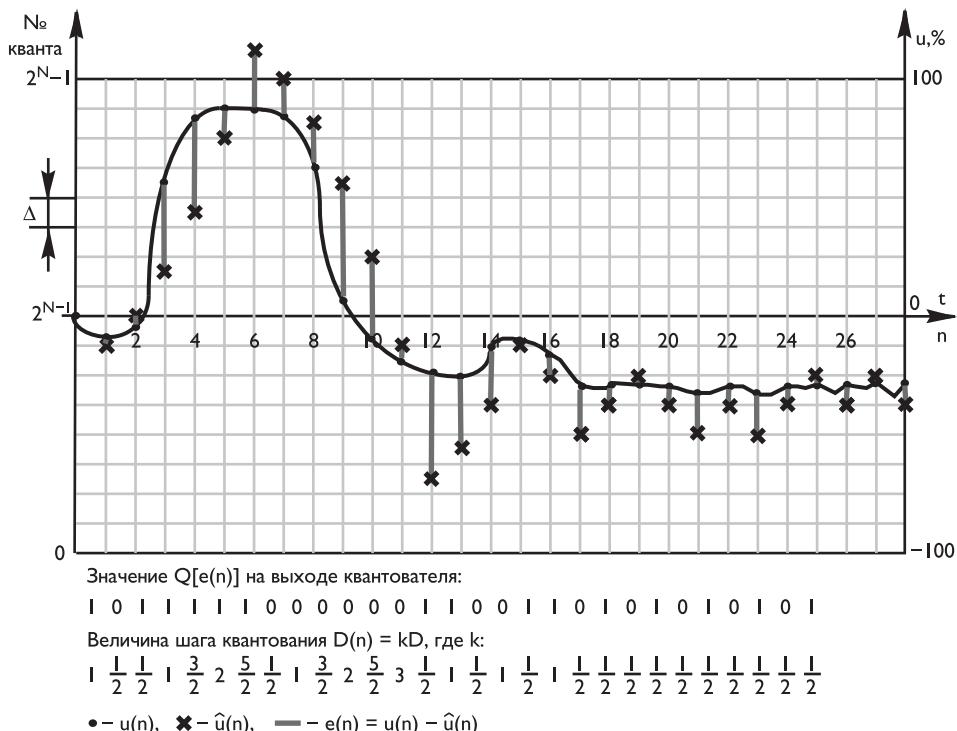


Рис. 15.19. Преобразование сигнала с использованием аддитивной дельта-модуляции

$$M = \begin{cases} 0, 5 & \text{при } 0 \leq |e(n-1)| \leq \Delta(n-1); \\ 1, 5 & \text{при } \Delta(n-1) < |e(n-1)| \leq 2\Delta(n-1); \\ 2, 5 & \text{при } 2\Delta(n-1) < |e(n-1)| \leq 3\Delta(n-1); \\ 3, 5 & \text{при } 3\Delta(n-1) < |e(n-1)|. \end{cases} \quad (15.30)$$

Данная схема эффективна, весьма проста и легко реализуется в цифровом виде. Существуют более сложные и более эффективные схемы реализации АОИКМ с многобитной функцией квантования $Q[e(n)]$.

15.6.3. Сигма-дельта-модуляция (СДМ)

Основой сигма-дельта-модуляции является не анализ приращений сигнала, а кодирование уровней самого преобразуемого сигнала, как при ИКМ [6.18–6.20].

Структурная схема сигма-дельта-модулятора приведена на рис. 15.20, на вход которого подаются дискретные отсчеты сигнала $u(n)$, относительные уровни которых могут изменяться в пределах от -1 до $+1$.

Кодер содержит однобитовый квантователь, на выходе которого формируется сигнал:

$$Q(n) = \begin{cases} +1, & \text{если } v_2(n) \geq 0, \\ -1, & \text{если } v_2(n) < 0, \end{cases} \quad (15.31)$$

где $v_2(n)$ – дискретные отсчеты сигнала на выходе квантователя, формируемые

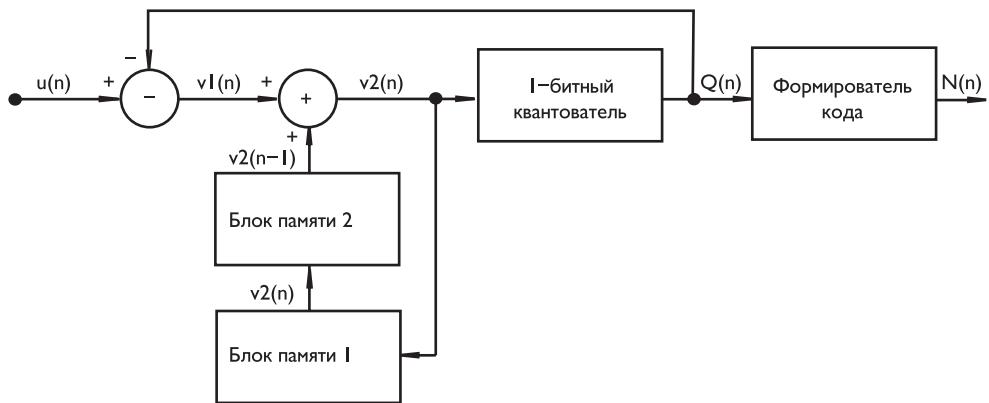


Рис. 15.20. Структурная схема сигма-дельта-модулятора

следующим образом:

- вычитатель, включенный на входе устройства, обеспечивает формирование разности отсчетов входного сигнала $u(n)$ и выходного сигнала квантователя $Q(n) - v_1(n) = u(n) - Q(n)$;
- отсчеты разностного сигнала $v_1(n)$ подаются на сумматор, с выхода которого отсчеты сигнала $v_2(n)$ поступают параллельно на входы квантователя и блока памяти БП1;
- с выхода блока БП1 с задержкой на такт отсчета сигнала подаются на вход блока памяти БП2, на выходе которого выделяются отсчеты сигнала предыдущего такта $v_2(n-1)$, которые поступают на второй вход сумматора;
- таким образом, на вход квантователя подается сигнал $v_2(n) = v_1(n) + v_2(n-1)$.

Формирователь кода, включенный на выходе квантователя, создает цифровую последовательность в соответствии с соотношением:

$$N(n) = \begin{cases} 1, & \text{если } Q(n) = 1; \\ 0, & \text{если } Q(n) = -1. \end{cases} \quad (15.32)$$

На рис. 15.21 представлен процесс кодирования синусоидального сигнала, для наглядности шаг дискретизации выбран достаточно большим. Работа схемы поясняется табл. 15.5, в которой величины отсчетов синусоидального сигнала округлены до двух знаков после запятой.

Таблица 15.5. Пояснение работы сигма-дельта-модулятора

№ такта	Входной сигнал $u(n)$	На выходе вычитателя $v_1(n)u(n) - -Q(n)$	На входе БП2 $v_2(n-1)$	На выходах слагателя и БП1 $v_2(n)v_1(n) + v_2(n-1)$	На выходе квантователя $Q(n)$
0	0,00	0,00	0,00	0,00	1
1	0,25	-0,75	0,00	-0,75	-1
2	0,48	1,48	-0,75	0,73	1
3	0,68	-0,32	0,73	0,41	1
4	0,84	-0,16	0,41	0,25	1

Таблица 15.5 (окончание)

5	0,95	-0,05	0,25	0,20	1
6	1,00	0,00	0,20	0,20	1
7	0,98	-0,02	0,20	0,18	1
8	0,90	-0,10	0,18	0,08	1
9	0,77	-0,23	0,08	-0,15	-1
10	0,59	1,59	-0,15	1,44	1
11	0,37	-0,63	1,44	0,81	1
12	0,13	-0,87	0,81	-0,06	-1
13	-0,13	0,87	-0,06	0,81	1
14	-0,37	-1,37	0,81	-0,56	-1
15	-0,59	0,41	-0,56	-0,15	-1
16	-0,77	0,23	-0,15	0,08	1
17	-0,90	-1,90	0,08	-1,82	-1
18	-0,98	0,02	-1,82	-1,80	-1
19	-1,00	0	-1,80	-1,80	-1
20	-0,95	0,05	-1,80	-1,75	-1
21	-0,84	0,16	-1,75	-1,59	-1
22	-0,68	0,32	-1,59	-1,27	-1
23	-0,48	0,52	-1,27	-0,75	-1
24	-0,25	0,75	-0,75	0,00	1
25	0,00	-1,00	0,00	-1,00	-1
26	0,25	1,25	-1,00	0,25	1
27	0,48	-0,52	0,25	-0,27	-1

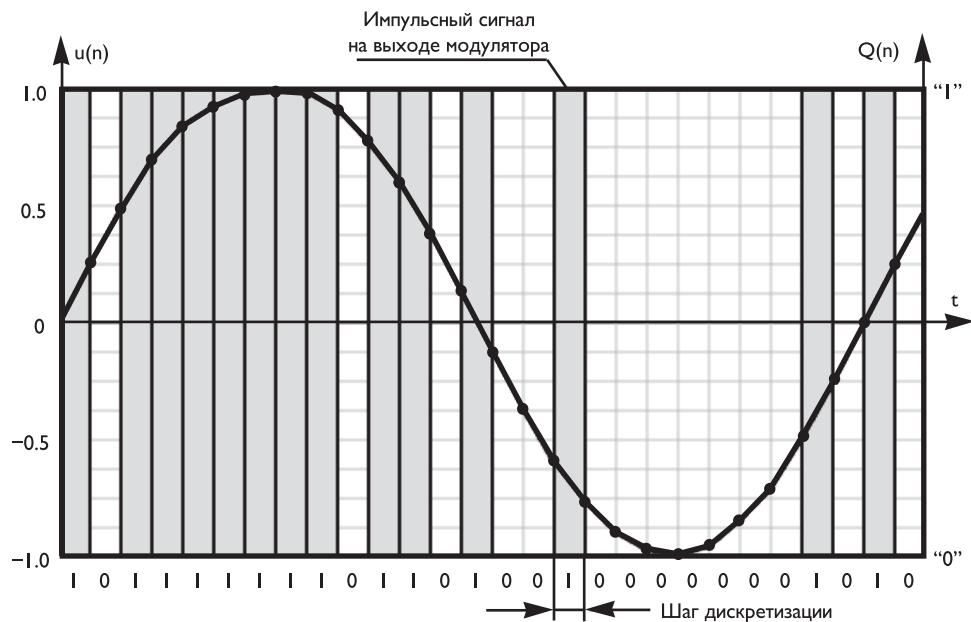


Рис. 15.21. Процесс СДМ-кодирования синусоидального сигнала

СДМ обычно применяется при весьма высоких частотах дискретизации, и шум квантования распространяется на значительно более широкую полосу частот, чем при применении ИКМ. По этой причине на область слышимых частот в диапазоне до 20 кГц приходится весьма малая часть энергии шума. Следует также заметить, что формат СДМ легко может быть преобразован в формат ИКМ [6.2].

15.7. Принципы кодирования речевой и звуковой информации

Оцифрованный аудиосигнал в форме одной из вариаций ИКМ является практически точной копией, но не компактной формой записи исходного аналогового сигнала. Поэтому преобразование относительно больших объемов аудиоданных, гарантирующих необходимое качество воспроизведения различных видов звуковой информации, требует применения различных методов кодирования, позволяющих существенно уменьшить избыточность информации для ее хранения или передачи по каналам связи.

Принципы кодирования речевой информации, основным требованием к которой является разборчивость воспроизводимого сигнала, и звуковой информации, гарантирующей достаточно хорошее качество звучания музыкальных и голосовых передач, существенно отличаются друг от друга.

15.7.1. Принципы кодирования речевой информации

В качестве международного стандарта для передачи речи принято использование полосы частот от 300 до 3400 Гц, достаточной для воспроизведения передаваемой информации. Решающими в выборе такой полосы были в основном экономические соображения. Правда, в цифровой телефонии существует принципиальная возможность использовать спектр речи за пределами полосы 300–3400 Гц и тем самым повысить качество, но эти методы не реализуются, так как они вычислительно более сложны. На основе этого стандарта построена всемирная сеть телефонной связи. В этом случае для описания формы сигнала его дискретизацию следует проводить с частотой порядка 8 кГц, а для получения нормального качества воспроизведения речи при равномерной шкале квантования необходимо использовать 13/14-разрядный квантователь.

В результате для передачи такой информации с помощью последовательности двоичных импульсов необходимо транслировать цифровой поток $8 \cdot 13(14) = 104(112)$ кбитов/с.

Указанное линейное квантование оказывается необходимым потому, что уровни аналоговых речевых сигналов могут изменяться в диапазоне 60 дБ. Но так как восприятие сигналов органами слуха человека пропорционально логарифму уровня сигнала, то сигналы высокого уровня целесообразно квантовать более грубо, а низкого уровня — более точно. Применяя нелинейное квантование с использованием логарифмического закона, можно обойтись восемью разрядами на отсчет, сохранив почти такое же качество передачи.

При этом используются соответствующие этой шкале соотношения A-Law и μ -Law [6.21]. Принятое в Европе соотношение A-Law обеспечивает преобразова-

ние 13-разрядной линейной шкалы градаций сигнала (от -2^{12} до $+2^{12}$) в 8-разрядную шкалу в соответствии с соотношением:

$$Y_A(m) = \text{round} \left[2^7 \cdot \begin{cases} \frac{A}{1+\ln A} \left(\frac{m}{m_f} \right), & \text{при } \left| \frac{m}{m_f} \right| \leq \frac{1}{A}; \\ \frac{\text{sign}(m)}{1+\ln A} \left(1 + \ln A \left| \frac{m}{m_f} \right| \right), & \text{при } \frac{1}{A} < \left| \frac{m}{m_f} \right| \leq 1 \end{cases} \right], \quad (15.33)$$

где $A = 87,6$; m и $m_f = 2^{12}$ — текущее и максимальное значение шкалы оцифрованного сигнала.

Соотношение μ -Law используется в Северной Америке и в Японии, и для преобразования линейной 14-разрядной шкалы в логарифмическую применяется формула:

$$Y_\mu(m) = \text{round} \left[2^7 \cdot \frac{\text{sign}(m)}{\ln(1+\mu)} \ln \left(1 + \mu \left| \frac{m}{m_p} \right| \right) \right], \quad (15.34)$$

где $\mu = 255$, m и $m_p = 2^{13}$ — текущее и максимальное значение шкалы оцифрованного сигнала.

Соотношения μ -Law и A-Law позволяют исходные многоразрядные цифровые сигналы преобразовывать в восьмиразрядные данные, что обеспечивает сжатие информации более чем на 30%. В результате скорость передачи двоичных разрядов окажется равной 64 кбитов/с. По сравнению с более эффективными методами сжатия речевой информации приведенное кодирование максимально нечувствительно к ошибкам в канале при достаточно высоком качестве восстановления. Поэтому данный алгоритм рекомендован для большинства систем цифровой передачи речи в качестве метода предварительного аналого-цифрового преобразования.

Дальнейшим усовершенствованием системы кодирования речи является применение адаптивного дифференциального метода импульсно-кодовой модуляции (АДИКМ). Преобразование и передача лишь разницы между реальным и предсказанным значениями сигнала позволяет уменьшить формируемый цифровой поток до 16–32 кбитов/с и заметно снизить требования к широкополосности канала. Следует иметь в виду, что метод не лишен серьезных недостатков: уровень шумов, связанный с квантованием сигнала, выше, а при резких изменениях уровня сигнала, превышающих диапазон АЦП, возможны серьезные искажения.

После того как аналоговый сигнал преобразован в цифровую форму, к нему можно применять различные способы обработки, которые обеспечивают сжатие речевой информации для достижения определенных целей — нужных скоростей битового потока, качества сигнала, задержки и сложности.

Для обеспечения эффективного кодирования используют разделение речевой информации на сигнальные фрагменты:

- так называемую частоту основного тона (ОТ) — периодическую подпитку энергией голосового тракта человека, который представляет собой объемный резонатор;
- формируемую голосовым трактом спектральную окраску речи, или ее формантную структуру, содержащую усиленные частотные области данного звука, позволяющие отличить его при слуховом восприятии от других звуков;
- переходные процессы при изменениях ОТ и формантных структур;

- паузы, средняя продолжительность которых составляет примерно 16% времени воспроизведения речи и порядка 50% времени при диалоге.

При передаче речи в цифровой форме каждый тип сигнала при одной и той же длительности и одинаковом качестве требует различного числа битов для кодирования и передачи. Следовательно, скорость передачи разных типов сигнала также может быть различной, что обуславливает применение кодеков с переменной скоростью. В основе кодека речи с переменной скоростью лежит классификатор входного сигнала, определяющий степень его информативности и, таким образом, задающий метод кодирования и скорость передачи речевых данных. Наиболее простым классификатором речевого сигнала является **детектор активной речи** (VAD — Voice Activity Detector), который выделяет во входном речевом сигнале активную речь и паузы. При этом фрагменты сигнала, классифицируемые как активная речь, кодируются каким-либо из известных алгоритмов (как правило, на базе метода АДИКМ) с базовой скоростью 4–8 кбитов/с. Фрагменты, классифицированные как паузы, кодируются и передаются с низкой скоростью порядка 0,1–0,2 кбита/с, либо не передаются вообще. Эта стратегия позволяет оптимизировать скорость кодирования до 2–4 кбита/с при достаточном качестве синтезируемой речи. При этом для особо критичных фрагментов речевого сигнала выделяется большая скорость передачи, для менее ответственных — меньшая. Вместе с тем необходимо отметить, что вокодер вносит дополнительную задержку, возникающую за счет использования буфера для накопления сигнала и учета статистики последующих отсчетов (алгоритмическая задержка) и выполнения алгоритмических преобразований речевого сигнала (вычислительная задержка).

В интервалы, когда в речи активного участника беседы наступает период молчания, терминалы слушающих могут просто отключить воспроизведение звука, но при этом в трубке может возникнуть «гробовая типина», и слушающему кажется, что соединение по каким-то причинам нарушилось. Избежать такого неприятного эффекта позволяет применение **генератора комфорtnого шума** (CNG — Comfort Noise Generator), параметры которого могут передаваться во время пауз.

Для того чтобы синтезировать речь на приемном конце системы связи, нужны генератор звуковой частоты с богатым спектром, генератор белого шума, набор формантных фильтров (их число невелико, так как гласных звуков немного, а каждый из них достаточно хорошо определяется двумя формантами) и модулирующие схемы. Располагая таким комплектом аппаратуры на приемном конце, можно передавать по каналу связи не речевой сигнал, а лишь команды, управляющие процессом синтеза речи. Таким образом, практическая задача сводится к тому, чтобы найти способ генерирования нужных команд.

Скорости передачи речевой информации, которую предусматривают используемые сегодня узкополосные кодеки, лежат в пределах 1,2–64 кбитов/с. Естественно, что от этого параметра прямо зависит качество воспроизведенной речи. Существует множество подходов к проблеме определения качества. Для прослушивания экспертам предъявляются разные звуковые фрагменты. Искажения оценивают путем опроса разных групп людей по пятибалльной шкале единицами субъективной оценки MOS (Mean Opinion Score). Оценки интерпретируют следующим образом:

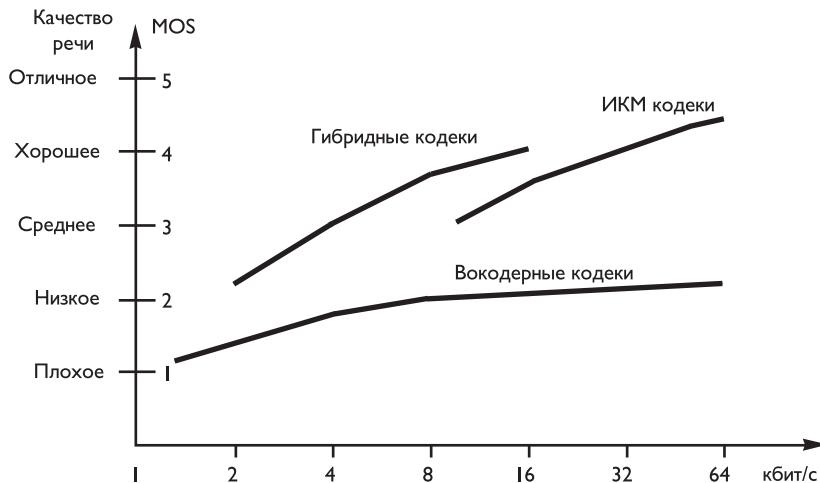


Рис. 15.22. Примерные характеристики кодеков речи

- 4–5 — высокое качество; аналогично качеству передачи речи в относительно широкополосных каналах связи;
- 3,5–4 — достаточное качество; аналогично качеству речи, передаваемой с помощью кодека АДИКМ при скорости 32 кбитов/с; такое качество обычно обеспечивается в большинстве телефонных разговоров;
- 3–3,5 — качество речи по-прежнему удовлетворительно, однако его ухудшение явно заметно на слух;
- 2,5–3 — речь разборчива, однако требует концентрации внимания для понимания; такое качество обычно обеспечивается в системах связи специального применения (например, в вооруженных силах).

В рамках существующих технологий достаточно высокое качество воспроизведения речи невозможно обеспечить при скоростях менее 4 кбитов/с.

На рис. 15.22 приведены примерные характеристики различных видов кодеков речи, оцениваемые в единицах MOS.

15.7.2. Принципы кодирования звуковой информации

Несколько иные проблемы возникают при **кодировании широкополосных звуковых сигналов**, реализующих технологии для музыкального творчества.

При первичном кодировании в студийном тракте обычно реализуется равномерное квантование отсчетов аудиосигналов с разрешением от 16 до 24 (и даже 32) битов/отсчет при частотах дискретизации 44,1, 48, 96 (и даже 192) кГц.

Считается, что в кодеках, использующих 16-битное линейное квантование отсчетов, при частоте дискретизации 48 кГц удается практически «идеально» преобразовывать аудиосигнал, обладающий спектром в полосе частот от 20 Гц до 20 кГц и динамическим диапазоном до 54 дБ. Скорость передачи одного такого сигнала составляет $48 \cdot 16 = 768$ кбитов/с. При стереоформате или звуковом формате 5.1 (или 3/2 плюс канал сверхнизких частот) цифровой поток может

составить соответственно 1,536 или 3,840 Мбит/с. Энтропийное кодирование такой информации позволяет сократить статистическую избыточность цифрового потока. Однако даже при использовании достаточно сложных алгоритмов обработки информации уменьшение статистической избыточности аудиосигналов позволяет уменьшить цифровой поток лишь на 20–50% по сравнению с его исходным значением.

Тем не менее такой способ преобразования, использующий многоразрядное квантование дискретных отсчетов сигнала с интервалом дискретизации, превышающим интервал Котельникова–Найквиста, при применении энтропийного кодирования цифровой информации считается **сжатием аудиосигналов без потерь** (Lossless Coding), поскольку этот способ позволяет осуществлять сто процентное восстановление исходных данных из сжатого потока. В данном случае под понятием «исходные данные» подразумевается исходный вид оцифрованных аудиоданных.

Кодирование аудиоинформации без потерь зачастую используется при ее хранении, но не способно обеспечить высокий уровень компрессии.

Как известно, органы слуха человека способны воспринимать информацию в объеме не более 100 бит/с и, следовательно, можно говорить о значительной избыточности закодированных без потерь звуковых цифровых сигналов. Существенная психоакустическая избыточность цифровых аудиосигналов дает возможность ее уменьшения.

Цель **сжатия аудиоинформации с потерями** (Lossy Coding) заключается в достижении максимально высокого коэффициента компрессии данных при сохранении качества их звучания на приемлемом уровне. Кодирование с потерями приводит к утрате некоторой части информации, и декодированный сигнал при воспроизведении звучит похоже на оригинал, но фактически перестает быть ему идентичным.

В основе большинства методов кодирования с потерями лежит использование психоакустических свойств слуховой системы человека. В частности, наиболее перспективными с этой точки зрения являются алгоритмы, учитывающие такие свойства слуха, как различного рода маскировка, выявление различных деталей звучания, которыми можно пренебречь, эффективные алгоритмы переквантования и передискретизации и др.

При кодировании звуковых сигналов наибольшее распространение получили три психоакустические модели, использующие различные алгоритмы обработки ИКМ-аудиоинформации [6.22].

Психоакустическая модель 1

Эта модель содержит девять блоков обработки первичного ИКМ-сигнала. **В блоке 1** производится расчет энергетического спектра звукового сигнала и его нормирование в соответствии с соотношениями:

$$F(k) = 20 \cdot R \cdot \lg \left| \frac{1}{N} \sum_{n=0}^{N-1} h(n) \cdot X(n) \cdot \exp \left(-2\pi j k \frac{n}{N} \right) \right| \text{дБ}, \quad (15.35)$$

где $X(n)$ — отсчеты сигнала; N — длина выборки БПФ (обычно 1024 или 512); $k = 0, 1, \dots, N/2$; $h(n) = \sin^2 \pi \frac{n}{N}$ — окно Ханна, используемое для уменьшения ис-

каждений, вызванных эффектом Гиббса; R — коэффициент, обеспечивающий нормировку максимальной по величине спектральной компоненте до уровня 96 дБ.

При частоте дискретизации $f_d = 48$ кГц и $N = 1024$ частотное разрешение линейчатого спектра БПФ составляет $\Delta f = 46,875$ Гц.

Во 2-м блоке вычисляется энергия сигнала выборки в каждой из субполосах кодирования по формуле:

$$E_{sb}(m) = \max \{ [F_{sb}(m)(k)]_{\max}, 20 \lg [32768 \cdot SCF_{\max}(m)] - 10 \} \text{ дБ}, \quad (15.36a)$$

или более точному выражению:

$$E_{sb}(m) = \max \{ [F_{sb}(k)], 20 \lg [32768 \cdot SCF_{\max}(m)] - 10 \} \text{ дБ}, \quad (15.36b)$$

где $F_{sb}(m) = 10 \lg \sum_{k \in m} 10^{F(k)/10}$ дБ, $m = 1, \dots, M$ — номера субполос (обычно $M = 32$); $[F_{sb}(m)(k)]_{\max}$ — уровень энергии максимальной по величине спектральной компоненты в m -ой субполосе; число 10 корректирует разность между пиковым и средним значениями уровней сигнала; $F_{sb}(k)$ — усредненная энергия всех спектральных компонент в каждой субполосе; $SCF_{\max}(m)$ — значение масштабного коэффициента для каждой из субполос, вычисляемых следующим образом:

- каждая из субполос разбивается на три части, называемых гранулами;
- в каждой грануле определяется максимальный отсчет спектральной компоненты;
- величина этих максимальных отсчетов в каждой грануле сравнивается с заданными таблично масштабными коэффициентами $SCF_i(m)$, $i=1, 2, 3$, и выбираются наиболее близкие;
- из трех выбранных масштабных коэффициентов $SCF_i(m)$ используется коэффициент наибольшей величины $SCF_{\max}(m)$.

3-й блок используется для выделения локальных максимумов спектра выборки отсчетов сигнала, реализуемого соотношениями:

$$F(k) > F(k-1) \text{ и } F(k) \geq F(k+1). \quad (15.37)$$

В 4-м блоке реализуется формирование списка тональных компонент путем исследований области частот вокруг каждой спектральной компоненты, являющейся локальным максимумом. Соответствующая ему спектральная составляющая включается в список тональных компонент $\{F_{tm}(k)\}_{tk}$, если в обследованной области частот она превышает любую компоненту, исключая две соседние с ней, не менее чем на 7 дБ: $F(k) - F(k+j) \geq 7$, дБ в обследуемой области частот при длине выборки $N = 512$ —

- $j = -2, +2$ для $2 < k < 63$;
- $j = -3, -2, +2, +3$ для $63 \leq k < 127$;
- $j = -6, \dots, -2, +2, \dots, +6$ для $127 \leq k < 255$;
- $j = -12, \dots, -2, +2, \dots, +12$ для $255 \leq k < 511$.

$$F_{tm}(k) = 10 \lg \sum_{j \in k} 10^{F(k+j)/10} \text{ дБ.} \quad (15.38)$$

Как правило, число тональных компонент невелико.

5-й блок формирует список нетональных (шумоподобных) областей — $\{F_{nm}(k)\}_{nk}$. С этой целью из исходного спектра выборки сигнала исключаются тональные и соседние с ними учтенные в блоке 4 компоненты. Затем спектр оставшихся компонент разделяется на полосы частот, заданные таблично и равные критическим полосам слуха.

В каждой из этих полос вычисляется суммарная энергия шумоподобных компонент $\{F_{nm}(k)\}_{nk}$, где

$$F_{nm}(k) = 10 \lg \sum_{\forall i} 10^{0.1F(i)}, \quad i \notin \{k-1, k, k+1\} \quad (15.39)$$

Далее все шумоподобные компоненты внутри i -й критической полосы слуха замещаются одной компонентой равной энергии $F_{nm}(k_i)$, расположенной в центре соответствующей критической полосы слуха.

Разделение спектра исходного сигнала на тональные и шумоподобные составляющие имеет важное значение, поскольку значения коэффициентов маскировки для них имеют различные величины.

В общем случае для оценки коэффициентов маскировки $K_M(i)$ внутри критической полосы слуха используется понятие индекса тональности α и соотношение:

$$K_M(i) = -[\alpha \cdot (14,5 + i) + 5,5 \cdot (1,0 - \alpha)] \text{ дБ}, \quad (15.40)$$

где значение $i = 1, \dots, 24$ Барк.

Для чистого тона индекс $\alpha = 1$ и значение коэффициента маскировки меняется от $K_{Mt}(1) = -14,5$ дБ для первой критической полосы до $K_{Mt}(24) = -38,5$ дБ при $i = 24$ (Tone masking Noise). Для шумоподобного сигнала индекс $\alpha \approx 0$ и значение коэффициента маскировки (Noise masking Tone) приблизительно равно $K_{Mn}(i) = -5,5$ дБ и почти не зависит от его положения по шкале Барков.

Маскировка вне критической полосы слуха (Extra-bend-masking) одинакова и для тональных, и для шумовых компонент; она оценивается с помощью индивидуальных кривых маскировки, учитывающих избирательные свойства слухового анализатора и взаимное маскирующее действие соседних спектральных компонент. Для построения этих кривых спектры тональных и нетональных компонент предварительно прореживаются.

Такое прореживание осуществляется в **блоке 6**.

Исходно исключаются все тональные $F_{tm}(k)$ и шумоподобные $F_{nm}(k)$ компоненты, лежащие ниже абсолютного порога слышимости. Кроме того, тональные компоненты дополнительно прореживаются с помощью окна шириной, например, равной 0,5 Барка. В других психоакустических моделях ширина окна выбирается иной. Если в окно попало две тональные компоненты, то меньшая по уровню из них исключается.

После прореживания формируется новая сетка спектральных компонент:

- в первых трех субполосах (0...2250 Гц) учитываются все спектральные компоненты;
- в 3...6 субполосах (2250...4500 Гц) — каждая вторая компонента;
- в 7...9 субполосах (4500...6750 Гц) — каждая четвертая компонента;
- в оставшихся 20 субполосах (6750...22500 Гц) — лишь каждая восьмая спектральная составляющая.

$$F_{tm/p}(i) = F_{tm/p}(k), \quad F_{tm/p}(k) = 0 \text{ при}$$

$$i = \begin{cases} k, & 1 \leq k \leq 48; \\ k + (k \bmod 2), & 49 \leq k \leq 96; \\ k + 3 - [(k-1) \bmod 4], & 97 \leq k \leq 232; \\ k + 3 - [(k-1) \bmod 8], & 233 \leq k \leq 512. \end{cases} \quad (15.41)$$

Таким образом, если исходный спектр содержал 512 спектральных компонент, то после такого прореживания в общей сложности остается 126 компонент.

В **7-м блоке** производится расчет коэффициентов маскировки и индивидуальных кривых маскировки для тональных и шумоподобных компонент спектра выборки аудиосигнала.

Коэффициенты маскировки тональных $K_{M1}[z(i)]$ и шумоподобных $K_{M2}[z(i)]$ компонент рассчитываются по формулам:

$$K_{M1}[z(i)] = -0,275z(i) - 6,025 \text{ дБ}, \quad (15.42)$$

$$K_{M2}[z(i)] = -0,175z(i) - 2,025 \text{ дБ}. \quad (15.43)$$

Индивидуальные кривые маскировки для каждой из компонент определяются в дБ соотношением:

$$M[z(i), z(j)] = \begin{cases} 17 \cdot [\Delta z(i, j) + 1] - \{0, 4F[z(i)] + 6\}, & -3 \leq \Delta z(i, j) < -1; \\ \{0, 4F[z(i)] + 6\} \cdot \Delta z(i, j), & -1 \leq \Delta z(i, j) < 0; \\ -17 \cdot \Delta z(i, j), & 0 \leq \Delta z(i, j) < 1; \\ -[\Delta z(i, j) - 1] \cdot \{17 - 0, 15F[z(i)]\} - 17, & 1 \leq \Delta z(i, j) < 8; \end{cases} \quad (15.44)$$

где $z(i)$ — высота тона маскирующей (тональной или шумоподобной) компоненты в Барках; $z(j)$ — высота тона маскируемой компоненты; $\Delta z(i) = z(i) - z(j)$ — разность высот тона маскирующей и маскируемой компонент.

На рис. 15.23 приведено семейство кривых маскировки, вычисленное по формуле (15.44).

В **блоке 8** производится расчет порогов маскировки для тональных и шумоподобных компонент спектра выборки аудиосигнала. При этом применяются следующие соотношения (для тональных и шумоподобных порогов соответственно):

$$N_{tm}[z(i)] = F_{tm}[z(i)] + K_{M1}[z(i)] + M[z(i), z(j)] \text{ дБ}; \quad (15.45)$$

$$N_{nm}[z(i)] = F_{nm}[z(i)] + K_{M2}[z(i)] + M[z(i), z(j)] \text{ дБ}. \quad (15.46)$$

В последнем **блоке 9** рассчитываются: кривая глобального порога маскировки выборки сигнала $N_{\text{ПМ}}(i)$ (путем суммирования порогов маскировки тональных и шумоподобных компонент); минимальное значение порога маскировки в каждой n -й субполосе кодирования $N_{\text{ПМ}}(n)$; отношение сигнал/маска $SMR(n)$ для каждой из субполос кодирования:

$$N_{\text{ПМ}}(i) = 10 \lg \left[10^{N_{\text{APC}}[z(i)]/10} + \sum_{\forall i, j} 10^{N_{tm}[z(i), z(j)]/10} + \sum_{\forall i, j} 10^{N_{nm}[z(i), z(j)]/10} \right] \text{ дБ}, \quad (15.47)$$

где $N_{\text{APC}}[z(i)]$ — значение абсолютного порога слышимости спектральной ком-

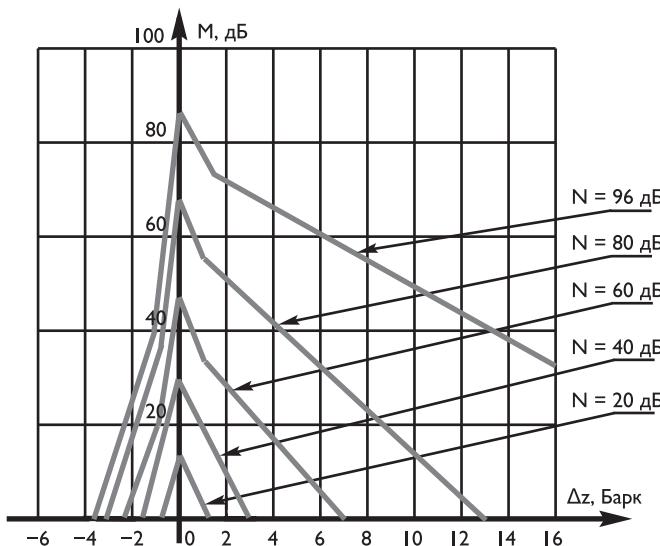


Рис. 15.23. Семейство индивидуальных кривых маскировки, соответствующих психоакустической модели 1

поненты с высотой тона $z(i)$:

$$N_{\text{ПМ}}(n) = [N_{\text{ПМ}}(i)]_{\min} \text{ дБ}; \quad (15.48)$$

$$'SMR(n) = E_{sb}(n) - N_{\text{ПМ}}(n) \text{ дБ}. \quad (15.49)$$

Еще раз следует указать, что величина $SMR(n)$ представляет собой выраженное в дБ отношение энергии полезного сигнала к максимально допустимому значению энергии искажений квантования в каждой n -й субполосе кодирования, при котором они еще маскируются полезным сигналом.

Психоакустическая модель 2

Эта модель содержит одиннадцать блоков обработки первичного ИКМ сигнала. Характерной особенностью этой психоакустической модели является разделение спектра выборки аудиосигнала на полосы психоакустического анализа, в которых осуществляются соответствующие вычисления; количество полос анализа зависит от используемой частоты дискретизации: при $f_d = 48$ кГц используется 62 полосы, при $f_d = 44,1$ кГц — 63 полосы и при $f_d = 32$ кГц — 59 полос.

В блоке 1 осуществляется расчет спектра выборки аудиосигнала:

$$\begin{aligned} F(k) &= \frac{1}{N} \sum_{n=0}^{N-1} h(n) \cdot X(n) \cdot \exp \left(-2\pi j k \frac{n}{N} \right), \\ R(k) &= |F(k)| = \sqrt{Re^2[F(k)] + Im^2[F(k)]}, \\ \varphi(k) &= \arctg \frac{Im[F(k)]}{Re[F(k)]}, \end{aligned} \quad (15.50)$$

где $X(n)$ — отсчеты сигнала; N — длина выборки БПФ (обычно 1024 или 256); $k = 0, 1, \dots, N-1$; $h(n)$ — окно Ханна.

Блок 2 реализует вычисление предсказанных значений амплитуды и фазы $\hat{R}(k)$, $\hat{\varphi}(k)$ для каждой спектральной компоненты сигнала текущей выборки по

массивам значений модулей и фаз спектральных составляющих двух блоков $t - 1$ и $t - 2$, предшествующих текущему блоку t :

$$\hat{R}(k) = 2 \cdot R_{t-1}(k) - R_{t-2}(k), \quad \hat{\varphi}(k) = 2 \cdot \varphi_{t-1}(k) - \varphi_{t-2}(k). \quad (15.51)$$

В **блоке 3** производится расчет меры непредсказуемости $C(k)$ спектральных компонент текущей выборки аудиосигнала.

При этом сначала определяются величины:

$$c(k) = \frac{\sqrt{R^2(k) + \hat{R}^2(k) - 2R(k)\hat{R}(k)\cos[\varphi(k) - \hat{\varphi}(k)]}}{R(k) + \hat{R}(k)}, \quad (15.52)$$

а затем рассчитывается мера непредсказуемости —

$$C(k) = \begin{cases} c(k), & 0 \leq k < 6; \\ c[(k+2)DIV4], & 6 \leq k < 206; \\ 0.4, & k \geq 206; \end{cases} \quad (15.53)$$

где знак DIV означает целочисленное деление с округлением к меньшей величине.

Блок 4 предназначен для вычисления энергии сигнала $E(m)$ и взвешенного значения непредсказуемости в полосах психоакустического анализа $c(m)$:

$$E(m) = \sum_{k=klow(m)}^{khig(m)} R^2(k), \quad c(m) = \sum_{k=klow(m)}^{khig(m)} R^2(k) \cdot C(k), \quad (15.54)$$

где $klow(m)$ и $khig(m)$ — соответственно нижняя и верхняя границы полос психоакустического анализа.

В **блоке 5** производится свертка энергии сигнала и взвешенного значения меры непредсказуемости с развертывающей функцией $M(i, j)$, представляющая собой индивидуальную кривую маскировки, учитывающую избирательные свойства уха человека:

$$EC(m) = \sum_{i,j \in m} E(i) \cdot M(i, j), \quad CT(m) = \sum_{i,j \in m} C(i) \cdot M(i, j), \quad (15.55)$$

где $M(i, j) = \begin{cases} 0, & ty < -100; \\ 10^{(tz+ty)/10}, & y \geq -100; \end{cases}$

$$tz = 8 \cdot \min \{(tx - 0,5)^2 - 2(tx - 0,5), 0\};$$

$$ty = 15,811389 + 7,5 \cdot (tx + 0,484) - 17,5 \cdot [1,0 + (tx + 0,474)^2]^{1/2};$$

$$tx = \begin{cases} 3,0 \cdot (j - i), & j \geq i; \\ 1,5 \cdot (j - i), & j < i; \end{cases}$$

i и j — соответственно значение высоты тона для развертываемого сигнала и значение высоты тона сигнала, который развертывается в полосу анализа i в Барках. Семейство рассчитанных развертывающих функций представлено на рис. 15.24.

Блок 6 реализует вычисление коэффициента хаоса и индекса тональности в полосах психоакустического анализа соответственно по формулам:

$$CN(m) = \frac{CT(m)}{EC(m)}; \quad \alpha(m) = -0,299 - 0,43 \ln[CN(m)]. \quad (15.56)$$

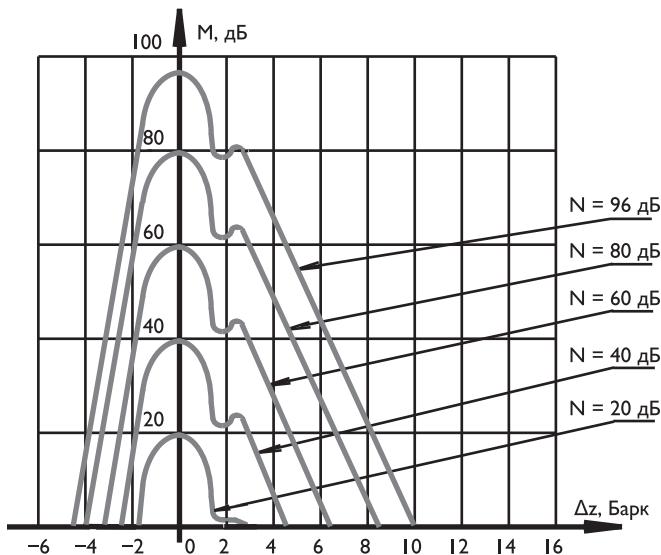


Рис. 15.24. Семейство индивидуальных кривых маскировки при психоакустической модели 2

Величина индекса тональности $\alpha(m)$ ограничивается пределами $0 \leq \alpha(m) \leq 1$. Если расчет дает значение больше единицы, индекс принимается равным 1; при отрицательной расчетной величине он устанавливается равным 0.

В блоке 7 производится расчет отношения сигнал/шум в каждой полосе психоакустического анализа:

$$SNR(m) = \max[\min val(m), SNR * (m)] \text{ дБ}, \quad BC(m) = 10^{-SNR(m)/10}, \quad (15.57)$$

где $SNR * (m) = \alpha(m) \cdot TMN(m) + [1 - \alpha(m)] \cdot NMT(m)$ дБ; $TMN(m)$ — коэффициент маскировки шума тоном, определяемый разностью уровней тона и шума; для разных стандартов эта величина для всех полос анализа принята равной 29 и 18 дБ; $NMT(m)$ — коэффициент маскировки тона шумом, определяемый разностью уровней шума и тона; для всех полос анализа эта разность принята равной 6 дБ.

Очевидно, чем меньше величина $SNR * (m)$, тем большим может быть допустимый уровень шума в полосе анализа. Нижней границей $SNR(m)$ служит табличная величина $minval(m)$ — поправочный коэффициент, отличающийся от нуля лишь на самых низких частотах.

В блоке 8 осуществляется расчет максимально допустимой энергии шума $ThR(m)$ — глобального порога маскировки, приходящейся на один коэффициент модифицированного дискретного косинусного преобразования (МДКП) в каждой полосе психоакустического анализа, при которой он еще маскируется полезным сигналом.

Используется два варианта такого расчета. В первой модели используется соотношение:

$$ThR(m) = \max [Nb(k), aThR(k)], \quad (15.58)$$

где $Nb(k) = \frac{Nb(m)}{khig(m) - klow(m) + 1}$; $Nb(m) = EN(m) \cdot BC(m)$; $Nb(m)$ — максимальное значение энергии шума в полосе анализа, при котором он маскируется полезным сигналом, $Nb(k)$ — то же самое, приходящееся на один элемент МДКП, $aThR(k)$ — значение абсолютного порога слышимости для спектральной компоненты с индексом k ; $khig(m)$ и $klow(m)$ — соответственно верхняя и нижняя границы блока психоакустического анализа; $EN(m)$ — энергия шума в полосе блока психоакустического анализа.

Во второй модели расчета определяется глобальный порог маскировки соответственно для длинных блоков ($N = 1024$):

$$\begin{aligned} ThR(m) &= \max\{aThR(m), \min[Nb(m), Nb_{t-1}(m), Nb_{t-2}(m)]\}; \\ Nb_{t-1}(m) &= 2 \cdot Nb(m); \\ Nb_{t-2}(m) &= 16 \cdot Nb(m); \end{aligned} \quad (15.59)$$

и для коротких блоков ($N = 256$):

$$ThR(m) = \max[aNhR(m), Nb(m)]. \quad (15.60)$$

Следующий **9-й блок** расчета обеспечивает вычисление допустимой энергии шума (порога маскировки) в полосах кодирования:

$$ThR(n) = \begin{cases} \sum_{k=klow(n)}^{khig(n)} ThR(k), & \alpha(n) = 1; \\ \min[ThR(k), klow(n) \leq k \leq khig(n)] \times \\ \times [khig(n) - klow(n) + 1], & \alpha(n) = 0; \end{cases} \quad (15.61)$$

где $khig(n)$ и $klow(n)$ — соответственно верхняя и нижняя границы полосы кодирования.

В **блоке 10** производится расчет энергии аудиосигнала в полосах кодирования:

$$E(n) = \sum_{k=klow(n)}^{khig(n)} R^2(k). \quad (15.62)$$

И, наконец, в **11-ом блоке** производится расчет отношения сигнал/маска $SNR(n)$ в полосах кодирования:

$$SMR(n) = 10 \cdot \lg \frac{E(n)}{ThR(n)}. \quad (15.63)$$

Иногда вычисления энергии сигнала и порога маскировки вычисляются по иным формулам, чем указанные в блоках 9 и 10:

$$\begin{aligned} E(n) &= w_1 E_u + \sum_{k=klow(n)+1}^{khig(n)-1} E(k) + w_2 E_o; \\ ThR(n) &= w_1 ThR_u + \sum_{k=klow(n)+1}^{khig(n)-1} ThR(k) + w_2 ThR_o; \end{aligned} \quad (15.64)$$

где $w_1, w_2, E_u, E_o, R_u, R_o$ — табличные данные, определенные для разных частот дискретизации сигнала.

Психоакустическая модель 3

В данной модели предварительно **выборка исходного ИКМ аудиосигнала преобразуется** с использованием модифицированного дискретного косинусного преобразования (МДКП):

$$F_D(k) = -\frac{2}{N} \sum_{n=0}^{N-1} X(n) \cos \frac{\pi}{4} (2k+1) \left(\frac{2n+1}{2N} + 1 + \alpha \right), \quad 0 \leq k < \frac{N}{2}, \quad (15.65)$$

где $N = 512$ или 256 ; $\alpha = 0$ при $N = 512$; $\alpha = -1$ и $\alpha = +1$ соответственно для первой и второй последовательных выборок при $N = 256$ для каждой из них.

Каждый коэффициент МДКП представляется в формате $F_D(k) = A(k) \cdot 2^{B(k)}$, где $A(k)$ — мантисса, $B(k)$ — экспонента (или порядок) k -го коэффициента преобразования.

Массивом входных данных для блока психоакустической модели являются значения порядков $B(k)$. Используется 50 полос психоакустического анализа и 20 полос кодирования.

Дискретность изменения величины порядка коэффициентов МДКП составляет величину 6 дБ, а наибольшее его значение равно 24, что определяет полный динамический диапазон сигнала в $24 \cdot 6 = 144$ дБ.

Шкала величин $B(k)$ предварительно преобразуется с использованием соотношения:

$$PSD(k) = 3072 - 128 \cdot B(k). \quad (15.66)$$

При этом новая шкала содержит 3072 градации; множитель в этой формуле уменьшает дискретность грубой шкалы $B(k)$ до величины шага, равного $6/128 = 0,046875$ дБ, что лежит существенно ниже порога различимости слуха по уровню сигнала. На рис. 15.25 приведена иллюстрация преобразования масштаба порядков интенсивностей коэффициентов МДКП.

Вычисления в психоакустической модели выполняются в так называемых полосах анализа, неодинаковых по ширине. До частоты 2531 Гц ширины полос психоакустического анализа выбраны так, что в каждую из них попадает только один коэффициент МДКП. При частоте дискретизации $f_d = 48$ кГц и длине выборки $N = 512$ отсчетов сигнала ширина каждой из этих полос составляет 93,75 Гц, а число полос на данном интервале равно 28. Далее ширина полос возрастает так, что они включают соответственно 3, 6 (по 6 полос) 12 и 24 (по 5 полос) коэффициента МДКП.

Суммарное значение энергии аудиосигнала в каждой полосе психоакустического анализа вычисляется по формуле:

$$\log(a+b) = \max[\log(a), \log(b)] + \log[1 + \exp(d)], \quad (15.67)$$

где $\log(a)$, $\log(b)$ — значения порядков соседних (в пределах каждой полосы психоакустического анализа) коэффициентов МДКП; $d = |\log(a) - \log(b)|$ — адрес в таблице, в которой вычислены значения величин $\log[1 + \exp(d)]$.

Если число коэффициентов в полосе анализа больше одного, предусматривается следующий механизм вычислений:

- сначала берутся значения порядков первых двух коэффициентов МДКП, и определяется максимальное значение; к нему добавляется величина, равная разности этих двух величин;

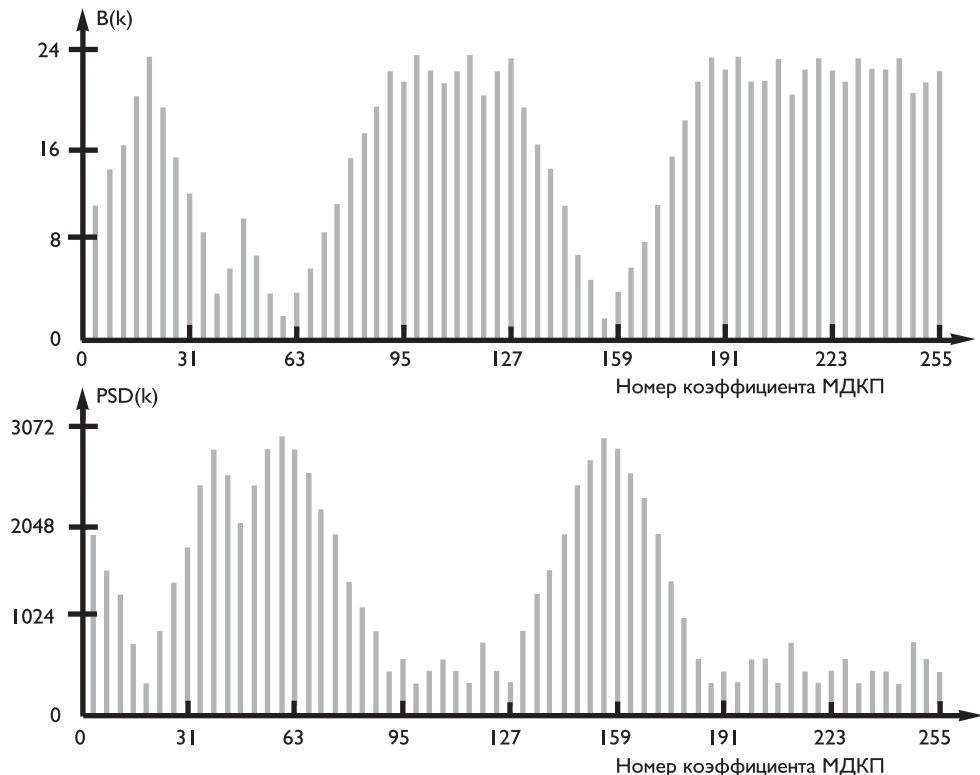


Рис. 15.25. Пример преобразования масштаба коэффициентов МДКП

- затем полученное число сравнивается со значением порядка следующего по номеру индекса коэффициента МДКП, и производится вновь указанная выше операция;
- процесс вычислений повторяется до тех пор, пока не будут использованы все коэффициенты МДКП в данной полосе анализа.

Формирование прототипа индивидуальной кривой маскировки осуществляется в частотной области. В основе выбора прототипа кривой маскировки лежат экспериментальные данные, изложенные, например, в [6.22, 6.23].

Параметром каждой кривой является абсолютный акустический уровень маскирующего тона, вычисленный относительно звукового давления $p_0 = 2 \cdot 10^{-5}$ Па и равный соответственно 40, 60, 80 и 100 дБ. Для каждого уровня маскирующего тона из экспериментальных данных определялся относительный порог слышимости шума, имеющего полосу частот, равную полосе частот психоакустического анализа, т. е. рассматривалась маскировка вне критической полосы слуха, при которой тон маскирует шум с полосой частот, примерно равной 0,5 Барка. Затем каждая полученная таким путем зависимость нормировалась к уровню маскирующего тона в соответствии с соотношением:

$$N_{RNT}(f) = N_{NT}(f) + N_{MT}(f) \text{ дБ}, \quad (15.68)$$

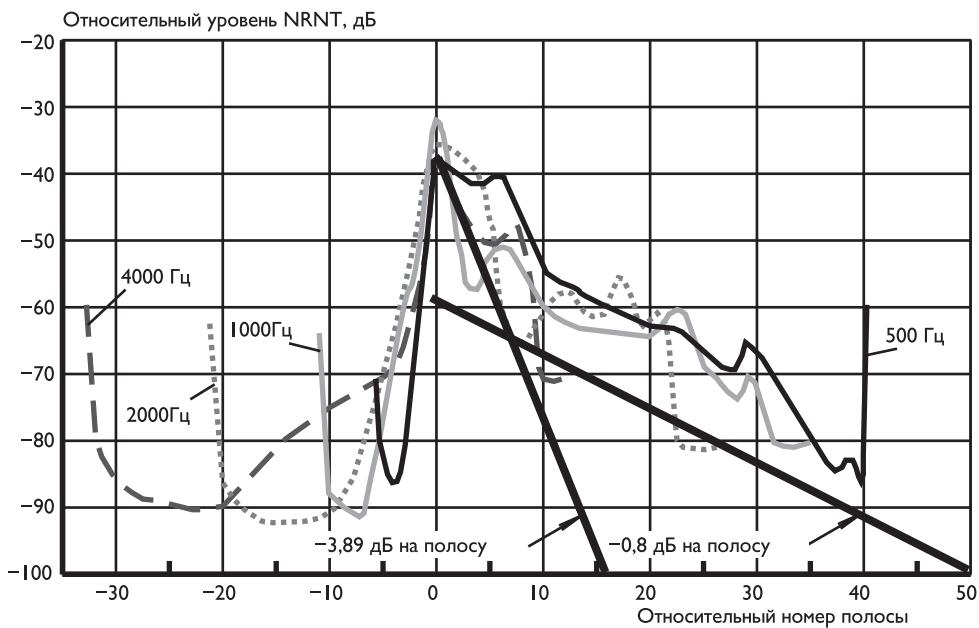


Рис. 15.26. К построению обобщенных кривых маскировки

где $NRNT(f)$ — нормированное по отношению к уровню маскирующего тона значение порога слышимости шума, $NNT(f)$ — порог слышимости шума, маскируемого тоном, $NMT(f)$ — уровень маскируемого тона.

Эти вычисления выполнялись для каждого значения частоты и уровня маскирующего тона. Совокупность полученных кривых преобразована в так называемые обобщенные кривые маскировки, представленные в качестве примера на рис. 15.26. По оси абсцисс на этом графике отложены значения высот тона в делениях, следующих через 0,5 Барка, что соответствует расстоянию между центрами соседних полос психоакустического анализа, а нуль этой шкалы соответствует высоте маскирующего тона в Барках. Параметром каждой такой обобщенной кривой является частота маскирующего тона.

Представленные различными цветами кривые являются развертывающими функциями или индивидуальными кривыми маскировки. Аппроксимация обобщенных кривых маскировки весьма сложна и по этой причине используются упрощения. При этом маскировка в сторону низких частот не учитывается.

Маскировка в сторону верхних частот с точностью, достаточной для практического использования, аппроксимируется двумя отрезками прямых линий (жирные линии на рис. 15.26):

- быстро затухающая прямая (Fast Upwards Masking) и
- медленно затухающая прямая (Slow Upwards Masking).

Формирование обобщенной кривой маскировки осуществляется с использованием четырех параметров, которые формируются кодером и передаются декодеру в поле данных психоакустической модели.

В состав этих параметров входят:

- крутизна наклона медленно затухающего сегмента (Slow Decay), определяемая величиной от $-0,7$ до $-0,98$ дБ на полосу анализа;
- вертикальное смещение t медленно затухающего сегмента (Slow Gain) от уровня маскирующей компоненты сигнала в пределах от -49 до -63 дБ;
- крутизна наклона быстро затухающего сегмента (Fast Decay), определяемая величиной от $-2,95$ до $-5,77$ дБ на полосу анализа;
- вертикальное смещение быстро затухающего сегмента (Fast Gain) от максимального уровня маскирующей компоненты сигнала в пределах от -6 до -48 дБ.

При этом синтез обобщенной кривой маскировки осуществляется с использованием включенных параллельно двух рекурсивных фильтров. Результирующее значение глобального порога маскировки $E_M(k)$ в k -й полосе психоакустического анализа определяется как наибольшее значение из выходных отсчетов этих двух фильтров:

$$\begin{aligned} x_0 &= [x_0(k) - d_0(k)] \oplus [E_C(k) - g_0(k)], \\ x_1 &= [x_0(k) - d_1(k)] \oplus [E_C(k) - g_1(k)], \\ E_M(k) &= \max(x_0, x_1), \end{aligned} \quad (15.69)$$

где $E_C(k)$ — энергия аудиосигнала в k -й полосе психоакустического анализа; $d_0(k)$ и $d_1(k)$ — крутизна наклона соответственно быстро и медленно затухающих сегментов обобщенной кривой маскировки; $g_0(k)$ и $g_1(k)$ — вертикальное смещение сегментов от максимального уровня k -й спектральной компоненты сигнала соответственно для быстро и медленно затухающего сегментов.

Далее полученные значения $E_M(k)$ корректируются с целью учета влияния маскирующего сигнала на величину порога маскировки. После коррекции значения глобального порога маскировки в каждой полосе психоакустического анализа сравниваются с величиной абсолютного порога слышимости и выбирается наибольшее из этих двух значений.

В результате выполнения этих операций формируется результирующая кривая маскировки, определяющая допустимые значения энергии шумов квантования в каждой полосе анализа.

Минимально допустимое отношение сигнал/шум для каждой полосы психоакустического анализа SNR_k вычисляется в дБ как разность величин энергий полезного сигнала и шумов квантования, находящихся на пороге слышимости. При расчете энергии полезного сигнала используются только значения порядков МДКП.

Величина $SNR(k)$, приведенная к одному члену МДКП в k -ой полосе психоакустического анализа вычисляется как

$$SNR(k) = SNR_k/n, \quad (15.70)$$

где n — число коэффициентов МДКП в данной полосе.

Массив данных $SNR(k)$ образует *кривую глобального порога маскировки*, которая определяет число битов, выделяемых на кодирование мантисс коэффициентов МДКП.

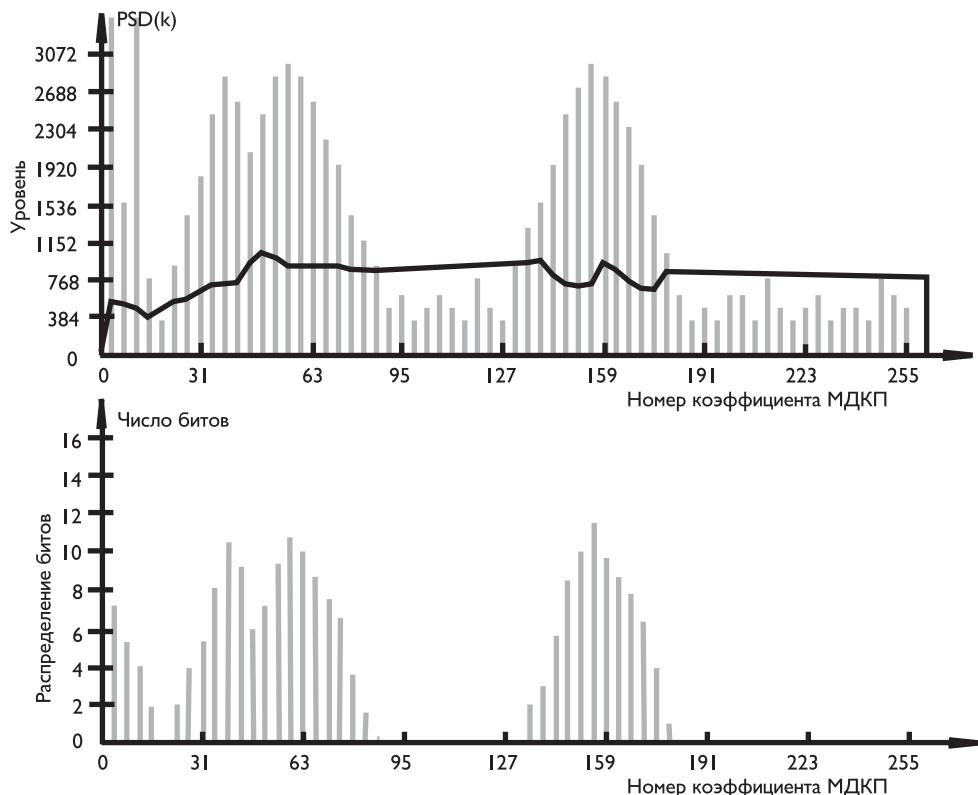


Рис. 15.27. Пример выделения битов для кодирования коэффициентов МДКП

На рис. 15.27 в качестве примера вверху приведена кривая ступенчатой формы. Коэффициенты МДКП, расположенные ниже этой кривой, не кодируются, поскольку лежат ниже относительного порога слышимости. В нижней части рис. 15.27 приведено число битов, которое требуется выделить для кодирования мантисс каждого из тех коэффициентов МДКП, которые необходимо передать в декодер.

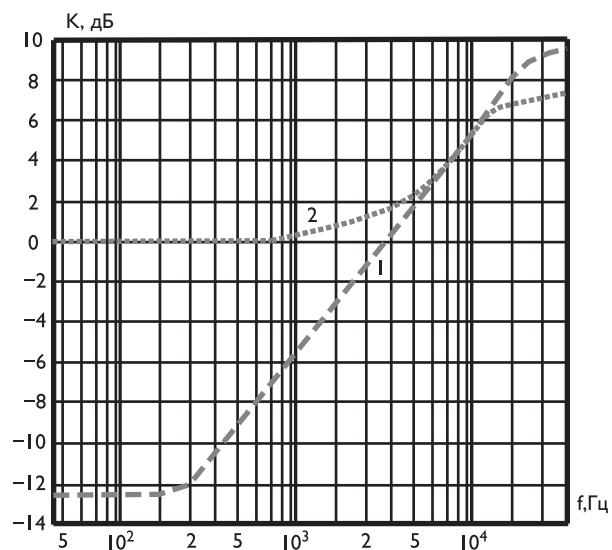
Предусмотрено «грубое» и «плавное» смещение кривой глобального порога маскировки соответственно $CSNR$ и $FSNR$. При этом шаг смещения кривой $FSNR$ установлен равным $3/16$ дБ, а кривой $CSNR$ — 3 дБ. Суммарная величина смещения глобального порога маскировки относительно ее исходного положения определяется формулой:

$$SNROFFSET = 4 \cdot [16 \cdot (CSNR - 15) + FSNR]. \quad (15.71)$$

15.8. Предыскажения звуковых сигналов при цифровом кодировании

При ИКМ1 преобразовании звукового сигнала с малым шагом квантования спектр шума имеет равномерный характер вплоть до половины частоты дискретизации $f_{\text{пд}}/2$, в то время как уровень спектральных составляющих сигнала уменьшается при увеличении их частоты. По этой причине для уменьшения влияния шумов квантования применяются частотные предыскажения на стороне кодера и восстановление сигнала при его декодировании (см. рис. 15.28).

Рис. 15.28. Характеристики цепей предыскажений



В соответствии с Рекомендацией 651 МККР [6.57] при линейном кодировании с 14-битовым разрешением, а также при нелинейном кодировании с почти мгновенным компандированием для уменьшения разрядности с 14 до 10 битов на отсчет звуковых сигналов с полосой спектра до 15 кГц предыскажения должны соответствовать Рекомендации J.17 МККТТ [6.58] с вносимым затуханием 6,5 дБ на частоте 0,8 кГц — рис. 15.28, кривая 1, соотношение (15.72), или характеристике 50/15 мкс — рис. 15.28, кривая 2, соотношение (15.73).

$$K(f) = 10 \lg \frac{1 + (2\pi f / 3000)^2}{75 + (2\pi f / 3000)^2}, \quad (15.72)$$

$$K(f) = 10 \lg \frac{1 + (0,05 \cdot 2\pi f)^2}{1 + (0,015 \cdot 2\pi f)^2}, \quad (15.73)$$

где f — частота в Гц.

В декодере эти частотные предыскажения компенсируются. Ввиду того что спектр шума квантования обрабатывается только восстанавливающей цепью, отношение сигнал/шум увеличивается примерно на 4 дБ.

ГЛАВА 16

ОСНОВНЫЕ СТАНДАРТЫ КОДИРОВАНИЯ РЕЧЕВОЙ ИНФОРМАЦИИ

Алгоритмы сжатия речи весьма гибки, используют различные методы обработки информации, что может привести к существенному различию в качестве декодируемой информации при одной и той же степени ее сжатия. Информация об основных стандартах кодирования речевой информации приведена в табл. 16.1.

Таблица 16.1. Основные параметры кодеков речи

Наименование стандарта	Тип кодека	Скорость передачи информации, кбитов/с	Оценка качества MOS	Год публикации
ITU-T G.711	ИКМ	64	4,12	1972
ITU-T G.722	АДИКМ	48; 56; 64	3,8–4,2	1993
ITU-T G.722.1	Siren7	16; 24; 32	3,8–4,2	1999
ITU-T G.722.2	AMR-WB	6,6–23,85	3,8–4,2	2002
ITU-T G.723.1	MP-MLQ	6,3	3,9	1996
ITU-T G.723.1	ACELP	5,3	3,65	1996
ITU-T G.726	АДИКМ	16–40	2–4,3	1984
ITU-T G.727	АДИКМ	40; 32; 24; 16	2,2–4,3	1986
ITU-T G.728	LD-CELP	16	3,61	1992
ITU-T G.729	CS-ACELP (без VAD)	8	3,92	1997
ITU-T G.729	двукратное кодирование	8	3,27	1997
ITU-T G.729	трехкратное кодирование	8	2,67	1997
ITU-T G.729a	CA-ACELP	8	3,7	1999
ETSI GSM FR	RPE-LTP / MPE-LTP	13	3,58	1992
ETSI GSM EFR	RPE-LTP / MPE-LTP	12,2	3,8	1999
iLBC	LPC-LSF-CD	13,33	3,7	1998
iLBC	LPC-LSF-CD	15,2	3,9	1999
Speex NB	CELP-VBR-VAD	2,15 – 24,6	2 – 4,4	2004
Speex WB	CELP-VBR-VAD	4 – 44,2	2,8 – 4,4	2004

16.1. G.711 — стандарт кодирования речи с применением ИКМ

G.711 — один из основных стандартов ITU-T для аудиокодеков. Он является частью более общих мультимедийных стандартов, используется в телефонии, определяет преобразование аудиосигнала с шириной полосы 3,4 кГц для передачи по информационным каналам со скоростью 64 кбитов/с.

Рекомендация предусматривает импульсно-кодовую модуляцию (ИКМ, Pulse Code Modulation — PCM) в диапазоне частот аналогового голосового сигнала [6.24]. Номинальная частота дискретизации сигнала — 8000 Гц, каждый отсчет представляется 8 битами.

Предлагаются два правила кодирования — A-law и μ -law, определяемые соотношениями (15.33 и 15.34) и заданные в Рекомендации таблично. По международным цифровым каналам связи между странами, в которых приняты различные правила кодирования, должны передаваться сигналы, закодированные в соответствии с правилом A-law. Если обе страны используют одно правило кодирования, то для передачи должно использоваться это правило кодирования. Все необходимые преобразования должны осуществляться странами, использующими правило μ -law. При последовательной передаче цифровых данных первым передается наиболее значимый бит.

Алгоритмы преобразования аудиосигнала стандарта G.711 рекомендованы для большинства систем цифровой передачи речи в качестве метода предварительного аналого-цифрового преобразования. Кодек поддерживается практически всеми устройствами IP-телефонии, не подлежит лицензированию, требует использования весьма скромных вычислительных ресурсов и обеспечивает хорошее качество, правда создавая при этом довольно высокую нагрузку на каналы передачи.

В 2008 г. ITU-T был принят стандарт G.711.1 [6.25], являющийся расширением стандарта G.711 и совместимый с ним при скорости цифрового потока 64 кбитов/с. Кодер G.711.1 может преобразовывать аудиосигналы со спектром 50–7000 Гц при частоте дискретизации 16 кГц с формированием цифрового потока 80 или 96 кбитов/с [6.25].

В 2009 г. ITU-T была принята рекомендация G.711.0 [6.26] безыскаженной дополнительной компрессии сигналов цифрового потока кодера G.711.

16.2. G.722 — кодирование речи со спектром до 7 кГц в цифровой поток 64 кбитов/с

Рекомендация определяет характеристики системы кодирования звука (50 Гц–7 кГц) с использованием аддитивной дифференциальной импульсно-кодовой модуляции (АДИКМ) при скорости выходного потока до 64 кбита/с. В методике обработки речевой информации (SB-ADPCM) частотная полоса делится на два поддиапазона (верхний и нижний), и сигнал в каждом поддиапазоне кодируется с помощью АДИКМ (ADPCM). Система имеет три основных режима работы, соответствующих скоростям выходного битового потока 64, 56 и 48 кбитов/с. По-

следние два режима позволяют передавать в канале с пропускной способностью 64 кбита/с дополнительные данные 8 или 16 кбитов/с соответственно [6.27].

В состав системы входит звуковая часть, которая преобразует звуковой сигнал в равномерно квантованный цифровой 14-битный сигнал с частотой дискретизации 16 кГц, и SB-ADPCM кодер, который уменьшает цифровой поток до скорости 64 кбита/с. Для приложений, использующих дополнительный канал данных, требуется устройство ввода этих данных на передающей стороне, которое, при необходимости, вводит 1 или 2 бита на октет в зависимости от режима работы, и устройство извлечения данных на приемной стороне, извлекающее дополнительные данные. В Рекомендации описаны три варианта декодера SB-ADPCM, оптимизированных для трех различных режимов работы. Когда режим работы не указывается, необходимо использовать режим 64 кбита/с.

Несовпадение режимов работы кодера и декодера не приводит к неправильной работе декодера, однако для получения максимального качества звука необходимо минимизировать время несовпадения режимов работы. Для совместимости различных типов звуковых терминалов рекомендуется поддерживать как минимум режим 64 кбита/с.

В состав кодера SB-ADPCM входят нижеследующие элементы.

- Два квадратурно-зеркальных нерекурсивных фильтра (QFM) с линейной фазовой характеристикой, которые делят частотный диапазон 0–8000 Гц на два поддиапазона: нижний (0–4000 Гц) и верхний (4000–8000 Гц). Частота дискретизации сигнала f_{IN} на входе QFM – 16 кГц, а на выходах фильтров f_L и f_H – 8 кГц.
- Кодер ADPCM нижнего поддиапазона, формирующий оценку S_L входного сигнала X_L , обеспечивает вычисление разностного сигнала E_L . Адаптивный 60-уровневый нелинейный квантователь используется для формирования 6-битного сигнала I_L с потоком 48 кбитов/с; в петле обратной связи два наименее значащих бита I_L удаляются, что дает 4-битный сигнал I_{Lt} , который используется для адаптации квантователя и для получения квантованного дифференциального сигнала D_{Lt} , получаемого после 15-уровневого адаптивного обратного квантования. Оценка сигнала S_L добавляется к этому сигналу, что дает восстановленное значение R_{Lt} входного сигнала нижнего поддиапазона. Работа с 4 битами вместо 6 в петле обратной связи как кодера, так и декодера ADPCM нижнего поддиапазона дает возможность введения дополнительных данных без нарушения работы декодера.
- Кодер ADPCM верхнего поддиапазона формирует оценку S_H сигнала X_H и вычисляет разностный сигнал E_H . Адаптивный 4-уровневый нелинейный квантователь используется для формирования 2-битного сигнала I_H с потоком 16 кбитов/с. Обратное адаптивное квантование создает дифференциальный сигнал D_H . Оценка сигнала S_H добавляется к этому дифференциальному сигналу, что дает восстановленное значение R входного сигнала верхнего поддиапазона.
- Мультиплексор, объединяющий сигналы I_H и I_L кодеров верхнего и нижнего поддиапазонов и формирующий полный сигнал I с потоком 64 кбита/с в виде октетов следующего формата:

$$I_{H1}I_{H2}I_{L1}I_{L2}I_{L3}I_{L4}I_{L5}I_{L6}.$$

В состав декодера SB-ADPCM входят нижеследующие элементы.

- Демультиплексор, разделяющий состоящий из октетов I_R сигнал с потоком 64 кбита/с на два сигнала I_L и I_H , которые затем поступают на входы декодеров ADPCM нижнего и верхнего поддиапазонов.
- Декодер ADPCM нижнего поддиапазона, работающий в одном из трех режимов. Часть, определяющая оценку сигнала S_L , включая адаптацию квантования, аналогична петле обратной связи кодера ADPCM нижнего поддиапазона. Восстановленный сигнал R_L получается при суммировании оценки сигнала с одним из трех квантованных разностных сигналов D_{L6} , D_{L5} или D_{L4} в зависимости от режима работы.
- Декодер ADPCM верхнего поддиапазона, аналогичный петле обратной связи кодера ADPCM верхнего поддиапазона, формирует восстановленный сигнал R_H .
- Квадратурные зеркальные нерекурсивные фильтры (QFM) с линейной фазовой характеристикой интерполируют сигналы R_L и R_H с частотами дискретизации 8 кГц до частоты 16 кГц и формируют выходной сигнал X_{OUT} .

Более новая версия кодека G.722 — кодек **G.722.1** (1999 г.) [6.28]. Он предназначен для сжатия широкополосного аудиосигнала и базируется на третьем поколении технологии сжатия Siren® компании Polysom. Этот стандарт обеспечивает передачу широкополосного аудиосигнала, близкого по качеству к FM-радио. G.722.1 определяет работу кодека на скоростях 24 и 32 кбитов/с при ширине спектра входного сигнала 50 Гц–7 кГц.

Кодек **G.722.1 Annex C** базируется на патентованной технологии Siren 14® компании Polysom. Качество аудиосигнала приближено к CD. Этот алгоритм сжатия обеспечивает сверхширокополосный аудиосигнал 14 кГц при скоростях передачи 24, 32 и 48 кбитов/с.

Часто используемый вариант кодека — кодек **G.722.2** (2002 г.), известный как «Адаптивный многоскоростной–широкополосный» (AMR-WB — Adaptive Multi Rate — WideBand). Он обеспечивает возможность быстрого изменения скорости сжатия при изменении пропускной способности сети передачи данных [6.29]. G.722.2 определяет 9 различных режимов скорости передачи от 6,6 кбитов/с до 23,85 кбитов/с, перечисленные ниже.

В обязательных многоскоростных конфигурациях:

- 6,60 кбитов/с (используется в системах коммутации каналов в GSM- и UMTS-соединениях; должен использоваться только временно в случае плохого радиосоединения и не считается предназначенным для передачи широкополосного речевого сигнала);
- 8,85 кбитов/с (используется в системах коммутации каналов в GSM- и UMTS-соединениях; должен использоваться только временно в случае плохого радиосоединения и не считается предназначенным для передачи широкополосного речевого сигнала; обеспечивает качество, равное G.722 при скорости передачи 48 кбитов/с для качественной речи);

- 12,65 кбитов/с (основная скорость; используется в системах коммутации каналов в GSM- и UMTS-соединениях; предлагает отличное качество аудиосигнала на этой и более высоких скоростях передачи; обеспечивает качество, равное G.722 при скорости передачи 56 кбитов/с для качественной речи).

Более высокие скорости для передачи речи в неблагоприятных условиях с повышенными шумами, для совместной передачи речи и музыки и для конференций со множеством участников:

- 14,25 кбитов/с;
- 15,85 кбитов/с;
- 18,25 кбитов/с;
- 19,85 кбитов/с;
- 23,05 кбитов/с (не предназначен для полноскоростных GSM каналов);
- 23,85 кбитов/с (обеспечивает качество, равное G.722 при скорости передачи 64 кбита/с для качественной речи; не предназначен для полноскоростных GSM каналов).

16.3. G.723.1 — речевой кодек с двумя скоростями потока — 5,3 и 6,3 кбитов/с

Рекомендация описывает кодирование речи или других звуковых компонент мультимедийных служб при низких скоростях передачи данных. Качество кодированного звука выше при скорости 6,3 кбитов/с. Возможно переключение между скоростями передачи на границе любого звукового кадра. Возможно дополнительное использование режимов передачи с паузами и заполнение шумом в интервалах отсутствия речевого сигнала [6.30].

Речь и другие звуковые сигналы обрабатываются кадрами, используется кодирование с линейным предсказанием, анализом и синтезом. Возбуждающий сигнал для высокоскоростного кодера — многоимпульсное квантование с максимумом правдоподобия (MP-MLQ), а для низкоскоростного — линейное предсказание с алгебраическим кодовым возбуждением (ACELP). Размер кадра равен 30 мс, также дополнительно учитывается последующий интервал 7,5 мс. Таким образом, общая задержка, вносимая алгоритмом, — 37,5 мс. Все дополнительные задержки кодера образуются из задержки при обработке (вычислениях), задержки системы передачи и задержки буферизации протоколов мультиплексирования.

Кодер предназначен для работы с цифровым сигналом, полученным после фильтрации в полосе речевого сигнала при дискретизации с частотой 8000 Гц и линейном квантовании с глубиной 16 битов на отсчет. Структурная схема кодера приведена на рис. 16.1.

В кодере, на который подаются отсчеты сигнала $y[n]$, обрабатываются блоки (кадры), по 240 отсчетов каждый (30 мс при частоте дискретизации 8 кГц). Каждый блок $s[n]$ сначала передается через фильтр высоких частот (ФВЧ) для устранения постоянной составляющей, а затем делится на четыре подкадра по 60 отсчетов $x[n]$ каждый.

Z -характеристика ФВЧ определяется соотношением:

$$H(z) = \frac{1 - z^{-1}}{1 - \frac{127}{128}z^{-1}}. \quad (16.1)$$

Для каждого подкадра вычисляется фильтр кодирования с линейным предсказанием (LPC) 10-го порядка, при этом используется необработанный входной сигнал. Его характеристика имеет вид:

$$A_i(z) = \frac{1}{1 - \sum_{j=1}^{10} a_{ij} z^{-j}}, \quad 0 \leq i \leq 3, \quad (16.2)$$

а коэффициенты a_{ij} определяются рекурсивно с применением алгоритма Левинсона–Дурбина (Levinson–Durbin).

Фильтр LPC для последнего подкадра квантуется с помощью разделяющегося векторного квантователя с предсказанием (LSP qantizator). Неквантованные коэффициенты LPC используются для построения короткого взвешивающего фильтра, который применяется для фильтрации всего кадра и для получения взвешенного речевого сигнала.

Для каждого двух подкадров (120 отсчетов) по взвешенному речевому сигналу рассчитывается период шага открытой петли L_{OL} . Оценка шага осуществляется по блокам из 120 отсчетов. Период шага определяется в диапазоне от 18 до 142 отсчетов. Далее речь обрабатывается по подкадрам из 60 отсчетов.

В состав кодера входит эквивалент декодера, содержащий LSP-декодер и LSP-интерpolator. Характеристика фильтра синтеза используется для того, чтобы генерировать декодированный речевой сигнал, и определяется формулой:

$$\tilde{A}_i(z) = \frac{1}{1 - \sum_{j=1}^{10} \tilde{a}_{ij} z^{-j}}, \quad 0 \leq i \leq 3, \quad (16.3)$$

где \tilde{a}_{ij} – коэффициенты, определяемые используемыми алгоритмами интерполяции.

Выделение воспринимаемых формант производится с использованием взвешивающего фильтра:

$$W_i(z) = \frac{1 - \sum_{j=1}^{10} \mathbf{a}_{ij} z^{-j\gamma_1^j}}{1 - \sum_{j=1}^{10} \mathbf{a}_{ij} z^{-j\gamma_2^j}}, \quad 0 \leq i \leq 3, \quad (16.4)$$

где $\gamma_1 = 0,9$; $\gamma_2 = 0,5$.

Определение основного тона осуществляется с использованием периода шага разомкнутого цикла L_{OL} и воспринимаемой взвешенной речи $f[n]$. Оценка периода шага осуществляется с использованием критерия взаимной корреляции согласно следующему выражению:

$$C_{OL}(j) = \frac{\left(\sum_{n=0}^{119} f[n] \cdot f[n-j] \right)^2}{\sum_{n=0}^{119} f[n-j] \cdot f[n-j]}, \quad 18 \leq j \leq 142. \quad (16.5)$$

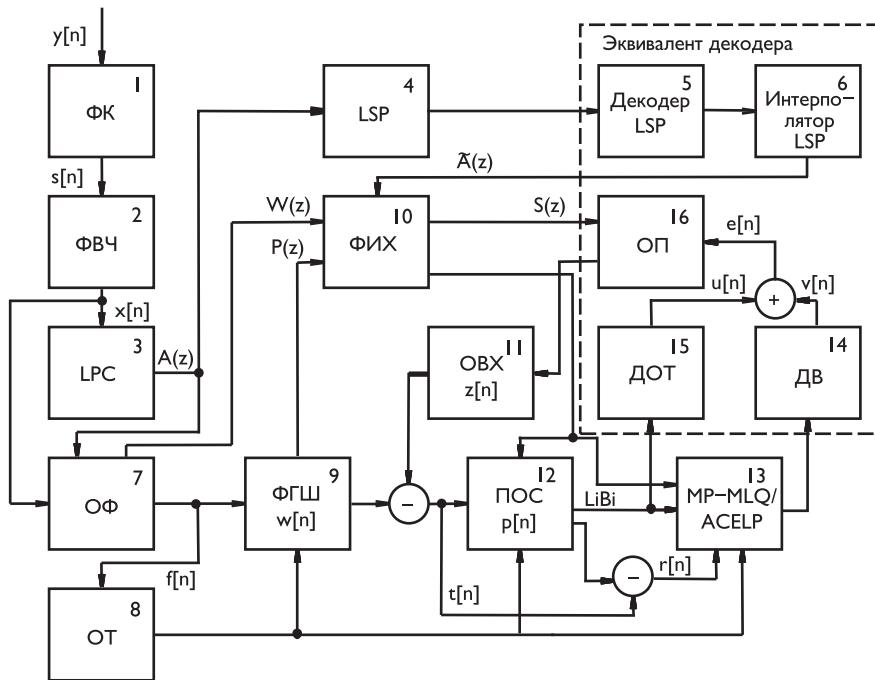


Рис. 16.1. Структурная схема кодирующего устройства G.723.1: 1 — формирователь кадров (ФК); 2 — высокочастотный фильтр (ФВЧ); 3 — кодер с линейным предсказанием (LPC); 4 — квантователь линейных спектральных пар (LSP); 5 — декодер LSP; 6 — интерполятор LSP; 7 — обработчик формант (ОФ); 8 — блок оценки основного тона (ОТ); 9 — формирователь огибающей гармонического шума (ФГШ); 10 — формирователь импульсной характеристики (ФИХ); 11 — формирователь отклика при отсутствии входного сигнала (ОВХ); 12 — предсказатель основного тона (ПОС); 13 — MP-MLQ/ACELP — многополосное квантование с максимумом правдоподобия / линейное предсказание с алгебраическим кодовым возбуждением; 14 — декодер возбуждения (ДВ); 15 — декодер основного тона (ДОТ); 16 — блок обновления памяти (ОП)

Индекс j , при котором взаимная корреляция максимальна, определяет значение величины воспринимаемого основного тона.

На основе оценки периода шага строится гармонический фильтр шума $w[n]$, улучшающий качество закодированной речи. Характеристика этого фильтра:

$$P_i(z) = 1 - \beta z^{-L}. \quad (16.6)$$

Оптимальная величина L для этого фильтра максимизирует критерий $C_{PW}(j)$ при соответствующих значениях $N(j)$:

$$N(j) = \sum_{n=0}^{59} f[n] \cdot f[n-j], \quad (16.7)$$

$$C_{PW}(j) = \frac{(N(j))^2}{\sum_{n=0}^{59} f[n-j] \cdot f[n-j]}, \quad L_1 \leq j \leq L_2, \quad (16.8)$$

где $L_1 = L_{OL} - 3$ и $L_2 = L_{OL} + 3$. Максимальное значение обозначено как C_L .

Оптимальная величина усиления фильтра G_{opt} и энергия речевого сигнала E равны:

$$G_{opt} = \frac{\sum_{n=0}^{59} f[n] f[n-L]}{\sum_{n=0}^{59} f[n-L] f[n-L]}, \quad (16.9)$$

$$E = \sum_{n=0}^{59} f^2[n]. \quad (16.10)$$

Тогда коэффициент β гармонического фильтра шума равен:

$$\beta = \begin{cases} 0,3125 \cdot G_{opt}, & \text{если } -10 \lg \left(1 - \frac{C_L}{E}\right) \geq 2,0, \\ 0,0, & \text{иначе.} \end{cases} \quad (16.11)$$

Вектор $w[n]$ вычисляется по формуле:

$$w[n] = f[n] - \beta f[n-L], \quad 0 \leq n \leq 59. \quad (16.12)$$

Комбинация синтезирующего LPC фильтра, взвешивающего фильтра формант и гармонического фильтра шума находится импульсный отклик:

$$S_i(z) = \tilde{A}_i(z) \cdot W_i(z) \cdot P_i(z), \quad 0 \leq i \leq 3, \quad (16.13)$$

используемый в дальнейших вычислениях.

Отсутствие входного сигнала определяется результатами вычисления $S_i(z)$. Результатом обработки является разность значений $\{w[n]\}$ и $\{z[n]\}$, $n = 0 \dots 59$; полученный вектор определяется как:

$$t[n] = w[n] - z[n]. \quad (16.14)$$

Предсказатель основного тона формирует сигнал подобия следующим образом. Для подкадров 0 и 2 задержка шага петли выбрана приблизительно соответствующей задержке шага разомкнутого цикла в диапазоне ± 1 . Для подкадров 1 и 3 задержка шага петли кодируется дифференцированно, используя 2 бита, и может отличаться от предыдущей задержки подкадра только на $-1, 0, +1$ или $+2$. Квантованные и декодированные значения задержки шага обозначаются L_i . Диапазон прогнозирующего блока – квантованный вектор, использующий две кодовые книги с 85 или 170 элементами старшего разряда и 170 элементами младшего разряда. 170 кодовых книг входа одинаковы для обеих норм. Если величина L_0 меньше, чем 58, для подкадров 0 и 1, или если величина L_2 меньше, чем 58, для подкадров 2 и 3, то используется кодовая книга с 85 элементами.

Для получения остаточного сигнала $\{r[n]\}$, $n = 0, \dots, 59$, результат работы прогнозирующего устройства $\{p[n]\}$, $n = 0, \dots, 59$, вычитается из целевого вектора $\{t[n]\}$, $n = 0, \dots, 59$:

$$\mathbf{r}[n] = \mathbf{t}[n] - \mathbf{p}[n]. \quad (16.15)$$

Этот остаточный новый целевой вектор при высокой скорости кодирования передается блоку **многополосного квантования с максимумом правдоподобия MP-MLQ**. Этот блок выполняет квантование вектора таким образом, что формируемый сигнал должен быть близким к величине:

$$r'[n] = \sum_{j=0}^n h[j] \cdot v[n-j], \quad 0 \leq n \leq 59, \quad (16.16)$$

где $v[n]$ — реакция фильтра $S(z)$ с импульсной характеристикой $h[n]$:

$$v[n] = G \sum_{k=0}^{M-1} \alpha_k \delta[n - m_k], \quad 0 \leq n \leq 59, \quad (16.17)$$

причем G — коэффициент усиления; $\delta[n]$ — функция Дирака, $\{\alpha_k\}$ и $\{m_k\}$, $k = 0, \dots, M-1$, — знаки (± 1) и позиции функций Дирака соответственно; M — число импульсов, равное 6 для четных подкадров и 5 — для нечетных подкадров.

Задача состоит в том, чтобы оценить неизвестные параметры G , $\{\alpha_k\}$ и $\{m_k\}$, $k = 0, \dots, M-1$, которые минимизируют средний квадрат ошибки сигнала:

$$err[n] = r[n] - r'[n] = r[n] - G \sum_{k=0}^{M-1} \alpha_k h[n - m_k]. \quad (16.18)$$

Параметр G_{\max} оценивается следующим образом. Во-первых, функция взаимной корреляции $d[j]$ между передаточной функцией $h[n]$ целевым вектором $r[n]$ вычисляется так:

$$d[j] = \sum_{n=j}^{59} r[n] \cdot h[n - j], \quad 0 \leq j \leq 59. \quad (16.19)$$

Тогда

$$G_{\max} = \frac{\max_{j=0, \dots, 59} \{ |d[j]| \}}{\sum_{n=0}^{59} h[n] \cdot h[n]}. \quad (16.20)$$

Это предполагаемое усиление логарифмически квантуется с возможным числом шагов квантования, равным 24, по 3,2 децибела каждый. Вокруг этой квантованной величины \tilde{G}_{\max} дополнительные значения усиления выбраны в пределах диапазона $[\tilde{G}_{\max} - 3,2, \tilde{G}_{\max} + 6,4]$. Конфигурация отсчетов в соотношении (16.18) выбирается исходя из минимума среднеквадратичной ошибки.

При низкой скорости кодирования используется **линейное предсказание с возбуждением алгебраическим кодом (ACELP)**.

Используется специальная кодовая книга для установленных значений $v[n]$. Каждый установленный вектор кода содержит, самое большое, четыре ненулевых импульса. Эти четыре импульса могут принять знаки и позиции, приведенные в табл. 16.2.

Таблица 16.2. ACELP — кодовая книга возбуждения

Значение	Позиция
± 1	0, 8, 16, 24, 32, 40, 48, 56
± 1	2, 10, 18, 26, 34, 42, 50, 58
± 1	4, 12, 20, 28, 36, 44, 52(, 60)
± 1	6, 14, 22, 30, 38, 46, 54(, 62)

Поиск кодовой книги осуществляется путем минимизации среднеквадратической ошибки между взвешенным речевым сигналом $r[n]$ и синтезированной взвешенной речью:

$$E_{\xi} = \|r - G\mathbf{H}\mathbf{v}_{\xi}\|^2, \quad (16.21)$$

где r – целевой вектор, состоящий из взвешенной речи после вычитания постоянной составляющей и дискретизации, G – коэффициент усиления кодовой книги; \mathbf{v}_ξ – алгебраический код при индексе ξ ; \mathbf{H} – треугольная матрица Тейлора с центральной диагональю $h(0)$, другими диагоналями $h(1), \dots, h(L-1)$ и с $h(n)$ импульсной передаточной функцией, реализующей взвешенную фильтрацию $S_i(z)$.

Оптимальный код при этом максимизирует величину:

$$\tau_\xi = \frac{C_\xi^2}{\varepsilon_\xi} = \frac{(\mathbf{d}^T \mathbf{v}_\xi)^2}{v_\xi^T \Phi \mathbf{v}_\xi}, \quad (16.22)$$

где $\mathbf{d} = \mathbf{H}^T \mathbf{r}$ – корреляция между целевым векторным сигналом $r[n]$ и импульсной передаточной функцией $h(n)$, $\Phi = \mathbf{H}^T \cdot \mathbf{H}$ – матрица ковариации импульсной передаточной функции.

Элементы вектора \mathbf{d} и матрицы Φ вычисляются по формулам:

$$d(j) = \sum_{n=j}^{59} r[n] \cdot h[n-j], \quad 0 \leq j \leq 59, \quad (16.23)$$

$$\Phi(i, j) = \sum_{n=j}^{59} h[n-i] \cdot h[n-j], \quad j \geq i, \quad 0 \leq i \leq 59. \quad (16.24)$$

Алгебраическая структура кодовой книги обеспечивает быструю процедуру поиска, так как вектор \mathbf{v}_ξ возбуждения содержит только 4 ненулевых импульса.

Максимум корреляции в уравнении (16.22) достигается при:

$$C = d'[m_0] + d'[m_1] + d'[m_2] + d'[m_3], \quad (16.25)$$

где m_k – позиция k -го импульса, а $d'[j] = d[j]s[j]$ и

$$\begin{aligned} s[2j] &= s[2j+1] = \text{sign}(d[2j]), & \text{если } |d[2j]| > |d[2j+1]|, \\ s[2j] &= s[2j+1] = \text{sign}(d[2j+1]), & \text{иначе.} \end{aligned}$$

Граница вычислений основана на оценке корреляции C . Максимальная абсолютная корреляция и средняя корреляция при воздействии первых трех импульсов, \max_3 и av_3 , находятся до поиска кодовой книги. Порог рассчитывается:

$$\text{th } r_3 = av_3 + \left(\max_3 - av_3 \right) / 2. \quad (16.26)$$

Четвертая составляющая может быть введена, если значение корреляции (при трех пульсах) превышает thr_3 .

Последним шагом после расчета последовательности $v[n]$ является квантование усиления G путем минимизации соотношения $|G - \tilde{G}_j|$, $0 \leq j \leq 23$.

По результатам обработки информации в блоках предсказателя основного тона и MP-MLQ/ACELP в эквиваленте декодера с помощью декодера основного тона и декодера возбуждения вклад абсолютной высоты тона $u[n]$ и вклад импульсов возбуждения $v[n]$ суммируются, формируя сигнал:

$$u[n] + v[n] = e[n], \quad (16.27)$$

поступающий на блок обновления памяти.

Таким образом, кодеки стандарта G.723.1 можно охарактеризовать как комбинацию АЦП/ЦАП и вокодера. Применение вокодера позволяет снизить скорость

передачи данных в канале, что принципиально важно для эффективного использования как в радиотракте, так и в IP-канале. Основной принцип работы вокодера — синтез исходного речевого сигнала посредством адаптивной замены его гармонических составляющих соответствующим набором частотных фонем и согласованными шумовыми коэффициентами. Кодек G.723.1 осуществляет преобразование аналогового сигнала в поток данных со скоростью 64 кбита/с (ИКМ), а затем при помощи многополосного цифрового фильтра/вокодера выделяет частотные фонемы, анализирует их и передает по IP-каналу информацию только о текущем состоянии фонем в речевом сигнале. Данный алгоритм преобразования позволяет снизить скорость кодированной информации до 5,3–6,3 кбитов/с без видимого ухудшения качества речи. Вариант кодирования со скоростью передачи 6,3 кбитов/с предназначен для сетей с пакетной передачей голоса и обеспечивает лучшее качество кодирования по сравнению с вариантом, реализующим скорость передачи 5,3 кбитов/с, но менее адаптирован к использованию в сетях со смешанным типом трафика (голос/данные).

16.4. G.726 — речевой кодек АДИКМ со скоростями 40, 32, 24, 16 кбитов/с

Структурная схема кодека приведена на рис. 16.2. В основе преобразования кода ИКМ, соответствующего стандарту G.726 (при двух правилах кодирования — A-law и μ -law), является использование нелинейной шкалы квантования, различной для разных цифровых потоков — от 40 до 16 кбитов/с [6.31].

При различных цифровых потоках используется различная нормализация входных/выходных характеристик квантования.

Рекомендация описывает алгоритм адаптивной дифференциальной импульсно-кодовой модуляции при скоростях выходного потока 40, 32, 24 и 16 кбитов/с.

Цифровые потоки со скоростью 24 и 16 кбитов/с предназначены в первую очередь для использования в перегруженных голосовых каналах в цифровом оборудовании мультиплексирования. Цифровые потоки со скоростью выходного потока 40 кбитов/с предназначены для передачи сигналов модемов в цифровом канальном оборудовании мультиплексирования, особенно для модемов, работающих на скоростях выше 4800 кбитов/с.

Кодер, структура которого приведена на рис. 16.2а, обеспечивает преобразование входного сигнала в равномерную ИКМ-последовательность и вычитание из него предсказанной оценки. Разностный сигнал квантуется с помощью адаптивного квантователя с 31, 15, 7 или 4 уровнями, что реализует 5, 4, 3 или 2 бита на отсчет соответственно, формируя выходной сигнал. Восстановленный исходный сигнал получается из суммы восстановленного с помощью обратного квантования разностного сигнала и оценки сигнала. Оценку сигнала формирует адаптивный предсказатель.

Декодер (рис. 16.2б) включает часть, аналогичную петле обратной связи кодера, обратное преобразование A-law или μ -law ИКМ и синхронную коррекцию кодирования. Синхронная коррекция кодирования предотвращает накопление ошибок, возникающих при определенных условиях в цепочке преобразований

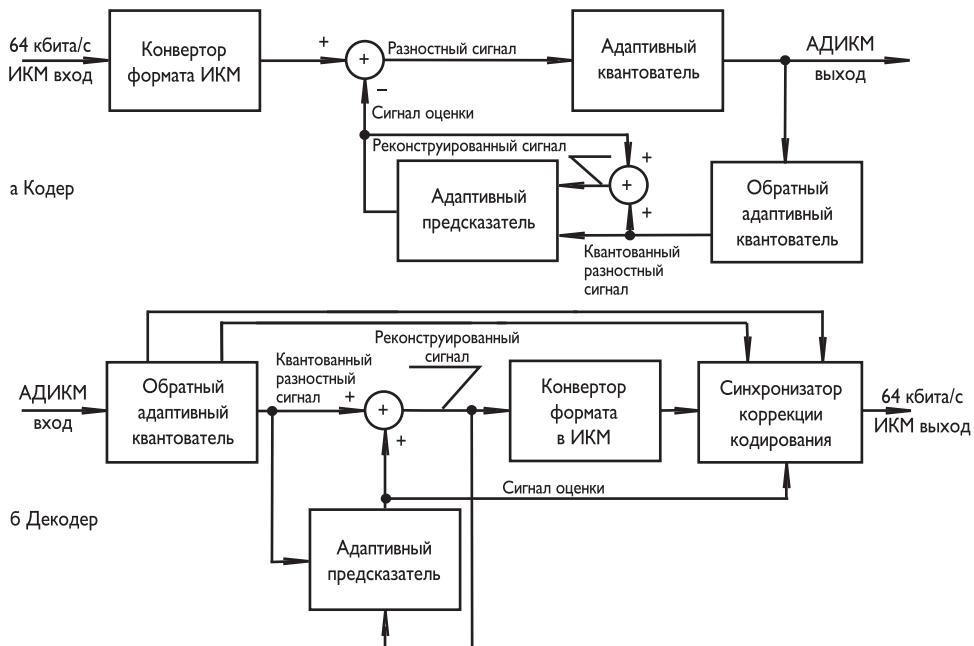


Рис. 16.2. Структурные схемы кодирующего (а) и декодирующего (б) устройств

АДИКМ-ИКМ-АДИКМ. Синхронная коррекция кодирования осуществляется с помощью изменения выходных кодов ИКМ таким образом, чтобы устранить ошибки квантования на следующей стадии кодирования АДИКМ.

16.5. G.727 — речевой кодек с алгоритмом вложенной АДИКМ

Данный алгоритм вложенной АДИКМ является расширением алгоритма, описанного в Рекомендации G.726, и реализует скорости выходного потока 40, 32, 24 и 16 кбитов/с (5, 4, 3 и 2 битах на отсчет) [6.32].

Алгоритмы вложенной АДИКМ реализуют преобразования с переменной скоростью кодированного битового потока и возможностью потери битов вне блоков кодера и декодера. Они состоят из ряда алгоритмов, в которых уровни квантования при низких скоростях являются подмножеством уровней квантования при высоких скоростях. Это позволяет уменьшать количество битов в любой точке сети без необходимости координации взаимодействия между передатчиком и приемником.

Вложенные алгоритмы могут приспосабливаться к непредсказуемым и пульсирующим характеристикам трафика, которые требуют уменьшения перегруженности. Так как необходимость уменьшения нагрузки может возникнуть после кодирования, вложенное кодирование отличается от кодирования с переменной

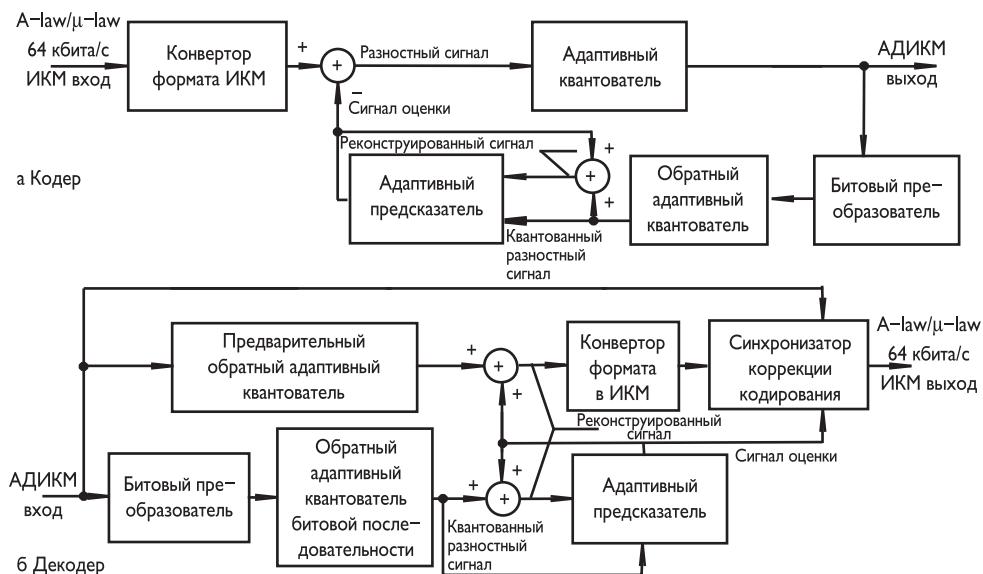


Рис. 16.3. Структурные схемы кодирующего (а) и декодирующего (б) устройств G.727

скоростью выходного потока, когда кодер и декодер должны использовать одинаковое количество битов на отсчет. В обоих случаях декодер должен знать количество битов на отсчет.

Вложенные алгоритмы формируют кодовые слова, содержащие основные биты и биты расширения. Тракт прямой обработки (FF) использует и основные биты, и биты расширения, в то время как тракт обратной связи (FB) использует только основные биты. Блоки обратного квантования/предсказания и кодера, и декодера используют основные биты. В этой схеме биты расширения могут быть потеряны или отброшены во время перегрузки сети. Однако количество основных битов в тракте обратной связи и кодера, и декодера должны оставаться одинаковыми во избежание рассогласования.

Четыре скорости вложенного АДИКМ — 40, 32, 24 и 16 кбитов/с. Наборы уровней квантования для скоростей 32, 24 и 16 кбитов/с являются подмножествами набора уровней для скорости 40 кбитов/с. Алгоритмы вложенного АДИКМ обозначаются парами (x, y) , где x относится к АДИКМ-битам тракта прямой обработки (основным и расширения), а y относится к АДИКМ-битам тракта обратной связи (основным). Например, если y равен двум битам, то пара $(5, 2)$ будет представлять вложенный алгоритм на 40 кбитов/с, $(4, 2)$ — на 32 кбитов/с, $(3, 2)$ — на 24 кбита/с, $(2, 2)$ — на 16 кбитов/с. Скорость выходного битового потока не может быть ниже 16 кбитов/с, потому что минимальное количество основных битов — два.

Рекомендация G.727 позволяет реализовать скорости кодирования 40, 32, 24 и 16 кбитов/с при скоростях основных битов 32, 24 и 16 кбитов/с. Это соответствует следующим парам: $(5, 2)$, $(4, 2)$, $(3, 2)$, $(2, 2)$; $(5, 3)$, $(4, 3)$, $(3, 3)$; $(5, 4)$, $(4, 4)$. Структурная схема кодека стандарта G.727 приведена на рис. 16.3.

16.6. G.728 — речевой кодек с выходным потоком 16 кбитов/с и малой задержкой

Речевой кодек G.728 обеспечивает выходной поток 16 кбитов/с при малой задержке, используя линейное предсказание с кодовым возбуждением.

Сущность методики линейного предсказания с кодом возбуждения (CELP) состоит в поиске по кодовой книге с помощью анализа и синтеза. Возбуждающий сигнал выбирается процедурой поиска, использующей закрытый цикл, и пропускается через синтезирующий фильтр. Синтезированный фрагмент сравнивается с сегментом оригинального речевого сигнала, искажения измеряются, процесс повторяется для всех кодовых векторов, содержащихся в кодовой книге. Индекс наилучшей возбуждающей последовательности передается на сторону декодера, который извлекает возбуждающие кодовые векторы из кодовой книги точно таким же образом, что и кодер [6.33].

Для большинства речевых кодеков, таких как, например, CELP, высокая степень сжатия достигается за счет обработки сигнала поблочно, это требует буферизации, составляющей обычно 20–30 мс в зависимости от длины блока. Такой процесс буферизации характерен для большинства низкоскоростных кодеков, это вызывает существенную общую задержку. Задержка очень критична для двустороннего общения в реальном времени. Большие задержки (свыше 150 мс) существенно ухудшают возможность общения. Низкая задержка также желательна еще и потому, что большие задержки усугубляют проблемы возникновения эха в телефонных сетях.

Требование низкой задержки конфликтует с другими желательными характеристиками кодека, такими как высокое сжатие, высокое качество синтезированной речи, низкая вычислительная сложность, устойчивость к канальным ошибкам. Тем не менее снижение задержки с минимальной деградацией других характеристик представляется достижимым, удачной попыткой можно считать технологию LD-CELP кодека G.728.

Для достижения низкой задержки LD-CELP применяет целый ряд стратегий, среди которых: уменьшенная длина блока, рекурсивная автокорреляционная оценка, внешнее предсказание внутри блока, обратное адаптивное линейное предсказание, высокий порядок фильтра предсказания.

На рис. 16.4 приведена структурная схема кодека по стандарту G.728.

Методика LD-CELP использует обратную адаптацию предсказателей и усиления для того, чтобы достичь алгоритмической задержки 0,625 мс. Передается только индекс (VQ) в кодовой книге возбуждений. Коэффициенты предсказателя уточняются на основе анализа с линейным предсказанием (LPC) ранее квантованной речи. Усиление возбуждения уточняется на основе информации об усилении ранее квантованного возбуждения. Размер блока вектора возбуждения и адаптации усиления — 5 отсчетов. Взвешивающий фильтр уточняется линейным предсказанием путем анализа неквантованной речи.

После преобразования A-law или μ -law ИКМ к равномерному ИКМ отсчеты кодируемой речи входного сигнала разделяются на блоки и подблоки. Каждый блок состоит из 20 отсчетов или четырех подблоков по 5 отсчетов. Поскольку длина блока невелика, то статистические свойства близких блоков отличаются мало.

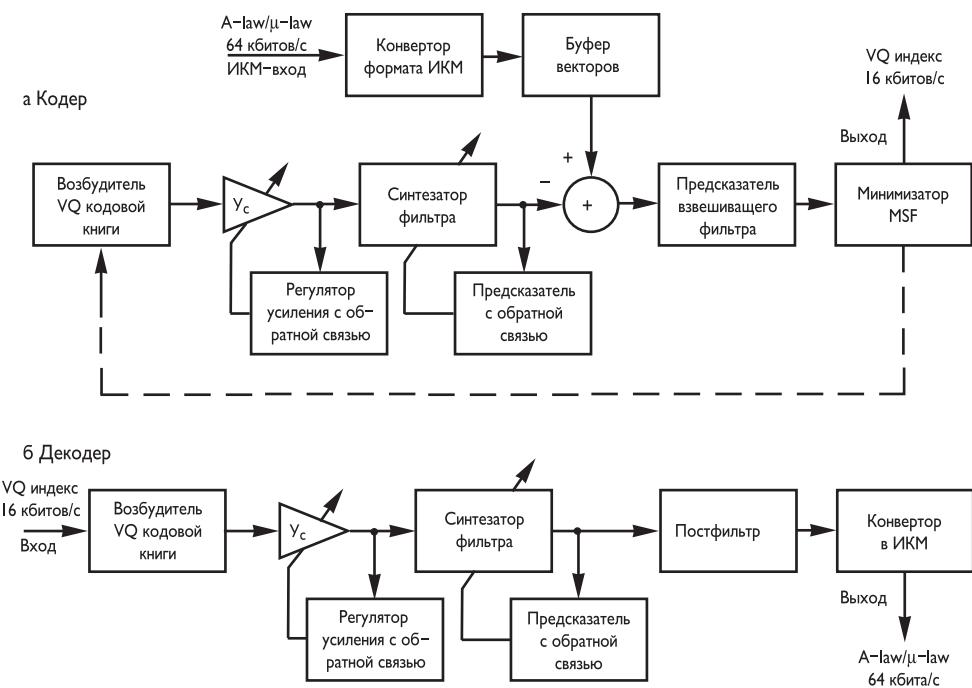


Рис. 16.4. Структурные схемы кодирующего (а) и декодирующего (б) устройств G.728

Поэтому возможно применять к текущему блоку линейное предсказание (ЛП) коэффициентов, полученных из предыдущего блока. Таким образом, нет необходимости буферизовать блок целиком, достаточно буферизовать только подблок (5 отсчетов).

Так как ЛП-анализ выполняется обратно-адаптивным образом (т. е. ЛП-коэффициенты получаются из синтезированной, а не оригинальной речи), то нет необходимости передавать коэффициенты, так как они могут быть получены на стороне декодера таким же образом из синтезированной речи.

Для каждого блока кодер анализирует каждый из 1024 кандидатов — векторов из кодовой книги возбуждений через устройство масштабирования усиления и синтезирующий фильтр. Из полученных 1024 кандидатов — квантованных векторов сигнала кодер выбирает один — минимизирующий частотно-взвешенную среднеквадратичную меру ошибки по отношению к вектору входного сигнала. Десятибитный индекс соответствующего вектора из кодовой книги передается декодеру. Этот наилучший вектор затем пропускается через устройство масштабирования усиления и синтезирующий фильтр для того, чтобы поправить память фильтра для кодирования следующего вектора сигнала. Коэффициенты синтезирующего фильтра и усиление периодически уточняются с использованием ранее квантованного сигнала и масштабированного возбуждения.

Взвешивающий фильтр имеет 10 коэффициентов линейного предсказания, полученных из оригинальной речи. Фильтр обновляется один раз на блок. Текущие коэффициенты рассчитываются из отсчетов предыдущего блока.



Синтезирующий фильтр соответствует авторегрессионному процессу 50-го порядка. Его коэффициенты рассчитываются из синтезированной речи предыдущих фреймов. Фильтр обновляется один раз на блок.

Коэффициент усиления обновляется на каждый подблок с использованием адаптивного линейного предсказателя 10-го порядка. Коэффициенты этого предсказателя обновляются на основе ЛП-анализа значений коэффициента усиления из предыдущих подблоков.

Поиск возбуждающей последовательности производится один раз на подблок. Процедура поиска генерирует ансамбль фильтрованных последовательностей: каждая возбуждающая последовательность используется как вход синтезирующего фильтра для получения выходной последовательности. Выбирается возбуждающая последовательность, минимизирующая ошибку.

Декодер производит восстановление. Синтезированная речь генерируется посредством пропускания возбуждающей последовательности через синтезирующий фильтр без всякого взвешивания. Так как коэффициент усиления и коэффициенты линейного предсказания обратноадаптивны, нет необходимости передавать эти параметры. Также может быть добавлен сглаживающий постфильтр для улучшения качества выходной речи.

После получения каждого десятибитного индекса декодер осуществляет поиск по таблице и извлекает соответствующий вектор из кодовой книги возбуждений. Полученный вектор затем обрабатывается устройством масштабирования усиления и синтезирующим фильтром для получения текущего декодированного вектора сигнала. Коэффициенты синтезирующего фильтра и усиление затем уточняются так же, как и в кодере. Декодированный вектор сигнала проходит через адаптивный постфильтр для улучшения качества восприятия звука. Коэффициенты постфильтра периодически уточняются с помощью информации, имеющейся на декодере. Затем пять отсчетов сигнала преобразуются в A-law или μ -law ИКМ.

16.7. G.729 — речевой кодек со скоростью 8 кбитов/с

Кодер предназначен для работы с цифровым сигналом, полученным после фильтрации в полосе телефонного аналогового сигнала, дискретизации с частотой 8000 Гц и линейного квантования с глубиной 16 битов на отсчет. Кодер использует технологию CS-ACELP (Conjugate-Structure Algebraic-Code-Exited Linear Prediction, сопряженное линейное предсказания с алгебраическим кодом возбуждения), которая основывается на модели CELP (линейное предсказание с кодовым возбуждением). Он оперирует с блоками длительностью 10 мс, что соответствует 80 отсчетам при частоте дискретизации 8 кГц. Для каждого блока речевой сигнал анализируется с целью извлечения параметров CELP-модели (коэффициенты фильтра линейного предсказания — ЛП, индексы и коэффициенты усиления для адаптивной и фиксированной кодовых книг). Эти параметры кодируются и передаются. Размещение битов кодека показано на табл. 16.3.

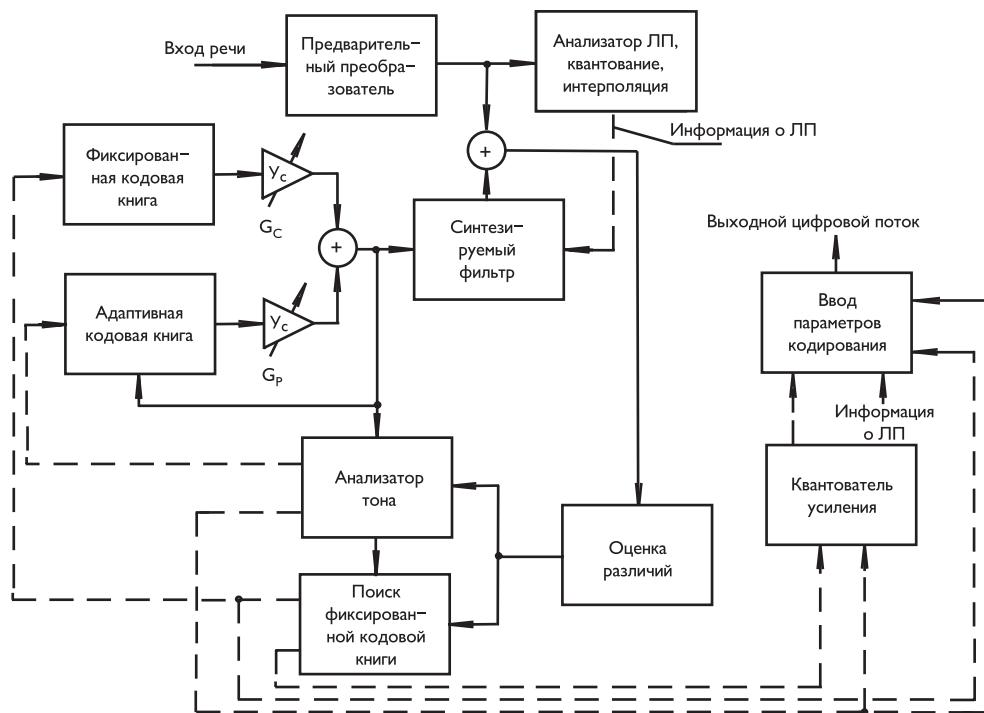


Рис. 16.5. Обобщенная структура кодирующего устройства по стандарту G.729

Таблица 16.3. Размещение битов кодека G.729

Параметр	1-й подблок	2-й подблок	Битов на блок
Коэффициенты ЛП			18
Индекс адаптивной книги	8	5	13
Бит четности	1		1
Индекс фиксированной книги	17	17	34
Индексы усиления	7	7	14
Всего			80

На стороне декодера эти параметры используются для извлечения возбуждающей последовательности и параметров синтезирующего фильтра. Речь реконструируется посредством пропускания возбуждающей последовательности через короткий синтезирующий фильтр линейного предсказания 10-го порядка. Длинный фильтр (фильтр синтеза шага) строится на основе так называемого метода аддитивной кодовой книги. После вычисления восстановленного речевого сигнала он подвергается постфильтрации с целью дальнейшего улучшения качества. Принцип работы кодера поясняется на рис. 16.5.

Входной сигнал речи проходит ВЧ-фильтрацию и масштабирование в блоке предварительной обработки. Затем осуществляется ЛП-анализ в каждом блоке длительностью 10 мс: вычисляются коэффициенты фильтра ЛП. Эти коэффициенты квантуются 18-битным векторным квантователем.

Сигнал возбуждения выбирается с помощью поисковой процедуры анализа-синтеза, в процессе выполнения которой минимизируется расхождение между исходным и реконструированным речевым сигналом с использованием взвешенной меры. Эта операция осуществляется с помощью фильтрации сигнала ошибки взвешивающим фильтром, коэффициенты которого определяются параметрами неквантованного ЛП-фильтра. Параметры возбуждения (фиксированной и адаптивной кодовых книг) определяются для подблока длительностью 5 мс (40 отсчетов). Коэффициенты квантованного и неквантованного ЛП-фильтра используются для второго подблока, а для первого используются интерполированные коэффициенты ЛП-фильтра. Задержка шага открытой петли производится для блока на основе взвешенного речевого сигнала. Основной тон оценивается методом открытого цикла один раз в блоке длительностью 10 мс с использованием взвешенного речевого сигнала. Далее для каждого подблока выполняются следующие операции:

- результирующий сигнал $x(n)$ вычисляется с помощью фильтрации остатка ЛП-фильтрации взвешенным синтезирующим фильтром;
- начальные состояния этих фильтров обновляются с помощью фильтрации разницы между остатками ЛП-фильтрации и возбуждением (это эквивалентно обычному подходу, при котором из взвешенного речевого сигнала вычитается отклик на нулевой сигнал);
- вычисляется импульсный отклик взвешенного синтезирующего фильтра $h(n)$;
- для нахождения значений задержки и усиления адаптивной кодовой книги в цепи с отрицательной обратной связью проводится анализ результирующего сигнала $x(n)$ и импульсного отклика $h(n)$, при этом поиск осуществляется в окрестности значения задержки, вычисляемой с помощью прямого предсказания; дробная часть задержки шага используется с точностью 1/3 интервала между отсчетами.

Задержка шага кодируется 8 битами в первом подблоке и дифференциально 5 битами во втором подблоке. Результирующий сигнал $x(n)$ уточняется с помощью вычитания профильтрованного вклада неадаптивной кодовой книги, и уточненный сигнал $x'(n)$ используется в неадаптивной кодовой книге для поиска оптимального возбуждения. Адаптивная кодовая книга (17 битов) используется для возбуждения фиксированной кодовой книги. Усиления вкладов адаптивной и фиксированной кодовых книг векторно квантуются (7 битов). Наконец, при помощи полученного сигнала возбуждения уточняются параметры фильтра.

Сигнал в декодере подвергается обратной процедуре обработки (рис. 16.6).

Изначально индексы параметров извлекаются из принятого битового потока. Эти индексы декодируются для получения параметров кодера относящихся к блоку длительностью 10 мс. К ним относятся: коэффициенты фильтра ЛП, две дробных задержки шага, два вектора фиксированной кодовой книги, два индекса адаптивной книги и два набора коэффициентов усиления. Коэффициенты ЛП выделяются путем интерполяции для получения коэффициентов для каждого подблока. Для каждого подблока длительностью 5 мс выполняются следующие шаги:

- сигнал возбуждения конструируется сложением векторов из адаптивной и фиксированной книг, масштабированных соответствующими коэффици-

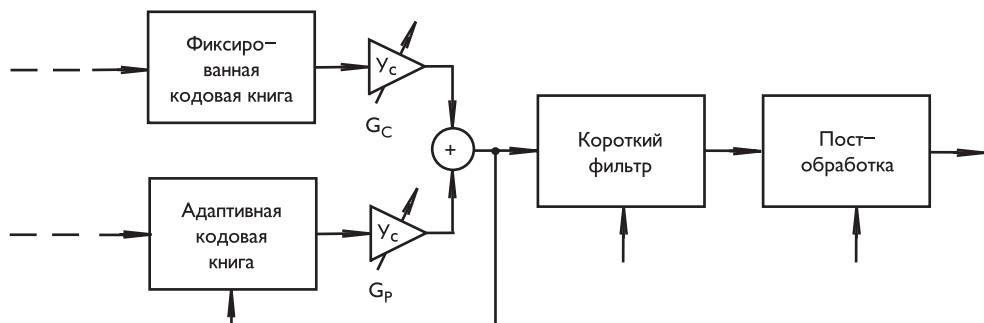


Рис. 16.6. К пояснению работы декодера по стандарту G.729

ентами усиления;

- речевой сигнал реконструируется пропусканием возбуждающего сигнала через синтезирующий фильтр линейного предсказания;
- реконструированный речевой сигнал проходит стадию постобработки, включющую адаптивный постфильтр, основанный на коротком и длинном синтезирующих фильтрах, высокочастотный фильтр и масштабирование.

Задержка системы кодирования по Рекомендации G.729 складывается из размера кадра (10 мс), предварительного просмотра 5 мс (общая алгоритмическая задержка — 15 мс), времени вычислений кодера и декодера, времени передачи по линии связи, времени задержки мультиплексирования звуковых и других данных.

Имеется множество модификаций кодера речи G.729, основными из которых являются Приложения G.729A и G.729B [6.35, 6.36].

Принцип кодирования G.729A близок к G.729, но требует меньше вычислений ввиду использования операций с фиксированной точкой, в отличие от G.729, который использует операции с плавающей точкой. Это упрощение приводит к незначительному ухудшению качества речи.

В модификации G.729B дополнительно к G.729 содержится детектор голосовой активности (VAD), который способен снижать битовый поток во время отсутствия речевого сигнала, заполняя паузы особым комфорtnым шумом, чтобы не вызывать дискомфорта у абонента воспроизведением абсолютной тишины.

В 2006 г. ITU-T принял Рекомендацию G.729.1 [6.37] масштабируемого широкополосного речевого кодера, обеспечивающего реализацию скорости передачи информации 8–32 кбитов/с и совместимого с кодерами стандарта G.729 и его модификациями.

Этот кодер был разработан с целью обеспечения лучшего качества воспроизведения звука и большей гибкости по сравнению с существующим кодером G.729. G.729.1 масштабируем в скорости передачи цифровой информации, акустической полосе пропускания и сложности. Кроме того, предлагаются различные кодирующие устройства и способы декодирования, включая поддержку частоты выборки входного сигнала, равной 8 и 16 кГц, совместимость с G.729B, а также уменьшение алгоритмической задержки.

Цифровой поток G.729.1 структурирован в 12 иерархических слоев. Первый слой (или основной слой) в 8 кбитов/с соответствует формату G.729. Второй слой

(12 кбитов/с) обеспечивает повышение качества низкочастотных составляющих речи. Третий слой (14 кбитов/с) является слоем расширения полосы пропускания. Дальнейшие слои (в шагах на 2 кбитов/с) обеспечивают повышение качества высокочастотных составляющих речи. Полоса пропускания канала G.729.1 составляет 50–4000 Гц при 8 и 12 кбитов/с, и 50–7000 Гц от 14 до 32 кбитов/с.

16.8. GSM — речевой кодек для систем сотовой связи

Кодек речевого сигнала стандарта GSM (Global System for Mobile Communications) цифровой сотовой связи использует долговременное предсказание с возбуждением регулярной последовательностью импульсов (Regular Pulse Excited Long Term Prediction — RPE-LTP). Он основан на применении двух способов кодирования: линейного предсказания с возбуждением остатком (Residual Excited Linear Prediction — RELP) и долговременного предсказания с импульсным возбуждением (Multi-Pulse Excited Long Term Prediction — MPE-LTP) [6.38].

Кодер создает битовый поток со скоростью 13 кбитов/с, он был выбран после многочисленных экспертных оценок. Преимущество RELP-кодека состоит в том, что он обеспечивает хорошее качество речи при сравнительно низкой сложности. Однако качество речи при использовании этого кодека ограничивается помехой, создаваемой при восстановлении высокочастотных компонентов речевого сигнала и ошибками при передаче битов по каналу связи. MPE-LTP-кодек обеспечивает очень хорошее качество речи при высокой сложности и не очень чувствителен к ошибкам в канале передачи [6.39, 6.40]. На рис. 16.7 приведена упрощенная функциональная схема кодека GSM.

Кодер содержит четыре основных блока обработки каждого сегмента речевого сигнала: предварительной обработки, кратковременного анализа (предсказания), долговременного анализа и формирования сигнала возбуждения в виде регулярной последовательности импульсов. На вход кодера с частотой следования 8 кГц поступают отсчеты речевого сигнала в виде 13-битовых кодовых слов ИКМ.

Блок предварительной обработки осуществляет:

- фильтрацию сигнала с использованием фильтра высоких частот, в результате чего подавляется постоянная составляющая сигнала и увеличивается уровень высокочастотных компонентов;
- разделение сигнала на сегменты по 20 мс (160 отсчетов в каждой выборке);
- взвешивание отсчетов каждого из сегментов временным окном Хэмминга.

Далее во втором блоке по 160 отсчетам сегмента вычисляются значения оценок параметров фильтра линейного предсказания при $M = 8$ (число оцениваемых параметров или порядок фильтра) и значения сигнала ошибки предсказания e .

Форма Z-характеристики анализатора кратковременного предсказания (в упрощенном варианте) имеет вид:

$$A(z) = 1 - \sum_{i=1}^M a_i z^{-i}, \quad (16.28)$$

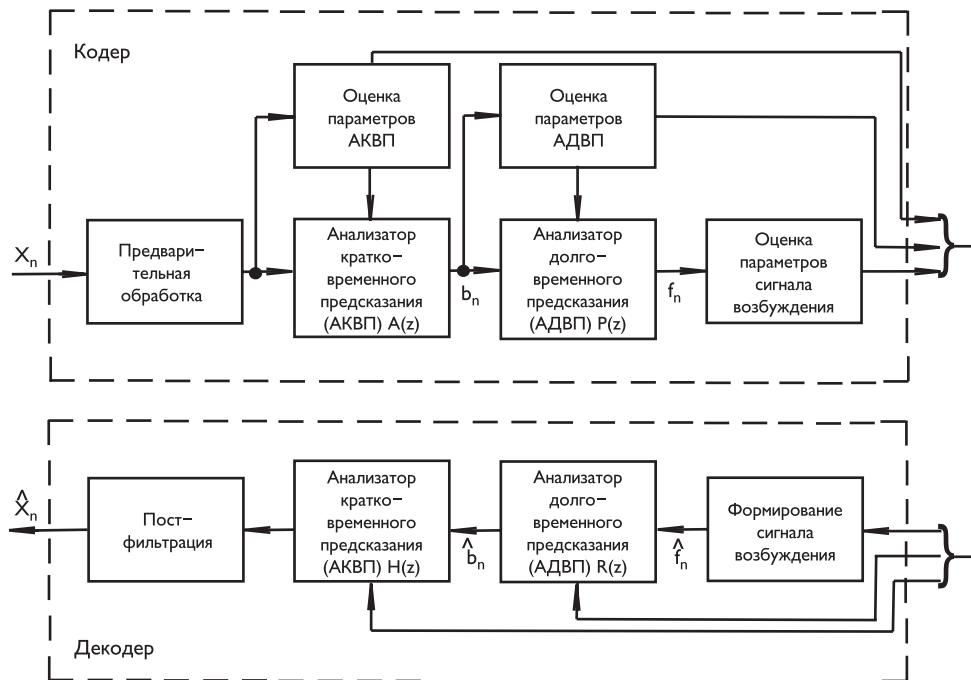


Рис. 16.7. Упрощенные функциональные схемы кодирующего (вверху) и декодирующего (внизу) устройств GSM

где a_i — коэффициенты предсказания, значения которых постоянны на интервале кодируемого сегмента речи 20 мс, они вычисляются из условия минимизации среднеквадратичного значения остатка предсказания e_n (см. рис. 16.8а).

Приравнивание к нулю частных производных суммы квадратов величины предсказания $(e_n)^2$ по каждому коэффициенту a_i приводит к системе M линейных уравнений с M неизвестными коэффициентами a_i . Матрица системы и метод ее решения оказываются несколько различными в зависимости от того, какими свойствами наделяется речевой сигнал на интервале преобразуемого сегмента речи.

В качестве побочного результата решения получаются значения так называемых коэффициентов отражения, или коэффициентов частичной корреляции k_i , $i = 1, \dots, M$, число которых равно числу коэффициентов линейного предсказания a_i и которые связаны с этими коэффициентами взаимно однозначными нелинейными функциональными соотношениями.

Коэффициенты отражения могут быть получены при использовании другой формы фильтра линейного предсказания — так называемым решетчатым, или лестничным, фильтром (рис. 16.8б). Коэффициенты отражения k_i более удобны, чем коэффициенты линейного предсказания a_i , для передачи по линии связи, так как в силу своих статистических характеристик в меньшей степени могут приводить к потере устойчивости фильтра при квантовании. Иначе говоря, они требуют меньшего числа разрядов при квантовании, т. е. приводят к более экономическому использованию линии связи. Иногда используются также функции от

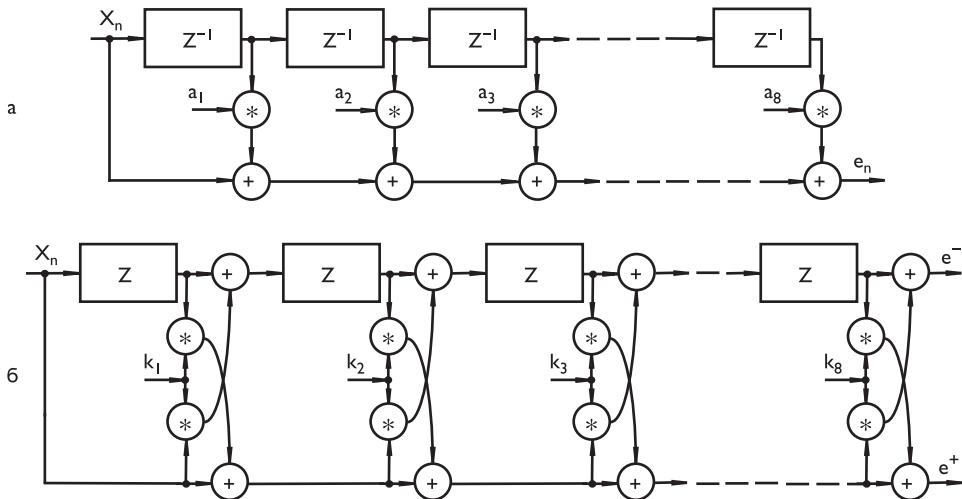


Рис. 16.8. Структура простого трансверсального фильтра-анализатора кратковременного преобразования (а) и решетчатого фильтра (б)

коэффициентов отражения — логарифмические отношения площадей:

$$r_i = \lg \frac{1 - k_i}{1 + k_i}. \quad (16.29)$$

В дополнение к кратковременному предсказанию, используется еще долговременное предсказание, в значительной мере устраниющее остаточную избыточность и приближающее остаток предсказания по своим статистическим характеристикам к белому шуму. Передаточная характеристика инверсного фильтра долговременного предсказания имеет вид:

$$P(z) = 1 - \sum_{k=-M_1}^{M_2} g_k z^{-(d+k)}, \quad (16.30)$$

где g_k — коэффициенты долговременного предсказания, порядок предсказания равен $M_1 + M_2 + 1$, а временная задержка d соответствует периоду основного тона (для вокализованных звуков).

Если на вход инверсного фильтра долговременного предсказания подается остаток кратковременного предсказания e_n , то на выходе получается остаток (ошибка) долговременного предсказания f_n , равный:

$$f_n = e_n - \sum_{k=-M_1}^{M_2} g_k e_{n-d-k}. \quad (16.31)$$

Ошибка f_n весьма близка к белому гауссовскому шуму, что облегчает экономичное формирование параметров сигнала возбуждения. Параметры долговременного предсказания g_k и d могут быть определены, например, из условия минимизации среднеквадратичного значения ошибки f_n на некотором интервале, составляющем 20–25% от длительности передаваемого сегмента речи. Задержка d обычно устанавливается в пределах 20–160 интервалов дискретизации сигнала, что соответствует диапазону частот основного тона 50–400 Гц.

Обычно порядок долговременного предсказателя устанавливается равным 1 и $M_1 = M_2 = 0$. При этом 160-выборочный сегмент остатка кратковременного предсказания e_n разделяется на 4 подсегмента, по 40 выборок в каждом, и параметры g , d оцениваются для каждого из подсегментов в отдельности, причем для оценки задержки d для текущего подсегмента используется скользящий подсегмент из 40 выборок, перемещающийся в пределах предшествующих 128 выборок сигнала остатка предсказания e_n . Сигнал e_n фильтруется фильтром-анализатором долговременного линейного предсказания, а выходной сигнал последнего — остаток предсказания f_n — фильтруется сглаживающим фильтром, и по нему формируются параметры сигнала возбуждения, в отдельности для каждого из 40-выборочных подсегментов.

Сигнал возбуждения одного подсегмента состоит из 13 импульсов, следующих через равные промежутки времени (втрое большие, чем интервал дискретизации исходного сигнала) и имеющих различные амплитуды. Для формирования сигнала возбуждения 40 импульсов подсегмента сглаженного остатка f_n обрабатываются следующим образом:

- последний (сороковой) импульс отбрасывается, а первые 39 импульсов разбиваются на три последовательности;
- в первой из последовательностей — импульсы 1, 4, …, 37, во второй — импульсы 2, 5, …, 38, в третьей — импульсы 3, 6, …, 39;
- в качестве сигнала возбуждения выбирается та из последовательностей, энергия которой больше;
- амплитуды импульсов нормируются по отношению к импульсу с наибольшей амплитудой, и нормированные амплитуды кодируются тремя битами каждая при линейной шкале квантования;
- абсолютное значение наибольшей амплитуды кодируется шестью битами в логарифмическом масштабе;
- положение начального импульса 13-элементной последовательности кодируется двумя битами, т. е. номером последовательности, выбранной в качестве сигнала возбуждения данного подсегмента.

Таким образом, выходная информация кодера речи для одного 20-миллисекундного сегмента речи включает:

- параметры фильтра кратковременного линейного предсказания — 8 коэффициентов, один набор на весь сегмент;
- параметры фильтра долговременного линейного предсказания — коэффициент предсказания g и задержка d — для каждого из четырех подсегментов;
- параметры сигнала возбуждения — номер последовательности n , максимальная амплитуда и нормированные амплитуды 13 импульсов последовательности — для каждого из четырех подсегментов.

В табл. 16.3 приведено число битов, используемых при кодировании передаваемых параметров 20 мс сегмента.

Таким образом, всего для одного 20-мс сегмента речи передается 260 битов информации, т. е. кодер речи осуществляет сжатие информации почти в 5 раз

$(1280/260 = 4,92)$. Перед выдачей в канал связи выходная информация кодера речи подвергается дополнительному канальному кодированию.

В декодере (рис. 16.7, внизу) блок формирования сигнала возбуждения, используя принятые параметры сигнала возбуждения, восстанавливает 13-импульсную последовательность сигнала возбуждения для каждого из подсегментов сигнала речи, включая амплитуды импульсов и их расположение во времени. Сформированный таким образом сигнал возбуждения фильтруется фильтром-синтезатором долговременного предсказания, передаточная характеристика $R(z)$ которого обратна $P(z)$ с точностью до скалярного коэффициента усиления F :

$$R(z) = F/P(z). \quad (16.32)$$

Таблица 16.4. Число битов при кодировании параметров 20-мс сегмента

Параметры	Число битов	Примечания
Анализатор кратковременного предсказания (отношения r_i , $i = 1, \dots, 8$)	36	r_1, r_2 – по 6 битов; r_3, r_4 – по 5 битов; r_5, r_6 – по 4 бита; r_7, r_8 – по 3 бита
Анализатор долговременного предсказания (коэффициент усиления g и задержка d для каждого из 4 подсегментов)	36	g – 2 бита, d – 7 битов
Сигнал возбуждения (номер последовательности n , максимальная амплитуда v , нормированные амплитуды импульсов b_i , $i = 1, \dots, 13$ для каждого из 4-х подсегментов)	188	n – 2 бита, v – 6 битов, b_i – по 3 бита
Всего за 20-мс сегмент	260	

На выходе этого фильтра формируется восстановленный остаток предсказания фильтра-анализатора кратковременного предсказания, который преобразуется решетчатым фильтром-синтезатором, причем параметры фильтра предварительно преобразуются из логарифмических отношений площадей r_i в коэффициенты частичной корреляции k_i . Передаточная характеристика фильтра-синтезатора $H(z)$ обратна характеристике анализатора $A(z)$ с точностью до скалярного коэффициента усиления G :

$$A(z) = G/H(z). \quad (16.33)$$

Далее сигнал фильтруется в блоке постфильтрации, восстанавливающим амплитудные соотношения частотных составляющих сигнала речи, т. е. компенсирующим предыскажение, внесенное входным фильтром блока предварительной обработки кодера. Сигнал на выходе постфильтра является восстановленным цифровым сигналом речи. Разработано несколько вариантов GSM кодеков.

Для кодирования вызова при временном разделении каналов используется стандартный кодек GSM FR (Full-rate).

Для увеличения возможности одновременного обслуживания двух абонентов разработан так называемый «половинный» кодек GSM HR (Half-rate). Этот кодек «укладывает» звук в среднем в 5,6 кбитов/с, естественно, с потерей качества относительно GSM FR. В результате такого использования возможно появление различных артефактов и «кваканья» при разговоре.

Далее семейство кодеков пополнилось сначала кодеком GSM EFR (Extended full rate), который обеспечивает лучшее качество голоса, чем GSM FR (при меньшей необходимой полосе пропускания 12,2 против 13,3 кбитов/с), а затем — кодеком GSM AMR (Adaptive multi-rate). Главное отличие кодека GSM AMR — способность работать, используя более широкий (но конечный) набор скоростей передачи данных при сопоставимом или более высоком, чем у базовых кодеков (GSM FR и GSM HR), качестве (в зависимости от доступной полосы пропускания).

16.9. GIPS iLBC — кодек, предназначенный для передачи речи по IP-сетям

Кодек GIPS iLBC (Global IP Sound internet Low-Bitrate Codec) рекомендован специальной комиссией интернет-разработок (Internet Engineering Task Force — IETF) для использования передачи речевой информации в Интернете. Кодек iLBC позволяет обеспечить относительно небольшую деградацию качества речи в случае потерь пакетов данных, которые происходят в канале связи из-за потерь или задержек IP-пакетов [6.41, 6.42].

Возможности кодека iLBC:

- цифровой поток 13,33 кбитов/с (399 битов, объединенные в пакет размером 50 байтов) при кодировании речи блоками по 30 мс и 15,2 кбитов/с (303 бита, объединенные в пакет размером 38 байтов) при кодировании речи блоками по 20 мс;
- качество выше, чем у кодека G.729A, высокая устойчивость к потерям пакетов;
- сложность вычислений находится на уровне кодека G.729A;
- кодек имеет свободные условия лицензирования.

На рис. 16.9а приведена упрощенная структурная схема кодирующего устройства iLBC.

Предварительная обработка речевого сигнала осуществляется с помощью фильтра верхних частот с граничной частотой, равной 90 Гц. При этом обеспечивается устранение постоянной составляющей и фоновых помех. Дискретизация сигнала осуществляется с частотой 8 кГц, используется исходно 16-битное линейное квантование.

При кодировании с линейным предсказанием (LPC) входная информация разделяется на блоки по 240 отсчетов (при интервале 20 мс) или 300 отсчетов (при интервале 30 мс), причем в первом случае 80 отсчетов используется из предыдущего блока и 160 — из текущего, а во втором случае 60 — из предыдущего и 240 — из текущего.

Каждый из блоков в 160/240 отсчетов разделяется на 4/6 подблоков, по 40 отсчетов сигнала каждый. Кодирование с линейным предсказанием проводится в несколько этапов.

Во-первых, производится расчет 11 коэффициентов корреляции следующим образом. Блок, состоящий из 240 отсчетов $X(i)$, предварительно умножается на

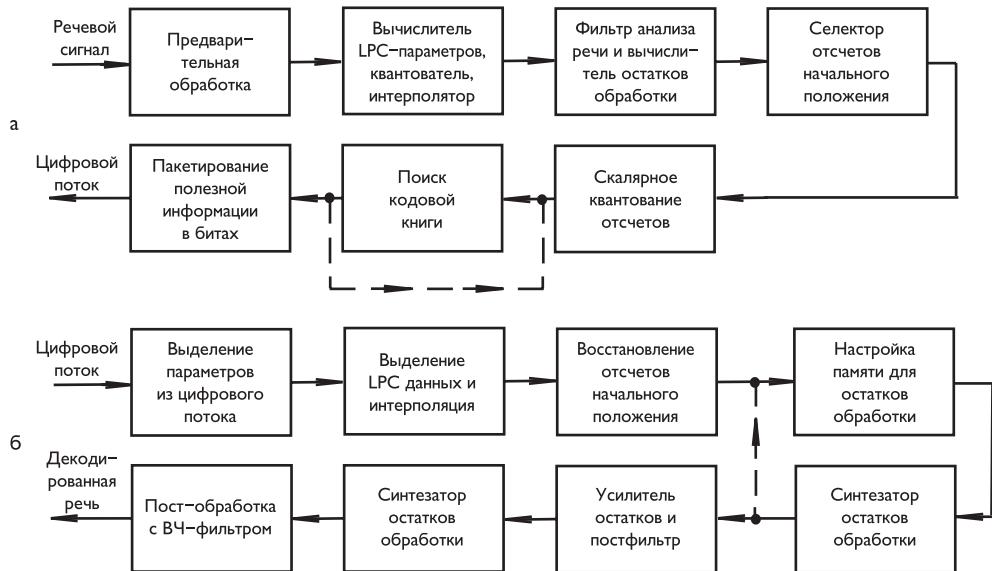


Рис. 16.9. Упрощенные структурные схемы кодирующего (а) и декодирующего (б) устройств стандарта iLBC

окно Ханна, центрированное относительно центра 3-го подблока в 40 элементов, а затем коэффициенты корреляции вычисляются следующим образом:

$$Ak_1(m) = \sum_{n=0}^{N-1-m} X_{G1}(n) \cdot X_{G1}(n+m), \quad (16.34)$$

где $X_{G1}(n)$ — отсчеты сигнала, умноженные на отсчеты окна Ханна, $N = 240$, $0 \leq m \leq 10$.

Затем коэффициенты $Ak(m)$ умножаются на экспоненциальное окно:

$$\hat{Ak}_1(m) = Ak_1(m) \cdot \exp \left[-\frac{1}{2} \cdot \left(\frac{2\pi \cdot 60m}{f_d} \right)^2 \right], \quad 0 \leq m \leq 10, \quad (16.35)$$

и к значению $\hat{Ak}(0)$ добавляется число 0,0001, соответствующее мощности шума на 40 дБ ниже мощности речевого сигнала; $f_d = 8000$ — частота дискретизации сигнала.

Второй набор коэффициентов автокорреляции, $Ak_2(m)$ формируется подобным образом. При этом используется новое окно, применяемое на весь блок к отсчетам 60–299 и состоящее из половины окна Ханна с полным интервалом в 440 отсчетов и косинусным спадом (центрированного относительно пятого подблока в 40 элементов):

$$X_{G2}(n) = X(n) \cdot \begin{cases} \sin^2 \frac{\pi(n+1)}{441}, & n = 0, \dots, 219; \\ \cos \frac{\pi(n-220)}{40}, & n = 220, \dots, 239. \end{cases} \quad (16.36)$$

Значения $Ak_2(m)$ и $\hat{Ak}_2(m)$ вычисляются также по формулам (16.34) и (16.35) при замене отсчетов $X_{G1}(n)$ на $X_{G2}(n)$.

Из $2 \cdot 11$ полученных коэффициентов автокорреляции при использовании известного рекурсивного метода Левинсона–Дурбина [6.43] вычисляются $2 \cdot 11$ коэффициентов линейного предсказания LPC — $lp_1(m)$ и $lp_2(m)$. Первый коэффициент всегда равен 1,0.

Для того чтобы сгладить возможные спектральные пики в коротком окне LPC, коэффициенты умножаются на уменьшающиеся коэффициенты:

$$\hat{lp}_1(m) = 0,9^m \cdot lp_1(m), \quad \hat{lp}_2(m) = 0,9^m \cdot lp_2(m), \quad m = 0, \dots, 10. \quad (16.37)$$

Два набора этих коэффициентов LPC характеризуют спектральные особенности речи на коротком интервале в пределах текущего блока. Они должны быть проквантованы и интерполированы. Но сначала более выгодно преобразовать параметры LPC в другой тип представления, так называемые линии спектральных частот (LSF). Параметры LSF лучше подвергаются квантованию и интерполяции, чем коэффициенты LPC.

Существует множество эффективных в вычислительном отношении методов для расчета коэффициентов LSF, один из которых приведен в стандарте [6.41]. Рассчитывается два массива по 10 таких коэффициентов — $isf_1(m)$ и $isf_2(m)$.

Эти LSF коэффициенты квантуются таким образом, что на их передачу в одном блоке речи должно приходиться 40 битов, т. е. по 20 битов для каждой из последовательностей $isf_1(m)$ и $isf_2(m)$. Использование векторного квантования (VQ) реализует преобразование $isf_1(m)$ и $isf_2(m)$ в $qlsf_1(m)$ и $qlsf_2(m)$, состоящих из трех подвекторов, с количеством 3–3–4 коэффициентов каждый (1–3, 4–6 и 7–10 коэффициенты). Для каждого из подвекторов имеются три кодовые книги размером {64, 128, 128}.

Каждый набор из трех коэффициентов $qlsf_1(m)$ или $qlsf_2(m)$ кодируется 6, 7 и 7 битами, так что общее количество битов равно 40.

Далее производится интерполяция как неквантованных, так и квантованных LSF-коэффициентов, учитывая, что коэффициенты $isf_1(m)$ и $qlsf_1(m)$ соответствуют 3-му подблоку, а коэффициенты $isf_2(m)$ и $qlsf_2(m)$ — 5-му подблоку 30-мс интервала речевой информации. Коэффициенты LSF определяются следующим образом:

- для 6-го подблока они устанавливаются равными коэффициентам 5-го подблока;
- для 4-го подблока вычисляются линейной интерполяцией между соответствующими коэффициентами 3-го и 5-го подблоков;
- для 1-го и 2-го подблоков вычисляются линейной интерполяцией между соответствующими коэффициентами 3-го подблока и 6-го подблока предыдущего блока.

Затем производится преобразование неквантованных и квантованных LSF-коэффициентов каждого подблока, соответственно $ak(m)$ и $\hat{ak}(m)$, в LPC коэффициенты с использованием линейных фильтров:

$$Ak(z) = 1 + \sum_{m=1}^{10} ak(m) \cdot z^{-m}, \quad \hat{Ak}(z) = 1 + \sum_{m=1}^{10} \hat{ak}(m) \cdot z^{-m}. \quad (16.38)$$

При используемом интервале в 20 мс вычисляется единственный набор коэффициентов автокорреляции блока $80 + 160 = 240$ отсчетов сигнала. Используется

такой же расчет десяти коэффициентов LSF, их разделение на три подвектора размера 3, 3 и 4, использование той же кодовой книги и принципа квантования. Интерполяция коэффициентов осуществляется между наборами коэффициентов подблоков текущего и предыдущего блоков.

Фильтр анализа речи с использованием вычисленных LPC-коэффициентов в каждом подблоке в 40 отсчетов обеспечивает вычисление остатка обработки сигнала.

Далее стандарт iLBC рекомендует использование взвешивающего фильтра, учитываяющего восприятие речевой информации, для каждого подблока:

$$W(z) = 1/\hat{A}k(z/0,4222). \quad (16.39)$$

При кодировании начального положения исходно отводится 6 битов на блок и 3 бита — на подблоки во взвешенной речевой области. Производится поиск 2 подблоков наибольшей мощности с применением взвешивающих коэффициентов $\{0,9\ 1,0\ 0,9\}$ для блоков длительностью 20 мс и $\{0,8\ 0,9\ 1,0\ 0,9\ 0,8\}$ для блоков длительностью 30 мс.

Положение 2 подблоков наибольшей мощности при интервале в 20 мс кодируется двумя битами — индексы 1, 2 и 3 (0- и 1-; 1- и 2-; 2- и 3-подблоки соответственно), а на интервале 30 мс — тремя битами — индексы 1, 2, 3, 4, 5 (0- и 1-; 1- и 2-; 2- и 3-; 3- и 4-; 4- и 5-подблоки соответственно). Для того чтобы сократить оценку начального состояния, вычисление производится, исключая 57 отсчетов для блоков в 20 мс и 58 отсчетов для блоков в 30 мс. Для определения, какой из отчетов 57/58 используется, отводится 1 бит.

Вычисление мощности осуществляется следующим образом:

- сначала вычисляется мощность первых 23/22 отсчетов (путем суммирования квадратов их величин) и последних 23/22 отсчетов;
- затем значение меньшей из этих двух величин отбрасывается.

Блок остаточных LPC коэффициентов подвергается фильтрации и квантованию. Фильтрация с применением фазодисперсионного фильтра осуществляется с целью сглаживания распределения величин отсчетов в остаточном сигнале. Фильтрация осуществляется циклической сверткой при начальной инициализации, равной нулю. Z-характеристика фильтра:

$$P_k(z) = \hat{A}kr(z)/\hat{A}k(z), \quad (16.40)$$

где $\hat{A}kr(z) = z^{-M} + \sum_{m=0}^{M-1} \hat{a}k(m) \cdot z^{m-(M-1)}$.

Из сформированного блока находится отсчет с максимальным значением, десятичный логарифм которого преобразуется 6-битовым квантователем до ближайшей величины q_{\max} . Все остальные отсчеты нормализуются и сравниваются со шкалой с дискретностью $4,5 \cdot 10^{-q_{\max}}$.

Нормализованные отсчеты преобразуются взвешивающим фильтром, учитываяющим восприятие звуковой информации, подвергаются ДИКМ и скалярно квантуются.

Последовательность отсчетов в блоке фильтруется с использованием фильтра с характеристикой $W_k(z)$, формируя взвешенные отсчеты $x[n]$. Затем из этих отсчетов вычитаются предсказанные значения $y[n]$, формируемые в цепи обратной

связи с применением фильтра с характеристикой $Pk(z) = 1 - 1/Wk(z)$, а целевые отсчеты $d[n] = x[n] - y[n]$ скалярно квантуются.

Закодированные отсчеты $u[n]$ получаются путем квантования $d[n]$ 3-битным квантователем с заданными параметрами.

Целью кодирования с использованием динамической кодовой книги является либо обработка остатков 23/22 отсчетов двух соответствующих подблоков, определяющих начало обработки, либо обработка 40 отсчетов подблока. Объем памяти книги равен 147 для двух/четырех подблоков с 40 отсчетами и 85 — для 23/22 отсчетов на подблок.

Поиск кодовой книги осуществляется на каждой стадии кодирования. Дополнительно к определению кодовых книг с учетом обновления взвешивающей цепи производится коррекция коэффициента усиления кодера, обеспечивающая максимальное подавление остаточного сигнала.

Последним этапом кодирования является пакетирование полезной информации. Структура битового потока на выходе кодера приведена в табл. 16.5.

Таблица 16.5. Структура цифрового потока стандарта iLBC

Параметры		Биты		
		Блок 20 мс	Блок 30 мс	
Коэффициенты LSF	LSF1	Часть 1	6	
		Часть 2	7	
		Часть 3	7	
	LSF2	Часть 1	—	
		Часть 2	—	
		Часть 3	—	
Общее количество		20	40	
Тип блока		2	3	
Позиция 22 отсчета в сегменте		1	1	
Фактор шкалы состояния кодирования		6	6	
Квантование остатков состояния отсчетов	Отсчет 0	3	3	
	Отсчет 0	3	3	
	- <<	- <<	- <<	
	- <<	- <<	- <<	
	Отсчет 56	3	3	
	Отсчет 57	—	3	
	Общее количество	171	174	
Кодовая книга для 22/23 отсчетов блока	Положение 1	7	7	
	Положение 2	7	7	
	Положение 3	7	7	
Общее количество		21	21	
Усиление для 22/23 отсчетов блока	Положение 1	5	5	
	Положение 2	4	4	
	Положение 3	3	3	
Общее количество		12	12	
Указатели для кодовых книг подблоков	Подблок 1	Положение 1	8	
		Положение 2	7	
		Положение 3	7	

Таблица 16.5 (окончание)

Указатели для кодовых книг подблоков	Подблок 2	Положение 1	8	8	
		Положение 2	8	8	
		Положение 3	8	8	
	Подблок 3	Положение 1	-	8	
		Положение 2	-	8	
		Положение 3	-	8	
	Подблок 4	Положение 1	-	8	
		Положение 2	-	8	
		Положение 3	-	8	
Общее количество			46	94	
Усиление для подблоков	Подблок 1	Положение 1	5	5	
		Положение 2	4	4	
		Положение 3	3	3	
	Подблок 2	Положение 1	5	5	
		Положение 2	4	4	
		Положение 3	3	3	
	Подблок 3	Положение 1	-	5	
		Положение 2	-	4	
		Положение 3	-	3	
	Подблок 4	Положение 1	-	5	
		Положение 2	-	4	
		Положение 3	-	3	
Общее количество			24	48	
Индикатор пустых кадров			1	1	
Суммарный поток			304	400	

Структурная схема декодирующего устройства стандарта iLBC приведена на рис. 16.9б. Декодирование для каждого блока начинается с восстановления и интерполяции LPC коэффициентов, затем расшифровывается начальное состояние. Подблоки декодируются в том же порядке, как они были закодированы. Постфильтрация устраниет постоянную составляющую декодированного сигнала. Нижняя граничная частота высокочастотного фильтра установлена равной 65 Гц.

16.10. Speex — свободный кодек, предназначенный для передачи речи по IP-сетям

Speex — свободный кодек для сжатия речевого сигнала. Он был создан в качестве альтернативы дорогим коммерческим кодекам для кодирования/декодирования голоса. Он оптимизирован для Интернет-приложений, Voice over IP (VoIP) и создания файлов со сжатым звуком. Кодек доступен по модифицированной свободной программной лицензии Университета Беркли (BSD license) и является частью проекта свободного программного обеспечения GNU. Он обладает множеством особенностей, которые отсутствуют в других кодеках. Speex является форматом сжатия с потерями, обеспечивает возможность изменения скорости

передачи цифровых данных (variable bitrate, VBR), т. е. может динамически менять скорость передачи данных в ответ на изменение условий сети [6.44–6.46].

В кодеке возможно использование трех частот дискретизации: 8, 16 и 32 кГц; допускается использование и других частот, но Speex не оптимизирован для использования с ними. Кодек может использоваться для каналов со скоростью передачи данных от 2,15 до 22,4 кбита/с.

К особенностям кодека Speex следует отнести:

- алгоритм кодирования — линейное предсказание с кодовым возбуждением (Code Excited Linear Prediction, CELP);
- переменный цифровой поток (VBR) и динамическое переключение скорости потока;
- использование детектора активности голоса (VAD);
- реализация кодирования стереофонического звука, использование подавителей эхосигналов и шума;
- устойчивость к потере данных;
- возможность передачи с перерывами (DTX);
- может быть инкапсулирован в медиаконтейнер Ogg, являющийся основным файловым и потоковым форматом для свободных мультимедиакодеков в Интернете.

Процесс кодирования речи контролируется специальным параметром, представляющим собой число в диапазоне от 0 до 10. В режиме с постоянным цифровым потоком (Constant bit-rate, CBR) параметр качества — целое число, а в режиме с переменным потоком — число с плавающей точкой.

Speex позволяет изменять сложность работы кодирующего устройства. Процесс управления осуществляется с помощью целого числа от 1 до 10. В обычных условиях уровень шума при сложности 1 на 1–2 дБ выше, чем при сложности 10, но загрузка процессора при сложности 10 выше приблизительно в 5 раз. На практике лучшим выбором, вероятно, будет диапазон чисел от 2 до 4, но для кодирования неречевых сигналов могут оказаться полезными и более высокие значения.

Speex поддерживает большое количество приложений, от потоковых приложений (телефонии) до видеоигр и программ обработки звука.

Speex VBR — кодек, реализующий динамически изменяющуюся скорость передачи звуковой информации (Variable Bit-Rate — VBR). Этот режим может обеспечить более низкую скорость передачи с постоянным качеством или лучшее качество при постоянной скорости передачи данных. Несмотря на свои преимущества Speex VBR имеет три основных недостатка. Во-первых, задание качества обработки звука не гарантирует определенную среднюю скорость передачи. Во-вторых, для некоторых приложений реального времени (например, VoIP), главной является максимальная скоростью передачи данных, что не всегда реализуется. И, в-третьих, шифрование Speex VBR-кодирования речи не может обеспечить полную конфиденциальность — передаваемую фразу можно определить, анализируя изменение скорости передачи.

Использование «средних скоростей» передачи данных (Average Bit-Rate — ABR) решает одну из проблем Speex VBR. Поскольку качество и скорость пере-

дачи данных настраиваются в режиме реального времени, общее качество будет несколько ниже, чем при реализации кодека Speex VBR.

Применение системы определения голосовой активности (Voice Activity Detection – VAD) реализует возможность анализа, кодируется ли речь или молчание/фоновый шум. VAD всегда активируется при кодировании в Speex VBR. В этом случае Speex обнаруживает неречевые интервалы и кодирует их малым количеством битов для воспроизведения фоновых шумов.

Значительным преимуществом Speex в дополнение к VAD/VBR является возможность разрывной передачи/приема.

Для улучшения восприятия (Perceptual enhancement) принимаемой информации при включении декодер пытается уменьшить уровень шума, возникающего в процессе кодирования/декодирования. В большинстве случаев процесс улучшения восприятия делает звук объективно более отличающимся от оригинала, но субъективно лучше воспринимаемым.

Кодек всегда вносит алгоритмическую задержку информации. Для кодека Speex эта задержка равна размеру и интервалу, необходимому для обработки каждого кадра. В узкополосной системе (частота дискретизации равна 8 кГц) задержка составляет 30 мс, а для широкополосной системы (16 кГц) она составляет 34 мс. Эти значения не учитывают процессорное время, необходимое для кодирования и декодирования.

На рис. 16.10 приведены графики сравнения качества кодеков Speex и других кодеков, используемых для передачи речи по Интернет-каналам.

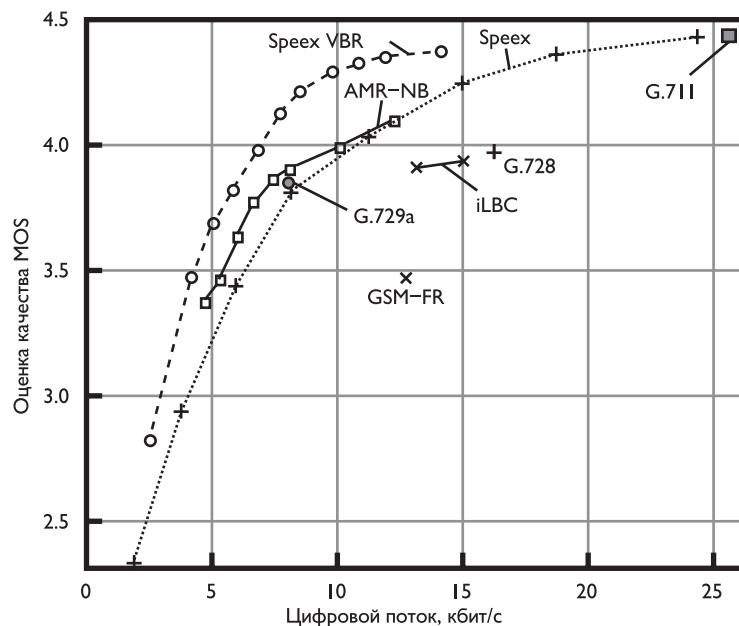


Рис. 16.10. Сравнение качества кодирования речи различными кодеками

ГЛАВА 17

СТАНДАРТЫ КОДИРОВАНИЯ ЗВУКОВОЙ ИНФОРМАЦИИ

Несмотря на многообразие используемых методов обработки звуковой информации в кодирующих устройствах с потерями их структуру можно представить в виде обобщенной структурной схемы рис. 17.1 [6.22]. На вход такой схемы поступает ИКМ-аудиосигнал с заданными частотой дискретизации f_d и линейной шкалой квантования дискретных отсчетов N бит/отсчет.

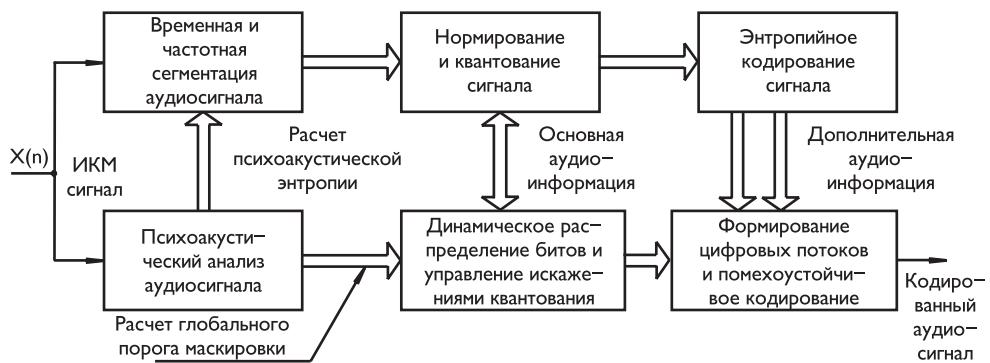


Рис. 17.1. Обобщенная структурная схема кодирующего устройства аудиосигнала с потерями

В блоке частотной и временной сегментации сигнал $X(n)$ разделяется на субполосные составляющие и сегментируется по времени. Длина кодируемой выборки зависит от формы временной функции аудиосигнала: в случаях резких изменений уровня сигнала используется выборка малой длительности, что обеспечивает высокое разрешение по времени; в противном случае применяется длительная по времени выборка, реализуется при этом более высокое разрешение по частоте.

Управление длиной выборки кодируемого сигнала осуществляется блоком психоакустического анализа. После сегментации субполосные сигналы нормируются, квантуются и кодируются. В ряде алгоритмов компрессии кодированию подвергаются не дискретные выборки сигнала, а элементы их преобразования с помощью модифицированного дискретного косинусного преобразования МДКП.

Обычно при компрессии аудиоинформации используется энтропийное кодирование, использование которого позволяет учитывать статистические характеристики сигнала. Учет слухового восприятия звуковых сигналов обеспечивается в процессе психоакустического анализа. Блок динамического распределения битов в соответствии с используемой психоакустической моделью выделяет мини-

мально необходимое количество битов для сигналов каждой субполосы кодирования.

В современных алгоритмах кодирования используются также специальные итерационные процедуры, позволяющие управлять величиной энергии искажений квантования в субполосах при недостаточном числе доступных для кодирования битов.

Для обеспечения правильного декодирования передаваемой информации к декодеру кроме основного цифрового потока данных передается также определенная дополнительная информация. Цифровые потоки основной и дополнительной информации форматируются в выходном блоке кодирующего устройства.

Алгоритмы сжатия высококачественных звуковых сигналов весьма гибки, что при различных реализациях может обеспечить существенное различие в качестве декодируемой информации при одной и той же степени ее сжатия.

Информация об основных стандартах кодирования аудиоинформации приведена в табл. 17.1.

Таблица 17.1. Стандарты кодирования аудиоинформации

Название	Стандарт	Год публикации	Частота дискретизации, кГц	Число каналов передачи	Скорость битового потока на канал, кбитов/с
MPEG-1 Audio, Layer-1	ISO/IEC 11172-3	1993	48	1; 2	32–448
MPEG-1 Audio, Layer-2	ISO/IEC 11172-3	1993	32; 44,1; 48	1; 2	32–384
MPEG-1 Audio, Layer-3	ISO/IEC 11172-3	1993	32; 44,1; 48	1; 2	32–320
MPEG-2 Audio	ISO/IEC 13818-3	1995	16, 22, ..., 48	1–5,1	32–384
MPEG-2 Audio	ISO/IEC 13818-7	1997	16, 22, ..., 48	1–5,1	16–384
MPEG-4 Audio	ISO/IEC 14496-3	1998	16, 22, ..., 48	1–5,1	2–64
Dolby AC-3	ATSC A/53	1995	44,1	1–561	32–384

17.1. MPEG-1 Audio — кодирование аудиоинформации для цифровых систем с потоком до 1,5 Мбит/с

Этот стандарт и его расширение MUSICAM (Layer-2), разработанное специалистами Corporate Computer Systems, Inc. (США), применяется для звукового сопровождения телевизионных программ MPEG-2.

В стандарте MPEG-1 содержатся две психоакустические модели, которые могут альтернативно служить в качестве основы для редукции аудиоданных при их обработке с целью сокращения избыточности передаваемых цифровых звуковых сигналов. Возможная редукция аудиоданных в обеих моделях различна. Однако для декодирования этих сигналов используется один и тот же декодер.

В стандарте MPEG-1 рекомендовано использование трех уровней (слоев) компрессии аудиоинформации: Layer-1, Layer-2 и Layer-3 [6.47].

Layer-1 применяется в профессиональных системах записи/перезаписи с высоким студийным качеством, характеризуется небольшой сложностью и невысокой степенью искажений при нескольких преобразованиях. Обеспечивает использование достаточно большой скорости цифрового потока — 192–256 кбитов/с, коэффициент компрессии — около 4, задержка сигнала при обработке — около 20 мс.

Layer-2 используется для высококачественного телерадиовещания; обладает средней сложностью преобразования сигналов и средней величиной их компрессии. Рекомендуемая скорость цифрового потока — 128 кбитов/с при кодировании звукового сигнала с полосой частот 40 Гц — 15 кГц, коэффициент компрессии — около 6, задержка сигнала при обработке — 40–50 мс.

Layer-3 применяется в основном для Интернет-вещания, отличается высокой сложностью обработки аудиоинформации, обеспечивая скорость передачи цифровой информации 64 кбитов/с при кодировании звукового сигнала с полосой частот 40 Гц — 15 кГц, время задержки — более 50 мс.

17.1.1. MPEG-1 Audio Layer-1 — профессиональная система кодирования аудиоинформации с высоким качеством

Этот вариант стандарта разработан с использованием метода субполосного кодирования системы компрессии аудиоинформации MUSICAM [6.48].

Исходный цифровой ИКМ-сигнал с частотой дискретизации $f_d = 48$ кГц, линейной шкалой квантования $N = 16$ бит/отсчет и цифровым потоком $v = 768$ кбитов/с разделяется банком полифазных квадратурных зеркальных фильтров (Polyphase Quadrature Mirror Filter — PQMF) на $M = 32$ субполосные составляющие (рис. 17.2). Аудиофрейм (кадр) состоит из 384 отсчетов звукового сигнала, интервал передачи аудиофрейма равен 8 мс. Все субполосы обладают постоянной шириной полосы, равной $f_d/M = 750$ Гц.

В каждой субполосе кодирования имеется $384/32 = 12$ отсчетов сигнала, при этом максимальное значение отсчета в каждой субполосе является масштабным коэффициентом (Scale Factor — SCF). Передаваемые декодеру значения SCF заданы в стандарте таблично, поэтому подбирается ближайшее наибольшее значение, а каждому табличному значению SCF соответствует свой индекс i_{SCF} , представляющий собой целое число, изменяющееся от 0 до 62. Он кодируется 6 битами. Для каждой субполосы кодируется и передается лишь один индекс, либо он вообще не передается, если отсчеты данной полосы не кодируются и не передаются. Информация о выбранных масштабных коэффициентах (Scalenefactors Selection Information — SCFSI) кодируется 2 битами для каждой субполосы. Определение значений SCF выполняется в блоке расчета масштабных коэффициентов.

После расчета масштабных коэффициентов производится нормировка субполосных отсчетов сигнала с использованием значений SCF в качестве нормирующих множителей. Затем отнормированные отсчеты равномерно квантуются и кодируются. Для каждой субполосы число ступеней квантования задается приведенной в стандарте табл. 17.2.

Для каждой частоты дискретизации f_d и скорости передачи информации v существует своя таблица набора квантователей и ограничения используемой макси-



Рис. 17.2. Структурная схема кодера стандарта MPEG-1 Audio Layer-1

мальной субполосы. Требуемая величина шага квантования Δ_m отсчетов аудиосигнала для каждой субполосы вычисляется с учетом отношения сигнал/маска $SMR(m)$, определенного с помощью психоакустической модели (см. раздел 15.7.2, психоакустическая модель-1) и определяющего минимально возможное отношение энергии полезного сигнала к энергии шумов квантования в субполосе m , при котором шум еще маскируется полезным сигналом.

Таблица 17.2. Набор квантователей для скоростей передачи 56–384 кбитов/с

Индекс субполосы кодирования	Число ступеней квантования (коэффициенты k^- и k^+ определяют индексы квантователя, равные $2^k - 1$ и $2^k + 1$)														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0, 1, 2	2 ⁻	3 ⁻	4 ⁻	5 ⁻	6 ⁻	7 ⁻	8 ⁻	9 ⁻	10 ⁻	11 ⁻	12 ⁻	13 ⁻	14 ⁻	15 ⁻	16 ⁻
3, ..., 10	2 ⁻	2 ⁺	3 ⁻	3 ⁺	4 ⁻	5 ⁻	6 ⁻	7 ⁻	8 ⁻	9 ⁻	10 ⁻	11 ⁻	12 ⁻	13 ⁻	16 ⁻
11, ..., 22	2 ⁻	2 ⁺	3 ⁻	3 ⁺	5 ⁻	16 ⁻									
23, ..., 26	2 ⁻	2 ⁺	16 ⁻												

В психоакустической модели для сокращения объема цифровой информации кодера MPEG-1 Audio производятся следующие вычисления:

- расчет спектральных компонентов с помощью БПФ (переход из временной в частотную область);
- определение уровня интенсивности сигнала в каждой частотной полосе;
- определение абсолютного порога слышимости;
- выделение тональных (синусоподобных) и нетональных частей (шумоподобных) спектральных компонент;
- редуцирование спектральных компонент;

- вычисление индивидуальных кривых маскировки, оставшихся после редуцирования спектральных компонент;
- вычисление глобального порога маскировки;
- вычисление наименьшего значения порога маскировки для каждой частотной субполосы;
- расчет SNR (отношения сигнал/шум) для каждой субполосы.

Число ступеней квантования при кодировании отсчетов сигнала в субполосах составляет 2^k , где k — число разрядов в кодовом слове отсчета, равное величине от 2 до 16.

Кодируется не число ступеней квантования, а соответствующий им индекс — в соответствии с табл. 16.2 кодовые слова для соответствующих субполос определяются следующим количеством битов:

- для субполос 0, 1, 2 — $4 \cdot 3 = 12$;
- для субполос 3, …, 10 — $4 \cdot 8 = 32$;
- для субполос 11, …, 22 — $3 \cdot 12 = 36$;
- для субполос 22, …, 26 — $2 \cdot 8 = 16$.

Следовательно, количество информации, необходимое для распределения битов по субполосам в аудиофрейме, в данном случае равно $12 + 32 + 36 + 8 = 88$ битов.

Процедура квантования отсчетов во фрейме выполняется в соответствии с соотношением:

$$S_{q, sb}(k) = A \cdot S_{sb}(k) + B, \quad (17.1)$$

где A и B — коэффициенты квантования, определяемые стандартом для каждого числа ступеней квантования, $S_{sb}(k)$ и $S_{q, sb}(k)$ — нормированные и квантованные k -е значения субполосного отсчета соответственно.

Последовательность операций, выполняемых в MPEG-1 Audio кодере, представлена на рис. 17.3.

После того как в кодере определены уровни сигнала в отдельных частотных субполосах, определяется абсолютный порог слышимости и анализируется, идет ли речь о тональных (синусоподобных) или нетональных (шумовых) частях сигнала. На основе использованной психоакустической модели идентифицируются доли (части) сигнала, которые в значительной степени маскируются, и затем вычисляется отношение SNR, при котором шум еще маскируется полезным сигналом. Эти расчеты выполняются для каждой субполосы. С помощью вычисленных значений SNR и уровней сигнала для каждой субполосы рассчитываются шаги квантования и необходимое для кодирования число битов. Общее число битов распределяется между субполосами так, чтобы выполнялись требования, относящиеся как к скорости передачи сигнала, так и к маскировке шумов в каждой субполосе.

В соответствии со стандартом MPEG-1 Audio Layer-1 основной единицей цифрового потока является аудиофрейм, структура данных внутри которого строго определена (рис. 17.4).

В преамбуле (заголовке) аудиофрейма (Header) передается служебная информация (32 бита), содержащая составляющую синхронизации из 12 битов (1111 1111 1111) и информацию о структуре данных фрейма (20 битов):

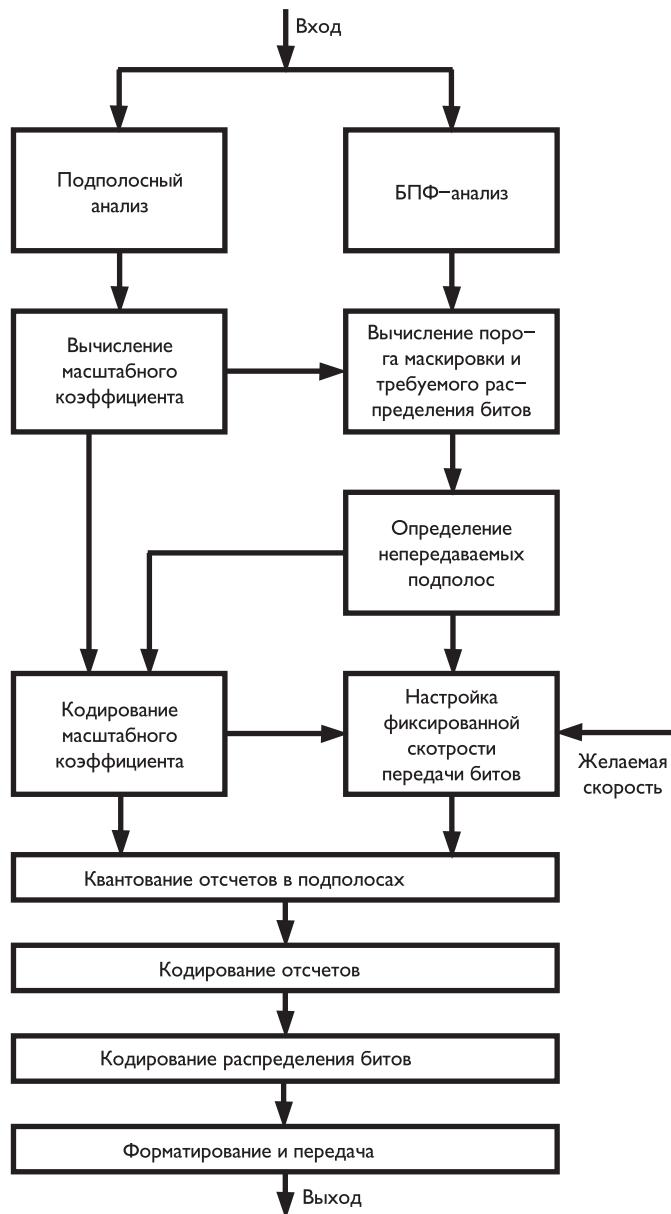


Рис. 17.3. Алгоритм кодирования звукового сигнала в соответствии со стандартом MPEG-1 Audio

- бит идентификации (ID), равный «1», если поток полностью соответствует стандарту ISO/IEC 11172-3, и «0» в противном случае;
- код уровня (2 бита), определяющий тип кодирования («Layer-1», «Layer-2», «Layer 3»);
- бит защиты, равный «1», если применяется помехоустойчивое кодирование, и «0» — в противном случае;

Пreamble (Header)	Помехоустойчивое кодирование (CRC)	Распределение битов по субполосам (BAL)	Индексы масштабных коэффициентов (SCF)	Коды субполосных нормированных коэффициентов (Subband Samples)	Дополнительные данные (AD)
Синх. 12 битов; Сист. инф. 20 бит	16 битов	4 бита	6 битов 	Одна субполоса содержит информацию о 32 отсчетах входного ИКМ-аудиосигнала	Дополнит. данные

Рис. 17.4. Структура цифровых данных аудиофрейма MPEG-1 Audio Layer-1

- значение скорости цифрового потока для каждого типа кодирования (4 бита);
- частоту дискретизации для каждого типа кодирования (2 бита);
- бит, равный «1», если частота дискретизации равна 44,1 кГц, и «0» — в противном случае;
- бит для использования в специальных целях, например при передаче дополнительной информации;
- код режима передачи: «Stereo», «Joint Stereo», «два независимых канала», «один канал» (2 бита);
- код режима расширения (2 бита) при использовании системы «Joint Stereo»;
- бит права копирования, равный «0», если копирование запрещено;
- бит, характеризующий тип передачи «оригинал/копия», равный «1», если передается оригинал;
- код предыскажения (2 бита).

После преамбулы 16 битов отводится для помехоустойчивого кодирования — обнаружения и коррекция возможных ошибок (CRC). Затем следуют данные компрессированного сигнала:

- сначала передается информация о распределении битов по субполосам (Bit Allocation-BAL); в Layer-1 поле этих данных содержит 4 бита;
- затем в аудиофрейме располагается информация о передаваемых масштабных коэффициентах (SCFSI) для каждой из субполос (по 2 бита);
- за этой информацией следует передача кодовых слов индексов самих SCF, для передачи одного масштабного коэффициента используется 6 битов;
- наконец, далее следуют кодовые слова субполосных нормированных отсчетов (Subband Samples), которые объединены в 12 блоков: $B(j)$, $j = 1, \dots, 12$.

Каждый из этих блоков $B(j)$ содержит нормированные отсчеты субполос под номерами от $3j - 2$ до $3j$, если отсчеты данной полосы передаются. Каждый блок содержит 30 подблоков: $C(k)$, $k = 1, \dots, 30$, каждый из которых содержит группу

из трех отсчетов субполосного сигнала. Таким образом, аудиофрейм стандарта MPEG-1 Audio Layer-1 содержит информацию о 384 отсчетах аудиосигнала. Завершается аудиофрейм интервалом, в котором могут передаваться дополнительные данные (Ansillare Data).

Скорость передачи цифровых данных и длина аудиофрейма определяют общее количество битов B_C , используемых на кодирование всей информации выборки сигнала. Из этой величины следует исключить биты преамбулы (B_{HDR} , 32 бита), биты помехоустойчивого кодирования (B_{CRC} , 16 битов CRC кода), данные, необходимые для передачи информации о распределении битов по субполосам (B_{BAL}), и биты, необходимые для передачи дополнительной информации (B_{ANC}). При этом общее количество битов, доступное в каждом аудиофрейме для кодирования информации о масштабных коэффициентах SCFSI, индексов самих масштабных коэффициентов SCF и субполосных отсчетах равно:

$$B_{AD} = B_C - (B_{HDR} + B_{CRC} + B_{BAL} + B_{ANC}). \quad (17.2)$$

Процедура выделения битов на кодирование аудиоинформации обычно производится итерационно. Этот процесс начинается с субполосы, в которой величина SNR, рассчитанная с использованием психоакустической модели, максимальна. Для этой субполосы с номером i сначала выделяются биты, требуемые для передачи информации о распределении коэффициентов по субполосам (SCFSI, 2 бита) и передаче значений этих коэффициентов (SCF, 6 бит). После этого выделяются биты для кодирования отсчетов данной субполосы, начиная с минимально возможной длины кодового слова, определяемой наименьшим числом возможных ступеней квантования отсчетов в данной i -ой субполосе при постепенном увеличении его длины в соответствии с имеющимся набором квантователей.

Процесс выделения бит для передачи данных i -й субполосы продолжается до тех пор, пока фактическое значение отношения SNR в этой полосе не станет больше минимального из множества этих значений, рассчитанных для остальных субполос психоакустической моделью. На этом процесс выделения битов на i -ю субполосу прерывается, а полученное для этой субполосы фактическое значение SNR_i заменяет требуемую величину SNR, рассчитанную ранее.

После этого биты начинают выделяться для новой j -й субполосы, где величина SNR оказывается максимальной. Процесс выделения битов при этом аналогичен вышеописанному и продолжается до тех пор, пока полученное фактическое значение SNR_j не станет больше, чем в какой-либо другой субполосе, включая i -ю субполосу.

Такой итерационный процесс продолжается до тех пор, пока не будет исчерпано все доступное количество битов.

Декодер сигналов MPEG-1 Audio Layer-1 проверяет входящие данные на ошибки, а также разделяет данные управления процессом декодирования и сжатые информационные аудиоданные. Прежде всего, разделенные на отдельные полосы сигналы экспандируются с помощью информации управления, в результате происходит их обратное преобразование в исходную форму. В инверсном блоке фильтров различные спектральные части звукового сигнала вновь объединяются в первоначальный сигнал. Результатом этого является цифровой поток аудиоданных на выходе декодера, который уже подготовлен для цифро-аналогового преобразования.

Режим стереовещания при достаточных скоростях передачи цифровой информации обычно осуществляется путем независимой трансляции правого R_n и левого L_n каналов аудиосигналов.

В случае же использования низких скоростей передачи используется так называемый режим «Joint Stereo», психоакустической основой которого являются следующие положения:

- частоты ниже 150–250 Гц практически не локализуются;
- частоты выше 8–10 кГц весьма часто не влияют на оценку азимута кажущегося источника звука;
- на оцену азимута кажущегося источника звука наибольшее влияние оказывают спектральные составляющие с максимальной энергией, чаще всего расположенные в среднечастотной области звукового сигнала; на частотах 1,5–2,5 кГц положение азимута кажущегося источника звука определяется не тонкой временной структурой сигнала, а его огибающей или соотношением энергий высокочастотных составляющих спектров сигналов стереопары.

Эти свойства и используются в режиме «Joint Stereo», при котором высокочастотные составляющие стереопары объединяются и их сумма передается в общем канале в виде монофонической добавки. Процедура распределения битов по субполосам при кодировании объединенного сигнала выполняется так же, как и при независимом кодировании.

17.1.2. MPEG-1 Audio Layer-2 — система кодирования аудиоинформации для высококачественного телерадиовещания

В MPEG-1 Audio Layer-2 возможно использование трех частот дискретизации — 32, 44,1 и 48 кГц. При $f_d = 48$ кГц аудиофрейм длительностью 24 мс образует выборка, состоящая из 1152 отсчетов звукового сигнала. Структурная схема кодера MPEG-1 Audio Layer-2 такая же, как для Layer-1 (см. рис. 17.2). В каждой из 32 субполос кодирования располагается $1152/32 = 36$ отсчетов аудиосигнала. Каждый блок из 36 отсчетов делится на три подблока — гранулы, и в каждой грануле, содержащей 12 отсчетов, определяется максимальный отсчет, являющийся масштабным коэффициентом SCF, общее количество которых равно $32 \cdot 3 = 96$. Полученные значения этих масштабных коэффициентов сравниваются с заданными в таблице стандарта, ближайшие значения принимаются за SCF каждой гранулы.

В зависимости от различий величин масштабных коэффициентов трех гранул SCF_1 , SCF_2 и SCF_3 субполосы может передаваться разное число этих коэффициентов — от одного до трех, либо ни одного, если отсчеты данной субполосы не кодируются.

Определение числа передаваемых коэффициентов осуществляется с использованием разностей:

$$Dscf_1 = SCF_1 - SCF_2, \quad Dscf_2 = SCF_2 - SCF_3. \quad (17.3)$$

Установлено 5 классов этих разностей, указанных в табл. 17.3 и определяющих количество передаваемых масштабных коэффициентов.

Таблица 17.3. Классы различий пар масштабных коэффициентов

Класс различий	Значение разности
1	$Dscf \leq -3$
2	$-3 < Dscf < 0$
3	$Dscf = 0$
4	$0 < Dscf < +3$
5	$Dscf \geq +3$

Набор квантователей масштабных коэффициентов для цифровых потоков 56–384 кбитов/с для Layer-2 также определяется табл. 17.2.

Процедуры нормирования, квантования и кодирования отсчетов аудиосигналов в кодере MPEG-1 Audio Layer-2 аналогичны соответствующим процедурам в Layer-1.

Структура цифровых данных аудиофрейма MPEG-1 Audio Layer-2 приведена на рис. 17.5.



Рис. 17.5. Структура цифровых данных кодера MPEG-1 Audio Layer-2

Служебная информация в преамбуле и биты, используемые для реализации помехоустойчивого кодирования CRC, аналогичны соответствующим данным цифрового потока кодера MPEG-1 Audio Layer 1.

Длина кодовых слов в поле распределения битов по субполосам (BAL) в кодере Layer 2 переменна, зависит от числа кодируемых субполосных сигналов, частоты дискретизации и изменяется в пределах от 2 до 4 битов. Информация о передаваемых масштабных коэффициентах (SCFSI) использует 2 бита для каждой из субполос. Далее следуют кодовые слова, несущие информацию об индексах самих масштабных коэффициентов (SCF), а затем аудиофрейм несет информацию о 1152 отсчетах звукового сигнала.

Качество передаваемого сигнала прямо пропорционально скорости цифрового потока на выходе кодера и обратно пропорционально ширине полосы частот, занимаемой исходным сигналом. Другими словами, для одного и того же исходного сигнала качество передачи будет тем выше, чем выше скорость цифрового потока

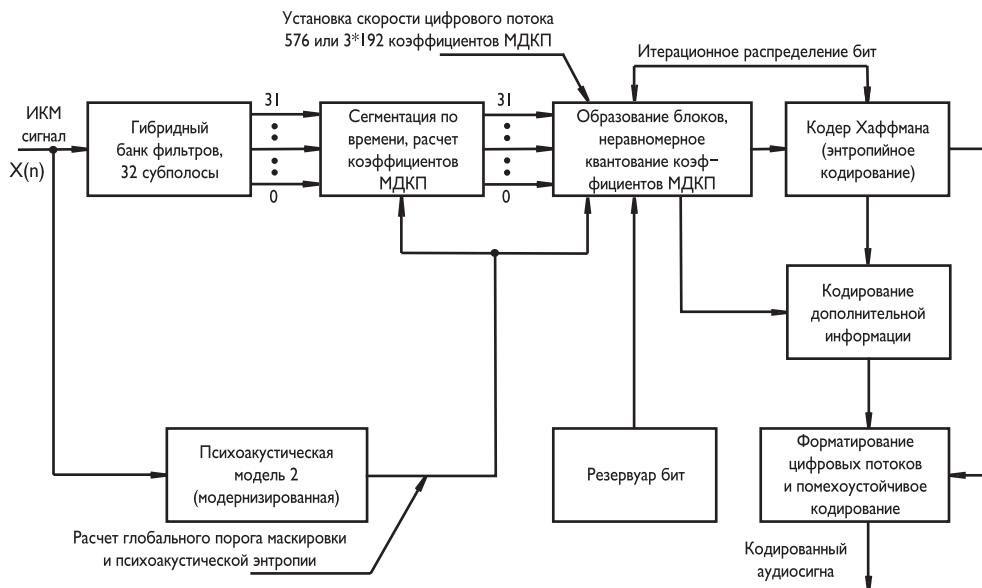


Рис. 17.6. Структурная схема кодера MPEG-1 Audio Layer-3

(меньше степень сжатия). И наоборот, при одной и той же степени сжатия более качественно будет передан сигнал, имеющий небольшой частотный диапазон. Алгоритм MPEG-1 Audio Layer-2 предусматривает различные степени сжатия данных с соответствующей скоростью цифрового потока на выходе: 56, 64, 112, 128, 192, 256 и 348 кбитов/с.

Алгоритм режима работы кодера «Joint-Stereo» аналогичен приведенному в описании кодера MPEG-1 Audio Layer-1.

17.1.3. MPEG-1 Audio Layer-3 — система кодирования аудиоинформации для Интернет-вещания

Специфической особенностью кодера MPEG-1 Audio Layer-3 является использование модифицированного дискретного косинусного преобразования МДКП, аудиофрейм образует выборка, состоящая из 1152 отсчетов сигнала длительностью 24 мс.

Структурная схема кодера MPEG-1 Audio Layer-3 приведена на рис. 17.6. Кодированию в данном случае подвергаются коэффициенты МДКП, в каждой из 32 субполос размещается по 18 коэффициентов.

Входной ИКМ-сигнал поступает на гибридный банк фильтров (Hybrid PQMF Filter Bank), содержащий помимо полифазных квадратурных зеркальных фильтров (PQMF) блок прямого МДКП (MDCT) и частотный корректор ЧК (рис. 17.7).

При помощи банка из 32 PQMF-фильтров 1152 отсчета аудиосигнала разделяются двумя блоками по 576 отсчетов на 32 субполосных составляющих, в каждой из которых выделяется $36 = 18 + 18$ отсчетов сигнала. Далее в каждом таком субканале выполняется МДКП, при этом длина ортогонального преобразования

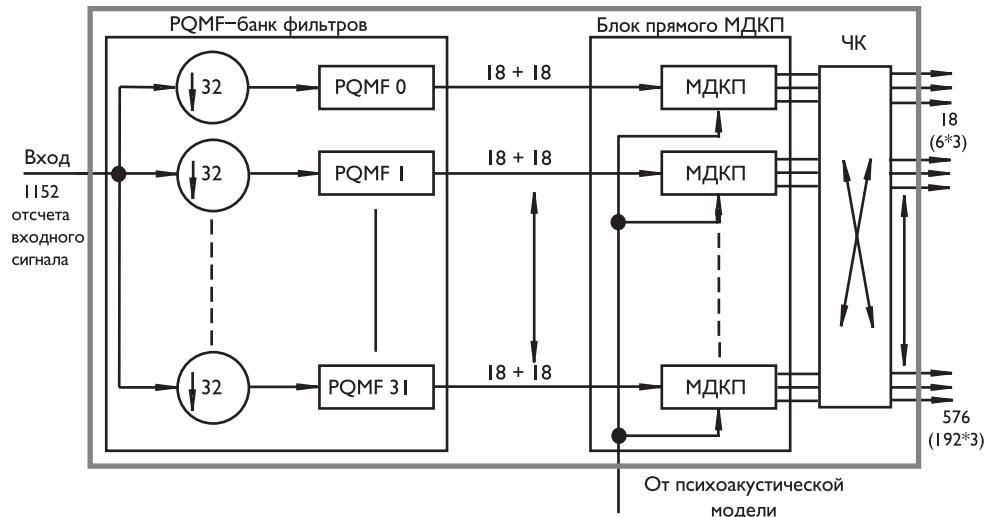


Рис. 17.7. Структурная схема гибридного банка фильтров

ния зависит от формы временной выборки звукового сигнала. При отсутствии всплесков в выборке МДКП выполняется над всей группой из 36 отсчетов, а при наличии всплесков отдельно для трех групп, содержащих по 12 отсчетов сигнала. При длинном преобразовании общее число коэффициентов МДКП во всех субполосах равно $18 \cdot 32 = 576$, а при коротком имеется $6 \cdot 32 = 192$ коэффициента в каждом коротком блоке. Три коротких блока объединяются в гранулу, в которой размещается также $192 \cdot 3 = 576$ коэффициентов МДКП.

Для изменения длины ортогонального преобразования используется набор оконных функций: синусное окно «long» — для длинного преобразования, синусное окно «short» — для короткого преобразования и два окна «start» и «stop», необходимые для уменьшения искажений при переходе от одного преобразования к другому (рис. 17.8).

Частотный корректор используется для уменьшения искажений, возникающих вследствие использования процедуры МДКП. Решение о выборе длины МДКП принимает психоакустическая модель кодера (см. раздел 15.7.2, психоакустическая модель 2). Рассчитанные коэффициенты МДКП масштабируются (нормируются), квантуются и кодируются.

В кодере MPEG-1 Audio Layer-3 применяется неравномерное квантование коэффициентов МДКП, которое выполняется при помощи двух итерационных циклов — внутреннего (Inner Iteration Loop), при котором производится квантование коэффициентов МДКП в соответствии с требованиями к установленной скорости передачи информации, и внешнего (Outer Iteration Loop), приводящего результаты квантования в соответствие с требованиями психоакустической модели. Для кодирования величин квантованных коэффициентов МДКП используются таблицы кодов Хаффмана.

Перед квантованием и кодированием рассчитывается приходящееся на одну гранулу среднее количество битов, вычисляемое по формуле:

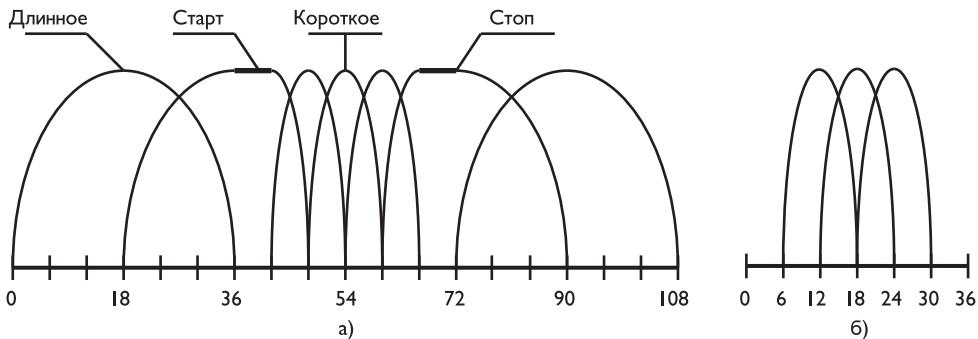


Рис. 17.8. Типы оконных функций, используемых в гибридном банке фильтров (а), и последовательность их переключения при наличии всплеска в выборке (б)

$$mean_bit = \frac{v \cdot N}{N_g \cdot f_d}, \quad (17.4)$$

где v — установленная скорость передачи цифровых данных, f_d — частота дискретизации, $N = 1152$ — число отсчетов в выборке сигнала, $N_g = 2$ — количество гранул в аудиофрейме.

Так как преамбула содержит 32 бита и для передачи дополнительной информации отводится 17 байтов (всего 136 битов), то среднее доступное количество битов, приходящееся на одну гранулу равно:

$$B_{cp}x = mean_bit - \frac{32 + 136}{2}. \quad (17.5)$$

Затем инициализируется количество итерационных циклов и вычисляется величина:

$$quantf = system_const \cdot \ln(smf), \quad (17.6)$$

где $system_const = 210$ — системная константа, smf — параметр, характеризующий спектральную неоднородность выборки сигнала, определяемый как

$$smf = \frac{\exp\left(\frac{2}{M} \sum_{m=0}^{M-1} \ln(xr_m)\right)}{\frac{2}{M} \sum_{m=0}^{M-1} \ln(xr_m)}, \quad (17.7)$$

$M = 576$ — суммарное количество спектральных коэффициентов в грануле, xr_m — m -е значение коэффициента МДКП в грануле. Значение $quantf$ является постоянной величиной для аудиофрейма.

Перед входом во внутренний цикл обработки предварительно проверяется, имеются ли ненулевые коэффициенты МДКП. Если все коэффициенты МДКП равны нулю, то внутренние и внешние циклы обработки не проводятся.

Квантование коэффициентов МДКП во внутреннем итерационном цикле осуществляется в соответствии с соотношением:

$$ix_m = \text{round} \left[\left(\frac{|xr_m|}{\sqrt[4]{2^{qquant} + quantf}} \right)^{3/4} - 0,0946 \right], \quad m = 0, \dots, 576, \quad (17.8)$$

где ix_m и xr_m — квантованное и исходное значение m -го коэффициента МДКП, $qquant$ — указатель шага квантования, определяющий степень компрессии, а $quantf$ — шаг изменения величины $qquant$.

Начальное значение шага квантования принимается равным нулю — $qquant = 0$, а затем изменяется дискретно с шагом 1: $qquant = qquant + 1$ до тех пор, пока не будет получена квантованная величина, лежащая в пределах самых больших значений из множества, заданного таблицей Хаффмана.

На следующем шаге определяется число пар квантованных значений коэффициентов МДКП, равных нулю и обозначаемых «*rzero*». Затем определяется количество четверок коэффициентов, квантованные значения которых не превышают по модулю единицы, включая коэффициенты «*rzero*». Эти четверки обозначаются «*count1*».

Далее определяется число битов, необходимое для кодирования «*count1*» — $\langle bitsum_count1 \rangle$. Для каждого четырехкомпонентного вектора $x_m^{(4)} = [0, 1, 1, 1]$ определяется длина кодовых слов в соответствии с таблицами А и В стандарта MPEG-1 Audio Layer-1. Например, четырехкомпонентному m -му вектору соответствует по таблице А шестибитовое кодовое слово «000100», а по таблице В — четырехбитовое слово «1000». То есть определяется число битов, необходимых для кодирования «*count1*», независимо сначала по таблице А — $\langle bitsum_table0 \rangle$, а затем по таблице В — $\langle bitsum_table1 \rangle$. Из этих двух значений выбирается минимальное и добавляются биты, необходимые для кодирования знака коэффициентов МДКП. Информация об используемой таблице «*count1table_selct*» кодируется одним битом, равным «0» при таблице А и «1» — при таблице В.

Квантованные коэффициенты МДКП — «*bigvalues*», не входящие в состав «*rzero*» и «*count1*», для каждой субполосы кодирования разбиваются на 2 региона при использовании окон «*short*», «*start*» и «*stop*» или 3 региона в противном случае.

После разбиений больших значений квантованных коэффициентов МДКП на регионы для каждого из них определяются кодовые таблицы из 32 таблиц Хаффмана.

Параметрами этих таблиц являются переменные x , y , $hlen$, $hcod$, $linbits$. Величины x и y — пара квантованных коэффициентов МДКП, кодируемых одним кодовым словом $hcod$ длиной $hlen$. В этих таблицах приводятся коды для величин коэффициентов МДКП, меньших 16. Для величин, равных или больших 16, имеются две специальные таблицы. Величина 15 является исключающим символом и кодируется дополнительным кодовым словом с длиной, обозначенной как $linbits$.

Далее подсчитывается общее количество битов, необходимое для кодирования всех коэффициентов МДКП. Если оно меньше доступной величины, то указатель шага квантования $qquant$ увеличивается на единицу и внутренний итерационный цикл повторяется с новым значением $qquant$. Этот процесс повторяется до тех пор, пока общее количество битов не будет чуть превышать установленную скорость цифрового потока.

Внешний итерационный цикл управляет величинами искажений, вызванными квантованием коэффициентов МДКП во внутреннем цикле. Прежде всего рассчитывается энергия искажений квантования, приходящаяся на один коэф-

фициент МДКП для каждой из субполос кодирования m :

$$xfsf_m = \sum_{k=ilow}^{ihigh} \frac{\left[|xr_k| - ix_k^{4/3} \cdot \sqrt[4]{2^{qquant+quantf}} \right]^2}{K_m}, \quad (17.9)$$

где $ilow$ и $ihigh$ — нижний и верхний индексы коэффициентов МДКП в субполосе m ; xr_k и ix_k — исходное и квантованное значение k -го коэффициента МДКП в субполосе m ; K_m — общее число коэффициентов МДКП в субполосе m .

Если в какой-либо из субполос кодирования m — величина энергии искажений квантования превышает значение, рассчитанное психоакустической моделью, то значения всех коэффициентов МДКП в данной субполосе корректируются следующим образом:

$$xr_k = xr_k \cdot \sqrt{2^{(1+scalefac_scale) \cdot scalefac_m}}, \quad k = ilow_m, \dots, ihigh_m, \quad (17.10)$$

где $scalefac_m$ — масштабный коэффициент для субполосы m , равный $scalefac_l_m$ или $scalefac_s_m$ соответственно для длинных и коротких блоков, $scalefac_scale$ — параметр, управляющий величиной шага изменения корректирующего множителя и имеющий только два значения: 0 или 1.

Величина масштабного коэффициента изменяется дискретно от нуля с шагом 1 до тех пор, пока его величина не превысит табличное значение, заданное стандартом.

Корректирующий множитель в соотношении (17.10) может иметь разные показатели степени: 1, 2, $pretab_m$, $2 \cdot pretab_m$ и $scalefac_m$ в зависимости от ряда условий.

Цикл управления шумами квантования исходно устанавливается $scalefac_scale = 0$ и затем к нему добавляется 1, если необходимо увеличить диапазон изменения корректирующих множителей.

Значения масштабных коэффициентов $scalefac_m$ и указатель шага квантования $qquant$ переаются в составе цифрового потока декодеру.

На завершающем этапе внешнего цикла проверяется условие его прекращения, при котором во всех субполосах кодирования отсутствуют искажения квантования с энергией, превышающей определенные психоакустической моделью допустимые значения. Однако не всегда удается выполнить такие условия, особенно при малых скоростях цифрового потока. В этом случае используются другие условия окончания внешнего итерационного процесса.

Каждый фрейм в структуре цифрового потока MPEG-1 Audio Layer-3 содержит преамбулу (заголовок), в которой передается служебная информация, блок дополнительной информации и блок аудиоданных, содержащих две гранулы (рис. 17.9). Каждая гранула содержит информацию о 576 коэффициентах МДКП.

В данном случае предполагается, что в первой грануле передается информация о последовательности коэффициентов МДКП длинного блока, а во второй грануле — последовательности из трех коротких блоков, с информацией о 192 коэффициентах МДКП в каждом.

Алгоритм компрессии «Joint Stereo» реализуется в виде двух режимов: суммарно-разностного кодирования (M/S) и объединения сигналов стереопары. Возможно раздельное и совместное использование этих режимов.

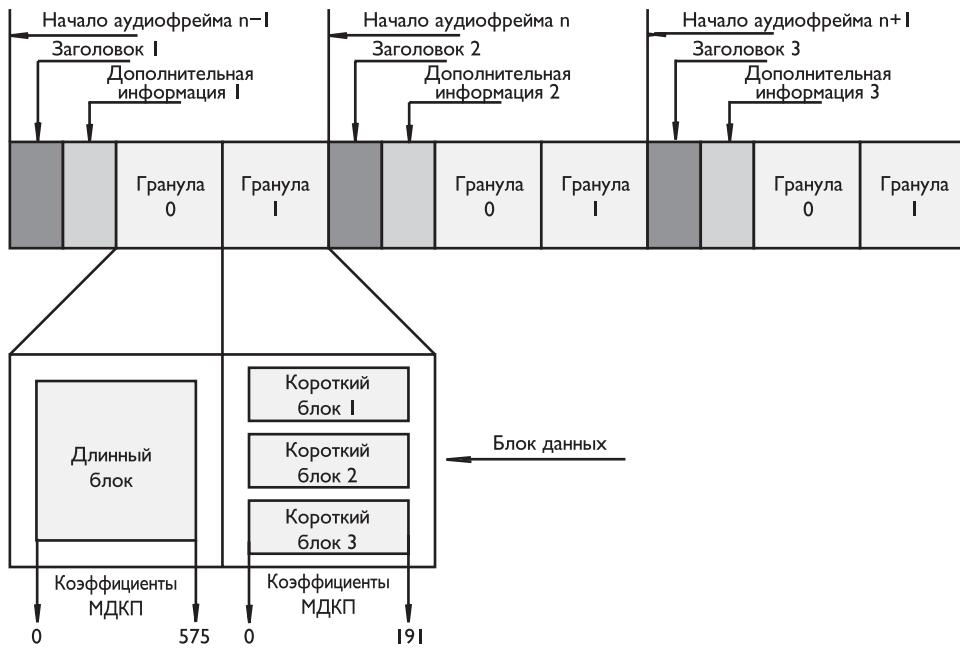


Рис. 17.9. Структура данных аудиофрейма стандарта MPEG-1 Audio Layer-3

Режим M/S кодирования применяется при условии, если

$$\sum_{w=0}^{576} (F_{w,L}^2 - F_{w,R}^2) < 0,8 \cdot \sum_{w=0}^{576} (F_{w,L}^2 + F_{w,R}^2), \quad (17.11)$$

где $F_{w,L}$ и $F_{w,R}$ — спектральные составляющие левого и правого сигналов стереопары.

При работе в данном режиме кодируются независимо суммарный M_m - и разностный S_m -сигналы стереопары в субполосе m по стандартной для Layer-3 процедуре.

В режиме объединения сигналов стереопары коэффициенты преобразования суммы M_m передаются в левом канале, а в правом канале для каждой субполосы передаются так называемые координаты объединения Ψ_m , необходимые для восстановления энергий исходных сигналов стереопары. Для передачи значений координат объединения используется 3 бита в пределах $\Psi_m = 0, \dots, 6$; число $\Psi_m = 7$ означает, что процедура объединения в данной субполосе не использовалась.

При декодировании сначала определяются только те субполосы m , в которых реализована процедура объединения, и определяются координаты Ψ_m для них. Затем находят энергетические коэффициенты q_m в объединенных субполосах:

$$q_m = \operatorname{tg} \left(\Psi_m \cdot \frac{\pi}{12} \right) = \sqrt{\frac{e_{m,L}}{e_{m,R}}}. \quad (17.12)$$

Исходные спектральные компоненты стереопары восстанавливаются следующим

образом:

$$F_{k,L.\text{вост}} = M_k \cdot \frac{q_m}{1 + q_m}, \quad F_{k,R.\text{вост}} = M_k \cdot \frac{1}{1 + q_m} \text{ для всех } k \text{ в субполосе } m. \quad (17.13)$$

Энергия всех восстановленных сигналов стереопары в субполосе m равна:

$$e_{m,L} = \sum_{k=ilow_m}^{ihigh_m} F_{k,L.\text{вост}}^2, \quad e_{m,R} = \sum_{k=ilow_m}^{ihigh_m} F_{k,R.\text{вост}}^2. \quad (17.14)$$

17.2. MPEG-2 Audio — стандарт кодирования аудиоинформации для многоканального стереофонического вещания

Этот стандарт расширяет использование стандарта MPEG-1 Audio на совокупность следующих форматов [6.49]:

- 3/2 — L, R, C (соответственно левый, правый и центральный фронтальные сигналы); LS и RS (левый и правый пространственные сигналы), а также канал сверхнизких частот СНЧ;
- 3/0 — L, R, C плюс 2/0 — L_2, R_2 (дополнительные вторые левый и правый сигналы);
- 3/1 — L, R, C и S (S — сигнал «окружения» — Surround);
- 2/0 — L, R плюс 2/0 — L_2, R_2 ;
- 3/0 — L, R, C;
- 2/1 — L, R, S;
- 2/0 — L, R (стандартное стереовещание);
- 1/0 — M (моновещание).

Предполагается, что сигналы L и R занимают полную полосу частот от 20 Гц до 20 кГц, полоса частот сигнала C ограничена сверху частотой 9 кГц, а частоты LS и RS имеют полосу частот от 100 Гц до 7 кГц. В стандарте MPEG-2 увеличено количество используемых частот дискретизации — возможно применение частот 32, 44,1, 48 кГц, а также 16, 22,05 и 24 кГц. Исходным является пятиканальный режим — L, R, C, LS и RS (см. рис. 17.10а).

С помощью матрицы $M1$ эти сигналы преобразуются определенным образом.

Два сигнала $T1$ и $T2$ являются сигналами обычной двухканальной стереофонии, причем их сумма должна обеспечить реализацию монофонического сигнала, т. е. $T1 + T2 = M$. Соотношения определяются следующим образом:

$$\begin{aligned} T1 &= \alpha(L + \beta C + \gamma LS); \quad T2 = \alpha(R + \beta C + \gamma RS); \\ T3 &= \alpha\beta C; \quad T4 = \alpha\gamma LS; \quad T5 = \alpha\gamma RS, \end{aligned} \quad (17.15)$$

где α, β, γ — коэффициенты, значения которых определяют 4 режима работы:

- режим 0 с параметрами $\alpha = 1/(1 + \sqrt{2}), \beta = 1/\sqrt{2}, \gamma = 1/\sqrt{2}$;
- режим 1 с параметрами $\alpha = 1/(1,5 + 0,5\sqrt{2}), \beta = 1/\sqrt{2}, \gamma = 0,5$;

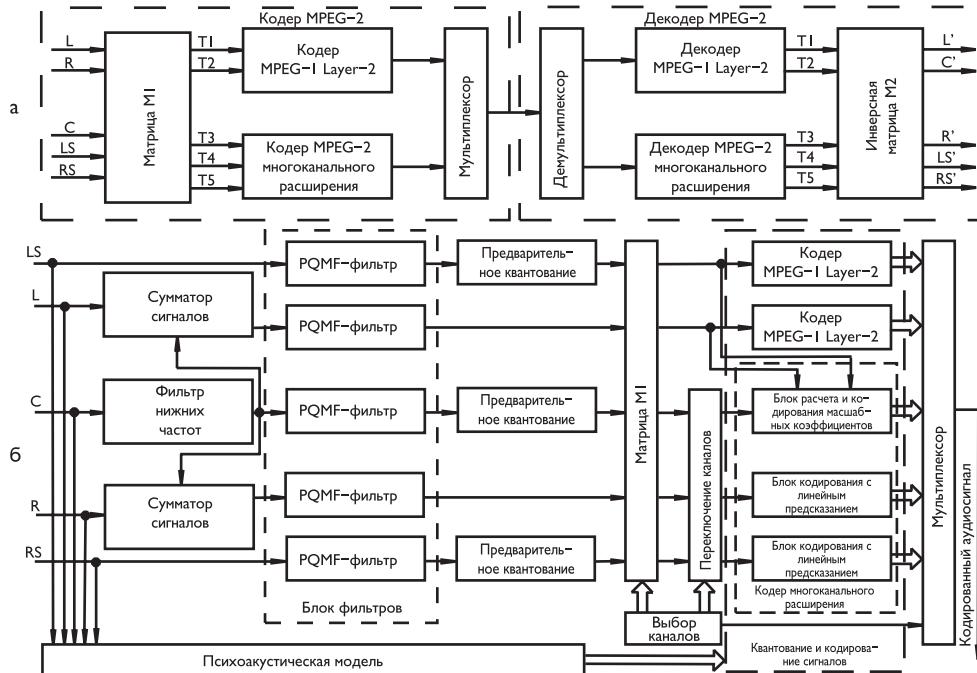


Рис. 17.10. Структурная схема кодека аудиосигналов стандарта MPEG-2

- режим 2 с параметрами $\alpha = 1/(1 + \sqrt{2})$, $\beta = 1/\sqrt{2}$, $\gamma = 1/\sqrt{2}$, при этом должен быть сформирован сигнал $S = (LS + RS)/2$ и в противофазе сложен с сигналами L и R ;
- режим 3 с параметрами $\alpha = \beta = \gamma = 1$.

После матрицирования сигналы кодируются, причем для кодирования составляющих T_1 и T_2 применяются алгоритмы стандарта MPEG-1 Audio, а составляющие T_3 , T_4 , T_5 кодируются отдельно кодером многоканального расширения MPEG-2.

При любом сочетании каналов возможна передача сигналов СНЧ, при котором используется частота дискретизации $f_d/96$, ширина полосы 125 Гц и разрешение больше 20 бит/отсчет.

В декодере передаваемые сигналы демультиплексируются, соответствующим образом декодируются и с применением матрицы M_2 преобразуются в требуемые сигналы.

При кодировании сигналов многоканального расширения используется относительно простая методика, поскольку при многоканальной стереофонии в каждый текущий момент времени передача всей информации не требуется. Для максимально возможной компрессии аудиоданных T_3 , T_4 и T_5 учитывается степень взаимной корреляции между основными и дополнительными сигналами. При большой корреляции между ними достаточно передавать только различие сигналов. Для еще большей компрессии данных применяется также процедура

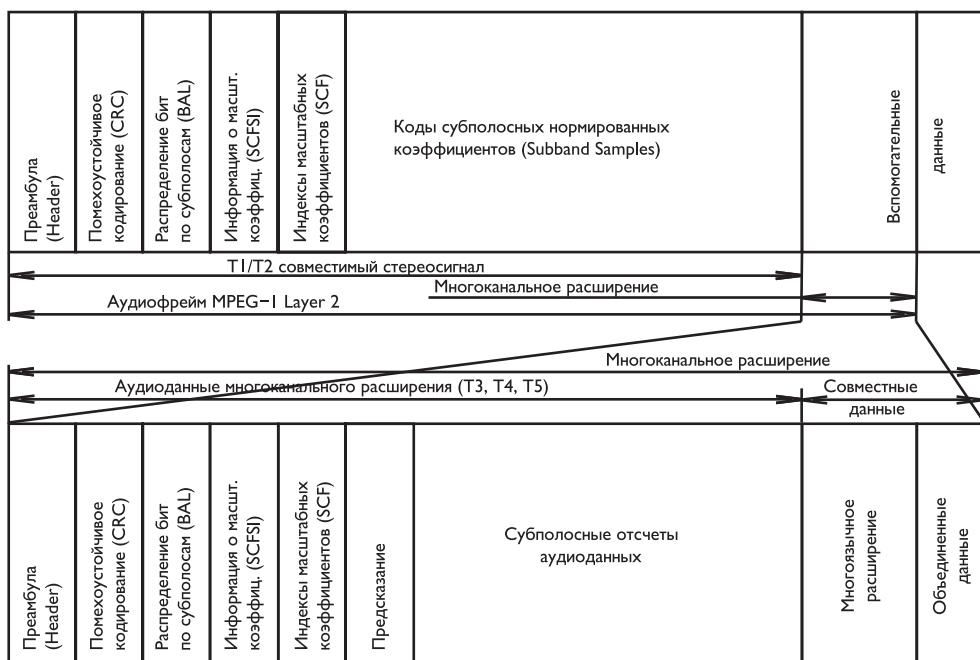


Рис. 17.11. Структура данных аудиофрейма стандарта MPEG-2, совместимая со стандартом MPEG-1 Audio Layer-2

ра динамического изменения переходного затухания между дополнительными сигналами, когда изменяются во времени лишь соотношения между уровнями воспроизводимых сигналов.

Психоакустическая модель кодера MPEG-2 соответствует стандарту MPEG-2 (рис. 17.10б). Два формата аудиофрейма, предусмотренные стандартом MPEG-2 Audio, приведены на рис. 17.11 и 17.12.

Первый формат совместим с форматом аудиофрейма стандарта MPEG-1 Audio. Здесь структура аудиоданных $T1$ и $T2$ соответствует формату аудиоданных MPEG-2 Audio, а информация о аудиоданных $T3$, $T4$ и $T5$ располагается в части аудиофрейма, относящейся к передаче дополнительных данных. В этой части также имеется своя преамбула (MC-Header), биты для помехоустойчивого кодирования (MC-CRC), часть для передачи аудиоданных $T3$, $T4$ и $T5$, а также часть для передачи дополнительной информации.

Структура аудиофрейма MPEG-2 Audio с дополнительным расширением приведена на рис. 17.12. Данные начинаются с преамбулы (Header) и битов помехоустойчивого кодирования (CRC). Далее следует информация о кодировании распределения битов по субполосам (BAL), распределении масштабных коэффициентов (SCFSI), величинах масштабных коэффициентов (SCF) и отсчетов субполосных составляющих, соответствующих по структуре формату данных MPEG-1 Audio Layer-3 для основной пары сигналов L и R .

Затем следуют дополнительные данные многоканального расширения (MC), включая преамбулу (MC-Header), данные помехоустойчивого кодирования (MC-

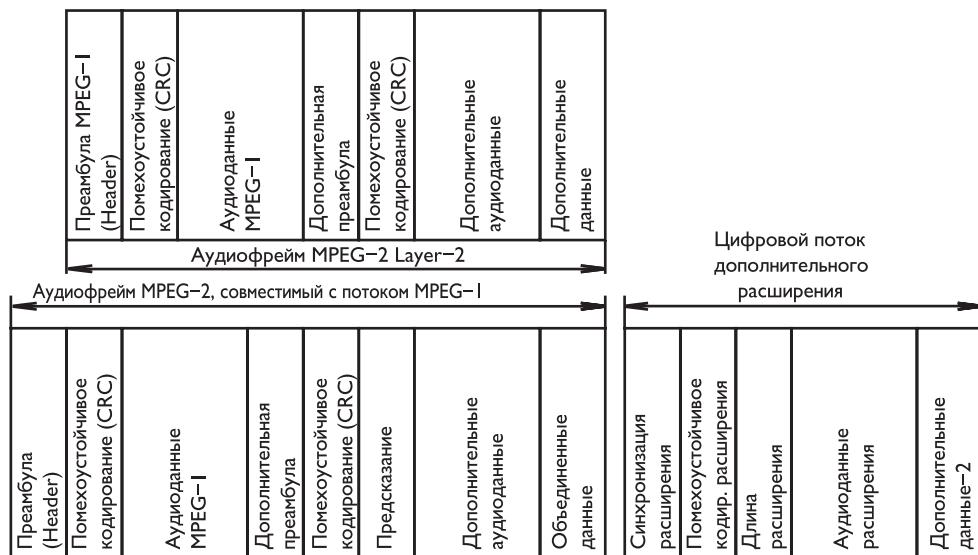


Рис. 17.12. Структура данных аудиофрейма стандарта MPEG-2 с дополнительным расширением

CRC), распределения бит по субполосам (MC-BAL), распределения масштабных коэффициентов (MC-SCFSI), величины масштабных коэффициентов (MC-SCF). Далее следуют аудиоданные: MC-предсказание, MC-данные групп частотных полос и возможные дополнительные данные, например многоязычные передачи. Формат дополнительных данных начинается с синхронизации части расширения (ext. Sync), помехоустойчивого кодирования (ext. CRC), указание о длине расширения (ext. Length), собственно аудиоданные расширения (MC-дополнительные данные-2).

17.3. MPEG-2 AAC — стандарт усовершенствованного аудиокодирования

Алгоритм компрессии AAC (Advanced Audio Coding) [6.50] базируется на стандартах MPEG-1 Layer-3 и MPEG-2 Layer-3, поддерживает все известные аудиоформаты: от монофонического звука до многоканального формата 5.1. При этом используется широкий набор частот дискретизации — 8; 11,025; 16; 22,05; 24; 32; 44,1; 48; 64; 88,2; 96 кГц.

Изменены форма и длины базисных функций — применяются окна Кайзера-Бесселя, длинное, включающее 2048 отсчетов, и короткое на 256 отсчетов аудиосигнала, реализованные с 50% перекрытием, что обеспечивает высокое разрешение по частоте.

Кодированию подвергаются коэффициенты МДКП, однако несколько изменена форма кривой компрессии при неравномерном квантовании, применены другие книги кодов Хаффмана, управление величиной искажений квантования,

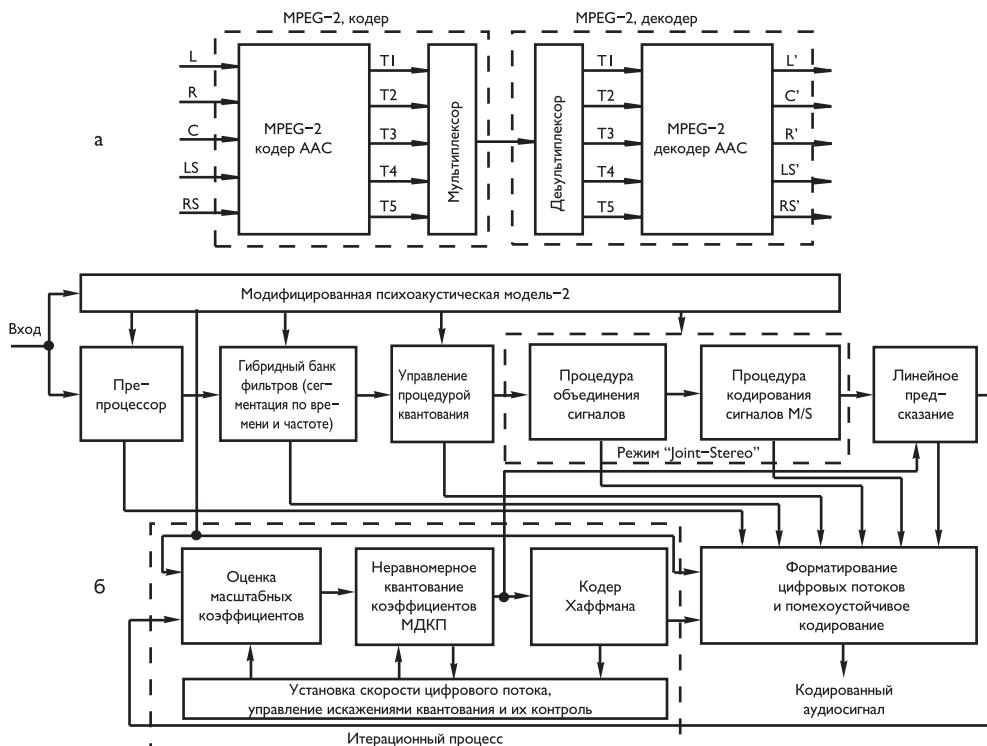


Рис. 17.13. Структурные схемы кодека MPEG-2 AAC

как и в стандарте MPEG-2 Layer-3, осуществляется двумя циклами — внутренним и внешним (рис. 17.13).

Для повышения качества компрессии применены специальные процедуры:

- управление микроструктурой искажений квантования внутри каждой из субполос (Temporal Noise Shaping — TNS);
- объединение субполосных сигналов при их кодировании (Coupling);
- возможность *M/S*-кодирования, при котором обработке подвергаются не исходные сигналы стереопары, а их сумма и разность — $M = (L + R)/\sqrt{2}$, $S = (L - R)/\sqrt{2}$;
- при линейном предсказании учитывается не только корреляция между отсчетами многоканального сигнала, но также форма спектра шумов квантования и ее изменение во времени;
- модифицированы дополнительные процедуры при расчете глобального порога маскировки в психоакустической модели кодера, основой которой все же являются алгоритмы психоакустической модели-2 (см. раздел 15.7.2).

В зависимости от области применения и вычислительной сложности стандарт MPEG-2 AAC предусматривает три возможные конфигурации.

Основная конфигурация (Main profile) используется, когда вычислительная сложность не является сдерживающим фактором, не используется банк PQMF-

фильтров, вся последовательность 2048 временных отсчетов непосредственно передается на блок ортогонального преобразования с 50%-и перекрытием. Порядок предсказания TNS составляет 20.

В конфигурации пониженной сложности (Low Complexity profile) также не используется банк PQMF-фильтров, не применяется блок линейного предсказания и порядок TNS сокращен до 12.

При адаптивной конфигурации кодер AAC использует банк PQMF-фильтров, блок линейного предсказания не применяется, а порядок TNS равен 12. Кроме того, не всегда требуется передача аудиосигнала с полной полосой спектра от 20 Гц до 20 кГц. Поэтому стандарт предусматривает возможность кодирования сигнала, например, со спектром, занимающим полосы от 20 Гц до 6 кГц, до 12 кГц, до 18 кГц.

Тестовая оценка показывает, что алгоритм компрессии AAC обеспечивает так называемое прозрачное кодирование при скорости цифрового потока 64 кбитов/с на канал.

При аудиоформате 5.1 искажения, вызванные компрессией, лежат ниже порогов их слуховой заметности при суммарной скорости цифрового потока 320–384 кбитов/с (при раздельном кодировании каналов).

17.4. MPEG-4 Audio — стандарт аудиокодера для мультимедийных приложений

В данном кодере впервые при компрессии цифровых данных высококачественных аудиосигналов предлагается использовать параметрическое кодирование, когда реальный звуковой сигнал представляется в виде модели, содержащей совокупность тональных и шумоподобных сигналов (рис. 17.14). В стандарте MPEG-4 Audio [6.51] приводятся две психоакустические модели, которые могут быть использованы в любом слое кодирования.

В случае применения алгоритма параметрического кодирования (рис. 17.15) исходный сигнал выборки $X(n)$ в блоке анализа/синтеза разделяется на тональные и шумоподобные составляющие. Затем оцениваются значения текущих частот, амплитуд и фаз тональных сигналов и уровней шумоподобных составляющих в определенных полосах частот. Значения указанных параметров квантуются и кодируются минимально возможным количеством битов, требуемое количество которых определяется психоакустической моделью.

Параметрическое кодирование, обладая очень сложными процедурами оценки параметров и требуя при реализации больших вычислительных затрат, позволяет получить скорость цифрового потока 16–24 кбитов/с при достаточно хорошем качестве воспроизведения аудиоинформации.

Вторым альтернативным алгоритмом, обеспечивающим более высокое качество, однако при большем цифровом потоке, является алгоритм кодера AAC, также включенный в стандарт MPEG-4 Audio.

Алгоритм «Joint-Stereo» стандарта MPEG-4 AAC очень похож на процедуру кодирования по стандарту MPEG-1 Audio Layer-3.

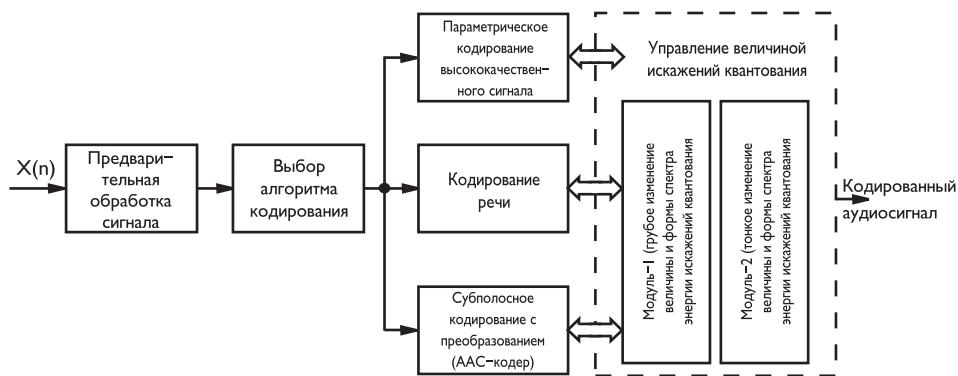


Рис. 17.14. Алгоритмы компрессии аудиоинформации в кодере MPEG-4 Audio

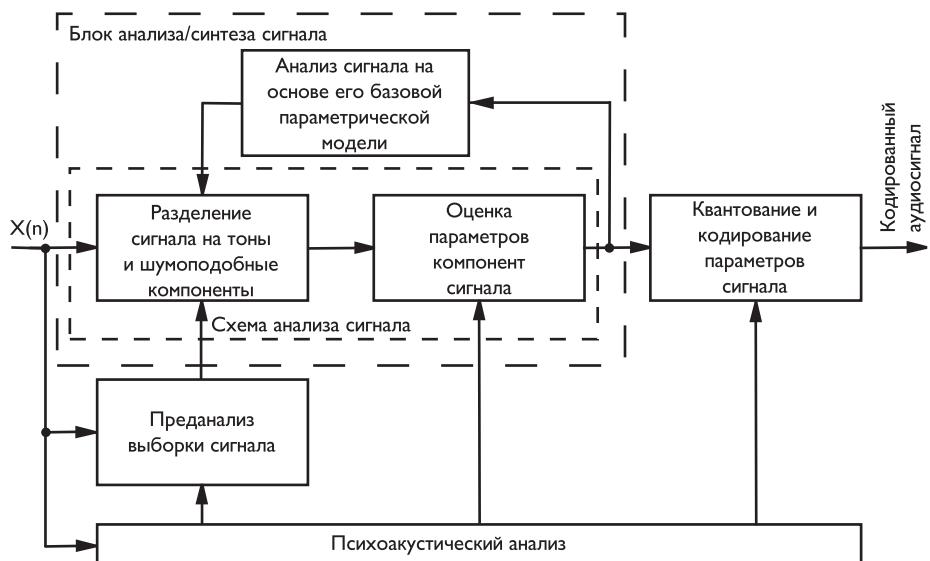


Рис. 17.15. Упрощенная структурная схема кодера MPEG-4 Audio, использующая метод параметрического кодирования

Для каждой полосы кодирования m путем суммирования квадратов амплитуд коэффициентов МДКП вычисляются энергии левого, правого и суммарного сигналов $e_{m,L}$, $e_{m,R}$, $e_{m,M}$, а затем рассчитываются значения координатного множителя Ψ_m и амплитуда каждого k -го коэффициента МДКП объединенного сигнала M_k .

Коэффициенты МДКП объединенного сигнала передаются вместо соответствующих компонент левого канала, а компоненты правого канала приравниваются к нулю. Далее выполняются стандартные процедуры квантования и кодирования коэффициентов МДКП объединенного канала с применением ДИКМ, т. е. кодируются разности текущих значений МДКП. При передаче объединенного

сигнала в правом канале используются специальные кодовые таблицы Хаффмана (INTENSITY_HCB и INTENSITY_HCB2), при этом их применение в левом канале запрещено. Указанные таблицы применяются при кодировании соответственно синфазных и противофазных составляющих объединяемых сигналов стереопары.

Информацию о соотношении фаз коэффициентов МДКП в режиме объединения субполосных сигналов можно получить посредством использования флага ms_used_m . Первоначальное соотношение фаз, идентифицированное кодовыми таблицами Хаффмана, меняется из синфазного на противоположное и наоборот, если соответствующий бит флага установлен для данной субполосы.

При декодировании используются два соображения:

- в системе AAC координатный множитель кодируется так же, как и масштабные коэффициенты, т. е. кодами Хаффмана с применением дифференциальных величин с двумя разностными значениями; если первое значение отсутствует, то оно считается равным нулю;
- дифференциальное декодирование происходит отдельно для масштабных коэффициентов и координатных множителей; для кодирования масштабных коэффициентов и координатных множителей используются одни и те же кодовые таблицы.

При декодировании объединенных каналов используются две функции:

$$h_m = \begin{cases} +1, & \text{если для субполос правого канала используется} \\ & \text{таблица INTENCSITY_HCB;} \\ -1, & \text{если для субполос правого канала используется} \\ & \text{таблица INTENCSITY_HCB2;} \\ 1 & \text{— в противном случае;} \end{cases}$$

$$\lambda_m = \begin{cases} 1 - 2 \cdot ms_used_m, & \text{если флаг } ms_mask_present = 1; \\ +1 & \text{— в противном случае;} \end{cases}$$

где флаг ms_used_m принимает значение, равное 1 или 0, а двухбитный флаг $ms_mask_present$ показывает присутствие маски MS .

Декодирование объединенных сигналов происходит следующим образом:

- сигнал левого канала принимается равным объединенному сигналу $F_{k.L} = M_k$;
- сигнал правого канала получается путем умножения сигнала левого канала на масштабный коэффициент $scale_m$:

$$F_{k.R} = scale_m \cdot F_{k.L}, \quad \text{где } scale_m = h_m \cdot \lambda_m \cdot 0,5^{\Psi_m/4}.$$

17.5. Dolby AC3 — стандарт аудиокодера для телевидения и других применений

В режиме многоканального стереофонического вещания стандарт Dolby AC3 предусматривает передачу в едином цифровом потоке фронтальных сигналов левого (Left), правого (Right) и центрального (Center) каналов, тыловых пространственных сигналов левого (Left Surround) и правого (Right Surround) ка-

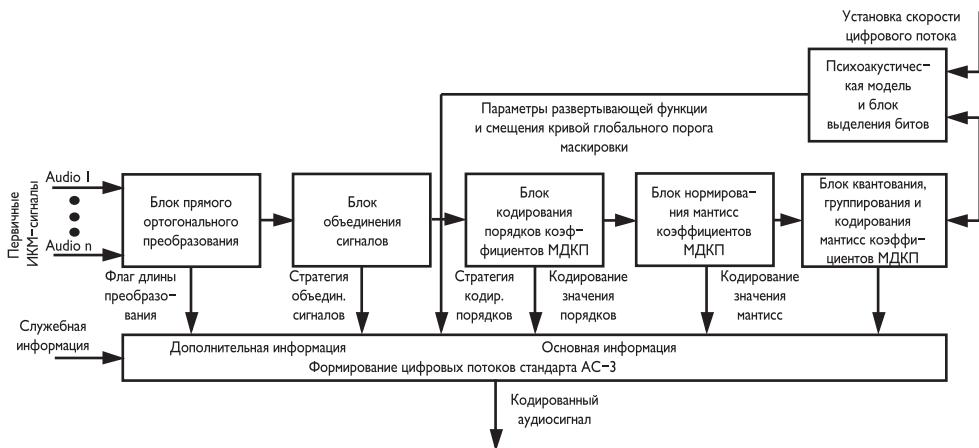


Рис. 17.16. Структурная схема кодера системы Dolby AC-3

налов, а также дополнительного сигнала канала сверхнизких частот СНЧ (Low Frequency). Возможно использование также форматов 2/0 (обычное стерео), 3/1 (Dolby-Stereo, Dolby-Surround) и 3/2 (Dolby-Pro-Logic).

Цифровой поток на выходе кодирующего устройства представляет собой последовательность аудиофреймов (Pack AC-3 Frame), содержащих информацию, которую можно разделить на две части: основную (Main information) и дополнительную (Side Information) [6.52] (рис. 17.16).

Аудиофрейм кодера содержит 6 блоков, в каждом из которых содержится информация о 512 отсчетах каждого из кодируемых сигналов (Audio 1, ..., Audio n). Вследствие 50% временного перекрытия в блок для каждого из сигналов включается 256 отсчетов предыдущего блока и 256 новых отсчетов. В шести блоках аудиофрейма передается $512 \cdot 6 = 3072$ обрабатываемых отсчета аудиосигнала, а если в аудиофрейм вводится информация о пяти кодируемых звуковых сигналах (формат 3/2), то общее число передаваемых отсчетов равно $3072 \cdot 6 = 15360$.

После сегментации по времени с 50%-м перекрытием отсчеты выборки сигнала преобразуются в частотную область с использованием МДКП в блоке частотного преобразования (Frequency Domain Transform). Перед ортогональным преобразованием отсчеты выборки подвергаются взвешиванию с помощью оконной функции, заданной таблично в стандарте AC-3 (рис. 17.17).

Расчет коэффициентов МДКП проводится с использованием соотношения (15.65), приведенного в описании психоакустической модели-3 (см. раздел 15.7.2). Флаг управления длиной обработки информации (Block Switch Flag — Blksw) переключает режим с длинного преобразования, при котором обеспечивается лучшее разрешение по частоте, на короткое — преобразование, реализующее лучшее разрешение по времени. При малых скоростях передачи цифровых данных предусмотрено использование процедуры объединения канальных сигналов (Coupling), позволяющей сократить количество битов, используемых при их кодировании.

В соответствии со стандартом Dolby AC-3 каждый k -й коэффициент МДКП представляется в формате с плавающей запятой двумя значениями: порядком

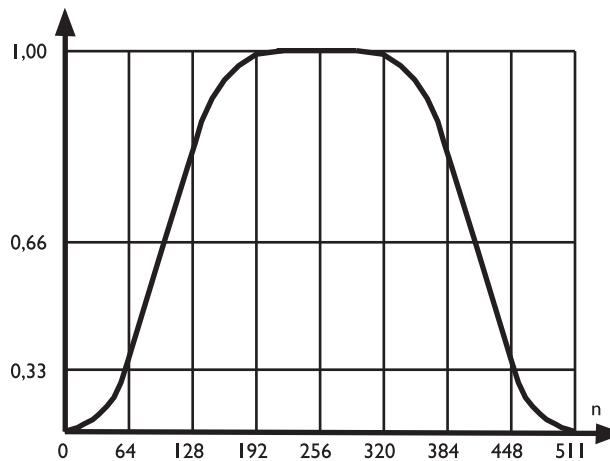


Рис. 17.17. Форма оконной функции стандарта Dolby AC-3

(экспонентой) — $B[k]$ и мантиссой — $A[k]$ (см. раздел 15.7.2, психоакустическая модель-3).

Порядок равен числу нулей перед первой единицей двоичного представления коэффициента МДКП — это и есть его масштабный коэффициент, или нормирующий множитель.

Например, пусть значение коэффициента МДКП $F_D[k]=0,158$ и его двоичное представление имеет вид 0,001010000110. Тогда значение порядка $B[k]=2$, а его мантисса в двоичном представлении равна 0,1010000110 или в десятичном виде $A[k]=0,6308$, т. е. $F_D[k] = A[k] \cdot 2^{-B[k]} = 0,6308 \cdot 2^{-2} = 0,158$. Мантиссы нормируются, квантуются и кодируются, значения порядков также кодируются. В блоке распределения битов учитывается эффект маскировки.

Порядок коэффициентов МДКП в кодере Dolby AC-3 может изменяться от 0 до 24, поэтому кодовое число должно было бы использовать пять разрядов. Однако в данном случае используется метод ДИКМ, при котором кодируется разность между порядками соседних коэффициентов МДКП, используется 4-битовое кодовое слово и дискретность значений разностей ограничена значениями: $-2, -1, 0, 1, 2$. Максимальное значение изменений порядков соседних коэффициентов МДКП составляет ± 2 , что соответствует изменениям в ± 12 дБ. Дифференциальные значения порядков коэффициентов МДКП объединяются в группы, определяемые тремя возможными стратегиями, обозначенными D15, D25 и D45. В стратегии D15 кодируется одним числом $M[k]$ каждое дифференциальное значение порядков коэффициентов МДКП независимо, при D25 кодируется одним числом каждая пара, а при D45 — каждая четверка указанных дифференциальных значений порядков коэффициентов МДКП. Выбор этих стратегий зависит от различий порядков: при резком их изменении применяется стратегия D15, а при плавном изменении — стратегия D45.

При всех режимах работы кодера наборы трех чисел $M[k]$, $M[k+1]$, $M[k+2]$ группируются и кодируются как одно 7-битовое число по правилу:

$$M[k, k+1, k+2] = 25 \cdot M[k] + 5 \cdot M[k+1] + M[k+2]. \quad (17.16)$$

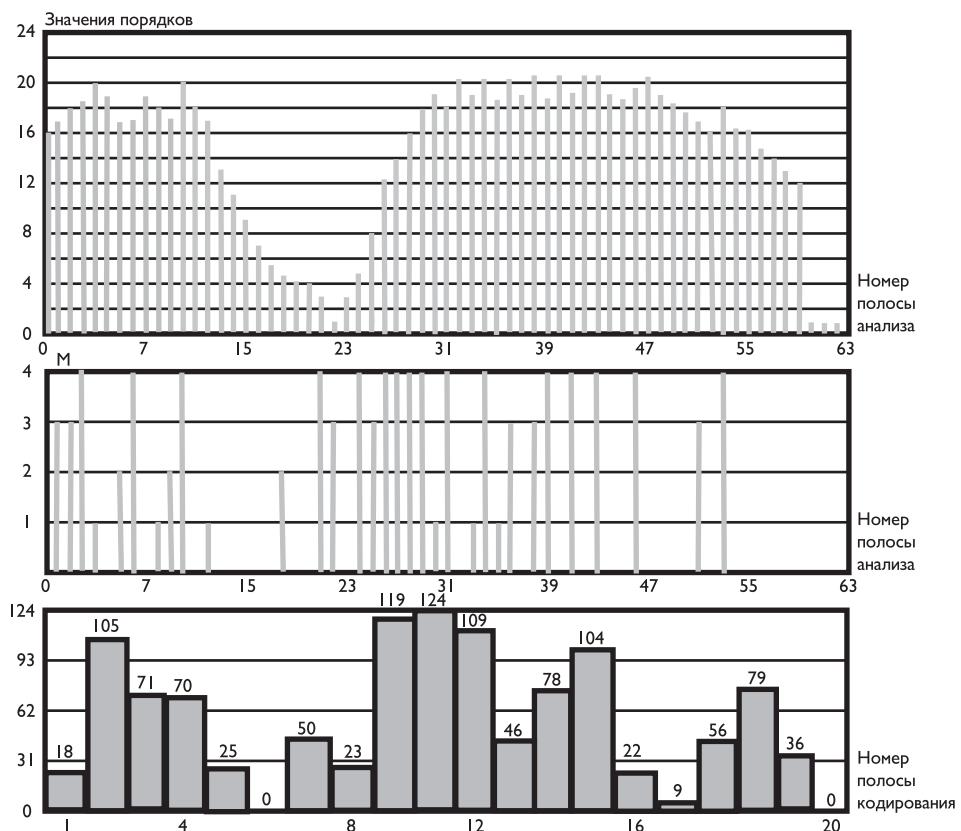


Рис. 17.18. Упаковка дифференциальных значений порядков в поток данных при стратегии D15

Эта процедура для стратегии D15 поясняется на рис. 17.18, где в верхней части приведены значения порядков коэффициентов МДКП, в средней части — величины чисел $M[k]$, а в нижней части — число битов, использованное для 20 полос их кодирования.

При кодировании значений мантисс коэффициентов МДКП (диапазон их изменения — от -1 до $+1$) учитываются следующие особенности:

- при 16 возможных ступенях квантования ($0, 3, 5, 7, 11, 15, 31, 63, 127, 255, 511, 1023, 2047, 4095, 16383, 65535$) используется равномерное квантование значений мантисс;
- при 5 ступенях квантования ($3, 5, 7, 11, 15$) применяется так называемое симметричное квантование, во всех остальных — асимметричное;
- при 3 ступенях квантования ($3, 5, 11$) кодовые слова объединяются в группы.

При симметричном квантовании вместо квантованных значений мантисс включаются значения их индексов, заданные таблицей. Например, если при числе ступеней квантования, равном трем, величина мантиссы находится в диапазоне

от -1 до $-1/3$, то декодеру будет передаваться число $-2/3$, соответствующее индексу $mc = 0$; при величине мантиссы в пределах от $-1/3$ до $+1/3$ передается число 0 , соответствующее индексу $mc = 1$, а при величине мантиссы в пределах от $+1/3$ до $+1$ передается число $+2/3$, соответствующее индексу $mc = 2$.

С целью уменьшения требуемого для кодирования индексов числа битов при 3, 5 и 11 ступенях квантования табличные индексы мантисс группируются по следующим правилам:

$$\begin{aligned} \text{Group_code}[3] &= 9 \cdot mc[a] + 3 \cdot mc[b] + mc[c], \\ \text{Group_code}[5] &= 25 \cdot mc[a] + 5 \cdot mc[b] + mc[c], \\ \text{Group_code}[11] &= 11 \cdot mc[a] + mc[b], \end{aligned} \quad (17.17)$$

где $mc[a]$, $mc[b]$, $mc[c]$ — значения табличных индексов мантисс коэффициентов МДКП с номерами a , b , c .

При работе в режиме объединения аудиосигналов высокочастотные части исходных сигналов в определенной полосе частот преобразуются кодером Dolby AC-3 в общий сигнал. Общий сигнал формируется путем сложения коэффициентов МДКП: коэффициенты с 37-го по 252-й группируются по 12 в каждой из 18 субполос. Суммарный сигнал кодируется так же, как и сигналы независимых каналов. Кодер дополнительно генерирует так называемые параметры объединения. После декодирования объединенные части сигналов будут иметь одинаковый спектральный состав и отличаться только уровнем.

Декодер системы Dolby AC-3 обеспечивает распаковку информации аудиофрейма и разделение ее на основную (Main Information) и дополнительную (Side Information) части (рис. 17.19). Далее в соответствующих блоках осуществляются процесс декодирования порядков, процедуры распаковки, деквантования и денормирования мантисс коэффициентов МДКП. Если в кодере была использована процедура объединения сигналов ряда каналов, то декодер реализует обратную операцию (De-Coupling). В блоке обратного МДКП осуществляется преобразование реконструированного сигнала во временную область.

Структура данных аудиофрейма стандарта Dolby AC-3 приведена на рис. 17.20а. Помимо заголовка и дополнительных данных аудиофрейм включает два 16-битовых слова помехоустойчивого кодирования, первое из которых следует в начале каждого фрейма после передачи сигнала синхронизации (0B77h или 0000 1011 0111 0111), а второе — в конце фрейма. Аудиофрейм содержит 6 аудиоблоков, структура данных каждого из них приведена на рис. 17.20б.

17.6. Совершенствование алгоритмов сжатия звука на основе стандарта AAC

Несмотря на широкое распространение различных форматов сжатия звуковой информации разработчики непрерывно занимаются их совершенствованием [6.53–6.55].

Алгоритмы усовершенствованного кодирования звуковой информации AAC организованы в профили (стандарт MPEG-2) или объектные типы (стандарт MPEG-4). MPEG-2 AAC предусматривает три различных профиля кодирования:

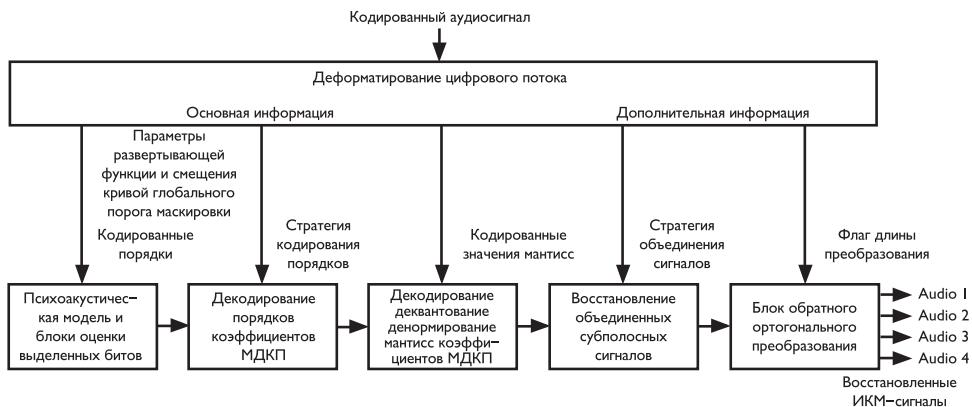


Рис. 17.19. Структурная схема декодера системы Dolby AC-3

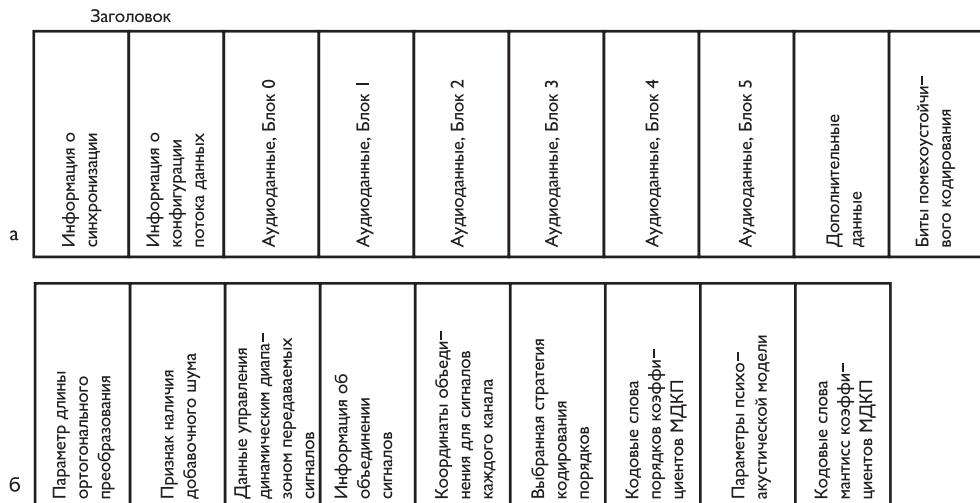


Рис. 17.20. Структура данных аудиофрейма (а) и данных аудиоблока (б) системы Dolby AC-3

- основной (AAC/Main);
- малой сложности (AAC/LC – Low Complexity);
- с масштабируемой величиной выборки (AAC/SSR – Scalable Sampling Rate).

В зависимости от того, какой профиль используется во время кодирования, изменяются время кодирования и качество формируемого цифрового потока. Наивысшее качество звучания (при самой медленной скорости компрессии) обеспечивает основной профиль AAC/Main. Это связано с тем, что этот профиль включает в себя все механизмы анализа и обработки входного потока. Профиль AAC/LC упрощен, что сказывается на качестве звучания восстанавливаемого цифрового потока, сильно отражается на скорости компрессии и, что более важно, декомпрессии. Профиль Main обеспечивает более качественную компрессию



по сравнению с LC, однако эта разница в качестве не всегда оправдывает разницу затрат мощности процессора при компрессии в Main и LC.

Профиль AAC/SSR кодирует данные в несколько слоев (layers) от очень низкого битрейта к высшим битрейтам (вплоть до компрессии без потерь). Такой подход позволяет, например, транслирующим серверам передавать пользователям звук с нужной скоростью без необходимости транскодирования данных.

О качестве кодирования звука, можно сказать, что поток AAC/Main 96 кбитов/с обеспечивает качество звучания, аналогичное потоку MPEG-1 Layer-3 128 кбитов/с. При компрессии AAC 128 кбитов/с, качество звучания ощущимо пре-восходит MPEG-1 Layer-3 128 кбитов/с.

Стандарт MPEG-4 AAC, предусматривающий правила и алгоритмы кодирования аудио, является продолжением MPEG-2 AAC. MPEG-4 AAC стандартизует следующие типы объектов:

- основной (AAC/Main);
- малой сложности (AAC/LC);
- с масштабируемой величиной выборки (AAC/SSR);
- с длительным предсказанием (LTP-AAC — Long Term Prediction);
- с малой задержкой (LD-AAC — Low Delay);
- версия 2 (Version-2);
- версия 3 (Version-3, включая HE-AAC);
- звуковой кодер без потерь (AAC/ALS — Audio Lossless Coding);
- масштабируемый звуковой кодер без потерь (AAC/SLS — Scalable Lossless Coding).

Как видно из этого перечня, первые три позаимствованы у кодера MPEG-2 AAC.

При этом новый стандарт является только дополнением к уже существующим и описанным в MPEG-2 алгоритмам звуковой обработки, а не отменяет его. В частности, профиль AAC-LC описан только в стандарте MPEG-2, являясь при этом базовым для нового высокоэффективного варианта кодека – HE-AAC (High Efficiency).

AAC/LTP основан на методах предсказания сигнала и является более сложным алгоритмом, нежели остальные, что сильно сказывается на скорости компрессии. Version-2 – это пакет стандартов, расширяющих некоторые механизмы кодирования стандарта MPEG-4. Version-3 – это еще одно расширение стандарта MPEG-4. Основным нововведением в стандарт MPEG-4, введенным Version-3, является стандартизация (с мая 2003 г.) алгоритма HE-AAC, известного также под именем aacPlus.

Существуют две версии расширения кодека AAC: «HE-AAC v.1» и «HE-AAC v.2». Первая версия (v.1) использует технологию спектрального «дублирования» (replication) нижней части спектра звукового сигнала в верхнюю: данный алгоритм носит название SBR – Spectral Band Replication. Вторая версия (v.2) полностью включает в себя всю функциональность первой и расширяет ее с помощью алгоритма параметрического кодирования стереосигнала, этот алгоритм носит название Parametric Stereo (PS).

Проведенные исследования подтвердили превосходство стандарта AAC над самым популярным на сегодняшний день форматом сжатия звука MP3 для всех

скоростей кодирования и всех типов звуковых сигналов. При этом установлено, что при скорости цифрового потока 24 кбитов/с кодек HE-AAC в состоянии обеспечить качество FM-стереовещания, а при скорости 32 кбитов/с — качество Audio-CD, при потоке 128 кбитов/с по качеству декодированного сигнала звук сравним с шестиканальным звуком стандарта «Dolby Digital Surround EX» (см. раздел 17.5).

Инструментарий MPEG-4 AAC содержит несколько интересных технологий. Например, принцип замены воспринимаемого шума (PNS — Perceptual Noise Substitution). Суть PNS довольно проста. Основываясь на идее, что один шум можно подменить другим, искусственным, PNS предполагает подмену шума, присущего в кодируемом сигнале, другим шумом, искусственным. Это позволяет генерировать искусственный шум на стадии декомпрессии сигнала (воспроизведения) и не передавать настоящий шум сигнала. Иными словами, при кодировании исходного сигнала происходит регистрация шума и запоминание его основных параметров (вместо непосредственного кодирования шумового сигнала), а на стадии воспроизведения — генерация искусственного шума вместо настоящего.

AAC/ALS — стандарт кодирования звука без потерь, включенный в MPEG-4 в конце 2005 года. Основными характеристиками этого варианта являются:

- поддержка PCM-потоков с разрешением до 32 бит и самыми разными частотами дискретизации (включая пары 16 бит/44,1 кГц, 16/48, 24/48, 24/96, 24/192);
- многоканальная/мультитрековая поддержка до 65536 каналов;
- поддержка данных в формате 32-bit IEEE с плавающей точкой.

AAC/SLS — совместная разработка Infocomm Research и института Fraunhofer. Он никак не связан с AAC/ALS. Расширение SLS предусматривает аудиопоток с несколькими уровнями данных, позволяющий его декодирование как в варианте без потерь (точное декодирование исходного аудиосигнала), так и с потерями при низкой скорости передачи. Иными словами, поток в формате SLS легко масштабируем в том смысле, что качество декодируемого сигнала находится в зависимости от того, сколько данных из этого потока было использовано для декодирования. Поток SLS стандартным неспециализированным MPEG-4-плеером воспринимается как формат AAC/LC. Более того, из файла в формате SLS можно извлечь файл в формате AAC/LC без необходимости перекодирования, а лишь просто средствами извлечения одного слоя данных из большого файла.

17.6.1. Общие алгоритмы кодека AAC

Общие алгоритмы обработки аудиосигнала в кодере AAC [6.50, 6.51, 6.56] поясняются с использованием структурной схемы, приведенной на рис. 17.21.

Регулировка усиления (*Gain control*) не является обязательной в структуре кодека и предназначена для регулировки уровней звукового сигнала. На стороне кодера происходит поблочная регулировка уровня звукового сигнала и передача декодеру информации, необходимой для восстановления исходного соотношения. В состав модуля регулировки усиления входит 4 полифазных квадратурных

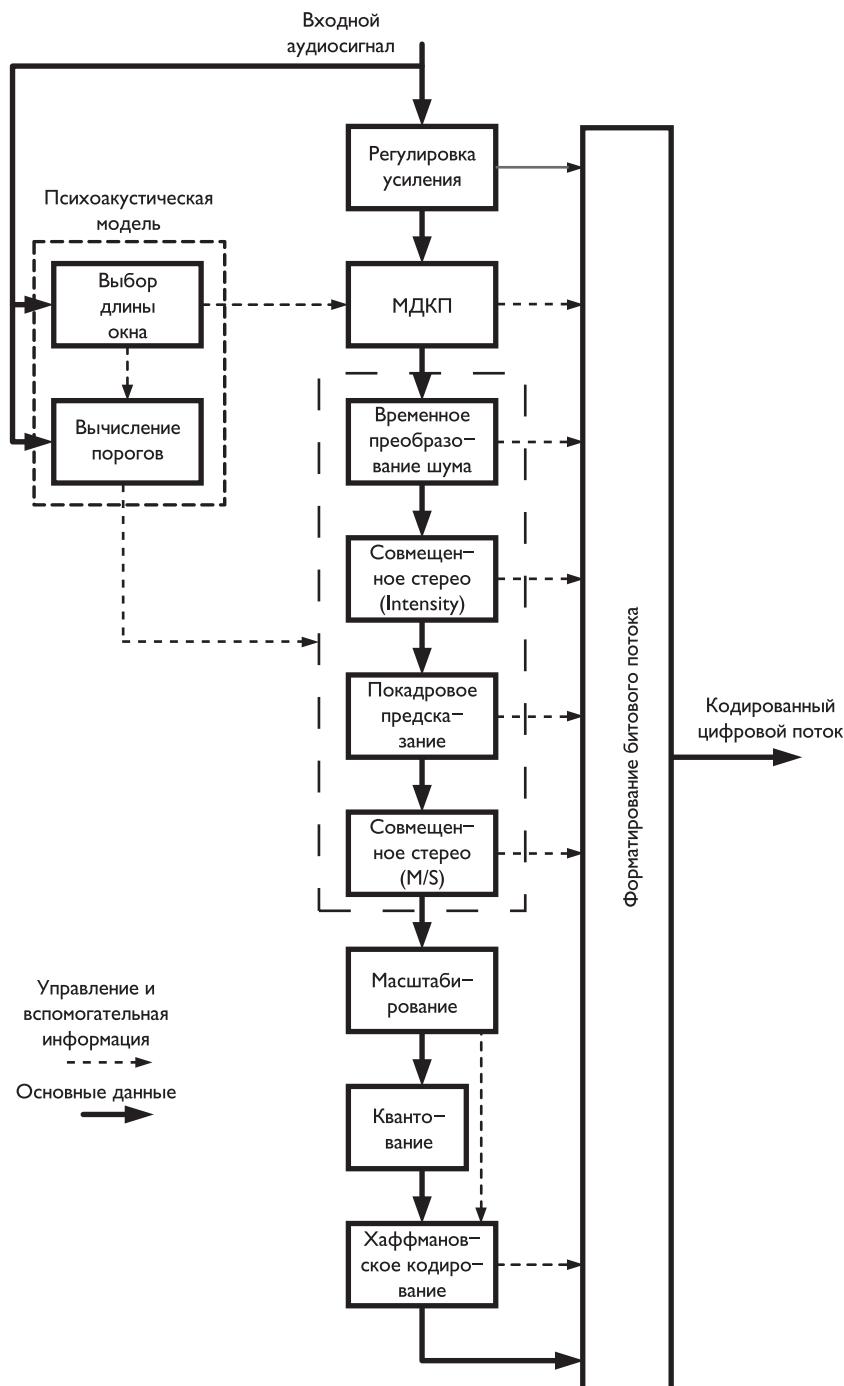


Рис. 17.21. Алгоритмическая структурная схема кодера AAC

фильтра (PQF), обеспечивающих четырехполосное разделение спектра сигнала с последующим контролем уровня в каждой из четырех полос во временной области.

Психоакустическая модель вычисляет максимальные значения ошибок, маскируемых (т. е. незаметных для человеческого уха) основным сигналом. Это — пороги ошибок, превышение которых заметно при воспроизведении декодированного звукового сигнала. В стандарте [6.50] приведены таблицы, в соответствии с которыми производятся вычисления указанных порогов. Кроме этого в блоке психоакустической модели осуществляется выбор типа и длины взвешивающего окна, используемого в дальнейшем непосредственно перед МДКП-преобразованием (см. раздел 17.4). Отметим, что выбор осуществляется между двумя типами окон — Кайзера-Бесселя и синусным. Выбор длины взвешивающего окна и, соответственно, длины МДКП для одного кадра осуществляется исходя из следующих альтернатив: при длине кадра 1024 отсчета — 2048 либо 256 отсчетов; при длине кадра 960 отсчетов — 1920 либо 240 отсчетов. Короткие преобразования (256 и 240) осуществляются на том же интервале кадра 8 раз подряд, при этом используются соответствующие короткие взвешивающие окна. При обнаружении участка резкого изменения уровня сигнала с целью локализации этой динамической особенности данный блок выносит решение о применении коротких взвешивающих окон. Здесь же происходит оценка количества битов, необходимых для кодирования очередного анализируемого кадра.

МДКП — *модифицированное дискретное косинусное преобразование* (*MDCT — Modified Discrete Cosine Transform*) осуществляет преобразование временного звукового сигнала в частотную область (см. раздел 15.7.2, психоакустическая модель-3).

Размерность входного сигнала в два раза больше размерности выходного. Это связано с тем, что МДКП, в отличие, например, от обычного косинусного преобразования, работает с перекрывающимися (на 50%) областями сигнала. Этим решается или по крайней мере ослабляется проблема блочности, т. е. появления немаскируемых артефактов на границах преобразования при последовательной обработке сигнала, длительность которого больше, чем длина окна используемого преобразования. Кроме этого для предотвращения разрывов фазы на границах окна преобразования и, как следствие, появления мешающих артефактов в этих областях в алгоритм AAC вводятся взвешивающие временные окна. Решение о длине окна и типе окна, как уже было сказано, принимается в блоке психоакустической модели кодека.

Метод временного преобразования формы шума квантования (*TNS — Temporal Noise Shaping*) служит для борьбы с последствиями процедуры квантования сигнала, что проявляется, например, в частом появлении немаскируемых, т. е. различимых человеческим ухом, преэхо сигналов при наличии в сигнале резких фронтов, особенно характерных для речевых сигналов. Суть метода состоит в том, чтобы на основе спектрального анализа звукового сигнала предсказать ошибку квантования (*Forward Prediction*, или опережающее предсказание) и ввести ее в сигнал («вычесть» ее). Таким образом при восстановлении сигнала немаскируемые артефакты, вызванные квантованием, будут частично подавлены.

Совмещенное стерео (*Joint stereo*): Intensity-стерео или M/S-стерео. Данный алгоритм служит для сокращения информационной избыточности, обусловлен-



ной высокой корреляцией между левым и правым каналами стереосигнала. При этом метод Intensity-стерео является алгоритмом сжатия с потерей информации, а M/S (middle/side = среднее/разность) — алгоритмом без ее потери. Соответствующему блоку декодера передается информация о выбранном типе стереокодирования. Алгоритм Intensity использует корреляционную зависимость между верхними частями спектров стереопары звукового сигнала. При этом в выходной поток передаются нижние части спектров левого и правого каналов в полосе до 6 кГц, а оставшаяся верхняя часть полосы передается в виде масштабированной невосстанавливаемой на стороне декодера суммы спектров левого и правого каналов. Алгоритм M/S анализирует возможность передачи вместо сигналов левого (L) и правого (R) каналов пары $M = (L + R)/2$ и $S = (L - R)/2$ с точки зрения минимизации выходного битового потока. При этом оцениваются распределения мощностей для обоих вариантов, и выносится решение о передаче пары L/R либо M/S .

Покадровое предсказание (Prediction) применяется для сокращения информационной избыточности, связанной с высокой степенью корреляции между спектрами последовательных кадров. При этом на стороне кодера используется так называемый алгоритм обратного предсказания (backward prediction), осуществляющий итеративное квантование и восстановление данных с целью предотвращения распространения ошибки предсказания. Добавим, что система кадрового предсказания включается только тогда, когда на уровне первоначального анализа звукового сигнала (на уровне психоакустической модели) принимается решение об отсутствии в данном кадре резких переходов и участков «нестационарности спектра» (так называемые *transients*).

Масштабирование спектральных компонент (Scaling) реализует умножение спектральных компонент на масштабные множители (Scalefactors). Данная процедура используется для управления в частотной области формой шумов квантования (во временной области этим занимается блок преобразования формы шума квантования TNS). Сами масштабирующие множители передаются вместе с потоком. Количество коэффициентов определяется разбиением спектра сигнала на так называемые «масштабные полосы» (scalefactor bands), каждой из которых соответствует один коэффициент.

Система квантования (Quantization) осуществляет неравномерное квантование спектральных компонент сигнала и разделена на 3 уровня. Верхний уровень занимается вычислением количества битов, в которые нужно «уложить» поток, проверяет, не равны ли все спектральные компоненты нулю и, если нет, вызывает процедуру следующего уровня — так называемый внешний итеративный цикл, который, в свою очередь, вызывает процедуру внутреннего итеративного цикла. Внутренний цикл квантует входной вектор, увеличивая шаг квантования до тех пор, пока выходной вектор все еще укладывается в отведенное ему количество битов. После выполнения внутреннего цикла внешний цикл оценивает искажения, вызванные последней итерацией квантования внутри каждой масштабной полосы (см. описание масштабирования спектральных компонент) и, если ошибка больше пороговой, уменьшает значение масштабирующего коэффициента для тех масштабных полос, в которых было допущено превышение порога ошибки.

Система хаффмановского кодирования (Huffman coding) представляет собой статистический кодер, работающий с парами или квартетами передаваемых кван-

тованных спектральных коэффициентов и с единичными значениями масштабных коэффициентов. Для этого предусмотрено 10 основных кодовых книг кодирования для спектральных компонент и одна — для масштабных коэффициентов, различающихся между собою тем, над чем они определены (пары или квартеты значений), максимальным значением входных величин, а также знаковостью самой кодовой книги (знаковая или беззнаковая). Порядок кодирования спектральных коэффициентов — от нижней частоты к верхней.

Формирование битового потока (Bitstream formatter) реализует компоновку выходного битового потока, который включает в себя следующие основные типы управляющей информации:

- квантованные спектральные коэффициенты звукового сигнала;
- масштабирующие коэффициенты модуля масштабирования спектральных компонент;
- информацию для восстановления входных уровней звукового сигнала в случае использования на стороне кодера блока регулировки усиления;
- информацию о типе и длине использовавшихся взвешивающих окон;
- информацию, необходимую для декодирования временного преобразования формы шума (TNS);
- информацию об использовавшемся типе объединенного стерео;
- информацию об использовавшихся на стороне кодера параметрах хаффмановского кодирования.

Обработка информации при декодировании осуществляется в обратном (относительно кодирования) порядке (рис. 17.22).

Вектор развития стандарта звукового кодека AAC можно обозначить с помощью таких символических формул:

$$\begin{aligned}\text{«HE-AAC v.1»} &= \text{«AAC»} + \text{«SBR»}; \\ \text{«HE-AAC v.2»} &= \text{«AAC»} + \text{«SBR»} + \text{«PS»}.\end{aligned}$$

17.6.2. Метод спектрального дублирования SBR

Метод спектрального дублирования, или отражения (SBR — Spectral Band Replication) основывается на двух вводимых в рамках данной концепции качественных постулатах:

- в верхней части спектра звукового сигнала содержится в значительной степени избыточная с точки зрения слухового восприятия информация; для слухового восприятия важны только мощностные характеристики высокочастотных составляющих, а фазовые соотношения могут быть искажены без потери воспринимаемого качества восстановленного звукового сигнала;
- на некотором временном интервале существует высокая корреляционная зависимость между мощностными характеристиками верхней и нижней половинами спектра звукового сигнала.

Оба эти утверждения непосредственно следуют из анализа алгоритмов, приведенных, например, в [6.51]. На рис. 17.23 приведена структурная схема НЕ-AAC кодера первой версии.

Суть SBR-алгоритма состоит в том, чтобы не передавать полностью верхнюю половину спектра (коэффициенты МДКП) звукового сигнала. Вместо этого

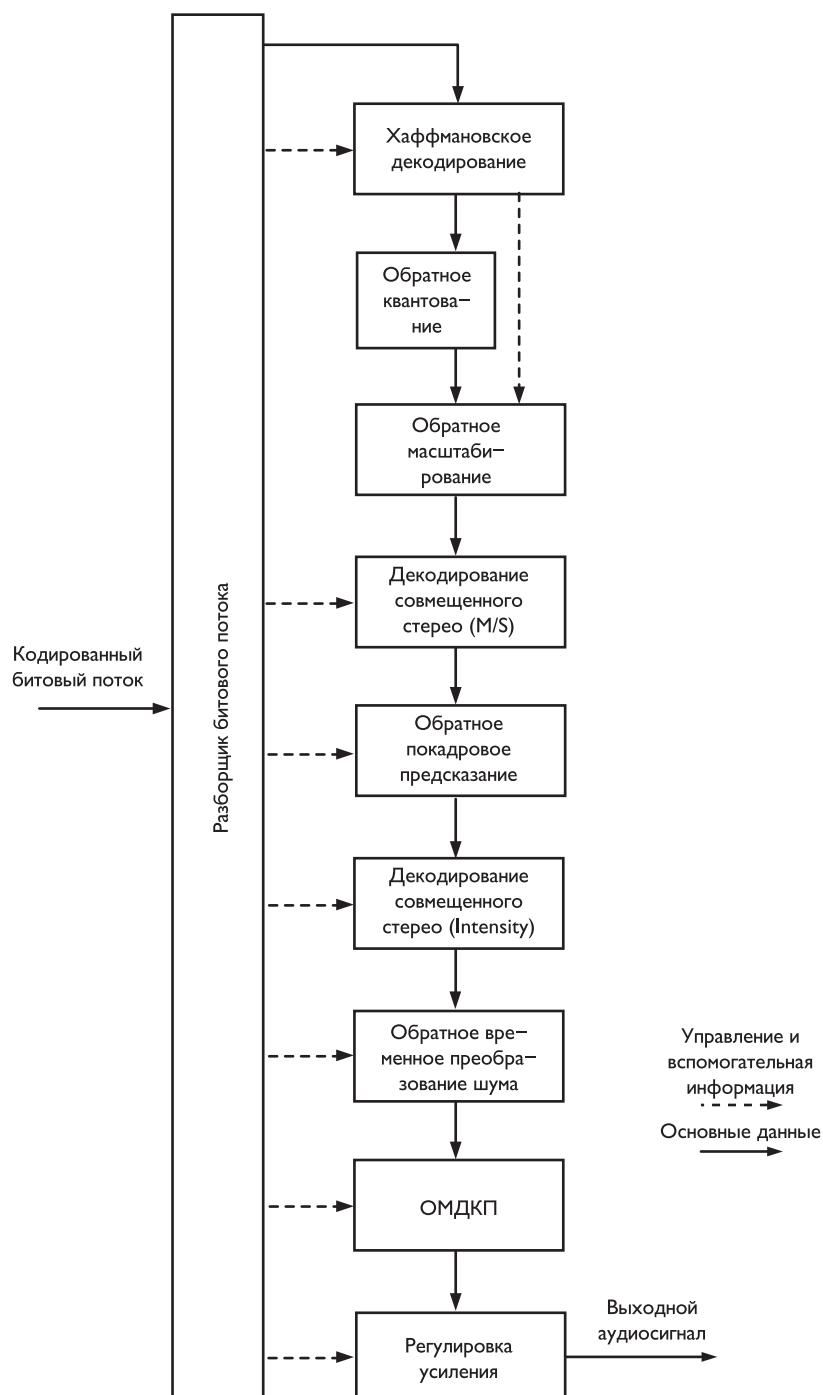


Рис. 17.22. Алгоритмическая схема декодера AAC

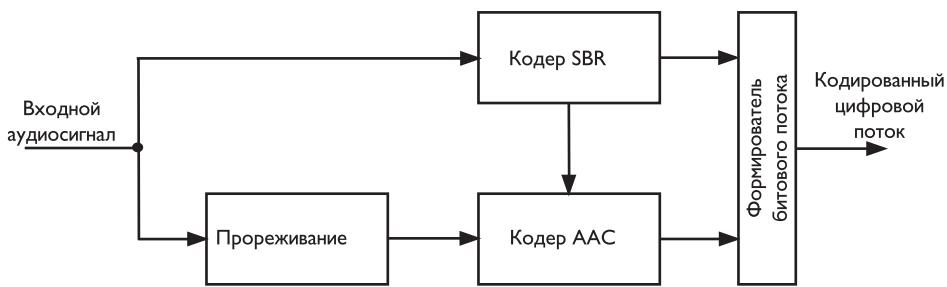


Рис. 17.23. Обобщенная структурная схема кодера HE-AAC v.1

передается информация о мощности спектральных компонент отдельных подполос верхней половины спектра. Кроме этого, на стороне передачи анализируется мощность шумовых составляющих этих подполос, и их уровень также передается декодеру для возможности восстановления исходного отношения сигнал/шум.

Мощность высокочастотных спектральных компонент оценивается линейным предсказателем 2-го порядка с помощью матрицы ковариаций огибающих спектра. SBR-кодер работает с непрореженным по времени сигналом, в то время как на AAC-кодер подается уже прореженный (обычно в два раза) сигнал. Важно отметить, что блок прореживания содержит НЧ-фильтр, предотвращающий зеркальное наложение верхней половины спектра на нижнюю при осуществлении собственно самого прореживания. Помимо перечисленных алгоритмов SBR-кодер отслеживает и оценивает возможность возникновения в высокочастотной части спектра мощных индивидуальных узкополосных сигналов, образующихся, например, вследствие возникновения в звуковом сигнале участков нестационарности спектра (так называемые *transients*). Информация о таких компонентах спектра также передается декодеру. Таким образом, SBR-декодер получает следующую информацию для восстановления звукового сигнала:

- мощностные коэффициенты подполос верхней половины спектра декодируемого сигнала (коэффициенты огибающей);
- мощностные коэффициенты шума для восстановления соотношения сигнал/шум;
- положение и мощность индивидуальных узкополосных сигналов, не поддающихся линейному предсказанию.

Заметим, что данная информация об огибающих спектра передается не напрямую, а в виде разностей оригинальных и предсказанных значений. Поэтому на стороне декодера вновь осуществляется линейное предсказание по низкочастотной части спектра, получаемой от AAC-декодера, после чего производится восстановление (по мощности) огибающих верхней половины спектра сигнала.

Таким образом, полностью передается только нижняя часть спектра сигнала, а высокочастотные составляющие восстанавливаются по сопроводительной информации, составляющей лишь небольшую долю общего потока.

На рис. 17.24 представлена структурная схема кодера SBR. Большинство модулей SBR-кодера оперируют моносигналом, однако некоторым из них требуется доступ к стереосигналу: в режиме работы объединения стерео вычисление энер-

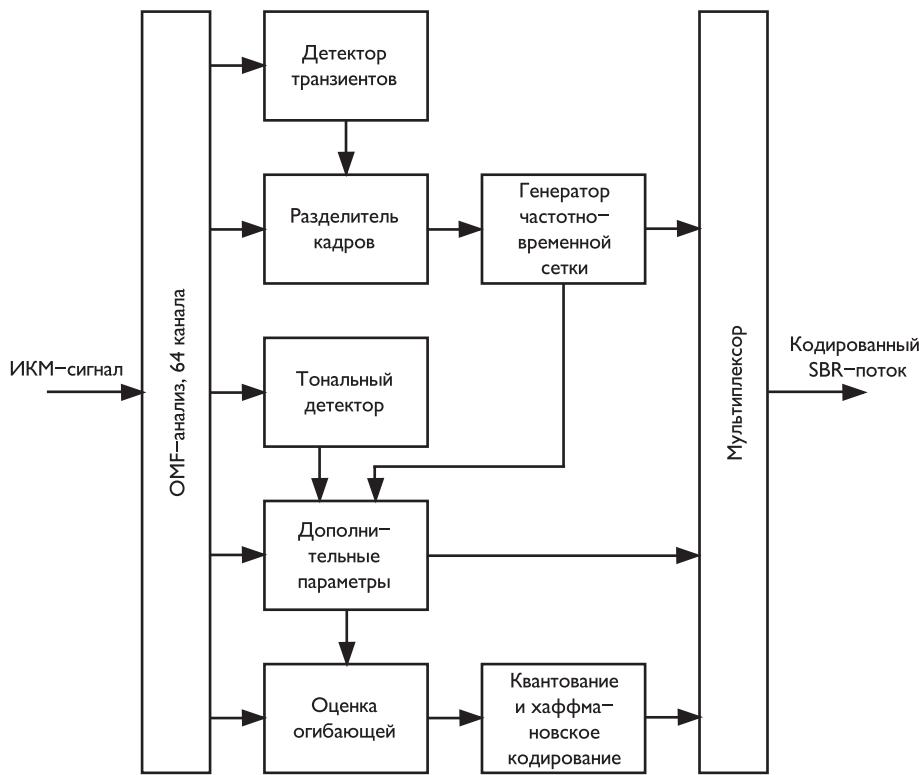


Рис. 17.24. Структурная схема кодера SBR

гетических коэффициентов осуществляется одновременно по отсчетам левого и правого каналов (модуль вычисления коэффициентов огибающей).

Входной ИКМ-сигнал поступает на вход *банка квадратурных зеркальных фильтров QMF*. В этом модуле формируются 32 подполосы, каждая из которых состоит из 64 частотных каналов; результат сохраняется в матрице $X(k, l)$, $0 \leq k < 64$, $0 \leq l < 32$. Выходной поток банка QMF-фильтров составляют подполосные комплексные отсчеты. Комплексность коэффициентов обуславливает вдвое большую скорость потока по сравнению с обычной реализацией QMF-банков. Используемая многополосная фильтрация включает в себя следующие шаги (отсчеты $x(n)$ включают 640 временных отсчетов звукового сигнала, при этом большие индексы у более «старых» отсчетов):

- сдвиг массива x на 64 позиции: старые 64 отсчета стираются, на их место записываются 64 новых отсчета;
- в окне из 640 отсчетов происходит умножение на весовое окно $c(n)$:

$$Z(n) = x(n) \cdot c(n), \quad 0 \leq n < 639;$$

в соответствии с формулой:

$$u(n) = \sum_{j=0}^4 Z(n + j \cdot 128), \quad n = 0, \dots, 127,$$

образуется массив из 128 значений;

- далее происходит формирование действительной и мнимой компонент в соответствии с формулами:

$$r(n) = u(n) - u(127 - n),$$

$$i(n) = u(n) + u(127 - n), \quad n = 0, \dots, 63;$$

вычисляются 64 новых комплексных подполосных отсчета $X = \mathbf{R} + j \cdot \mathbf{I}$ с помощью соответственно косинусного и синусного преобразований:

$$\mathbf{R}(k) = \sum_{i=0}^{63} r(n) \cdot \cos \left[\frac{\pi}{64} \left(k + \frac{1}{2} \right) n \right],$$

$$\mathbf{I}(k) = \sum_{i=0}^{63} i(n) \cdot \sin \left[\frac{\pi}{64} \left(k + \frac{1}{2} \right) n \right], \quad 0 \leq k < 64.$$

Каждый цикл данного алгоритма производит 64 комплексных подполосных отсчета, которые являются выходным отсчетом одной из 64 подполос банка QMF-фильтров. Для каждого SBR-кадра банк фильтров продуцирует 32 подполосных отсчета в каждой из 64 своих подполос.

Детектор транзиентов. Под транзиентами в цифровой обработке звуковых сигналов понимают короткие участки резкого изменения характера звука. В частности, под транзиентом можно понимать резкое изменение локального спектра сигнала. Такое изменение спектра наблюдается, например, при смене одного музыкального инструмента другим: флейта–клавесин–флейта. Детектор транзиентов предназначен для нахождения временной границы этой смены. Он предоставляет возможность разбить звуковой сигнал на условно локальные по спектру участки. На основе результатов работы этого модуля осуществляется построение частотно-временной сетки в модуле разделения кадра.

Модуль разделения кадра активируется только в случае отсутствия транзиентов в текущем кадре. Работа данного модуля осуществляется над подполосными отсчетами длительностью полтора кадра. Выходом разделителя кадров является решение о необходимости разбиении текущего кадра (свободного от транзиентов) на две огибающие одинаковой длительности.

Генератор кадров занимается построением частотно-временной сетки SBR-кадра. На вход поступают результаты работы детектора транзиентов и разделителя кадров. Выходами генератора являются: структура описания частотно-временного разделения текущего кадра и информация о внутреннем представлении частотно-временной сетки, используемая для оценки огибающей и шумового порога.

При отсутствии в текущем кадре транзиентов используется модель с постоянными границами (FIXFIX). Для данной модели генератор принимает решение об использовании либо одной, либо двух огибающих. При обнаружении разнесенных транзиентов (разделенных одним или несколькими кадрами), используются модели FIXVAR и VARFIX. При обнаружении двух последовательных кадров, содержащих транзиенты, используется модель VARVAR.

Модуль оценки огибающей использует частотно-временную сетку, построенную соответствующим генератором. Коэффициенты выходной матрицы QMF-

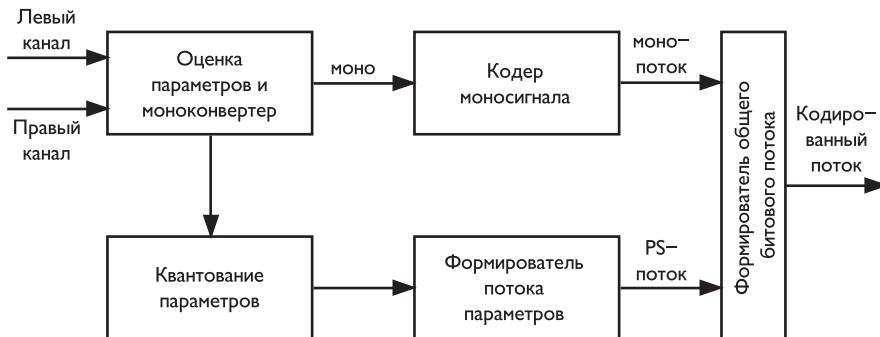


Рис. 17.25. Структурная схема параметрического стереокодирования

банки группируются в частотной и временной областях для вычисления коэффициентов огибающей.

В случае отсутствия гармоник в какой-либо подполосе, вместо усредненного значения энергии, используется ее максимальное значение.

17.6.3. Параметрическое стереокодирование PS

Параметрическое стереокодирование (PS – Parametric Stereo) осуществляет преобразование стереосигнала в моносигнал (downmix), сопровождая его дополнительным набором параметров, позволяющих на стороне декодера восстановить стереообраз. Данное преобразование относится к классу кодирования с информационными потерями. На рис. 17.25 изображена схема PS-кодера.

В блоке «Оценка параметров и моноконвертер» осуществляется оценка следующих четырех типов параметров:

- межканальных разностей уровней (IID – Inter-channel Intensity Differences);
- межканальных разностей фаз (IPD – Inter-channel Phase Differences);
- общего фазового сдвига (OPD – Overall Phase Difference);
- межканальной когерентности (IC – Inter-channel Coherence).

Перечисленных параметров достаточно для того, чтобы восстановить стереообраз на стороне приема. Для восстановления этого стереообраза без заметных на слух искажений под поток этой дополнительной информации достаточно отвести 9–10 кбитов/с. При этом в канале будет передаваться только моносигнал (после процедуры преобразования моно–стерео). Перечисленные 4 параметра квантуются (блок «Квантование параметров») и кодируются как дополнительные части монопотока (в блоке «Формирователь общего битового потока») таким образом, чтобы обеспечить обратную совместимость при воспроизведении данного потока монокодеками, игнорирующими эту дополнительную информацию.

Общая схема HE-AAC-кодека второй версии, включающего в себя кодеры Spectral Band Replication (SBR) и Parametric Stereo (PS), приведена на рис. 17.26.

Заметим здесь еще раз, что системы параметрического кодирования (PS) и спектрального дублирования (SBR) работают с двойной частотой дискретизации сигнала относительно кодера AAC.

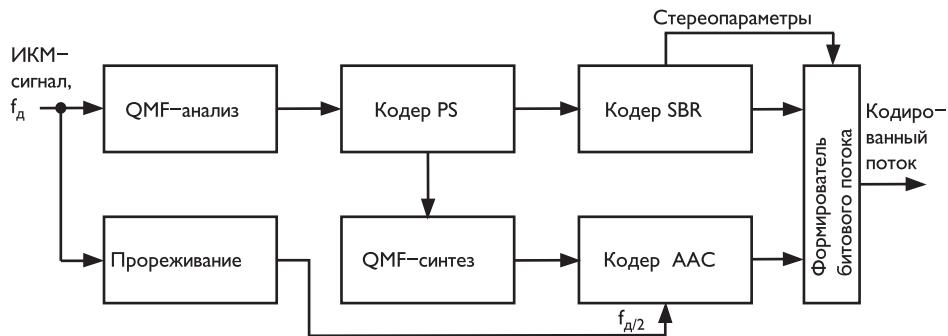


Рис. 17.26. Структурная схема кодера HE-AAC v.2

17.6.4. Конфигурации режимов работы кодеков AAC и HE-AAC

Варианты реализации кодеков стандарта AAC и HE-AAC v.2 приведены соответственно в табл. 17.4 и табл. 17.5 [6.59]. Из этих таблиц видно, насколько кодеки стандарта HE-AAC более эффективны, чем кодеки стандарта AAC.

Таблица 17.4. Режимы работы кодека AAC

Режим каналов	Скорость потока, кбитов/с	Частота дискретизации, кГц	Границчная частота спектра сигнала, кГц
1 канал — моно	16	24	5,2
1 канал — моно	20	24	7,2
1 канал — моно	24	24; 32	7,2; 7,2
1 канал — моно	28	24; 32	10,0; 10,0
1 канал — моно	32	24; 32; 44,1; 48	10,0; 10,0; 10,0; 10,0
1 канал — моно	40	24; 32; 44,1; 48	12,0; 12,4; 12,4; 12,4
1 канал — моно	48	24; 32; 44,1; 48	12,0; 13,5; 13,5; 13,5
1 канал — моно	56	24; 32; 44,1; 48	12,0; 15,5; 15,5; 15,5
1 канал — моно	64	32; 44,1; 48	15,5; 15,5; 15,5
1 канал — моно	80	32; 44,1; 48	16,0; 17,6; 17,6
1 канал — моно	96	32; 44,1; 48	16,0; 22,1; 24,0
1 канал — моно	112	32; 44,1; 48	16,0; 22,1; 24,0
1 канал — моно	128	32; 44,1; 48	16,0; 22,1; 24,0
1 канал — моно	160	32; 44,1; 48	16,0; 22,1; 24,0
2 канала — стерео	24	24	6,6
2 канала — стерео	28	24	6,6
2 канала — стерео	32	24	6,6
2 канала — стерео	40	24; 32	8,5; 8,5
2 канала — стерео	48	24; 32	8,5; 8,5; 8,5
2 канала — стерео	56	24; 32; 44,1; 48	12,0; 12,5; 12,5; 12,5
2 канала — стерео	64	32; 44,1; 48	12,5; 12,5; 12,5
2 канала — стерео	80	32; 44,1; 48	13,5; 13,5; 13,5
2 канала — стерео	96	32; 44,1; 48	15,5; 15,5; 15,5

Таблица 17.4 (окончание)

Режим каналов	Скорость потока, кбитов/с	Частота дискретизации, кГц	Границная частота спектра сигнала, кГц
2 канала — стерео	112	32; 44,1; 48	16,0; 16,0; 16,0
2 канала — стерео	128	32; 44,1; 48	16,0; 16,0; 16,0
2 канала — стерео	160	32; 44,1; 48	16,0; 17,6; 17,6
2 канала — стерео	192	32; 44,1; 48	16,0; 22,1; 24,0
2 канала — стерео	224	32; 44,1; 48	16,0; 22,1; 24,0
2 канала — стерео	256	32; 44,1; 48	16,0; 22,1; 24,0
2 канала — стерео	320	32; 44,1; 48	16,0; 22,1; 24,0

Таблица 17.5. Режимы работы кодека HE-AAC v.2

Режим каналов	Скорость потока, кбитов/с	Частота дискретизации, кГц	Границная частота спектра сигнала, кГц
1 канал — моно	8	24	8,3
1 канал — моно	10	24; 32	10,9; 11,0
1 канал — моно	12	24; 32	11,4; 11,04 10,7
1 канал — моно	16	24; 32; 44,1	12,3; 11,7; 12,0
1 канал — моно	20	32; 44,1; 48	14,5; 14,8; 15,4
1 канал — моно	24	32; 44,1; 48	15,3; 14,8; 15,4
1 канал — моно	28	32; 44,1; 48	16,0; 16,2; 16,9
1 канал — моно	32	32; 44,1; 48	16,0; 16,2; 16,9
1 канал — моно	40	32; 44,1; 48	16,0; 17,6; 18,4
1 канал — моно	48	32; 44,1; 48	16,0; 20,3; 20,3
1 канал — моно	56	32; 44,1; 48	16,0; 20,3; 20,3
1 канал — моно	64	32; 44,1; 48	16,0; 20,3; 20,3
2 канала — стерео	24	32; 44,1; 48	13,8; 12,7; 13,1
2 канала — стерео	28	32; 44,1; 48	15,3; 14,8; 15,4
2 канала — стерео	32	32; 44,1; 48	15,3; 14,8; 15,4
2 канала — стерео	40	32; 44,1; 48	16,0; 16,2; 16,9
2 канала — стерео	48	32; 44,1; 48	16,0; 16,2; 16,9
2 канала — стерео	56	32; 44,1; 48	16,0; 17,6; 18,4
2 канала — стерео	64	32; 44,1; 48	16,0; 20,3; 22,1
2 канала — стерео	80	32; 44,1; 48	16,0; 20,3; 22,1
2 канала — стерео	96	32; 44,1; 48	16,0; 20,3; 22,1
2 канала — стерео	112	32; 44,1; 48	16,0; 20,3; 22,1
2 канала — стерео	128	32; 44,1; 48	16,0; 20,3; 22,1

На рис. 17.27 приведены результаты тестирования качества восстановленного аудиосигнала при его кодировании с использованием наиболее распространенных кодеков: MP3 (MPEG-1 Layer-3), AAC, MP3Pro (усовершенствованный кодек MP3, использующий технологию SBR) и HE-AAC. Тестированию подвергались кодеки стереосигналов при одинаковом цифровом потоке, равном 48 кбитов/с. Организации MPEG и EBU (Европейский союз радиовещателей), осуществлявшие сравнение результатов кодирования, признали явным победителем стандарт HE-AAC [6.60].

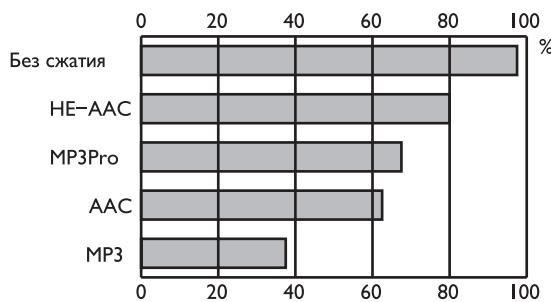


Рис. 17.27. Результаты тестирования стандартов кодирования стереосигналов при цифровом потоке 48 кбитов/с

ГЛАВА 18

ПОДАВЛЕНИЕ АКУСТИЧЕСКОГО ЭХА В СИСТЕМАХ ТЕЛЕКОНФЕРЕНЦСВЯЗИ

Эхо — это задержанная и искаженная копия оригинального акустического или электрического сигнала, отраженная обратно к своему источнику.

В телекоммуникационных сетях существует два типа эха, различные по природе происхождения: акустическое и электрическое. Электрическое эхо присутствует в телефонной сети (PSTN) и выходит за рамки анализа в этой главе. Стоит отметить, однако, что в большинстве случаев подавление электрического эха по ряду причин является значительно более простой задачей, чем подавление эха акустического. В сложных же случаях, например при большой длине эхопути или нелинейностях в канале, задачи подавления акустического и электрического эха идентичны.

В последнее время большое распространение получают так называемые hands-free коммуникации с использованием громкоговорящей акустической системы (динамика) и микрофона вместо привычной телефонной трубки. Это приводит к появлению сильной акустической связи между динамиком и микрофоном.

18.1. Основные положения

Акустическое эхо образуется как благодаря прямому распространению акустической волны от динамика к микрофону, так и благодаря отражениям звука от стен помещения и различных предметов (см. рис. 18.1). Если отраженный сигнал возвращается к источнику звука через очень короткое время (единицы миллисекунд), то это воспринимается как реверберация. Реверберация может несколько уменьшить разборчивость речи, но обычно существенно на комфортность общения не влияет. Однако если задержка начинает превышать десятки миллисекунд, то возвращенный сигнал воспринимается говорящим как эхо, и в ряде случаев делает невозможным общение.

Сигнал от «удаленного конференц-зала», поступающий в «ближний конференц-зал», образует акустическую связь между громкоговорителями и микрофонами: воспроизведенный акустическими системами и многократно отраженный от стен, он улавливается микрофонами «ближнего конференц-зала» и возвращается обратно в «удаленный конференц-зал» с определенной задержкой. Таким образом, участники мероприятия в удаленном конференц-зале будут слышать помимо полезной речи и собственную задержанную и искаженную речь.

Единственный модуль в ближнем конференц-зале, где можно устраниТЬ акустическое эхо, обозначен на рис. 18.1 цифрой «1». Этот модуль отсекает акусти-

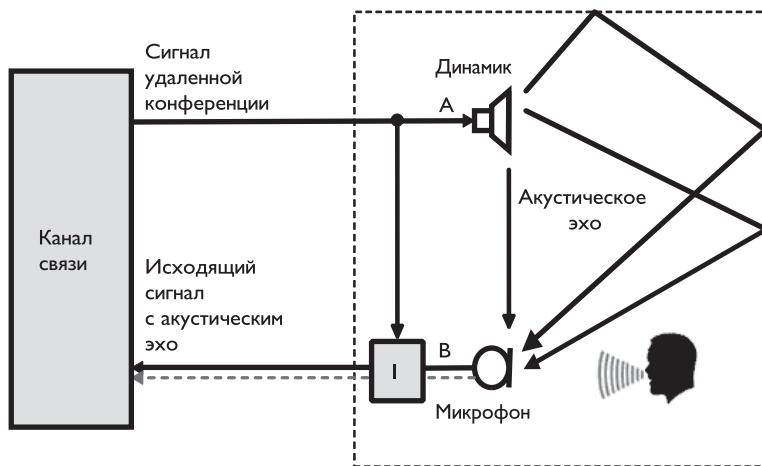


Рис. 18.1. Образование акустического эха в системах телеконференции

ческое эхо от полезного сигнала из ближнего конференц-зала, используя сигнал, приходящий от удаленной конференции. При этом в выигрыше оказываются только участники удаленной конференции, для участников ближней конференции никаких изменений не произойдет. Для того чтобы избавить участников ближней конференции от выслушивания собственного эха, необходимо установить аналогичную систему эхоподавления на удаленной стороне.

Определение и характеристики эхопути

Акустическая часть пути от динамика до микрофона как таковая для создания системы эхоподавления малоинтересна, поскольку требуется смоделировать общий путь прохождения сигнала между точками, доступными для оценки [6.61–6.64].

Под термином эхопуть здесь и далее будет пониматься общий путь сигнала между точками А и В (см. рис. 18.1), включая:

- внутренние буферы цифровые ресемплеры и прочие цифровые алгоритмы обработки звука в программной части системы воспроизведения;
- цифро-аналоговый преобразователь;
- усилитель;
- акустическую систему (динамик);
- пути распространения акустической волны в помещении (включая отражения);
- микрофон;
- микрофонный усилитель;
- аналого-цифровой преобразователь;
- внутренние буферы, цифровые ресемплеры и прочие цифровые системы обработки звука в программной части системы звукозахвата.

Таким образом, эхопуть не является чисто акустическим. Сигнал при прохождении по эхопути подвергается преобразованию сначала из цифровой формы

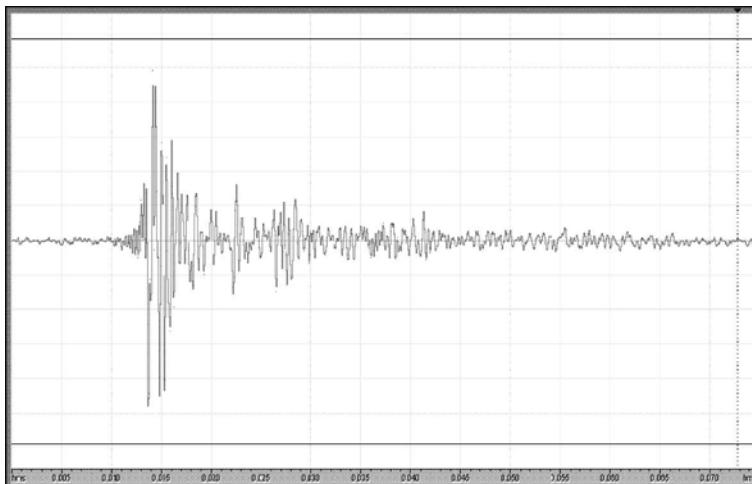


Рис. 18.2. Типичный вид импульсной характеристики эхопути

в аналоговую электрическую, затем в акустическую, затем снова в аналоговую, затем опять в цифровую. В общем виде эхопуть не является линейным, однако в первом приближении может быть описан линейным фильтром с импульсной характеристикой $h(t)$. Это достаточно грубое упрощение, которое не всегда приводит к приемлемому уровню эхоподавления. Обобщение на случай существенно нелинейных систем будет рассмотрено далее.

Эффективная длина импульсного отклика $h(t)$ зависит в основном от акустических свойств помещения. В типичных офисных помещениях длина $h(t)$ обычно не превышает 200–300 мс, однако в больших залах, ангарах и т. д. длина эхопути может достигать нескольких секунд. Типичный внешний вид импульсного отклика приведен на рис. 18.2.

Как можно видеть из графика, реальная импульсная характеристика не содержит в явном виде компонентов, соответствующих прямому прохождению акустического сигнала и основным отражениям, а имеет довольно сложную форму.

Принципиальная схема подавления акустического эха

Рассматривается только одна сторона в телекоммуникационной системе (см. рис. 18.3). Дальний сигнал, приходящий с удаленной стороны (far-end) на рис. 18.3 обозначен $x(n)$ и доступен для прямой оценки.

Сигнал $x(n)$, прошедший весь эхопуть преобразуется в эхосигнал $y(n) = \mathbf{h} \cdot \mathbf{x}$. Сигнал, появляющийся на выходе микрофона $z(n)$ определяется суммой ближней речи диктора $s(n)$, эха $y(n)$ и фонового шума $v(n)$. Не делая предположений о статистических характеристиках $s(n)$ и $v(n)$, отделить их друг от друга не представляется возможным и не является задачей акустического эхоподавления. Задача акустического эхоподавления — исключить из сигнала $z(n)$ эхо $y(n)$, основываясь на известных $z(n)$ и $x(n)$.

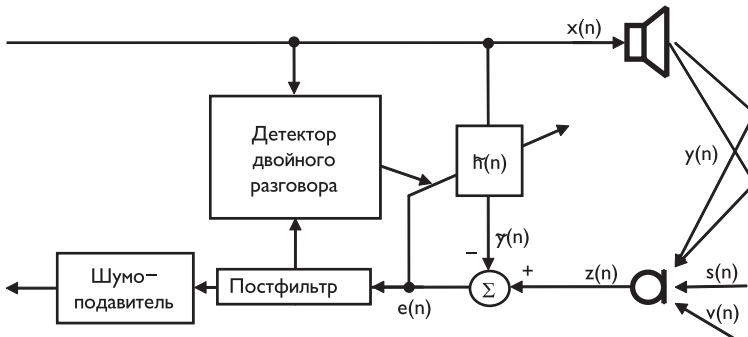


Рис. 18.3. Принципиальная схема подавителя акустического эха

Относительно понятий «эхокомпенсация» и «эхоподавление»

Существуют два основных подхода для подавления акустического эха. В иностранных источниках эти подходы имеют устоявшиеся названия: Acoustic Echo Cancellation (AEC) и Acoustic Echo Suppression (AES). В русскоязычной литературе разделять эти подходы не принято, устоявшихся и общепринятых названий для этих подходов не существует. Более того, довольно часто оба подхода называют термином «акустическое эхоподавление», что является некорректным словесным переводом термина «Acoustic Echo Suppression». Общую задачу борьбы с акустическим эхом, независимо от подхода, назовем в соответствии с заглавием: «Подавление акустического эха», термин «Acoustic Echo Cancellation» переведем как эхокомпенсация, а «Acoustic Echo Suppression» — как эхоподавление. Эти названия, на взгляд авторов, довольно точно отражают суть указанных подходов, за исключением упоминания акустической природы эха.

18.2. Алгоритмы подавления акустического эха

Эхокомпенсация

Эхокомпенсация является традиционным подходом для решения проблемы акустического эха. Эхокомпенсатор подавляет эхо путем вычитания смоделированного эха (удаленного сигнала, пропущенного сквозь фильтр, моделирующий эхопуть) из микрофонного сигнала.

Эхопуть имитируется линейным фильтром длиной L , $\mathbf{h} = \{h_1, \dots, h_L\}^T$. Тогда микрофонный сигнал записывается в виде:

$$z(n) = \mathbf{h}^T \cdot \mathbf{x}(n) + v(n) + s(n), \quad (18.1)$$

где $\mathbf{x}(n) = \{x(n-L+1), \dots, x(n)\}^T$ — сигнал от дальней стороны, $v(n)$ — шум на ближней стороне, $s(n)$ — полезный сигнал на ближней стороне.

Моделирующий фильтр $\tilde{\mathbf{h}} = \{\tilde{h}_1, \dots, \tilde{h}_L\}^T$ используется для аппроксимации истинного эхопути \mathbf{h} . Определив его, можно оценить эхосигнал:

$$\tilde{y}(n) = \tilde{\mathbf{h}}^T \mathbf{x}(n). \quad (18.2)$$

Для поиска $\tilde{\mathbf{h}}$ используются адаптивные алгоритмы.



Эхосигнал может быть успешно исключен, если моделирующий фильтр близок к реальному. В практических приложениях, как правило, моделирующий фильтр, и реальный эхопуть довольно значительно различаются по многим причинам, например связанным с нелинейностью характеристики динамика, быстрыми изменениями акустической среды или рассинхронизацией устройств захвата и воспроизведения звука. Все это приводит к неполному подавлению эха и появлению так называемого остаточного эха.

Эхоподавление

Помимо эхокомпенсации существует другой подход к борьбе с акустическим эхом: эхоподавление. В телефонии эхоподавление реализовывалось как полудуплексный эхозаградитель, пропускающий сигнал по каналу связи только в одну сторону, основываясь на сравнении энергий ближнего и удаленного сигналов. Таким образом, один из абонентов в любой момент времени может либо говорить, либо слушать своего собеседника. Это чрезвычайно простой, но неэффективный метод. Современные требования к качеству звука в телеконференциях и комфорту общения требуют полнодуплексных схем, т. е. позволяющих одновременный встречный разговор. Эхоподавление, реализованное в спектральном домене, аналогично традиционным алгоритмам одноканального шумоподавления, например, спектральному вычитанию, делает возможной реализацию полнодуплексной связи.

Для понимания принципов эхоподавления рассмотрим алгоритм одноканального шумоподавления на базе спектрального вычитания.

Шумоподавление на базе спектрального вычитания

Для подавления аддитивного широкополосного шума в одном канале разработано множество методов. Одним из них является спектральное вычитание, принцип работы которого заключается в следующем.

Строится статистическая модель шума (амплитудный спектр и вариации). Статистика собирается в каждой частотной полосе в моменты отсутствия полезного сигнала. Далее амплитудный спектр шума вычитается из амплитудного спектра входного сигнала с некоторым масштабированием уровня шума. Блок-схема алгоритма представлена на рис. 18.4.

Зашумленный сигнал $x(n)$ состоит из полезного сигнала $s(n)$ и аддитивного стационарного широкополосного шума $v(n)$:

$$x(n) = s(n) + v(n) \quad (18.3)$$

После перехода в спектральный домен имеем:

$$X(f) = S(f) + V(f). \quad (18.4)$$

Для амплитуд спектральных составляющих можно записать приблизительное соотношение:

$$|X(f)| \approx |S(f)| + |V(f)|. \quad (18.5)$$

Тогда амплитуда полезного речевого сигнала может быть представлена как разница амплитуды входного сигнала и амплитуды модели шума:

$$|S(f)| \approx |X(f)| - |\tilde{V}(f)|. \quad (18.6)$$

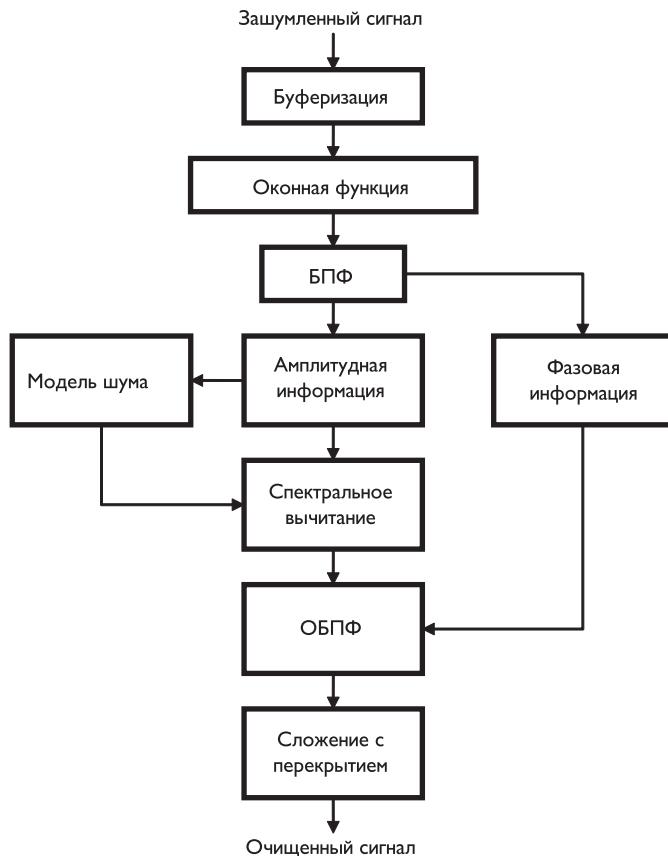


Рис. 18.4. Блок-схема шумоподавления на базе спектрального вычитания

Фазовую информацию восстановить невозможно, поэтому фаза очищенного сигнала полагается равной фазе входного сигнала.

Комбинируя фазовую и амплитудную информации, мы получаем выходной сигнал:

$$S(f) = X(f) \cdot \left(\frac{\max(0, |X(f)|^\alpha - \beta \cdot |V(f)|^\alpha)}{|X(f)|} \right)^{\frac{1}{\alpha}} = G(f) \cdot X(f), \quad (18.7)$$

где α и β — параметры алгоритма, $G(f)$ — коэффициент аттенюации амплитуды спектральной составляющей.

Вычисления $S(f)$ производятся для последовательных перекрывающихся фреймов, которые после перехода во временной домен складываются с перекрытием, что минимизирует краевые эффекты от использования окна конечной длины.

Спектральное вычитание имеет серьезный недостаток: поскольку шум носит случайный характер, он содержит узкополосные всплески амплитуды. Соотношение энергии всплесков и среднего уровня шума невелико, поэтому в общем сигнале они неразличимы. После вычитания среднего значения шума уровень шума в целом резко понижается и соотношение энергий оставшихся всплесков и сред-



него уровня шума резко вырастает, несмотря на то что соотношение сигнал–шум увеличилось.

Всплески воспринимаются как короткие периодические призвуки, т. н. музикальный шум, и довольно неприятны на слух.

Есть множество способов борьбы с музикальным шумом, основанных как на сглаживании по времени и частоте $G(f)$, так и на постфильтрации коротких узкополосных помех.

Эхоподавление на базе спектрального вычитания

В отличие от эхокомпенсации эхоподавление модифицирует спектральную амплитуду сигнала, оставляя неизменной (искаженной) фазу.

По аналогии с одноканальным шумоподавлением можно записать формулу для аттенюации спектральной амплитуды:

$$G(f) = \left(\frac{\max(0, |Z(f)|^\alpha - \beta \cdot |\tilde{Y}(f)|^\alpha)}{|Z(f)|} \right)^{\frac{1}{\alpha}}, \quad (18.8)$$

где $Z(f)$ — спектр микрофонного сигнала, $\tilde{Y}(f)$ — спектр оценки эха.

Для получения $\tilde{Y}(f)$ необходимо использовать адаптивный фильтр.

Сложение с перекрытием

Для корректной работы в частотном домене используется так называемый алгоритм сложения с перекрытием (Overlap-Add). Рассмотрим его более подробно, поскольку этот алгоритм также можно успешно использовать не только для эхоподавления, но и для эхокомпенсации в частотной области.

Если взять конечной длины БПФ от косинусоидального сигнала одной частоты $A \cdot \cos(\omega t)$, то формируется ряд ненулевых спектральных составляющих с частотами, в том числе отличными от ω . Это происходит благодаря краевым эффектам, поскольку используется простейшее прямоугольное окно анализа. Использование такого окна обеспечивает максимальное разрешение по частоте, но это свойство полностью нивелируется слабым подавлением боковых лепестков. Чаще всего для подавления боковых лепестков в качестве окна анализа используется простейшее окно Ханна:

$$w(n) = 0,5 - 0,5 \cdot \cos\{2 \cdot \pi \cdot n/N\}, \text{ где } n = [0, \dots, N-1].$$

Оно также обладает полезным свойством: сложение окон Ханна с шагом $N/2$ дает в сумме единицу.

Таким образом, метод сложения с перекрытием можно представить следующим образом:

- используются фреймы исходного сигнала $\mathbf{z}(n) = \{z(n) \dots z(n+N-1)\}^T$ длиной N с шагом $N/2$;
- каждый фрейм помножается на окно \mathbf{w} с коэффициентами $w(n)$, и вычисляется БПФ:

$$Z(f) = FFT\{\mathbf{z}(n) \cdot \mathbf{w}\};$$

- производятся операции над амплитудным спектром $Z(f)$;

- вычисляются ОБПФ;
- затем полученный сигнал складывается с предыдущими фреймами также с шагом $N/2$.

Сравнение эхокомпенсации и эхоподавления

Метод эхоподавления имеет как свои плюсы, так и минусы. Минус очевиден: эхоподавление не способно восстановить фазу полезного сигнала. При соотношениях сигнал–шум менее нуля дБ фазовые искажения становятся настолько большими, что эхоподавление вносит существенные искажения в речь. Эхокомпенсация, являясь линейной операцией, в принципе не способна внести нелинейные искажения.

Достоинства эхоподавления в более широкой области применения, большей устойчивости и более высокой степени подавления эха. В случае когда возникает рассинхронизация ввода–вывода, например из-за наличия в программной части захвата–воспроизведения звука ресемплеров, подстраивающих выходную частоту в зависимости от наполненности буфера, эхокомпенсация оказывается неэффективной. Резкое снижение уровня громкости на аппаратном уровне (например, регулятором на динамике) может вызвать некорректную работу эхокомпенсатора: некоторое время он будет генерировать так называемое паразитное эхо, поскольку оценочная импульсная характеристика адаптивного фильтра не соответствует реальному эхопути и вычитание будет производиться с избыtkом. В аналогичной ситуации эхоподавление реагирует излишним подавлением полезного сигнала, и ни при каких условиях не сможет сгенерировать паразитное эхо. Вообще, эхокомпенсация очень чувствительна к взаимным сдвигам сигналов x и z относительно друг друга, эхоподавление же успешно справляется с такими сдвигами, пока они значительно меньше длины фрейма.

Также в случае, когда моделирующий фильтр по соображениям экономии вычислительных ресурсов оказывается короче реальной длины эхопути, эхоподавление имеет существенное преимущество перед эхокомпенсацией. Естественно, оба метода можно комбинировать.

18.3. Принципы линейной адаптивной фильтрации

Адаптивный фильтр — важнейший компонент алгоритма подавления эха. Его задача — вычислить линейную аппроксимацию эхопути, наиболее близкую к реальному, причем все вычисления должны быть адаптивными, чтобы учитывать изменяющуюся во времени акустическую обстановку, например перемещения людей по комнате.

Далее удаленный сигнал пропускается сквозь коэффициенты вычисленного фильтра и вычитается из микрофонного сигнала (для эхокомпенсации), либо используется для вычисления спектральной аттенюации (для эхоподавления).

Важнейшей характеристикой адаптивного фильтра является скорость сходимости, которая определяет, насколько быстро фильтр сходится к лучшей аппроксимации эхопути. В алгоритмах эхокомпенсации обычно используются КИХ-фильтры.

Для эхоподавления применяется множество различных алгоритмов адаптивной фильтрации. Наиболее распространенным до сих пор является нормализо-

ванный алгоритм NLMS (Normalized Least Mean Square), требующий относительно небольших вычислительных ресурсов и имеющий приемлемую в большинстве случаев скорость сходимости. Существует множество разновидностей этого алгоритма, реализованных на базе быстрого преобразования Фурье: FDNLMS, PFDNLMS и др. Есть еще более простой, но медленно сходящийся LMS, ресурсоемкий RLS, алгоритмы, основанные на аффинных проекциях, и т. д.

Винеровская фильтрация

Фильтр Винера является оптимальным в смысле среднеквадратической ошибки. Он минимизирует функцию затрат, параметром которой является вектор \mathbf{w} весовых коэффициентов адаптивного фильтра:

$$J(\mathbf{w}) = E\{e^2\}, \quad (18.9)$$

где $E\{e^2\}$ — средняя энергия сигнала ошибки $e(n)$.

Оптимальные коэффициенты \mathbf{w}_{opt} минимизируют функцию затрат:

$$J(\mathbf{w}_{opt}) = \min(E\{e^2\}). \quad (18.10)$$

Сигнал ошибки определяется как разница между желаемым сигналом и выходом адаптивного фильтра:

$$e(n) = z(n) - y(n) = z(n) - \mathbf{w}^T \cdot \mathbf{x}(n), \quad (18.11)$$

где $\mathbf{x}(n) = \{x(n-L+1) \dots x(n)\}^T$ — вектор входного сигнала фильтра, L — длина фильтра.

Введем обозначения:

$\mathbf{R} = E\{\mathbf{x}^T \mathbf{x}\}$ — автокорреляционная матрица;

$\mathbf{p} = E\{\mathbf{x}d\}$ — вектор кросскорреляции.

Тогда функцию затрат можно переписать в виде:

$$J(\mathbf{w}_{opt}) = \sigma_d^2 - 2 \cdot \mathbf{p} \cdot \mathbf{w}^T + \mathbf{R} \cdot \mathbf{w}^T \cdot \mathbf{w}. \quad (18.12)$$

Минимум квадратичной функции может быть определен по нулевому градиенту $J(\mathbf{w}_{opt})$:

$$\nabla_w \{J(\mathbf{w})\} = -2 \cdot \mathbf{p} + 2 \cdot \mathbf{R} \cdot \mathbf{w} = 2 \cdot (\mathbf{R} \cdot \mathbf{w} - \mathbf{p}) = 0, \quad (18.13)$$

что приводит к уравнению Винера–Хопфа:

$$\mathbf{w}_{opt} = \mathbf{p} \mathbf{R}^{-1}, \quad (18.14)$$

которое дает коэффициенты винеровского фильтра, оптимальные в смысле минимизации среднеквадратической ошибки.

Коэффициенты фильтра Винера зависят от статистики \mathbf{R} и \mathbf{p} , которую можно собирать для отслеживания изменений \mathbf{w}_{opt} .

Инверсия матрицы \mathbf{R} является вычислительно трудноосуществимой задачей, поэтому были предложены несколько способов упрощенного вычисления, которые и привели к алгоритмам (N)LMS и RLS.

Алгоритм LMS

Алгоритм LMS использует метод наискорейшего спуска. Вместо прямого вычисления точки \mathbf{w}_{opt} гораздо проще следовать градиенту функции ошибки (вернее,

в противоположном направлении), что в конечном счете итеративно приведет к оптимуму, поскольку $J(\mathbf{w})$ — положительно определенная квадратичная функция.

На каждом шаге вектор весов обновляется:

$$\mathbf{w}(k+1) = \mathbf{w}(k) - c \cdot \nabla_w \{J(\mathbf{w})\}, \quad (18.15)$$

$$\begin{aligned} \nabla_w \{J(\mathbf{w})\} &= -2 \cdot \mathbf{x}(k) \cdot d(k) + 2 \cdot \mathbf{x}(k) \cdot \mathbf{x}^T(k) \cdot \mathbf{w} = \\ &= -2 \cdot \mathbf{x}(k) \cdot [d(k) - \mathbf{x}^T(k) \cdot \mathbf{w}] = -2 \cdot \mathbf{x}(k) \cdot e(k). \end{aligned}$$

Это приводит к алгоритму LMS:

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu \cdot \mathbf{x}(k) \cdot e(k), \quad (18.16)$$

где $\mu = 2c$ — параметр, контролирующий скорость сходимости фильтра.

Алгоритм NLMS

NLMS появился как дальнейшее развитие алгоритма LMS. В LMS трудно подобрать оптимальное значение параметра μ . Поскольку энергия входного сигнала изменяется во времени, то и шаг между двумя итерациями коэффициентов фильтра, т. е. скорость сходимости, будет меняться. Скорость сходимости LMS будет мала на слабых сигналах и может быть слишком большой, т. е. проскаакивать искомый минимум, — на больших. Идея алгоритма в нормализации скорости сходимости и подстройке шага к энергии входного сигнала:

$$\mu = \frac{2\alpha}{\|\mathbf{x}^2(k)\|}, \quad (18.17)$$

где α — новый параметр, ответственный за скорость сходимости, но теперь устойчивость алгоритма гарантируется для $0 < \alpha < 1$. NLMS сходится значительно быстрее LMS при несколько большем количестве вычислений.

18.4. Детектор двойного разговора

Детектор двойного разговора — важный модуль в системе акустического эхоподавления, необходимый для работы в полнодуплексном режиме. В моменты наличия двойного разговора сигнал, снимаемый с микрофона, содержит не только эхо, но и ближний сигнал. Это приводит к расхождению коэффициентов адаптивного фильтра. Поэтому в моменты возникновения двойного разговора необходимо замораживать адаптацию коэффициентов фильтра. Управление адаптацией и есть задача детектора двойного разговора.

Общая схема детектирования двойного разговора

Детектирование двойного разговора основывается на статистике, формируемой из доступных для наблюдения сигналов: сигнал, отправляемый на динамики (дальний сигнал), сигнал, захваченный с микрофона, и выходной сигнал. По этой статистике тем или иным способом вычисляется некая метрика ξ , которая сравнивается с фиксированным порогом Thr . По результату сравнения выносится решение о наличии двойного разговора и необходимости заморозки процесса адаптации.

Существует достаточно много разновидностей детекторов двойного разговора, основанных на различных алгоритмах вычисления метрики ξ . Наиболее простым является алгоритм Гейгеля. Также популярны алгоритмы, основанные на нормализованной кросскорреляции и на динамике коэффициентов адаптивного фильтра (Variance Impulse Response).

Алгоритм Гейгеля

Это наиболее простой алгоритм, который чаще всего используется для детектирования двойного разговора в случае электрического эха. Он основан на сравнении энергий дальнего и ближнего сигналов. Поскольку импульсный отклик помещения обычно ослабляет дальний сигнал, то в случае если амплитуда микрофонного сигнала, деленная на максимальную амплитуду дальнего сигнала, превышает определенный порог, выносится решение о наличии двойного разговора. Метрика ξ вычисляется следующим образом:

$$\xi = \frac{|z(t)|}{\max(|x(t)|, |x(t-1)|, \dots, |x(t-L+1)|)}. \quad (18.18)$$

Если ξ превышает некий фиксированный порог, то детектируется наличие двойного разговора.

Выбор порога Thr представляет определенную сложность, поскольку он существенно влияет на точность детектора.

Алгоритм Гейгеля требует очень малых вычислительных ресурсов, но в силу своей низкой точности практически не используется в системах подавления акустического эха.

Детектирование двойного разговора по нормализованной кросскорреляции

Более точным методом является алгоритм, основанный на нормализованной кросскорреляции. Микрофонный сигнал представим в виде суммы эхосигнала и ближней речи (наличие шума в данном случае игнорируется):

$$z(t) = y(t) + s(t). \quad (18.19)$$

Предположим, что импульсный отклик помещения — \mathbf{h} , тогда эхосигнал представим в виде:

$$y(t) = \mathbf{h}^T \cdot \mathbf{x}(t). \quad (18.20)$$

Энергия микрофонного сигнала может быть записана как:

$$\sigma_z^2(t) = \mathbf{h}^T \cdot \mathbf{R}_{xx} \cdot \mathbf{h} + \sigma_s^2(t), \quad (18.21)$$

где $\mathbf{R}_{xx} = E\{\mathbf{x}^T \cdot \mathbf{x}\}$. Кросскорреляция микрофонного и дальнего сигналов может быть записана как:

$$\mathbf{r}_{xy} = E\{\mathbf{x}y\} = \mathbf{R}_{xx} \cdot \mathbf{h} \quad (18.22)$$

и, следовательно,

$$\mathbf{h} = \mathbf{r}_{xy} \cdot \mathbf{R}_{xx}^{-1} \quad (18.23)$$

При этом энергия микрофонного сигнала может быть переписана как:

$$\sigma_z^2(t) = \mathbf{r}_{xy}^T \cdot \mathbf{R}_{xx}^{-1} \cdot \mathbf{r}_{xy} + \sigma_s^2(t). \quad (18.24)$$

При отсутствии ближней речи, т. е. когда $s(t) = 0$, $z(t) = y(t)$,

$$\sigma_z^2(t) = \mathbf{r}_{xz}^T \cdot \mathbf{R}_{xx}^{-1} \cdot \mathbf{r}_{xz}, \text{ где } \mathbf{r}_{xz} = E\{\mathbf{xz}\}. \quad (18.25)$$

Метрика детектирования двойного разговора определяется как:

$$\xi = \left[\frac{\mathbf{r}_{xz}^T \mathbf{R}_{xx}^{-1} \mathbf{r}_{xz}}{\sigma_z^2(t)} \right]^{\frac{1}{2}}. \quad (18.26)$$

Числитель в этом соотношении является не чем иным, как энергией микрофонного сигнала в отсутствие ближней речи, в то время как знаменатель — реальная энергия микрофонного сигнала. При отсутствии ближней речи $\xi \sim 1$, иначе $\xi < 1$.

Решающее правило, таким образом, формулируется следующим образом:

$$\begin{aligned} \xi < Thr &— двойной разговор присутствует, \\ \xi \geq Thr &— двойной разговор отсутствует. \end{aligned}$$

Пороговое значение Thr выбирается между нулем и единицей.

В представленном выше виде алгоритм вычислительно практически нереализуем, поскольку требует не только вычисления \mathbf{r}_{xz} и \mathbf{R}_{xx} , но и инвертирования ковариационной матрицы дальнего сигнала \mathbf{R}_{xx} . Практическая реализация требует дополнительных упрощений. Импульсный отклик помещения \mathbf{h} приблизительно (в случае сопредшегося фильтра) можно аппроксимировать коэффициентами этого фильтра \mathbf{w} . Тогда:

$$\xi = \sqrt{\frac{\mathbf{r}_{xz}^T \mathbf{w}}{\sigma_z^2(t)}} = \sqrt{\frac{\mathbf{w}^T \mathbf{R}_{xx}^T \mathbf{w}}{\sigma_z^2(t)}} = \sqrt{\frac{\sigma_y^2(t)}{\sigma_z^2(t)}}. \quad (18.27)$$

В числителе — энергия оцененного эхосигнала, в знаменателе — реальная энергия микрофонного сигнала. В таком виде алгоритм реализуем без заметных вычислительных затрат. Очевидной проблемой этого алгоритма является зависимость от стартовой оценки коэффициентов адаптивного фильтра \mathbf{w} .

Детектирование двойного разговора по динамике коэффициентов

Алгоритм известен под аббревиатурой VIRE DTD (Variance Impulse Response). Он использует максимальное значение коэффициентов адаптивного фильтра. Основная идея алгоритма состоит в следующем: при отсутствии ближней речи коэффициенты фильтра должны сходиться к оптимальной точке, причем со временем шаг сходимости будет уменьшаться. При наличии двойного разговора коэффициенты фильтра будут испытывать значительные флуктуации.

Итак, максимальное значение коэффициентов фильтра определяется:

$$\gamma = \max\{w_1, \dots, w_L\}. \quad (18.28)$$

Проводится усреднение по времени с параметром λ :

$$\gamma'(n) = \lambda \cdot \gamma' \cdot (n-1) + (1-\lambda) \cdot \gamma \quad (18.29)$$

и вычисляется метрика детектирования двойного разговора:

$$\xi(n) = \lambda \cdot \xi(n-1) + (1-\lambda) \cdot [\gamma - \gamma']^2. \quad (18.30)$$

Решающее правило так же формулируется с помощью фиксированного порога:

$\xi \geqslant Thr$ — двойной разговор присутствует,

$\xi < Thr$ — двойной разговор отсутствует.

Данный детектор двойного разговора, также как и предыдущий, работает для всего фрейма, т. е. вычисляется в конце каждого фрейма.

18.5. Подавление акустического эха на базе фильтра Калмана

Фильтры Калмана применительно к задаче подавления акустического эха нельзя рассматривать как еще один адаптивный алгоритм, поскольку они позволяют существенно изменить общепринятую схему построения системы эхоподавления, а именно полностью избавиться от самого ненадежного блока в системе: детектора двойного разговора.

Ранее было указано, что разделение полезного сигнала $s(n)$ и шумового сигнала $v(n)$ не является задачей эхоподавления и это разделение невозможно без информации о статистических свойствах $s(n)$ и $v(n)$. В реальной телеконференции можно столкнуться с ситуацией, когда на ближнем конце шум будет присутствовать постоянно, например шум от вентилятора блока питания персонального компьютера. То есть система эхоподавления будет вынуждена всегда работать в режиме двойного разговора.

Рассмотренная выше схема в таких условиях будет неработоспособна, поскольку адаптация коэффициентов фильтра будет всегда заморожена детектором двойного разговора. Можно изменить порог срабатывания детектора, но тогда порог придется подстраивать под каждый вид шума.

Рассмотрим классическую задачу оценки состояния с помощью фильтра Калмана. Уравнение процесса выглядит следующим образом:

$$\mathbf{x}_k = \mathbf{F}_k \cdot \mathbf{x}_{k-1} + \mathbf{B}_k \cdot \mathbf{u}_k + \mathbf{w}_k, \quad (18.31)$$

где \mathbf{x}_k — вектор искомого (неизвестного) состояния системы; \mathbf{F}_k — известная матрица эволюции процесса/системы, \mathbf{x}_{k-1} — вектор состояния системы в момент $(k-1)$; \mathbf{B}_k — известная матрица управления, \mathbf{u}_k — также известный вектор управляющих воздействий, \mathbf{w}_k — нормальный случайный процесс с нулевым математическим ожиданием и ковариационной матрицей \mathbf{Q}_k .

Уравнение, описывающее наблюдение (оценку), выглядит следующим образом:

$$\mathbf{z}_k = \mathbf{H}_k \cdot \mathbf{x}_k + \mathbf{v}_k, \quad (18.32)$$

где \mathbf{z}_k — оцениваемый (наблюдаемый) вектор; \mathbf{H}_k — матрица наблюдений (известная); \mathbf{v}_k — шум наблюдения, нормальный случайный процесс с нулевым математическим ожиданием и ковариационной матрицей \mathbf{R}_k .

Задача оценки состояния состоит в нахождении оценки $\hat{\mathbf{x}}_k$ по наблюдаемой \mathbf{z}_k и при известных \mathbf{F}_k , \mathbf{B}_k , \mathbf{H}_k .

Применительно к задаче подавления акустического эха мы можем перейти в спектральный домен и рассматривать каждую частотную полосу независимо. Пе-

реход обратно во временную область будет осуществляться методом сложения с перекрытием. Это позволит существенно уменьшить длину векторов и, следовательно, общее количество вычислений. Поскольку частотные полосы полагаются независимыми, ниже индекс, соответствующий номеру частотной полосы, указываться не будет.

Пусть коэффициенты адаптивного фильтра в одной частотной полосе задаются вектором $\mathbf{W}(k)$ с длиной L , равной соотношению длины моделируемого эхопути к шагу алгоритма сложения с перекрытием.

Будем рассматривать коэффициенты адаптивного фильтра $\mathbf{W}(k)$ как искомое состояние, а изменение коэффициентов как случайный гауссовский процесс с нулевым средним и известной ковариационной матрицей:

$$\mathbf{W}(k+1) = \mathbf{W}(k) + \Delta\mathbf{W}(k), \quad (18.33)$$

где $\Delta\mathbf{W}(k)$ — вектор шума процесса с нулевым средним, описывающий случайные вариации эхопути.

Переоценка ковариационной матрицы производится во время работы алгоритма, начальная оценка — диагональная матрица с одинаковыми коэффициентами на главной диагонали. Начальное значение влияет только на первые доли секунды выходного сигнала.

В уравнениях процесса и наблюдения сделаем подстановки в соответствии с принятыми нами обозначениями:

$$\begin{aligned} \mathbf{x}_k &\rightarrow \mathbf{W}(k); \\ \mathbf{F}_k &\rightarrow 1; \\ \mathbf{B}_k &\rightarrow 0; \\ \mathbf{H}_k &\rightarrow \mathbf{X}(k); \\ \mathbf{v}_k &\rightarrow S(k) + V(k) — ближний сигнал (полезный + шум); \\ \mathbf{z}_k &\rightarrow S(k) + V(k) + Y(k) — микрофонный сигнал. \end{aligned}$$

Используя решение задачи оценки состояния можно записать:

$$\begin{aligned} E(k) &= Z(k) - \mathbf{X}(k)\mathbf{W}(k-1); \\ \Psi_{EE}(k) &= \alpha \cdot \Psi_{EE}(k-1) + (1-\alpha) \cdot |E(k)|^2; \\ \mathbf{K}(k) &= \mathbf{P}(k-1) \cdot \mathbf{X}^H(k) \cdot [\mathbf{X}(k) \cdot \mathbf{P}(k-1) \cdot \mathbf{X}^H(k) + \Psi_{EE}(k)]^{-1}; \\ \Delta\mathbf{W}(k) &= E(k) \cdot \mathbf{K}(k); \\ \Psi_{\Delta\Delta}(k) &= \beta \cdot \Psi_{\Delta\Delta}(k) + (1-\beta) \cdot \Delta\mathbf{W}(k); \\ \mathbf{W}(k) &= \mathbf{W}(k-1) + \Delta\mathbf{W}(k); \\ \mathbf{P}(k) &= \mathbf{P}(k-1) - \mathbf{K}(k) \cdot \mathbf{X}(k) \cdot \mathbf{P}(k-1) + \text{diag}\Psi_{\Delta\Delta}(k), \end{aligned}$$

где $Z(k)$ — спектр микрофонного сигнала; $\mathbf{X}(k)$ — вектор-строка, состоящая из L последних значений спектра удаленного сигнала $X(k)$; $\mathbf{W}(k)$ — искомый вектор-столбец весов адаптивного фильтра; $E(k)$ — спектр ошибки, он же является спектром выходного сигнала для эхокомпенсации.

Принципиальное отличие подхода, основанного на фильтре Калмана, от описанных ранее состоит в изначальном учете наличия ближнего сигнала. (N)LMS строится на предположении, что $S(k) = 0$ и $V(k) = 0$ в любой момент времени, и при несоблюдении этого условия адаптивный фильтр расходится. Здесь же



допущение значительно более слабое. Полагается, что шум оценки (ближний сигнал) является случайным процессом, имеет нулевое среднее и диагональную ковариационную матрицу (в данном случае она вырождается в скаляр $\Psi_{EE}(k)$). Это допущение достаточно грубое и для речевого сигнала в общем случае неверное, но при работе в одной независимой частотной полосе дает удовлетворительные результаты.

Если для обратного БПФ берется напрямую сигнал ошибки $E(k)$, то мы получаем схему эхокомпенсации, если для модификации входного сигнала $Z(k)$ используется спектральное вычитание, то реализуется схема эхоподавления.

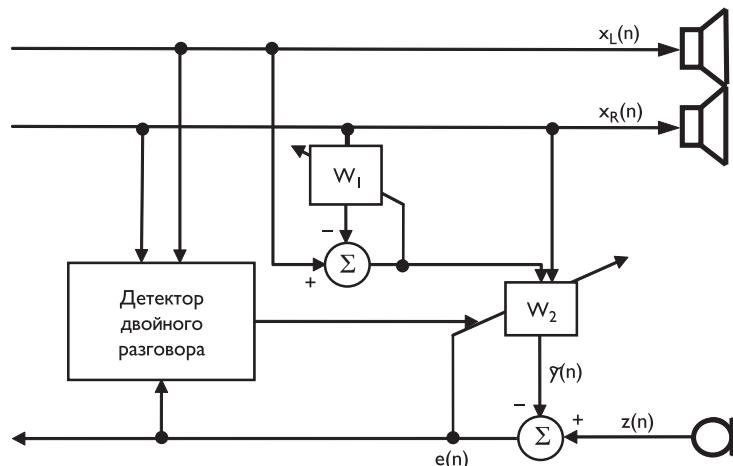
18.6. Особые случаи

Стереоэхоподавление

В случае когда удаленный сигнал является одноканальным, наличие двух или более динамиков в системе воспроизведения никак не влияет на задачу эхоподавления. При удаленном стереосигнале $x_L(n)$, $x_R(n)$ можно использовать один набор коэффициентов адаптивного фильтра. Для каждого канала должна быть своя версия адаптивного фильтра. Проблема такого подхода заключается в том, что каналы в стереозвуке, как правило, сильно коррелированы между собой. То есть все допущения о диагональности корреляционных матриц приращений весов являются неприменимыми.

Одним из способов решения этой проблемы является декорреляция каналов перед системой эхоподавления, которая также может быть проведена с помощью адаптивного фильтра. Одна из возможных схем построения такой системы приведена на рис. 18.5.

Рис. 18.5. Схема эхоподавления в системе стереоконференции



Нелинейная постфильтрация

Независимо от того, насколько хорошо работает эхоподавитель, эхо никогда не будет подавлено полностью, и некоторая ослабленная часть эха, так называемое остаточное эхо, еще будет присутствовать в выходном сигнале. Причинами

возникновения остаточного эха, как правило, являются нелинейности в эхопути и ограниченность длины моделирующего фильтра.

Одним из наиболее простых методов подавления остаточного эха является центральное клиппирование выходного сигнала. Принцип метода предельно прост: каждый сэмпл выходного сигнала, чей модуль не превышает определенного фиксированного порога, приравнивается к нулю. В случаях когда остаточное эхо обычно имеет довольно низкий уровень, метод работает весьма эффективно. Порог часто делают адаптивным в зависимости от уровня фонового шума.

18.7. Метрологические требования к подавлению акустического эха

Рекомендации МСЭ [6.65, 6.66] не определяют конкретные требования к архитектуре эхокомпенсаторов и эхоподавителей. Они содержат минимальные требования по скорости сходимости, уровню подавления и описания соответствующих тестов. Поскольку эти рекомендации ориентированы в первую очередь на подавление эха в телефонных линиях, то для создания подавителей акустического эха они особого интереса не представляют. Но эти стандарты определяют основные термины для количественного описания работы систем эхоподавления.

Уровень подавления эха

Критерием эффективности подавления акустического эха служит величина дополнительного подавления эха системой, измеряемая в дБ (Echo Return Loss Enhancement или ERLE):

$$ERLE = 10 \cdot \lg[E\{z^2(k)\}] - 10 \cdot \lg[E\{e^2(k)\}], \quad (18.34)$$

где $E\{z^2(k)\}$ — оценка энергии микрофонного сигнала; $E\{e^2(k)\}$ — оценка энергии выхода системы эхоподавления.

В случае когда алгоритм содержит параметры, позволяющие регулировать уровень подавления эха, необходимо иметь средства контроля за качеством выходного полезного сигнала, поскольку увеличение уровня подавления эха неизбежно приведет к увеличению искажений ближней речи. Для оценки качества речи при наличии записи эталонного (неискаженного) сигнала может использоваться рекомендация МСЭ [6.67].

Скорость сходимости

Время сходимости адаптивного фильтра можно определить как время, достаточное для достижения коэффициентами фильтра стационарного состояния, т. е. время, после которого градиент изменения коэффициентов будет описываться случайным процессом с нулевым средним.

К сожалению, время сходимости зависит не только от типа и параметров адаптивного алгоритма, но и от входного сигнала. Обычно время сходимости измеряют при отсутствии ближнего сигнала, поскольку при наличии двойного разговора алгоритм LMS и производные от него расходятся, алгоритм, основанный на калмановском фильтре, существенно снижает скорость сходимости.

Типичное время сходимости для рассмотренных выше адаптивных алгоритмов при отсутствии ближнего сигнала — от 0,5 до 5 секунд.

Оценка надежности детектора двойного разговора

Некорректная работа детектора двойного разговора приводит к некорректной работе системы в целом. Ошибки детектора двойного разговора можно разделить на два вида:

- ложная тревога (когда дальняя речь при отсутствии ближней детектируется как двойной разговор),
- пропуск двойного разговора (когда фрейм, содержащий двойной разговор, детектируется как не содержащий ближней речи).

Ошибки первого типа приводят к замедлению сходимости адаптивного алгоритма, поскольку в моменты, когда алгоритм мог корректно модифицировать коэффициенты фильтра адаптация была заморожена.

Ошибки второго типа имеют еще более неприятные последствия, поскольку при этом происходит разбегание коэффициентов адаптивного фильтра, и, как следствие, увеличение уровня остаточного эха.

Регулированием порога срабатывания детектора двойного разговора можно найти оптимальное соотношение между этими двумя ошибками. Если построить график зависимости вероятности ошибки первого типа от ошибки второго типа для всех допустимых значений порога, то получится график, аналогичный представленному на рис. 18.6. Чем меньше площадь под кривой, тем более надежным можно считать детектор.



Рис. 18.6. График надежности работы детектора двойного разговора

Литература к части VI

- 6.1. *Алдошина И.А.* Основы психоакустики. — <http://aldan-theatre.my1.ru/project-studio/Psychoacoustics-Aldoshina.pdf>, <http://inform-hifi.narod.ru/3.htm>.
- 6.2. *Радзиневский А.Ю.* Основы аналогового и цифрового звука. М.-СПб-Киев: Изд. дом «Вильямс», 2006.
- 6.3. *Попов О.Б., Рихтер С.Г.* Цифровая обработка сигналов в трактах звукового вещания. М.: Горячая линия-Телеком, 2007.
- 6.4. *Вемян Г.В.* Передача речи по сетям электросвязи. М.: Радио и связь, 1985.
- 6.5. Радиовещание и электроакустика /Под ред. Ю.А. Ковалгина. М.: Радио и связь, 1999.
- 6.6. *Кирн П.* Реальный мир. Цифровой звук. М.-СПб-Киев: Изд. дом «Вильямс», 2008.
- 6.7. *Музыченко Е.* FAQ(Часто задаваемые вопросы) по электронному созданию и обработке звука. — <http://websound.ru/articles/theory/soundfaq.htm>.
- 6.8. *Музыченко Е.* FAQ (Часто задаваемые вопросы) по цифровому представлению звуковых сигналов. — <http://websound.ru/articles/theory/dgaudfaq.htm>.
- 6.9. ГОСТ 21185-75. Измерители уровня квазипиковые. Типы и основные параметры. Методы испытаний. М.: Изд. стандартов, 1976.
- 6.10. *Vigovsky A.* Громкость — теоретические сведения. — <http://ac3filter.net/>, 2006.
- 6.11. *Дэвис Г., Джонс Р.* Звук: теория, устройства, практические рекомендации// Цифровое видео. № 4–6, 2002; № 1–4, № 6, 2003. — www.digitalvideo.ru.
- 6.12. Общие данные о звуке и шуме. — <http://www.hvacinfo.ru/toolbox/ac.php>.
- 6.13. Бинауральный эффект звука. Звукозапись. История развития. — <http://covers.by.ru/theory/x4.htm>.
- 6.14. *Балабан А.* Многоканальный окружающий звук. — <http://www.muzoborudovanie.ru/articles/sur/surround.php>.
- 6.15. *Закаменский В.* Три измерения звука. — <http://www.technofresh.ru/technology/poligon/3d-sound-technologies.html>.
- 6.16. Технология создания позиционируемого 3D звука. / iXBT — <http://www.ixbt.com/multimedia/3dsound-tech.html>
- 6.17. *Cummiskey P., Jayant N.S., Flanagan J.L.* Adaptive Quantization in Differential PSM Coding of Speech// Bell Syst. TECH Jour. Vol. 52. Sept., 1973.
- 6.18. Application Note AN-283: Sigma-Delta ADCs and DACs. — Applications Reference Manual., Analog Devices, 1993.
- 6.19. *Хоровиц П., Хилл У.* Искусство схемотехники. Т. 2 / Пер. с англ. М.: Мир, 1983.
- 6.20. *Голуб В.С.* Цифровая обработка сигналов: Сигма-дельта АЦП// Электроника: Наука. Технология. Бизнес. 2002. № 4.



- 6.21. Методы и стандарты передачи речи по трактам связи, применяемые в современном оборудовании (7 кГц). —
http://www.communications.su/equipment/stand_pered_rechi.html.
- 6.22. *Ковалгин Ю.А., Вологдин Э.И.* Цифровое кодирование звуковых сигналов. СПб.: КОРОНА принт, 2004.
- 6.23. *Fastl H., Zwicker E.* Psycho-Akustiks// Springer-Verlag. Berlin, Heidelberg, 1990, 1999, 2007.
- 6.24. ITU-T. Recommendation G.711. Pulse Code Modulation (PCM) of Voice Frequencies. (1972), 1988, 1993.
- 6.25. ITU-T. Recommendation G.711.1. Wideband embedded extension for G.711 pulse code modulation. 03/2008.
- 6.26. ITU-T. Recommendation G.711.0. Lossless compression of G.711 pulse code modulation. 09/2009.
- 6.27. ITU-T. Recommendation G.722. 7 kHz Audio-Coding within 64 kbit/c. 1988, 1993.
- 6.28. ITU-T. Recommendation G.722.1. Low complexity coding at 24 and 32 kbit/s for hands-free operation in systems with low frame loss. 05/2005.
- 6.29. ITU-T. Recommendation G.722.2. Wideband coding of speech at around 16 kbit/s using Adaptive Multi-Rate Wideband (AMR-WB). 03/2004.
- 6.30. ITU-T. Recommendation G.723.1. Dual rate speech coder for multimedia communications transmitting at 5.3 and 6.3 kbit/s. 05/2006.
- 6.31. ITU-T. Recommendation G.726. 40, 32, 24, 16 kbit/s Adaptive Differential Pulse Code Modulation (ADPCM), 1990.
- 6.32. ITU-T. Recommendation G.727. 5-, 4-, 3- and 2-bits Sample Embedded Adaptive Differential Pulse Code Modulation (ADPCM). 1990.
- 6.33. ITU-T. Recommendation G.728. Coding of Speech 16 kbit/s using Low-Delay Code Excited Linear Prediction. 09/1992.
- 6.34. ITU-T. Recommendation G.729. Coding of Speech at 8 kbit/s Using Conjugate-Structure Algebraic-Code-Excited Linear Prediction (CS-ACELP). 03/1996.
- 6.35. ITU-T. Recommendation G.729. Annex A. Reduced Complexity 8 kbit/s CS-ACELP Speech Codec. 11/1996.
- 6.36. ITU-T. Recommendation G.729. Annex B. A silence compression scheme for G.729 optimized for terminals conforming to Recommendation. V.70. 02/1998.
- 6.37. ITU-T. Recommendation G.729.1. G.729-based embedded variable bit-rate coder: An 8–32 kbit/s scalable wideband coder bitstream interoperable with G.729. 05/2006.
- 6.38. ETSI EN 300 726 V8.01. Digital Cellular Telecommunications System(Phase 2+) (GSM); Enhanced Full Rate (EFR) Speech Transcoding (GSM 06/60 version 8/01 Release 1999). 11/2000.
- 6.39. Кодирование речи. —
http://study.ustu.ru/view/aid/2/1/DOTS_DSP_SC.html.

- 6.40. *Vary P.* GSM Speech Codec// Conference Proceedings DCRC, 12–14 October, 1988. Hagen FRG.
- 6.41. IETF Rec. 3951 on Internet Low Bit Rate Codec (iLBC). 12/2004.
- 6.42. IETF Rec. 3952. Real-time Transport Protocol (RTP) Payload Format for internet Low Bit Rate Codec (iLBC) Speech. 12.2004.
- 6.43. Рекурсия Левинсона. Академические словари и энциклопедии — <http://perevod.yandex.ru/en/?url=http%3A%2F%2Fdic.academic.ru%2Fdic.nsf%2Fenwiki%2F141007>.
- 6.44. *Valin, J.M.* The Speex Codec Manual [Electronic resource]// Speex documentation. 2008.— Mode of access: <http://www.speex.org/docs/>, 31.07.2008.
- 6.45. Speex: A Free Codec For Free Speech. — <http://www.speex.org/>.
- 6.46. Speex: A free codec for free speech. — <http://www.speex.org/comparison/>.
- 6.47. ISO/IEC 11172-3. Information Technology - Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1.5 Mbit/s. Part 3: Audio./ Ed.1, JTS 1/ SC 29, 1993.
- 6.48. *Theile G., Stoll G.* MUSICAM Surround: A Universal Multi-channel Coding System Compatible with ISO 11172-3//Preprints AES 93RD Cjnvention tht Moscone Cjnvention Center. San-Francisco, Cflifornia, USA, 1992. October, 1–4 (N 3403).
- 6.49. ISO/IEC 13818-3. Information Technology . Generic Coding of Moving Pictures and Associated Audio Information. Part 3: Audio / Ed.1, JTS 1/ SC 29, 1994.
- 6.50. ISO/IEC 13818-7. Information Technology - Generic Coding of Moving Pictures and Associated Audio Information. Part 7: Advanced Audio Coding (AAC)./ Ed.1, JTS 1/ SC 29, 1997.
- 6.51. ISO/IEC 14496-3: Information technology — Coding of audio-visual objects — Part 3: Audio, AMENDMENT 1: HD-AAC profile and MPEG Surround signaling/ JTC1/SC29/WG11, 2005.
- 6.52. ATSC Digital Television Standard.//Doc. A/53., 1995.
- 6.53. Радзиневский А.Ю. Стандарты MPEG и их история.— <http://websound.ru/faq.htm>;
<http://audiophilesoft.ucoz.ua/publ/theory/mpeg/6-1-0-30>.
- 6.54. MPEG-2/4 Кодирующее устройство AAC-LC и Декодер MPEG-2/4 AAC-LC Encoder and Decoder.— http://perevod.yandex.ru/en/?url=http%3A%2F%2Fwww.adaptivedigital.com%2Fproduct%2Faudio%2Faac_lc64x_encdec.htm.
- 6.55. О стандартах MPEG и об их истории. Обзор форматов сжатия аудио. — http://3-51.ru/blog_post.php?blogid=8565&p=210263.
- 6.56. *Иртюга В.А.* AAC — прогрессивный метод цифрового кодирования звука. «Цифровая обработка сигналов». 2007. № 3.
- 6.57. Rec. ITU-R BO.651 Digital PCM coding for the emission of high-quality sound signals in satellite broadcasting (15 kHz nominal bandwidth). 07/86.

- 6.58. Rec. CCITT J.17 Pre-emphasis used on sound-programme circuits/ Blue Book Fascicle III.6. 11/88.
- 6.59. Проектирование и инсталляция радиостудий. ПИРС. — http://www.radiopirs.ru/optcodpc_p3.htm.
- 6.60. Формат сжатия звука aacPlus. Stereo&Video 1/2005. — http://www.stereo.ru/whatiswhat.php?article_id=258.
- 6.61. *Enzner Gerald, Vary Peter.* Frequency domain adaptive Kalman filter for acoustic echo control in hands-free telephones. Signal Processing 86, 2006.
- 6.62. Malik S. and Enzner G.: «Model-based vs. Traditional Frequency-Domain Adaptive Filtering in the Presence of Continuous Double-Talk and Acoustic Echo Path Variability», Proc. of Intl. Workshop on Acoustic Echo and Noise Control (IWAENC), Seattle (Washington). Sept. 2008.
- 6.63. *Ephraim Y. and Malah D.* «Speech enhancement using minimum mean square error log-spectral amplitude estimator»// IEEE Trans. Vol.ASSP-33. № 2, p. 443–445. April, 1985.
- 6.64. Cohen I. and Berdugo B. »Speech enhancement for non-stationary noise environments»// Signal Processing. Vol.81. № 11, p. 2403–2418. Nov., 2001.
- 6.65. Rec. ITU-T G.164 (11/88). Echo suppressors. Doc. E 7071, 1990-03-08.
- 6.66. Rec. ITU-T G.165 (03/93). Echo cancellers. Doc. E 4246, 1994-03-09.
- 6.67. Rec. ITU-T P.862 (02/01): Perceptual Evaluation of Speech Quality (PESQ): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs. Doc. E 22134, 2004-06-07.

ЧАСТЬ VII

ОСНОВНЫЕ МЕТОДЫ И СИСТЕМЫ КАНАЛЬНОГО КОДИРОВАНИЯ И МОДУЛЯЦИИ ЦИФРОВОЙ ИНФОРМАЦИИ

Введение

Широкое внедрение цифровых систем связи и вещания обусловлено несколькими причинами. В частности, только цифровые системы обеспечивают передачу требуемого или максимального количества информации за заданное время, с требуемым качеством (достоверностью), при минимальных энергетических затратах.

Многие источники информации являются цифровыми (информация в ПК, аудио- и видеоносители и др.), цифровые способы хранения данных обеспечивают возможность долгосрочного сохранения информации без потерь, реализуют более высокие характеристики. Используемые принципы цифровой обработки сигналов позволяют достаточно просто применять такие алгоритмы обработки, которые нереализуемы или практически недостижимы с использованием аналоговых устройств (например, специальные характеристики фильтров).

Цифровые методы позволяют существенно повысить характеристики приемно-передающей аппаратуры. Ввиду этого достигается значительно большая функциональность цифровых систем по сравнению с аналоговыми, так как смена выполняемых функций, стандарта связи, вида модуляции, обновление алгоритмов обработки до более современной версии осуществляется простой перезагрузкой программы.

Создание единого мирового информационного пространства неразрывно связано с широким использованием преобразования различных видов информации



в цифровую форму, ее хранением и передачей по проводным и беспроводным каналам связи.

Реализация цифровых методов обработки, передачи, приема и восстановления различных видов информации позволяет устранить многие недостатки аналоговых систем, в том числе искажения сигналов, накапливающиеся при увеличении числа преобразований.

Круг задач, решаемых в системах преобразования информации от ее источника до потребителя, поясняется с помощью представленной на рис. VI.1 структурной схемы, в которой не показаны преобразователь непрерывного сигнала на входе источника дискретной информации и преобразователь дискретного сигнала в аналоговую информацию у потребителя [7.1]. Конечно, реальная система передачи/приема дискретной информации содержит большое число функциональных элементов, не отраженных на рис. VI.1, к которым необходимо отнести задающие генераторы, системы управления, блоки синхронизации, усилители мощности, радиоприемники и др. Без этих «вспомогательных» элементов реализация действующей системы невозможна.

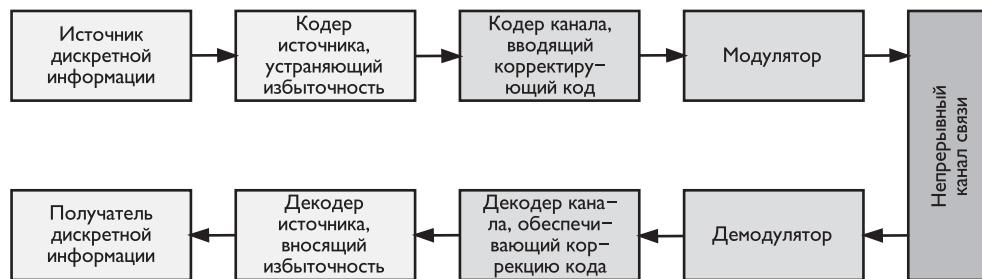


Рис. VI.1. Структурная схема передачи/приема дискретной информации

Цифровой источник сигналов генерирует цифровой сигнал — последовательность битов (символов) дискретной информации. Цифровым источником может быть, к примеру, персональный компьютер, цифровой носитель аудио- или видеоинформации.

Цифровая последовательность поступает на кодер источника сигнала, основной задачей которого является сжатие информации — чем меньше объем информации, который необходимо передавать через радиоканал в единицу времени, тем меньше ошибок возникнет при приеме информации, меньше требуемая полоса частот и энергия, которую необходимо затратить на передачу.

Кодер канала используется практически во всех современных системах цифровой связи и предназначен для повышения достоверности информации на приемной стороне.

Однако увеличение достоверности передачи информации происходит не безвоздемдно, а путем добавления избыточности к передаваемой информации, что приводит к некоторому снижению скорости передачи.

Процесс добавления избыточности к исходной информации с целью повышения достоверности передачи выполняет функцию помехоустойчивого кодирования.

Достоверность передачи информации в цифровых системах характеризуется статистической величиной — вероятностью ошибочного приема при передаче

одного бита информации, усредненной для статистически большого объема передаваемой информации (BER – Bit Error Rate). Величиной, характеризующей эффективность цифровой системы связи, является пропускная способность (бит/с). Пропускная способность характеризует количество информации, которое может быть передано в системе связи в единицу времени (со 100%-й достоверностью). Верхняя граница пропускной способности в системе при заданном отношении сигнал/шум и доступной полосе передачи устанавливается теоремой Шеннона.

На практике достигается лишь некоторое приближение к пределу, устанавливаемому этой теоремой, поскольку она устанавливает величину пропускной способности для канала, единственным видом искажений в котором является аддитивный белый гауссовский шум. На практике же в канале присутствует множество других видов искажений.

В практических системах не требуется передавать информацию со 100%-й достоверностью. Поэтому вместо величины пропускной способности, характеризующей достоверную передачу информации, оперируют величиной «скорость передачи при заданном уровне битовой ошибки».

Воспроизведение исходной дискретной информации на приемной стороне реализуется двумя декодирующими устройствами — декодером канала, осуществляющим коррекцию кода, и декодером источника, вносящим устраниенную избыточность исходной информации.

Модулятор преобразует цифровую информацию в аналоговый радиосигнал.

Существует два основных критерия сравнения эффективности различных видов модуляции (манипуляции) — спектральная и энергетическая эффективность. Спектральная эффективность характеризует полосу частот, необходимую для передачи информации с определенной скоростью. Энергетическая эффективность характеризует мощность, необходимую для передачи информации с заданной достоверностью (вероятностью ошибки).

Спектр модулированного сигнала на радиочастоте с точностью до постоянного множителя совпадает со спектром модулирующего сигнала, однако центр спектра радиосигнала размещен на несущей частоте, а не на нулевой. Поэтому, как правило, анализируются спектральные плотности модулирующих сигналов, центрированные относительно нулевой частоты.

Используется несколько различных видов цифровой модуляции — амплитудная, фазовая или частотная манипуляция или их сочетание.

Модуляция может быть линейной или нелинейной. Для линейных типов модуляции, к которым относятся амплитудная и фазовая манипуляции, справедливо линейное соотношение между спектром модулирующего сигнала и спектром модулированного колебания. Также линейны соотношения между амплитудой модулированного сигнала и исходным информационным сигналом и полной фазой модулированного сигнала и информационным сигналом. Частотная манипуляция является нелинейной.

Для линейных процессов справедлив принцип суперпозиции, поэтому для них можно параллельно изменять два параметра несущего колебания — амплитуды и фазы несущей частоты.

Модуляция с таким видом преобразования радиосигнала наиболее широко используется в системах цифровой связи и называется квадратурной амплитудной модуляцией (QAM — Quadrature Amplitude Modulation).

ГЛАВА 19

ПОМЕХОУСТОЙЧИВОЕ КОДИРОВАНИЕ В СИСТЕМАХ ПЕРЕДАЧИ ЦИФРОВОЙ ИНФОРМАЦИИ

Основной задачей помехоустойчивого кодирования является обеспечение высокой достоверности передачи данных по неидеальным каналам связи с ограниченной пропускной способностью за счет использования методов избыточного кодирования.

Прежде чем перейти к рассмотрению вопросов построения систем эффективного канального кодирования цифровой информации, следует более подробно изложить вопросы кодирования источников сообщений.

19.1. Принципы кодирования источника сообщений

Средства связи выполняют задачу передачи символов источника сообщений с использованием собственного алфавита символов, который наилучшим образом согласован с условиями физической передачи и характеристиками среды распространения. В силу этого обстоятельства характерным для цифровых систем связи является преобразование символов источника сообщений $A \in X = \{x_1, x_2, \dots, x_m\}$ в символы Q другого алфавита, состоящего из d элементов (рис. 19.1). Согласно [7.2], преобразователь, осуществляющий кодирование, обеспечивает согласование источника с каналом связи в статистическом смысле.

Процесс сопоставления символов A множества X символам Q другого алфавита U называется кодированием сообщений источника. Множество $U = \{u_1, u_2, \dots, u_d\}$ называется *алфавитом кода*, а его элементы $u_j, j = \overline{1, n}$ — *кодовыми символами*. Каждый из полученных символов Q размерности n , отображающих сообщения A , называется *кодовой последовательностью* (кодовым словом), а полное множество слов объемом d^n — *кодом*. В подавляющем числе современных систем связи основание алфавита кода d равно 2, что соответствует формированию кодером источника двоичных кодов.

В указанных выше условиях задача кодирования источника будет состоять во взаимно-однозначном назначении различных двоичных комбинаций длины n битов символам A источника сообщений.

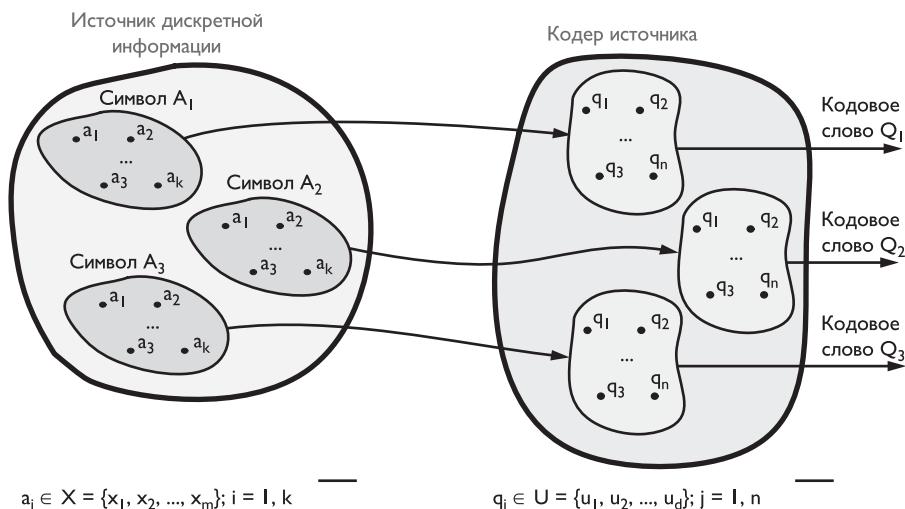


Рис. 19.1. Преобразование символов источника сообщений в символы алфавита кодера

19.1.1. Теоретические предпосылки помехоустойчивого кодирования

При наличии идеального дискретного канала связи, вероятность возникновения ошибки в котором пренебрежимо мала, возникает вопрос: при каких условиях будет обеспечена без потерь наиболее эффективная передача по этому каналу связи информации дискретного источника с энтропией $H(X)$. Ответ на этот вопрос дает основная теорема Шеннона для дискретного канала связи без помех [7.2].

Теорема Шеннона для канала связи без помех. Если имеется дискретный канал связи с пропускной способностью C (бит в секунду), то сообщения любого источника с энтропией H (бит на символ) можно так закодировать, что окажется возможным передавать эти сообщения со скоростью, сколь угодно близкой к C/H символов в секунду или, что то же самое, со скоростью C битов в секунду. Передавать символы со скоростью больше, чем C/H , невозможно.

Как было отмечено выше, кодер обеспечивает согласование источника с каналом связи в статистическом смысле. Согласно теореме Шеннона, точное согласование характеристик в общем случае невозможно, к нему можно приблизиться сколь угодно близко. Мерой эффективности подобного согласования может выступать отношение фактической скорости передачи символов источника R к пропускной способности канала связи C .

В качестве иллюстрации выполнения теоремы Шеннона можно рассмотреть следующий пример [7.3]. Пусть имеются: источник сообщения со статистически независимыми элементами a_i (дискретный источник без памяти — ДИБП) с основанием алфавита $m = 10$, кодер источника с основанием $d = 2$ и идеальный канал связи с пропускной способностью $C = 1$ бит/с. При однобуквенном представлении символов источника ($k = 1$) энтропия источника составляет $H_1 H_{\max} = \log_2 10 \approx 3,332$ бита. Согласно теореме Шеннона, скорость передачи символов Q в таком канале может быть сколь угодно близка к $C/H \approx 1/3,332$. При $k = 1$



одной букве $a_i \in \{0, 1, 2, \dots, 9\}$ кодером ставится в соответствие кодовая комбинация размерности n (рис. 19.1). Для представления одной десятичной цифры требуется $n = 4$ кодовых символа $q_j \in \{0, 1\}$, тогда как, в силу теоремы, можно осуществить более эффективное кодирование, приближаясь к пределу 3,332 бита на символ источника.

В условиях ДИБП и фиксированной размерности кода n эффективное кодирование можно обеспечить путем укрупнения алфавита источника ($k > 1$). При двухразрядном десятичном представлении символа A источника одной букве $(a_i a_{i+1}) \in \{0, 1, 2, \dots, 99\}$ будет соответствовать минимальное значение длины кода $n = 7$. При таком кодировании на каждый знак a_i сообщения будет приходиться в среднем уже 3,5 бита кодера. При трехбуквенном представлении ($k = 3$) одному символу источника $(a_i a_{i+1} a_{i+2}) \in \{0, 1, 2, \dots, 999\}$ будет соответствовать минимальное значение длины кода $n = 10$ (так как $2^{10} = 1024$). Следовательно, при таком кодировании потребуется в среднем $n/k \approx 3,333$ бита кодера, что существенно ближе к теоретическому пределу 3,332. Продолжая укрупнение алфавита источника, можно еще больше приблизиться к пределу Шеннона, но никак не превзойти его.

Полученный пример показывает, что в идеальном канале с пропускной способностью C можно, применяя равномерный код (код с постоянной длиной n кодовых комбинаций), передавать сообщения любого источника с алфавитом m со скоростью, сколь угодно близкой к $C/H_{\max}C/\log_2 m$ символов в секунду со средней длиной кодового слова на один символ сообщения:

$$n_{\text{cp}} = \frac{n}{k} \geq \left[\frac{H(X)}{H(Q)} + \delta \right] = \left[\frac{\log_2 m}{\log_2 d} + \delta \right], \quad (19.1)$$

где $\delta > 0$ — бесконечно малая величина.

Практический вывод, сделанный на основе рассмотренного выше примера, обобщается следующим утверждением [7.4]:

Для заданных ДИБП с энтропией $H(X)$ и кодового алфавита из d символов возможно так закодировать блоки из k букв источника, что средняя длина кодовых слов на одну букву источника n_{cp} будет удовлетворять условию:

$$\frac{H(X)}{\log d} \leq n_{\text{cp}} \leq \frac{H(X)}{\log d} + \frac{1}{k}. \quad (19.2)$$

Источник сообщения в виде модели ДИБП является идеальным и не имеет избыточности, его производительность максимальна и равна $H_{\max} = \log m$. С помощью рассмотренной выше процедуры сообщение может быть закодировано для передачи по идеальному каналу связи с пропускной способностью, большей H_{\max} .

В случае источника с избыточностью $H < H_{\max}$ возникает необходимость такого кодирования, при котором будет устранена или существенно уменьшена избыточность источника. Избыточность источников информации может быть связана с неравномерным распределением вероятностей алфавита или может определяться статистической зависимостью последовательных знаков источника.

Как правило, в качестве физических процессов, несущих информацию, выступают речевые и видеосигналы, различные текстовые и голосовые сообщения. Подобные источники характеризуются очень сильной статистической связью между соседними символами $a_i, a_{i-1}, a_{i-2}, \dots, a_{i-h}$ и достаточно точно описываются марковской моделью. Интуитивно понятно, что любая вероятностная связь

между символами означает снижение среднего количества информации, приходящегося на символ a_i . В свою очередь, это приводит к неэкономному расходованию канального ресурса в виде назначения символам источника кодовых комбинаций избыточной длины. Как следствие — увеличение времени передачи сообщений или потребность в большей пропускной способности канала связи по сравнению со статистически независимыми символами сообщения.

Избыточность, связанная с неравномерным распределением вероятностей алфавита, характерна, например, для текстовых сообщений.

В случае когда источник формирует элементарные сообщения a_i с разными вероятностями и независимо друг от друга, подобная избыточность может быть полностью или частично устранена, если при кодировании представлять наиболее вероятные знаки короткими кодовыми комбинациями, а менее вероятные — более длинными. Такие неравномерные коды требуют наличия специальных разделителей между символами или должны формироваться так, чтобы начало символа для каждой кодовой комбинации было уникальным. Коды, удовлетворяющие вышеуказанному условию, называются префиксными (неприводимыми) кодами.

Для случая неравномерного кода из теоремы Шеннона следует правило, по которому каждому символу дискретного источника с энтропией H ставится в соответствие среднее количество битов для передачи сообщения по каналу связи [7.4]:

$$n_{cp} = \frac{H(X)}{\log d} + \varepsilon, \quad n_{cp} = H(X) + \varepsilon, \quad d = 2, \quad (19.3)$$

где $\varepsilon > 0$ — бесконечно малая величина, d — основание кода. Из выражения (19.3) следует, что минимальная средняя длина двоичного кодового слова Q , приходящаяся на один символ сообщения A , не может быть выше энтропии источника $H(X)$.

Таким образом, теорема Шеннона указывает на существование такого способа кодирования, при котором каждый символ кода обладал бы максимальной информативностью. Следовательно, для двоичного алфавита каждый элементарный символ должен принимать значения 0 или 1 с примерно равными вероятностями и каждый символ не должен зависеть от значений предшествующих знаков. Код, для которого средняя длина кодовых комбинаций равна наименьшему из возможных значений, является *оптимальным* с точки зрения максимальной информативности передаваемого символа.

В каналах с помехами эффективным средством повышения достоверности передачи сообщений является помехоустойчивое кодирование. Оно основано на применении специальных кодов, которые корректируют ошибки, вызванные действием помех. Для того чтобы код обладал корректирующими способностями, в кодовой последовательности должны содержаться дополнительные (избыточные) символы, предназначенные для коррекции ошибок. Чем больше избыточность кода, тем выше его корректирующая способность. Коды, обеспечивающие выполнение этих задач, называются *помехоустойчивыми*, а сам процесс кодирования — *помехоустойчивым* (избыточным) кодированием.

В основе помехоустойчивого кодирования лежит фундаментальная теорема Шеннона для дискретного канала с помехами.

Теорема Шеннона для канала связи с помехами. Если источник с энтропией H создает сообщения со скоростью R , меньшей пропускной способности

С дискретного канала связи с помехами, то при любом $\delta > 0$ существует способ кодирования и декодирования источника, при котором сообщения передаются получателю с вероятностью ошибки, меньшей, чем δ , и, в среднем, без растущих задержек во времени. Если $R > C$, то такого способа кодирования не существует.

Важной особенностью теорем Шеннона является то обстоятельство, что они носят характер теорем существования и не указывают на практические способы реализации процедур кодирования и декодирования.

Идею помехоустойчивого кодирования можно проиллюстрировать на примере равномерных блочных кодов следующим образом.

Пусть из набора элементарных (в частности, двоичных) символов составляется множество V размерности N двоичных кодовых комбинаций. При этом только N_0 кодов могут быть использованы для передачи — такие кодовые комбинации называются *разрешенными*. Остальные $N - N_0$ кодов не используются для передачи сообщений — они называются *запрещенными* (рис. 19.2).

При таком способе организации кодера канала появляется потенциальная возможность обнаруживать и исправлять ошибки. Принцип обнаружения ошибок состоит в следующем.

Получателю известно множество *разрешенных* и множество *запрещенных* комбинаций. Очевидно если при искажении информации в канале связи с помехами переданная разрешенная комбинация переходит в запрещенную, то ошибка обнаруживается, поскольку получатель знает, что такая комбинация не может быть передана.

Процедура исправления ошибок в процессе декодирования сводится к определению переданной комбинации по известной принятой. Минимальное значение количественной меры близости принятой запрещенной комбинации к любому из разрешенных кодов может рассматриваться как критерий исправления существующей ошибки. Если же одна разрешенная комбинация переходит в другую разрешенную, то такая ошибка не может быть обнаружена. Следовательно, не все ошибки обнаруживаются избыточными кодами — они имеют ограниченную возможность по обнаружению и коррекции ошибок.

Кодовые комбинации с выхода кодера поступают на модулятор, который с помощью определенного метода модуляции реализует их отображение в аналоговый сигнал $S(t)$. В соответствии с заданным видом модуляции в модуляторе осуществляется преобразование блока битов $\{b_0, b_1, \dots, b_{k-1}\}$, $k = \log_2 M$ (M — позиционность модуляции) в канальный символ $S_j(t)$, $j = 1, M$. Выбор вида канальной модуляции определяется компромиссом между результирующей пропускной способностью системы и требованием высокой достоверности передачи данных (см. главу 20). Наиболее распространены следующие виды модуляции: BPSK (Binary Phase-Shift Keying), QPSK (Quadrature Phase-Shift Keying), m -QAM (Quadrature Amplitude Modulation), $m = 16, 64, 256$. С увеличением позици-

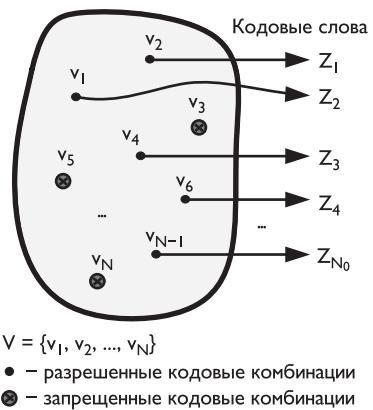


Рис. 19.2. Иллюстрация конечного множества кодовых комбинаций кодера канала

онности модуляции растет спектральная эффективность — способность передачи данных с большей скоростью в фиксированной полосе частот. К примеру, при переходе от BPSK сигналов к QPSK скорость передачи цифрового потока возрастает в 2 раза при неизменной занимаемой полосе частот, при переходе от QPSK к 16-QAM — в 4 раза. Цена подобного улучшения — существенное увеличение необходимого для обеспечения высокой достоверности передачи данных отношения сигнал/шум на входе приемника.

В физическом канале сигнал $S(t)$ подвергается воздействию шума $n(t)$. Для количественной оценки степени влияния шума $n(t)$ на сигнал $S(t)$ обычно используют *отношение сигнал/шум*:

$$\frac{P_{\text{пс}}}{P_{\text{шш}}} = \frac{E_s/T}{N_0\Delta F} = \frac{E_s}{N_0},$$

определенное как отношение мощности сигнала P_c к мощности шума $P_{\text{шш}}$; E_s , T — энергия и длительность канального символа соответственно, N_0 — спектральная плотность шума в полосе частот $\Delta F = 1/T$, занимаемой символом $S_j(t)$.

В приемном устройстве осуществляются процессы демодуляции, детектирования и декодирования принятого из канала сигнала $\tilde{S}(t)$. Если детектор вырабатывает решение о переданном блоке битов $\{b_0, b_1, \dots, b_{k-1}\}$ на основании принятого символа $\tilde{S}_j(t)$, то говорят, что детектор выносит *жесткое решение* относительно переданных символов. В случае если детектор формирует оценку принятого блока в виде квантованных значений с числом уровней квантования, превышающим позиционность модуляции M , то считается, что он вырабатывает мягкие решения.

Сигнал с выхода детектора поступает на декодер канала, который, используя внесенную кодером избыточность, осуществляет обнаружение и исправление ошибок и, таким образом, идентифицирует переданное источником сообщение.

19.1.2. Классификация помехоустойчивых кодов

Известно обширное количество различных классов помехоустойчивых кодов, отличающихся друг от друга функциональным назначением, структурой, алгоритмами кодирования и декодирования, энергетической эффективностью и множеством других свойств. В соответствии с этим существует много вариантов классифицирующих признаков помехоустойчивых кодов. Один из вариантов классификации помехоустойчивых кодов приведен на рис. 19.3. [7.5].

В соответствии с этой классификацией корректирующие коды делятся на две группы: блочные и непрерывные. Блочные коды характеризуются тем, что последовательность передаваемых символов перед кодированием разделяется на блоки. Операции кодирования и декодирования в каждом блоке производятся независимо. Это означает, что кодер для блочного кода является устройством без памяти, отображающим последовательности из k входных символов в последовательности из n выходных символов. Термин «без памяти» указывает на то, что каждый выходной блок кодера из n символов зависит только от соответствующего входного блока из k символов и не зависит от других блоков. Основными параметрами блочного кода являются размерность кода n , длина информационной части k , скорость кода rk/n и минимальное кодовое расстояние d_{\min} . Если все комбинации имеют одинаковую длину, блочный код называется равномерным, в противном случае код является неравномерным.

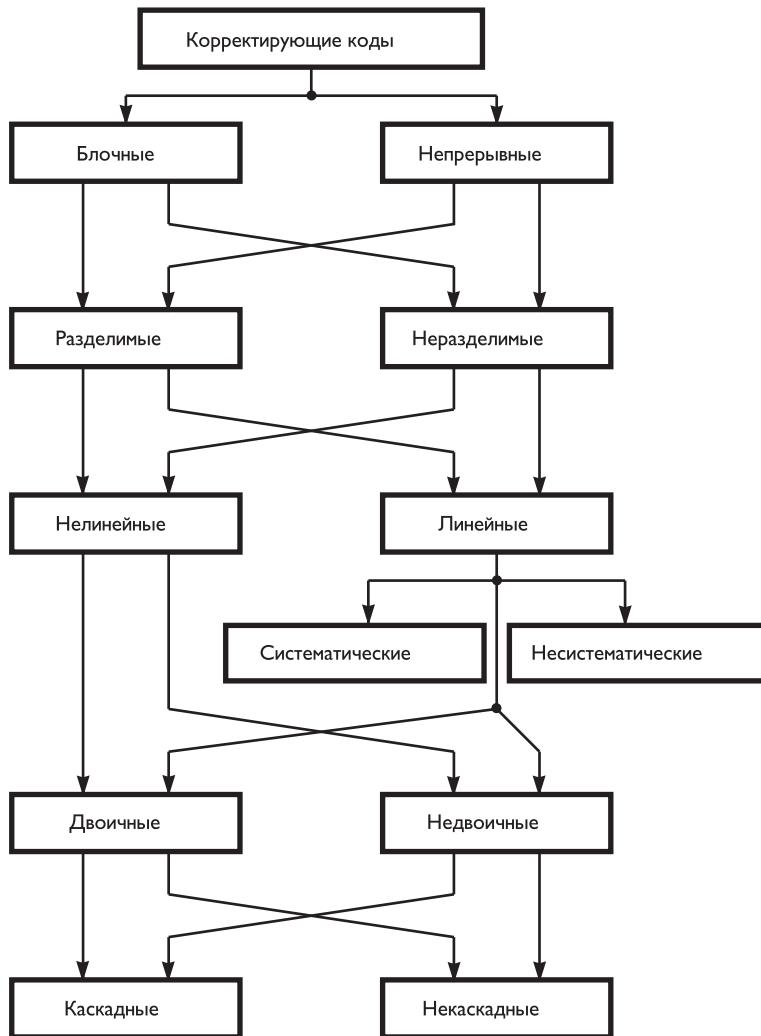


Рис. 19.3. Классификации помехоустойчивых кодов

Отличительной особенностью непрерывных кодов является то, что обработка поступающих символов производится непрерывно, без разделения на блоки. В таких кодах избыточные символы размещаются в определенном порядке между информационными символами.

Кодер для непрерывного кода является устройством с памятью, в котором из группы k входных символов формируются группы из n выходных символов. Каждая группа выходных символов непрерывного кодера зависит от текущей входной группы символов и от $K - 1$ предыдущих входных наборов. Параметр K называется конструктивной длиной непрерывного (сверточного) кода, а величину $K \cdot n$ — длиной кодового ограничения.

Корректирующие коды делятся на разделимые и неразделимые. В *разделимых кодах* можно выделить *информационные* и *проверочные* символы, которые

занимают одни и те же позиции. В *неразделимых* кодах деление на информационные и проверочные символы отсутствует.

Наиболее многочисленный класс разделимых кодов составляют *линейные* коды. Основная их особенность состоит в том, что избыточные символы образуются как линейные комбинации информационных символов.

В свою очередь, линейные коды могут быть разделены на два подкласса: *систематические* и *несистематические*. Все двоичные систематические коды являются групповыми. Это означает, что все кодовые комбинации, относящиеся к группе, обладают тем свойством, что сумма по модулю два любой пары кодов снова дает кодовую комбинацию, принадлежащую этой группе. Линейные коды, которые не могут быть отнесены к подклассу систематических, называются несистематическими. *Нелинейные коды* свойствами групповых кодов не обладают.

Наиболее известными представителями *систематических* кодов являются циклические коды и коды Хемминга.

19.2. Линейные блочные коды

Важное семейство кодов линейных блочных кодов образуют двоичные блочные коды. Особенность этих кодов заключается в том, что процессы кодирования и декодирования описываются с помощью аппарата линейной алгебры, при этом компонентами соответствующих векторов и матриц являются символы «0» и «1». Операции над двоичными компонентами производятся по стандартным правилам двоичной арифметики (табл. 19.1).

Таблица 19.1. Правила двоичной арифметики

Сложение			Умножение		
	0	1		0	1
0	0	1	0	0	0
1	1	0	1	0	1

19.2.1. Математическое описание линейных блочных кодов

С математической точки зрения на основании определения операций с двоичными символами согласно табл. 19.1 было построено поле Галуа (Galois Field) характеристики $p = 2$: $\mathbf{GF}(2)[7.6, 7.7]$. Под полем Галуа характеристики 2 подразумевается конечное поле, состоящее всего из двух элементов $\{0, 1\}$, в котором операции сложения и умножения (по модулю 2) должны удовлетворять коммутативному, ассоциативному и дистрибутивному законам.

Одно из основных свойств конечного поля состоит в том, что в нем всегда присутствует по крайней мере один примитивный элемент $\alpha \in \mathbf{GF}(p)$, степени которого порождают все ненулевые элементы поля $b_i = \alpha^i \in GF(p)$, $0 \leq i \leq p-2$. Для поля $\mathbf{GF}(2)$ примитивный элемент $\alpha = 2$, а элементы поля определяются как $\{0, \alpha^1\}$.



Рис. 19.4. Кодовое векторное пространство

Общая теория полей Галуа позволяет строить поля характеристики p порядка $m = \mathbf{GF}(p^m)$, где p — простое, m — любое конечное целое. Переход к расширенным полям $\mathbf{GF}(p^m)$ дает возможность формировать коды, обладающие по сравнению с двоичными кодами рядом новых свойств. В частности, коды Рида–Соломона с символами из $\mathbf{GF}(2^m)$, $m > 2$ находят широкое применение в телекоммуникационных системах. Подробнее вопросы, связанные с алгеброй полей Галуа, можно изучить по монографиям [7.7, 7.8].

Исходным множеством для построения кодовых конструкций служит n -мерное двоичное векторное пространство \mathbf{V}_n , в котором заданы операции арифметики по модулю 2 (табл. 19.1). В это пространство вложено k -мерное замкнутое линейное пространство \mathbf{V}_k , содержащее 2^k кодовых слов (рис. 19.4).

Блочный двоичный код длины n с 2^k разрешенными комбинациями является линейным (n, k) -кодом, если его кодовые слова образуют k -мерное подмножество \mathbf{V}_k векторного n -мерного пространства \mathbf{V}_n [7.9]. Это означает, что кодер двоичного блокового (n, k) -кода отображает множество 2^k возможных двоичных информационных слов в множество 2^k n -мерных кодовых слов. Вместо k битов информационного вектора в канал передаются n битов кодового вектора. В этом случае говорят об избыточном кодировании со скоростью $r = k/n$; чем ниже скорость, тем больше избыточность кода и его корректирующая способность.

Любое кодовое слово $\mathbf{v} \in \mathbf{V}_k$ может быть записано как линейная комбинация k элементов $(\mathbf{g}_0, \mathbf{g}_1, \dots, \mathbf{g}_{k-1})$ базиса пространства \mathbf{V}_k :

$$\mathbf{v} = c_0 \mathbf{g}_0 + c_1 \mathbf{g}_1 + \dots + c_{k-1} \mathbf{g}_{k-1}, \quad (19.4)$$

где $\mathbf{c} = (c_0, c_1, \dots, c_{k-1})$ — вектор-сообщение на входе кодера.

Уравнение (19.4) отражает процесс кодирования блочного кода \mathbf{c} и может быть записано через *порождающую матрицу* \mathbf{G} :

$$\mathbf{v} = \mathbf{c}\mathbf{G},$$

где векторы $\mathbf{g}_0, \mathbf{g}_1, \dots, \mathbf{g}_{k-1}$ образуют строки *порождающей матрицы* кода

$$G_{k \times n} = \begin{pmatrix} g_0 \\ g_1 \\ \vdots \\ g_{k-1} \end{pmatrix} = \begin{pmatrix} g_{0,0} & g_{0,1} & \dots & g_{0,n-1} \\ g_{1,0} & g_{1,1} & \dots & g_{1,n-1} \\ \vdots & \vdots & \ddots & \vdots \\ g_{k-1,0} & g_{k-1,1} & \dots & g_{k-1,n-1} \end{pmatrix}. \quad (19.5)$$

Кодовые слова $\mathbf{v} \in \mathbf{V}_k$ могут быть представлены в систематической форме $\mathbf{v} = (\mathbf{c}, \mathbf{e})$, которая образуется отдельно вектором-сообщением \mathbf{c} и проверочной частью из $tn - k$ проверочных символов. Порождающая матрица такого систематического кода имеет вид:

$$G_{k \times n} = [I_k \mid P_{k \times (n-k)}] = \begin{pmatrix} 1 & 0 & \cdots & 0 & g_{0,0} & g_{0,1} & \cdots & g_{0,m-1} \\ 0 & 1 & \cdots & 0 & g_{1,0} & g_{1,1} & \cdots & g_{1,m-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & g_{k-1,0} & g_{k-1,1} & \cdots & g_{k-1,m-1} \end{pmatrix}, \quad (19.6)$$

где I_k — единичная матрица размерности $k \times k$, предназначенная для формирования информационной части кодового слова; $P_{k \times (n-k)}$ — матрица элементов линейно независимых векторов \mathbf{g} , определяющая проверочные символы.

Для кода $\mathbf{v} \in \mathbf{V}_k$ в пространстве \mathbf{V}_n существует дуальный код \mathbf{v}_d , такой, что скалярное произведение любой пары векторов, один из которых относится к коду \mathbf{v} , а другой к \mathbf{v}_d , всегда равно нулю. Это значит, что векторы кода \mathbf{v}_d ортогональны векторам кода \mathbf{v} . Дуальное векторное подпространство образуется $n - k$ линейно независимыми базисными векторами $(\mathbf{h}_0, \mathbf{h}_1, \dots, \mathbf{h}_{n-k-1})$. Эти векторы образуют строки *проверочной матрицы*:

$$H_{(n-k) \times n} = \begin{pmatrix} h_0 \\ h_1 \\ \vdots \\ h_{n-k-1} \end{pmatrix} = \begin{pmatrix} h_{0,0} & h_{0,1} & \cdots & h_{0,n-1} \\ h_{1,0} & h_{1,1} & \cdots & h_{1,n-1} \\ \vdots & \vdots & \ddots & \vdots \\ h_{n-k-1,0} & h_{n-k-1,1} & \cdots & h_{n-k-1,n-1} \end{pmatrix}.$$

Если порождающая матрица задана в систематической форме (19.6), то систематическая форма проверочной матрицы имеет вид:

$$H = [P_{k \times (n-k)}^T \mid I_{n-k}]. \quad (19.7)$$

В силу ортогональности векторов кода \mathbf{v}_d векторам кода \mathbf{v} выполняется следующее свойство:

$$\mathbf{G} \mathbf{H}^T = 0. \quad (19.8)$$

Таким образом, для каждого кодового слова $\mathbf{v} \in \mathbf{V}_k$ в силу (19.8) справедливо соотношение

$$\mathbf{s} = \mathbf{v} \cdot \mathbf{H}^T = \mathbf{c} \cdot \mathbf{G} \cdot \mathbf{H}^T = 0. \quad (19.9)$$

Вектор \mathbf{s} называется синдромом. Любому вектору, не принадлежащему коду \mathbf{v} , будет соответствовать отличный от нуля синдром.

Уравнение (19.9) является фундаментальным для декодирования линейных кодов. Пусть кодовое слово $\mathbf{v} \in \mathbf{V}_k$, переданное по дискретному каналу с помехами, принято как $\mathbf{r} = \mathbf{v} + \mathbf{e}$, где вектор ошибок $\mathbf{e} \in \{0, 1\}^n$. Вектор ошибок \mathbf{e} создается каналом связи в процессе передачи кодового слова \mathbf{v} .

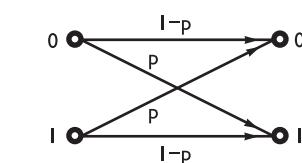


Рис. 19.5. Модель двоичного симметричного канала; p — вероятность искажения символа

Простейшей моделью канала является двоичный симметричный канал (ДСК), отражающий применение двоичной модуляции в канале связи с аддитивным белым шумом (рис. 19.5).

Тогда процедура декодирования в соответствии с (19.8) будет состоять в определении *синдрома* путем решения системы проверочных уравнений

$$\mathbf{s} = \mathbf{r} \cdot \mathbf{H}^T = (\mathbf{v} + \mathbf{e}) \cdot \mathbf{H}^T = \mathbf{e} \cdot \mathbf{H}^T. \quad (19.10)$$

Если значение синдрома $\mathbf{s} = 0$ при $\mathbf{e} = 0$, то принятая кодовая комбинация \mathbf{r} является разрешенной и, следовательно, не содержит ошибки. В случае когда значение синдрома $\mathbf{s} = 0$ при $\mathbf{e} \neq 0$, то имеется факт *необнаружимой* ошибки. Очевидно, что это произойдет только тогда, когда вектор \mathbf{e} будет принадлежать коду \mathbf{v} . В противном случае ошибка, содержащаяся в \mathbf{r} , будет обнаружена и при возможности исправлена.

Декодер может выдавать потребителю ошибочное информационное слово тогда и только тогда, когда в канале связи произошли необнаружимые ошибки или кратность канальной ошибки превышает корректирующую способность кода. Эффективность конкретного кода зависит от области его применения и от свойств канала связи. К примеру, если в канале связи с аддитивным белым гауссовским шумом отношение сигнал/шум достаточно велико, то в силу независимости ошибок в кодовом слове вероятность одиночной ошибки во много раз превышает вероятность ошибок высших кратностей. Поэтому использование в таком канале простых кодов с исправлением однократной ошибки может оказаться весьма эффективным. В каналах связи, где преобладают многократные ошибки (пакеты ошибок — в каналах с замираниями), применение кодов, исправляющих одиночные ошибки, эффективно лишь в сочетании с процедурами глубокого перемежения/деперемежения, сводящими многократные ошибки к одиночным.

19.2.2. Свойства линейных блочных кодов

Корректирующая способность кода и расстояние Хемминга

Корректирующая способность помехоустойчивого кода определяется свойствами векторного пространства и зависит от расстояния между кодовыми словами. Для наглядности на рис. 19.6 представлена геометрическая интерпретация векторного пространства. На нем показаны n -мерные кодовые слова $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$, соответствующие разрешенным кодовым комбинациям, векторы \mathbf{r} , означающие возможные принимаемые комбинации, и вектор ошибки \mathbf{e}_1 . Окружностями с центрами в точках, соответствующих изображениям $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$ на плоскости, обозначены области декодирования. Это означает, что если принятый вектор \mathbf{r} окажется внутри области декодирования кода \mathbf{v}_1 , то декодер выдаст решение о приеме \mathbf{v}_1 .

В случае двоичных равномерных кодов количественная мера отличия будет определяться числом символов, которыми отличаются комбинации одна от другой:

$$d(\mathbf{v}_i, \mathbf{v}_j) = \omega_H(\mathbf{v}_i \oplus \mathbf{v}_j) = \sum_{l=0}^{n-1} v_{i,l} \oplus v_{j,l}. \quad (19.11)$$

Расстояние, измеренное таким образом, называется *кодовым расстоянием*, или *расстоянием Хемминга*. Величину $d(\mathbf{v}_i, \mathbf{v}_j)$ можно также выразить через *вес кода* $\omega_H(\cdot)$, означающего число отличных от нуля компонент кодового вектора (в данном случае — вектора, образованного скалярной суммой векторов \mathbf{v}_i и \mathbf{v}_j). Значение $d(\mathbf{v}_i, \mathbf{v}_j)$ определяется количеством единиц в сумме этих комбинаций по модулю два и не может превышать значения длины кодовой комбинации.

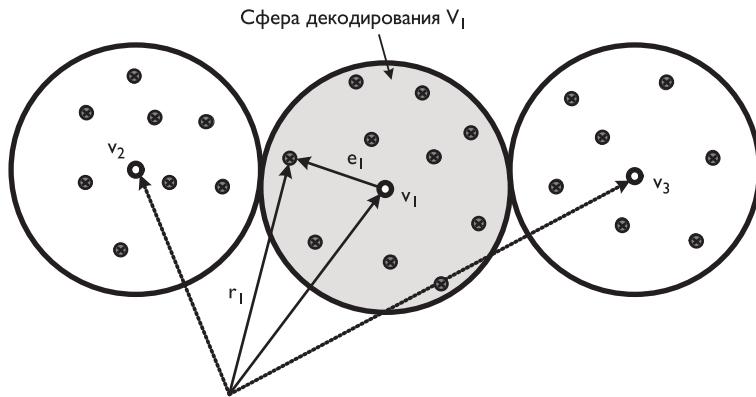


Рис. 19.6. Сфера декодирования векторного пространства

Например:

$$\begin{aligned} 110011 &\rightarrow \mathbf{v}_i \\ 011101 &\rightarrow \mathbf{v}_j \\ \hline 101110 & d(\mathbf{v}_i, \mathbf{v}_j) = 4. \end{aligned}$$

Важнейшей характеристикой помехоустойчивых кодов является минимальное расстояние Хэмминга, которое определяется как минимальное кодовое расстояние между всеми возможными комбинациями кода:

$$d_{\min} = \min\{d(\mathbf{v}_i, \mathbf{v}_j)\} \forall i, j. \quad (19.12)$$

По минимальному кодовому расстоянию можно определить кратность (число ошибок в кодовой комбинации) обнаруживаемых и исправляемых ошибок.

Корректирующая способность линейного кода определяется следующим образом: линейный двоичный (n, k) -код с минимальным расстоянием Хэмминга $d_{\min} \geq 2t + 1$ может обнаруживать ошибки кратности $d_{\min} - 1$ и исправлять t ошибок. Если код исправляет все ошибки кратности меньше t , то области декодирования представляют собой сферы радиуса t в n -мерном пространстве (рис. 19.6).

Геометрическая интерпретация кодового расстояния и граница Хемминга

Геометрической моделью n -разрядного безызбыточного двоичного кода являются вершины n -мерного куба с единичными ребрами. Так, например, для трехразрядного ($n = 3$) кода $c_1c_2c_3$ используется модель, изложенная на рис. 19.7.

Геометрическая модель позволяет интерпретировать кодовое расстояние как минимальную сумму длин ребер, соединяющих вершины куба, координаты которых представляют кодовые комбинации. Нетрудно видеть, что минимальное расстояние между вершинами куба равно 1, т. е. минимальное кодовое расстояние для безызбыточного кода равно $d_{\min} = 1$.

Геометрическая модель кода с проверкой на четность при $n = 3$ представлена на рис. 19.8, где черные точки — запрещенные комбинации, а голубые — разрешенные. Очевидно, что минимальное расстояние между разрешенными комбинациями равно двум, т. е. минимальное кодовое расстояние для этого кода $d_{\min} = 2$.

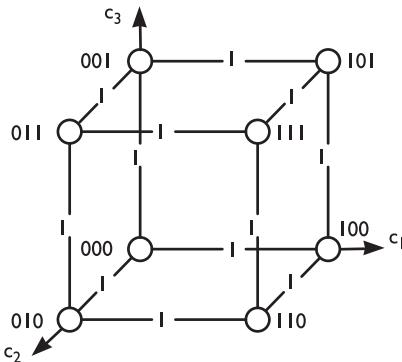


Рис. 19.7. Геометрическая модель кода с проверкой на четность при $n = 3$ и кодовом расстоянием $d_{\min} = 1$

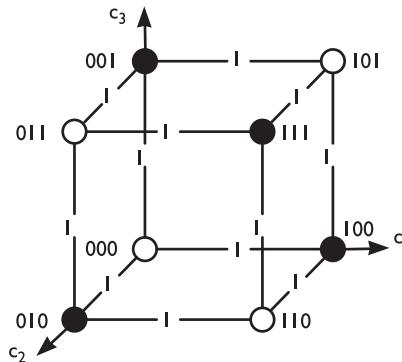


Рис. 19.8. Геометрическая модель кода с проверкой на четность при $n = 3$ и кодовом расстоянием $d_{\min} = 2$

Коды, в которых непересекающиеся сферические области декодирования охватывают все векторное пространство размерности n , называются *совершенными*, или *плотноупакованными*. При использовании совершенных кодов всегда возможна коррекция ошибок [7.6].

Соотношения между параметрами совершенных двоичных (n, k) -кодов, способных исправлять t ошибок, можно определить следующим образом: область декодирования совершенного (n, k) -кода с $d_{\min} = 2t + 1$ образуют 2^k непересекающихся сфер радиуса t в n -мерном векторном пространстве; поскольку пространство совершенных кодов является плотноупакованным, следовательно, каждая сфера содержит единственную разрешенную n -мерную кодовую комбинацию и M остальных n -мерных векторов, находящихся на расстоянии l от разрешенного кодового слова, причем $0 < l < t$.

Таким образом, каждой сфере принадлежит

$$\sum_{l=1}^t \binom{n}{l} = \binom{n}{1} + \binom{n}{2} + \dots + \binom{n}{t}$$

n -мерных векторов, где

$$\binom{n}{l} = \frac{n!}{l!(n-l)!}.$$

Так как общий объем непересекающихся сфер плотноупакованного пространства не может превышать объем n -мерного векторного пространства, то для двоичных кодов становится справедливым выражение:

$$2^n \geq 2^k \sum_{l=1}^t \binom{n}{l} \quad \text{или} \quad 2^{n-k} \geq \sum_{l=1}^t \binom{n}{l}. \quad (19.13)$$

Равенство имеет место только для совершенных двоичных кодов. Выражение (19.13) называется *границей Хэмминга*. Эта граница является нижней оценкой необходимого числа проверочных символов двоичного кода длины n , способного исправлять t ошибок.

Таким образом, в каждой сфере плотноупакованного пространства кодов помимо разрешенной комбинации содержится $M = 2^{n-k} - 1 = 2^m - 1$ векторов. Для (n, k) -кода величина $mn - k$ соответствует разрядности синдрома s и величину M

можно трактовать как количество различных отличных от нуля значений синдрома.

Вероятность ошибки декодирования

Как следует из рис. 19.6, ошибка при передаче кода не обнаруживается, если посланное кодовое слово в канале переходит в другую разрешенную кодовую комбинацию. Из свойства замкнутости векторного пространства \mathbf{V}_k относительно операции сложения (сумма любой пары векторов является вектором этого пространства) и в соответствии с соотношением (19.10) следует, что в этом случае сам вектор ошибки \mathbf{r} должен являться кодовым словом из векторного пространства \mathbf{V}_k . Следовательно, вероятность необнаружимой ошибки будет определяться суммой вероятностей независимых событий $\mathbf{r} = \mathbf{v}_i$, $1 \leq i \leq 2^k$.

Вероятность события, при котором вектор ошибки совпадет с конкретным n -мерным двоичным кодом \mathbf{v} , можно определить как $p^i \cdot (1-p)^{n-i}$, где i — число единиц в коде, p — вероятность одиночной ошибки (рис. 19.5). Таким образом, вероятность необнаружимой ошибки для ДКС без памяти будет равна:

$$p_r = \sum_{i=d_{\min}}^n A_i p^i (1-p)^{n-i}, \quad (19.14)$$

где A_i — количество кодовых комбинаций (n, k) -кода \mathbf{v} веса i (с количеством i ненулевых элементов). Не зная распределения весов A_i , вероятность необнаружимой ошибки можно оценить сверху как [7.6]:

$$p_r = p^{d_{\min}} \sum_{i=d_{\min}}^n A_i p^{i-d_{\min}} (1-p)^{n-i} < (2^k - 1) p^{d_{\min}}, \quad (19.15)$$

где $p^{i-d_{\min}} (1-p)^{n-i} < 1$, $\sum_{i=d_{\min}}^n A_i = 2^k$.

Для оценки эффективности помехоустойчивого кодирования используются разные критерии. Наибольшее распространение получил энергетический критерий, состоящий в расчете отношения сигнал/шум SNR (Signal-to-Noise Ratio) на бит передаваемой информации, обеспечивающего заданную вероятность битовой ошибки BER (Bit Error Rate) p при передаче без кодирования и отношения сигнала/шума на кодовый символ для передачи с кодированием при условии $BER=p$. После этого находится SNR на бит полезной информации с учетом скорости кода. Разность энергетических затрат на бит передаваемой информации при передаче без кодирования и с кодированием называется *энергетическим выигрышем кода* (ЭВК). Использование кодов с большой избыточностью позволяет достичь ЭВК до 6 дБ и выше при $p = 10^{-5}$ [7.6].

19.2.3. Практические примеры блочных кодов

Систематические коды

Систематические коды являются одними из наиболее распространенных линейных блочных кодов. Это разделимые коды, в которых проверочные символы определяются путем линейного преобразования информационных символов. Обозначим информационные символы c_i , $i = \overline{1, k}$, а проверочные b_j , $j = \overline{1, m}$. Тогда

любую кодовую комбинацию (n, k) -кода можно представить в систематической форме:

$$\underbrace{c_1 c_2 \dots c_k}_k \underbrace{b_1 b_2 \dots b_m}_m, \quad n=k+m$$

где $c_i \in \{0, 1\}$, $b_j \in \{0, 1\}$ могут принимать значения «0» и «1».

Процесс кодирования состоит в формировании проверочных символов из информационных по следующему правилу:

$$b_j = \sum_{i=1}^k \oplus g_{ij} C_i, \quad j = \overline{1, m}, \quad (19.16)$$

где $g_{ij} \in \{0, 1\}$ — элементы порождающей матрицы (19.5); $\sum \oplus$ — знак суммирования по модулю 2.

Пример. Рассматривается код $(7, 4)$, $n = 7$, $k = 4$, $mn - k = 3$. Для одного из вариантов этого кода коэффициенты g_{ij} имеют значения, указанные в табл. 19.2.

Таблица 19.2. Варианты кодов коэффициентов g_{ij}

j	g_{ij}			
	$i = 1$	$i = 2$	$i = 3$	$i = 4$
1	0	1	1	1
2	1	0	1	1
3	1	1	0	0

С учетом этого правила (19.16) приобретает вид:

$$\begin{aligned} b_1 &= c_2 \oplus c_3 \oplus c_4; \\ b_2 &= c_1 \oplus c_3 \oplus c_4; \\ b_3 &= c_1 \oplus c_2 \oplus c_4. \end{aligned} \quad (19.17)$$

И в связи с этим конкретная кодовая комбинация, например, $C=(1101)$, будет дополнена проверочными символами:

$$\begin{smallmatrix} 1 & 1 & 0 & 1 \\ c_1 & c_2 & c_3 & c_4 \end{smallmatrix} \rightarrow \begin{smallmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 1 \\ c_1 & c_2 & c_3 & c_4 & b_1 & b_2 & b_3 \end{smallmatrix}$$

поскольку $b_1 = 1 \oplus 0 \oplus 1 = 0$; $b_2 = 1 \oplus 0 \oplus 1 = 0$; $b_3 = 1 \oplus 1 \oplus 1 = 1$.

Декодирование линейных систематических кодов может осуществляться различными способами. Один из них — метод контрольных чисел. Он состоит в том, что из принятых информационных символов $c'_1 c'_2 \dots c'_k$ образуется новая группа проверочных символов $b''_j = \sum_{i=1}^k \oplus g_{ij} c'_i$, $j = \overline{1, m}$, которая сравнивается с принятыми проверочными символами b'_j путем суммирования по модулю 2:

$$X = \frac{b'_1 b'_2 \dots b'_m}{\overline{x_1 x_2 \dots x_m}}$$

где $b'_1 b'_2 \dots b'_m$ — принятые проверочные символы; $b''_1 b''_2 \dots b''_m$ — символы, вновь сформированные по принятым c'_j , $j = \overline{1, m}$; X — контрольное число, или синдром.

С помощью синдрома можно обнаружить или обнаружить и исправить ошибки. Если ошибок нет, то $b'_j = b''_j$ и контрольное число $X = 0$.

При наличии ошибочного символа c'_i (или b'_j) синдром $X \neq 0$. Для исправления ошибок знание того, что $X \neq 0$, недостаточно. Чтобы исправить ошибку, нужно указать номер разряда, в котором произошла ошибка. Эта задача решается по виду синдрома.

Для этого каждой подлежащей исправлению ошибке присваивается соответствующее значение синдрома, что позволяет по полученному синдрому определить положение ошибочно принятых символов. Так как синдром представляет собой комбинацию с числом разрядов, равным m , то общее количество различных отличных от нуля значений синдрома равно $M = 2^m - 1$.

Величина M не должна быть меньше числа различных сочетаний ошибок, подлежащих исправлению, поскольку каждому сочетанию должно соответствовать свое значение контрольного числа X . То есть число ненулевых синдромов должно быть больше числа исправляемых ошибок.

Предположим, код предназначен для исправления всех однократных ($t = 1$) ошибок. Число таких ошибок равно числу сочетаний $\binom{n}{i} = n$. Тогда, в соответствии с (19.13), необходимо выполнить условие

$$M = 2^m - 1 \geq n.$$

Это условие позволяет определить при заданном числе информационных символов k необходимое число проверочных символов m .

Порождающая матрица систематического кода строится исходя из заданной величины d_{\min} . Поскольку строки матрицы являются разрешенными кодовыми комбинациями, то для них должны выполняться следующие требования:

- каждая строка матрицы \mathbf{G} должна быть удалена от нулевой комбинации на расстояние, большее d_{\min} , т. к. число единиц в каждой строке должно быть как минимум d_{\min} ;
- любые две строки матрицы \mathbf{G} должны быть удалены друг от друга на кодовое расстояние, большее d_{\min} .

Коды Хэмминга

Одним из самых простых линейных блочных (n, k) -кодов являются коды Хэмминга [7.1]. Для каждого натурального $m > 3$ существует двоичный код Хэмминга со следующими параметрами:

- длина кода $n = 2^m - 1$;
- число информационных разрядов $k = 2^m - 1 - m$;
- число проверочных разрядов $m n - k$.

Коды Хэмминга являются совершенными и обладают кодовым расстоянием $d_{\min} = 3$, поэтому способны исправлять только одну или обнаружить две ошибки.

Для задания кодов Хэмминга обычно используется проверочная матрица \mathbf{H} , содержащая m строк и $2^m - 1$ столбцов, причем столбцами являются все возможные ненулевые двоичные векторы длины m . Конструкция кодов Хэмминга определяется следующими свойствами проверочной матрицы вида (19.7):

- так как минимальное расстояние кода Хэмминга $d_{\min} = 3$, все столбцы проверочной матрицы должны быть попарно различными, т. е. проверочная матрица не должна содержать одинаковых столбцов;
- из $d_{\min} = 3$ следует, что каждая строка порождающей матрицы должна содержать как минимум три единицы, так как строки порождающей матрицы, в свою очередь, являются кодовыми словами; если порождающая матрица представлена в виде (19.6), то это значит, что строки матрицы \mathbf{P} должны содержать как минимум две единицы.

Рассмотрим построение девятиразрядного кода Хэмминга $(9, 5)$, каждая комбинация которого содержит пять информационных и четыре проверочных символа. Этот код удовлетворяет неравенству (19.13): $2^4 - 1 = 15 \geq k + m = 9$. Проверочную матрицу $(9, 5)$ кода Хэмминга можно представить в виде (19.7):

$$\mathbf{H} = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \quad (19.18)$$

Соответственно, порождающая матрица такого кода получается путем объединения единичной матрицы \mathbf{I}_k и подматрицы, включающей k первых строк транспонированной матрицы \mathbf{H} :

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix}. \quad (19.19)$$

В соответствии с этим четыре контрольных символа образуются по общему правилу (19.16):

$$b_j = g_{j1}c_1 \oplus g_{j2}c_2 \oplus g_{j3}c_3 \oplus g_{j4}c_4 \oplus g_{j5}c_5.$$

Декодирование осуществляется путем вычисления четырех проверок на четность по методу контрольных чисел (см. п. 19.2.3):

$$\begin{aligned} x_1 &= b'_1 \oplus b''_1 = b'_1 \oplus \sum_{i=1}^5 g_{1i}c'_i, \\ x_2 &= b'_2 \oplus b''_2 = b'_2 \oplus \sum_{i=1}^5 g_{2i}c'_i, \\ x_3 &= b'_3 \oplus b''_3 = b'_3 \oplus \sum_{i=1}^5 g_{3i}c'_i, \\ x_4 &= b'_4 \oplus b''_4 = b'_4 \oplus \sum_{i=1}^5 g_{4i}c'_i. \end{aligned}$$

Поскольку $m = 4$, то имеется всего 16 комбинаций $Xx_1x_2x_3x_4$. Нулевая комбинация X соответствует случаю правильного приема, а остальные пятнадцать характеризуют факт появления ошибки и должны использоваться для определения ее местоположения.

Кроме описанных кодов Хэмминга существуют укороченные и расширенные коды Хэмминга. Укороченные коды Хэмминга получаются при исключении какой-либо проверки (т. е. удалении одной строки проверочной матрицы). Расширен-

ные коды Хэмминга формируются путем введения дополнительной проверки на четность всех символов кодового слова. В результате их минимальное расстояние увеличивается до $d_{\min} = 4$, что позволяет данным кодам исправлять одну и обнаруживать две или только обнаруживать три ошибки.

На рис. 19.9 представлены экспериментальные зависимости вероятности битовой ошибки P_b от отношения сигнал/шум на бит E_b/N_0 для кодов Хэмминга с $m = 3, \dots, 7$ [7.10] при работе в ДСК, декодируемых с помощью декодера максимального правдоподобия, вырабатывающего жесткие решения. Под декодером максимального правдоподобия подразумевается декодер, осуществляющий выбор из всех возможных разрешенных кодовых слов то, которое находится на минимальном расстоянии от принятого из канала связи вектора.

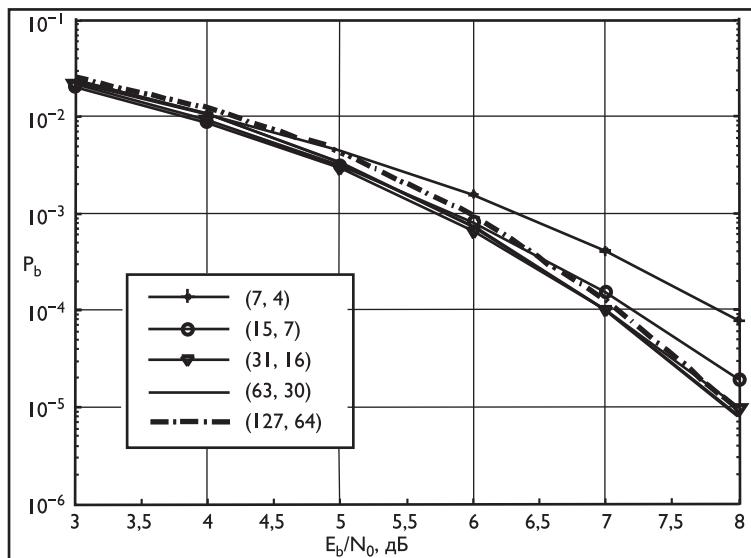


Рис. 19.9. Характеристики кодов Хэмминга для различных значений t

Как следует из этого рисунка, коды Хэмминга обладают слабой корректирующей способностью и в непосредственном виде практически не используются. Однако применение данных кодов в составе каскадных схем кодирования (например, Turbo Product Codes) позволяет получить очень хорошие результаты. Более подробное описание кодов Хэмминга представлено в [7.1].

19.3. Циклические коды

19.3.1. Понятие циклического кода

Циклические коды являются подмножеством линейных кодов. Они были получены в результате поиска наиболее простой технической реализации помехоустойчивого кодирования. Благодаря хорошим корректирующим свойствам, относительно малой избыточности, простоте схемной реализации устройств ко-

дирования и декодирования циклические коды получили относительно широкое распространение.

При описании таких кодов ограничимся только двоичными циклическими кодами, все операции с компонентами двоичных циклических кодов производятся по правилам арифметики по модулю 2. Двоичные циклические коды образуют линейные векторные пространства над полем Галуа $\mathbf{GF}(2)$. На практике широко используются циклические коды с компонентами из расширенных полей Галуа $\mathbf{GF}(2^m)$. Такими кодами являются коды Бозе–Чоудури–Хоквингема (БЧХ) и коды Рида–Соломона (РС).

Линейный (n, k) -код $\mathbf{v} \in \mathbf{V}$ является циклическим, если циклический сдвиг любого кодового слова из \mathbf{V} также принадлежит коду, т. е. циклический сдвиг разрешенной кодовой комбинации приводит к другой разрешенной кодовой комбинации. Пусть имеется n -мерная кодовая комбинация циклического кода $\mathbf{c} = (c_0, c_1, \dots, c_{n-1})$ с компонентами $c_i \in \{0, 1\}$. Циклический сдвиг реализуется с помощью регистра сдвига длины n с обратной связью (рис. 19.10) и соответствует сдвигу всех компонент вектора \mathbf{c} на один разряд вправо, причем крайний правый символ займет положение крайнего левого:

$$\mathbf{c}^{(1)} = (c_{n-1}, c_0, c_1, \dots, c_{n-2}).$$

При i -кратном циклическом сдвиге получается

$$\mathbf{c}^{(i)} = (c_{n-i}, \dots, c_{n-1}, c_0, c_1, \dots, c_{n-i-1}).$$



Рис. 19.10. Регистр сдвига с обратной связью

При описании циклических кодов n -разрядные кодовые комбинации можно представить в виде многочленов (полиномов) степени $(n - 1)$ фиктивной переменной x

$$v(x) = c_{n-1}x^{n-1} + c_{n-2}x^{n-2} + \dots + c_0x^0. \quad (19.20)$$

Например, семиразрядной кодовой комбинации $c_6c_5c_4c_3c_2c_1c_0 = 1001101$ соответствует полином

$$v(x) = 1 \cdot x^6 + 0 \cdot x^5 + 0 \cdot x^4 + 1 \cdot x^3 + 1 \cdot x^2 + 0 \cdot x^1 + 1 \cdot x^0 = x^6 + x^3 + x^2 + 1.$$

Степени переменной x при полиномиальной форме записи используются только для обозначения места соответствующей компоненты кодового вектора в регистре сдвига и никакой иной смысловой нагрузки не несут.

Над полиномами, описывающими двоичные кодовые комбинации, можно выполнять все алгебраические операции, причем коэффициенты при одинаковых

степенях складываются по модулю два (без переноса в старший разряд), а операция вычитания эквивалентна сложению. Операция умножения на полином степени m соответствует сдвигу кодовой комбинации на m разрядов влево. Проиллюстрируем это на примере. Пусть $m = 3$ и полином $v(x) = x^3 + x^2 + 1$ умножается на x^3 . Полиному $v(x) = x^3 + x^2 + 1$ соответствует комбинация 1101. Полиному $v(x) = (x^3 + x^2 + 1) \cdot x^3 = x^6 + x^5 + x^3$ — комбинация 1101000.

19.3.2. Порождающая и проверочная матрицы циклического кода

Основным свойством циклических кодов является то, что все полиномы $v(x)$, представляющие разрешенные кодовые комбинации, должны делиться без остатка на некоторый полином $G_m(x)$ минимальной степени m , который называется порождающим полиномом кода [7.6, 7.9]. Следовательно, любое произведение $G_m(x)$ на многочлены, степень которых не превышает $k - 1$, образует линейный циклический (n, k) -код. Свойство делимости комбинации циклического кода на порождающий полином положено в основу циклического кодирования и декодирования.

Как показано в [7.6], одним из основных свойств порождающего полинома $G_m(x)$ является то, что он делит бином $(x^n + 1)$ без остатка. Таким образом, чтобы найти некоторый порождающий полином, необходимо знать разложение $(x^n + 1)$ на *неприводимые* множители $\phi_j(x)$, $j = \overline{1, l}$:

$$x^n + 1 = \phi_1(x)\phi_2(x) \dots \phi_l(x). \quad (19.21)$$

Многочлен $\phi(x)$ степени m является *неприводимым* в поле $\mathbf{GF}(2)$, если он не делится ни на какой многочлен с коэффициентами из $\mathbf{GF}(2)$ степени меньшей m , но большей 0.

В соответствии с (19.21) порождающим может быть полином, образованный как каждым *неприводимым* множителем $\phi_j(x)$, $j = \overline{1, l}$, так и любой комбинацией их произведений, которые будут также являться делителями $x^n + 1$:

$$G_m(x) = \prod_{j \in \{1, 2, \dots, l\}} \phi_j(x). \quad (19.22)$$

Для компактной записи двоичного циклического кода (n, k) -кода используется *порождающий* (образующий) полином $G_m(x)$ степени $mn - k$ с коэффициентами g_i , $i = \overline{1, n - k}$ из поля Галуа $\mathbf{GF}(2)$:

$$G_m(x) = g_0 + g_1x + g_2x^2 + \dots + g_mx^m.$$

Неприводимые полиномы играют роль, сходную с простыми числами в теории чисел. Вид неприводимого полинома зависит от числа проверочных символов m . Имеются таблицы неприводимых полиномов, некоторые из них представлены в табл. 19.3.

При больших значениях n двучлен $x^n + 1$ может иметь несколько делителей степени m . В связи с этим возникает неоднозначность в выборе порождающего полинома: какой из этих делителей будет порождать наилучший код? На этот вопрос не существует однозначного ответа. Тем не менее во многих случаях целесообразно пользоваться таблицей наилучших двоичных циклических кодов, которые применяются в различных Рекомендациях ITU-T (табл. 19.4).

Таблица 19.3. Таблица некоторых неприводимых полиномов

m	$G_m(x)$	Двоичный эквивалент	Десятичный эквивалент
1	$G_1(x) = x + 1$	11	3
2	$G_2(x) = x^2 + x + 1$	111	7
3	$G_3(x) = x^3 + x + 1$ $G_3(x) = x^3 + x^2 + 1$	1011 1101	11 13
4	$G_4(x) = x^4 + x + 1$ $G_4(x) = x^4 + x^3 + 1$ $G_4(x) = x^4 + x^3 + x^2 + x + 1$	10001 11001 11111	17 25 31

В соответствии с основным свойством циклических кодов, каждый многочлен циклического кода может быть представлен в виде произведения

$$v(x) = C_k(x) \cdot G_m(x) = c_0 G_m(x) + c_1 x G_m(x) + \dots + c_{k-1} x^{k-1} G_m(x). \quad (19.23)$$

Таблица 19.4. Таблица эффективных двоичных циклических кодов ITU

Код	Порождающий полином $G_m(x)$	Примечание
CRC-4	$1 + x + x^4$	Применяется в ISDN
CRC-8	$(1+x)(1+x^2+x^3+x^4+x^5+x^6+x^7) = 1+x+x^2+x^8$	Используется в системах асинхронной передачи данных по технологии ATM
CRC-12	$(1+x)(1+x^2+x^{11}) = 1+x+x^2+x^3+x^{11}+x^{12}$	
CRC-16	$(1+x)(1+x+x^2+x^3+x^4+x^{12}+x^{13}+x^{14}+x^{15}) = 1+x^5+x^{12}+x^{16}$	Является стандартом CCITT для HDLC и LAPD
CRC-32	$1+x+x^2+x^4+x^5+x^7+x^8+x^{10}+x^{11}+x^{12}+x^{16}+x^{22}+x^{23}+x^{26}+x^{32}$	Применяется в HDLC

Учитывая произведение $x^s G_m(x)$, $s = \overline{0, k-1}$, каждое слагаемое в (19.23) представляет собой сдвиг порождающего полинома $G_m(x)$ на s элементов вправо, которому соответствует вектор

$$g = (1, g_1, g_2, \dots, g_{m-1}, 1),$$

где $g_0 = 1$ на основании соответствующего свойства полиномов минимальной степени [7.6]. В матричной форме кодовый вектор \mathbf{v} , соответствующий многочлену $V(x)$, может быть представлен в виде произведения информационного вектора \mathbf{c} на порождающую матрицу $\mathbf{G}_{k \times n}$:

$$\mathbf{v}_{1 \times n} = \mathbf{c}_{1 \times k} \oplus \mathbf{G}_{k \times n}, \quad (19.24)$$

где порождающая матрица $\mathbf{G}_{k \times n}$ имеет вид:

$$\mathbf{G}_{k \times n} = \begin{pmatrix} 1 & g_1 & g_2 & \dots & g_{m-1} & 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & g_1 & \dots & g_{m-2} & g_{m-1} & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & \dots & \dots & g_{m-1} & 1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \dots & 1 & g_1 & g_2 & \dots & \dots & \dots & g_{m-1} & 1 \end{pmatrix}. \quad (19.25)$$

Каждая строка порождающей матрицы $\mathbf{G}_{k \times n}$ имеет размерность n и является кодовым словом; в каждом столбце (19.26) присутствует циклический сдвиг вектора $g = (1, g_1, g_2, \dots, g_{m-1})$.

В общем случае матрица $\mathbf{G}_{k \times n}$ порождает несистематический циклический код. Однако заметим, что в силу линейности кодового пространства \mathbf{V}_n , замена любой строки суммой этой строки с другой, отличной от нее, не меняет кодового пространства. Это свойство позволяет привести $\mathbf{G}_{k \times n}$ к систематическому виду.

Проверочная матрица циклического кода может быть получена на основании того, что многочлен $G_m(x)$ делит $x^n + 1$ без остатка, следовательно,

$$x^n + 1 = G_m(x) \cdot H(x). \quad (19.26)$$

Введя проверочную матрицу следующего вида

$$\mathbf{H}_{(n-k) \times n} = \begin{pmatrix} h_k & h_{k-1} & h_{k-2} & \cdots & h_1 & h_0 & 0 & 0 & \cdots & 0 \\ 0 & h_k & h_{k-1} & \cdots & h_2 & h_1 & h_0 & 0 & \cdots & 0 \\ 0 & 0 & h_k & \cdots & \cdots & \cdots & h_1 & h_0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \cdots & 0 & h_k & h_{k-1} & \cdots & \cdots & h_1 & h_0 \end{pmatrix},$$

получим известное матричное уравнение (19.9) для синдромного декодирования:

$$\mathbf{v} \oplus \mathbf{H}^T = 0. \quad (19.27)$$

19.3.3. Систематическое кодирование циклическими кодами

Приведенные в табл. 19.3 порождающие полиномы образуют несистематический код. Однако путем некоторой модификации алгоритма кодирования (19.24) можно получить систематический циклический код с теми же параметрами [7.6].

Пусть имеется информационный полином степени $k - 1$ и его $mn - k$ кратный сдвиг в n -разрядном сдвиговом регистре:

$$C_k(x) = c_0 + c_1x + c_2x^2 + \dots + c_{k-1}x^{k-1}, \quad (19.28)$$

$$x^m C_k(x) = c_0x^m + c_1x^{m+1} + c_2x^{m+2} + \dots + c_{k-1}x^{n-1}. \quad (19.29)$$

Такой сдвиг не вызывает переполнения регистра сдвига и соответствует заполнению k правых крайних двоичных разрядов регистра информационным вектором $C_k(x)$. При этом m оставшихся разрядов остаются свободными. Эти разряды необходимо заполнить таким образом, чтобы вектор, содержащийся в регистре, принадлежал циклическому коду, т. е. делился без остатка на порождающий полином $G_m(x)$ минимальной степени m . Поэтому многочлен $x^m C_k(x)$ можно представить в следующем виде:

$$x^m C_k(x) = Q(x) \cdot G_m(x) + R(x), \quad (19.30)$$

где $R(x)$ — остаток от деления вектора $x^m C_k(x)$, содержащегося в регистре сдвига, на $G_m(x)$.

В (19.30) и далее операция сложения производится по правилам арифметики по модулю 2, при которой операция вычитания эквивалентна операции сложения. Из этого соотношения также следует

$$x^m C_k(x) + R(x) = Q(x) \cdot G_m(x) = v(x), \quad (19.31)$$

а из (19.31) вытекает следующий алгоритм кодирования систематического циклического (n, k) -кода:

- подлежащая кодированию k -разрядная кодовая комбинация описывается полиномом $C_k(x)$ степени $k - 1$;
- полином $C_k(x)$ умножается на x^m , что эквивалентно сдвигу k -разрядной комбинации влево (в сторону старших разрядов) на m разрядов или добавлению m нулей справа;
- полученный полином $x^m C_k(x)$ делится на порождающий полином $G_m(x)$, имеющий степень, равную числу проверочных символов m , в результате чего получается целая часть $Q(x)$ и остаток от деления $R(x)$

$$\frac{C_k(x)x^m}{G_m(x)} = Q(x) + \frac{R(x)}{G_m(x)}; \quad (19.32)$$

- формируется n -разрядная разрешенная комбинация систематического циклического кода, соответствующая полиному $v(x)x^m C_k(x) + R(x)$, для чего в свободные разряды регистра записывается комбинация, соответствующая остатку от деления $R(x)$.

Можно показать, что комбинации, соответствующие полиному $v(x)$, являются разрешенными. С этой целью умножим обе части уравнения (19.32) на $G_m(x)$:

$$C_k(x)x^m = Q(x)G_m(x) + R(x).$$

Вычитая из обеих частей $R(x)$ и учитывая, что операции вычитания и суммирования по модулю два эквивалентны, получим:

$$x^m C_k(x) + R(x) = Q(x)G_m(x).$$

Выражение в левой части этого равенства — это кодовое слово $v(x)$. Следовательно, $\frac{v(x)}{G_m(x)} = Q(x)$ — целая часть, т. е. полученный полином $v(x)$ делится на $G_m(x)$ без остатка, а значит, представляет собой разрешенную комбинацию циклического кода.

Пример. Построим кодовую комбинацию циклического кода $(7, 4)$, соответствующую комбинации 1001. Общая длина комбинации $n=7$, число информационных символов $k = 4$, проверочных $mn - k = 3$. Из таблицы неприводимых полиномов табл. 19.3 для $m = 3$ выберем образующий полином $G(x)x^3 + x + 1$. В соответствии с рассмотренным выше алгоритмом кодирования получаем:

$$1. 1001 \rightarrow C_4(x) = x^3 + 1.$$

$$2. x^3 C_4(x) = x^6 + x^3.$$

3. Вычисление остатка от деления $x^3 C_4(x)$ на порождающий полином $G_m(x)$:

$$\begin{array}{c} \oplus \quad x^6 + x^3 \\ \underline{x^6 + x^4 + x^3} \\ \oplus \quad x^4 \\ \underline{x^4 + x^2 + x} \\ \underbrace{x^2 + x}_{R(x)} \end{array} \quad \left| \begin{array}{c} x^3 + x + 1 \\ \hline x^3 + x \end{array} \right.$$

$$\Rightarrow (101)_2$$

4. Формирование разрешенной комбинации циклического кода:

$$v_7(x) = x^3 C_4(x) + R(x) = x^6 + x^3 + x^2 + x \rightarrow 1001110.$$

19.3.4. Обнаружение ошибок

Процедура обнаружения ошибки сводится к делению принятой комбинации на порождающий полином $G_m(x)$ и анализу остатка от деления $R(x)$. Поскольку разрешенные кодовые комбинации делятся на образующий полином без остатка, то значение $R(x) = 0$ означает либо факт отсутствия ошибок, либо случай, когда кратность ошибки выше обнаруживающей способности циклического кода. Если $R(x) \neq 0$, то произошел прием с ошибкой (обнаруживаемой). Таким образом, остаток от деления $R(x)$ играет роль синдрома в коде Хэмминга.

Пример процесса декодирования

Пусть полученная комбинация 1001110 совпадает с переданной, т. е. искажений нет. Известно, что комбинация образована с помощью порождающего полинома $G_3(x)x^3 + x^2 + 1$. В результате деления полинома, описывающего эту комбинацию на порождающий полином $G_3(x)$ получается:

$$\begin{array}{r} \oplus \quad x^6 + x^3 + x^2 + x \\ \underline{x^6 + x^4 + x^3} \\ \oplus \quad x^4 + x^2 + x \\ \underline{x^4 + x^2 + x} \\ 0 \end{array} \quad \left| \begin{array}{c} x^3 + x + 1 \\ x^3 + x \end{array} \right. \Rightarrow R(x) = 0.$$

Пусть в результате приема искажен второй символ кода, т. е. принята комбинация 1101110. Тогда в результате расчета $R(x)$ фиксируется ошибка:

$$\begin{array}{r} \oplus \quad x^6 + x^5 + x^3 + x^2 + x \\ \underline{x^6 + x^4 + x^3} \\ \oplus \quad x^5 + x^4 + x^2 + x \\ \underline{x^5 + x^3 + x^2} \\ \oplus \quad x^4 + x^3 + x \\ \underline{x^4 + x^2 + x} \\ \oplus \quad x^3 + x^2 \\ \underline{x^3 + x + 1} \\ x^2 + x + x \end{array} \quad \left| \begin{array}{c} x^3 + x + 1 \\ x^3 + x^2 + x + 1 \end{array} \right. \Rightarrow R(x) \neq 0.$$

19.3.5. Исправление однократных ошибок

Определение места ошибки в принятой комбинации циклического кода основано на анализе остатка $R(x)$ от деления полинома $v'(x)$ принятой комбинации на порождающий полином $G_m(x)$. Принятую с ошибкой в i -м разряде n -разрядную кодовую комбинацию можно представить полиномом:

$$v'(x) = v(x) + x^{n-i},$$

где $i = \overline{1, n}$ — номер искаженного разряда, отсчитываемый от старшего; $v(x)$ — полином комбинации CRC-кода, принятой без ошибки; x^{n-i} — полином однократной ошибки (ошибку можно представить как n -разрядную комбинацию с «1» в i -м разряде; например, при $i = 2$ и $n = 5$ ошибка представляется комбинацией 01000, которой соответствует полином x^3).

После деления $v'(x)$ на $G_m(x)$ получим

$$\frac{v'(x)}{G_m(x)} = \frac{v(x)}{G_m(x)} + \frac{x^{n-i}}{G_m(x)}.$$

Согласно свойству циклического кода $v(x)$ делится на $G_m(x)$ без остатка. Следовательно, остаток от деления $v'(x)$ равен остатку от деления x^{n-i} , т. е. не зависит от вида $v(x)$, а определяется только положением принятого с ошибкой символа. Это позволяет определить место ошибки путем последовательных сдвигов принимаемой комбинации в сторону старшего разряда и сравнения остатков $R_j(x)$, $j = \overline{0, n-1}$ (j — число сдвигов) от деления сдвигаемой комбинации с остатком $R'_0(x)$ (выделенным синдромом) от деления на $G_m(x)$ полинома x^{n-1} , описывающего ошибку в старшем разряде.

Согласно одному из методов исправления однократных ошибок при совпадении остатков $R_j(x) = R'_0(x)$ на j -м шаге принимается решение об искажении символа C_{j+1} , номер которого на единицу больше числа сдвигов j .

Рассмотрим работу этого алгоритма на примере.

Пусть комбинация циклического кода $(7, 4)$ 1001110 , сформированная с помощью образующего полинома $G(x)x^3 + x + 1$ принята с ошибкой в третьем разряде: $10\underline{1}1110$. Для исправления ошибки определяется выделенный синдром $R'_0(x)$ путем выполнения деления полинома ошибки в старшем разряде x^6 на порождающий полином $G(x)$ в двоичном эквиваленте:

$$\begin{array}{c} \oplus \quad 100000 \quad | \quad 1011 \\ \underline{1011} \quad \quad \quad | \quad 1011 \\ \oplus \quad 1100 \quad \quad \quad | \quad \\ \underline{1011} \quad \quad \quad | \quad \\ \oplus \quad 1110 \quad \quad \quad | \quad \\ \underline{1011} \quad \quad \quad | \quad \\ 101 \quad \Rightarrow \quad R'_0(x). \end{array}$$

Для определения искаженного символа вычисляются остатки $R_j(x)$ от деления принятой кодовой комбинации на двоичный эквивалент порождающего полинома $G(x)$:

$$\begin{array}{c} \oplus \quad 101\underline{1}110 \quad | \quad 1011 \\ \underline{1011} \quad \quad \quad | \quad 10001 \\ \oplus \quad \underline{110} | 0 \Rightarrow R_0(x) \neq R'_0(x) \\ \oplus \quad \underline{101} | 1 \\ \underline{111} | 0 \Rightarrow R_1(x) \neq R'_0(x) \\ \oplus \quad \underline{101} | 1 \\ 101 \quad \Rightarrow R_2(x) = R'_0(x). \end{array}$$

19.3.6. Общий принцип декодирования циклических кодов

Рассмотрим модель передачи данных по каналу связи с ошибками (рис. 19.11). При передаче по каналу связи с ошибками к кодовому слову $v(x)$ добавляется многочлен ошибок $e(x)$.



Рис. 19.11. Модель передачи данных по каналу связи с ошибками

В результате многочлен принятого кодового слова имеет вид:

$$r(x) = v(x) + e(x) \quad (19.33)$$

или в соответствии с (19.31):

$$r(x) = Q(x)G_m(x) + s(x), \quad (19.34)$$

где $s(x)$ является синдромом, представляющим собой остаток от деления $r(x)$ на порождающий полином $G_m(x)$:

$$s(x) = r(x) \bmod (G_m(x)) = e(x) \bmod (G_m(x)). \quad (19.35)$$

Если $r(x)$ является кодовым словом, то $s(x)$ — нулевой многочлен. На рис. 19.12 представлена обобщенная структура декодера циклического кода.



Рис. 19.12. Обобщенная структура декодера циклического кода

Синдром $s(x)$ используется для определения полинома ошибок $e(x)$. Проблема декодирования в этом случае состоит в поиске полинома ошибок $e(x)$ по известному синдрому $s(x)$ согласно уравнению (19.35). При наличии многократных ошибок задача декодирования будет рассматриваться уже как решение системы уравнений вида (19.35). Эффективная реализация алгоритмов декодирования циклических кодов, позволяющих обнаруживать и исправлять многократные ошибки, достигается при использовании арифметики конечного поля — поля Галуа.

19.3.7. Арифметика полей Галуа

Возможность построения хороших циклических кодов определяется умением находить порождающие полиномы. В настоящее время существует ограниченное количество методов конструирования порождающих полиномов циклических кодов с предсказуемой исправляющей способностью и приемлемой скоростью кодирования. Наиболее известный из методов конструирования порождающих полиномов основан на теории полей Галуа.

Как известно, в обычной алгебре операции сложения/вычитания и умножения/деления могут дать результат, который выходит за пределы некоторого за-

данного диапазона. В конечном поле любые операции с любыми элементами этого поля в результате дают элемент, принадлежащий этому же полю. Соответственно, в конечных полях вообще отсутствуют такие проблемы, как переполнение при умножении и потеря точности при делении, делая тем самым алгебру \mathbf{GF} наименее естественной с точки зрения компьютерных вычислений, которые имеют конечную разрядность. Конечное поле — это некоторая ограниченная совокупность чисел, над которыми определены четыре арифметические операции, результат выполнения которых не выходит за пределы этой совокупности.

Пример. Рассмотрим построение поля Галуа $\mathbf{GF}(2^8)$, состоящее из $q = 2^8 = 256$ чисел от 0 до $2^8 - 1$, упорядоченных особым образом.

Согласно теории Галуа, любой элемент произвольного поля можно представить в виде i -й степени одного (или нескольких) примитивного элемента α : $b_i = \alpha^i \in GF(p)$, $0 \leq i \leq p - 2$. В поле $\mathbf{GF}(2^8)$ имеется единственный примитивный элемент, в качестве которого выступает простое число 2. Следовательно, любой ненулевой элемент поля b_i может быть получен возведением 2 в степень i , $0 \leq i \leq 2^8 - 2$. Нетрудно заметить, что, начиная с $i = 8$, получается результат, который выходит за пределы множества $[0, 2^8 - 1]$. В поле $\mathbf{GF}(2^8)$ эта ситуация может быть решена особым правилом создания поля.

Согласно теории конечных полей, в любом поле должны быть определены элементы 0, $b_0 = \alpha^0 = 1$ и $b_n = \alpha^n = 1$, где $np^m - 1$, для $\mathbf{GF}(2^8)$: $b_{255} = \alpha^{255} = 1$. Остальные элементы поля $\mathbf{GF}(2^8)$ определяются следующим образом:

$$b_i = \begin{cases} \alpha \cdot \alpha^{i-1}, & \alpha^{i-1} < 2^{m-1} = 128, \\ \alpha \cdot \alpha^{i-1} \bmod \xi, & \alpha^{i-1} \geq 2^{m-1} = 128, \end{cases}$$

где ξ — некоторое число, которое в теории Галуа задается в полиномиальной форме.

Для примера, поле $\mathbf{GF}(2^8)$ может быть порождено полиномом $f(x)x^8 + x^4 + x^3 + x^2 + 1$, который используется в коде Рида–Соломона (204, 188) на физическом уровне телевизионного стандарта DVB-T. Этот полином соответствует в двоичной форме числу $\xi = 0011101B$ и в десятичной — $\xi = 285$.

Согласно теории [7.7, 7.8], любой порождающий поле полином $f(x)$ должен быть неприводимым. Таким образом, конечное поле вида $\mathbf{GF}(2^m)$ может быть построено заданием некоторого двоичного неприводимого полинома $f(x)$ порядка m , который по смыслу соответствует роли порождающего полинома $G_m(x)$ циклических кодов.

Элементы теории полей Галуа

Поле — это множество элементов A , в котором определены операции сложения, вычитания, умножения и деления. Сложение и умножение должны удовлетворять коммутативному, ассоциативному и дистрибутивному законам: для любых элементов α, β, γ . В таком поле должны выполняться соотношения:

- коммутативность: $\alpha + \beta = \beta + \alpha$; $\alpha \cdot \beta = \beta \cdot \alpha$;
- ассоциативность: $\alpha + (\beta + \gamma) = \alpha + \beta + \gamma$; $\alpha \cdot (\beta \cdot \gamma) = (\alpha \cdot \beta) \cdot \gamma$;
- дистрибутивность: $\alpha \cdot (\beta + \gamma) = \alpha \cdot \beta + \alpha \cdot \gamma$;
- замкнутость: $\alpha + \beta \in A$,

а также должны существовать элементы 0, 1, $-\alpha$, обратный к $\alpha \neq 0$ элемент α^{-1} , такие, что:

$$\begin{array}{lll} \alpha + 0 = \alpha; & -\alpha + \alpha = 0; & 0 \cdot \alpha = 0; \\ 1 \cdot \alpha = \alpha; & \text{если } \alpha \neq 0, & \text{то } \alpha^{-1} \cdot \alpha = 1. \end{array}$$

Число элементов конечного поля называется *порядком* поля. Согласно теории [7.8], число элементов q конечного поля всегда удовлетворяет условию

$$q = p^m \quad (19.36)$$

где p — простое число, называемое *характеристикой* поля; $m = 1, 2, \dots$. Другими словами, если число элементов q некоторого множества не удовлетворяет условию (19.36), то для этого множества невозможно определить операции сложения и умножения, удовлетворяющие аксиомам поля. Так, например, невозможно образовать поле с числом элементов, равным 6, 10, 12, 14 и т. д., но можно построить поле с числом элементов, равным 2, 3, 4, 5, 7, 8, 9, 11 и т. д.

Наиболее просто операции сложения и умножения выполняются в поле с числом элементов, равным p ($m = 1$). В этом случае поле называется *простым*; операции сложения и умножения в нем определены по модулю p ($\bmod p$), а сами элементы образуют последовательность чисел $\{0, 1, 2, \dots, p-1\}$. Простейшим *простым* полем Галуа является двоичное поле $\mathbf{GF}(2)$, сложение и умножение в котором выполняются по правилам арифметики «по модулю 2». Можно заметить, что правила арифметики табл. 19.1 удовлетворяют перечисленным выше правилам, с учетом того что обратным элементом к «1» по сложению и умножению является сам элемент «1».

Сложнее обстоит дело с построением *расширенных* полей Галуа $\mathbf{GF}(p^m)$, где $m > 1$. Так как теория помехоустойчивого кодирования основывается, как правило, на полях характеристики 2, то ниже будут рассматриваться основные методы построения полей $\mathbf{GF}(2^m)$. Каждый элемент такого поля можно представить в виде слова длины m над $\mathbf{GF}(2)$ или многочлена с двоичными коэффициентами, степень которого меньше чем m . К примеру, любой элемент $\alpha \in \mathbf{GF}(2^3)$ можно записать как двоичное слово или как многочлен $a_2x^2 + a_1x + a_0$, $\{a_2, a_1, a_0\} \in \{0, 1\}$, где формальная переменная x является указателем позиции соответствующего коэффициента. В этом случае, если сложение элементов из $\mathbf{GF}(2^m)$ будет выполняться по правилу сложения представляющих их многочленов в поле $\mathbf{GF}(2)$, а умножение элементов поля определить как умножение представляющих эти элементы многочленов по модулю некоторого заданного неприводимого многочлена над $\mathbf{GF}(2)$ степени m , то тем самым будет построено поле Галуа $\mathbf{GF}(2^m)$.

Если рассмотреть степени некоторого ненулевого элемента $\alpha \in \mathbf{GF}(q)$ вида $\alpha^1, \alpha^2, \dots, \alpha^l, \dots$, то нетрудно заметить, что все они являются элементами поля $\mathbf{GF}(q)$ и в силу конечности поля для некоторых l и s ($l \neq s$) будет справедливо равенство $\alpha^l = \alpha^s$, а значит, $\alpha^{l-s} = 1$. Минимальное натуральное число d_α , для которого $\alpha^{d_\alpha} = 1$, является мультипликативным порядком элемента α .

В теории конечных полей доказывается, что мультипликативный порядок любого ненулевого элемента α поля $\mathbf{GF}(q)$ делит значение $q - 1$. Например, в поле $\mathbf{GF}(2^3)$ число ненулевых элементов поля является простым: $q - 1 = 7$, следовательно, делителями $q - 1$ могут быть только 1 и 7. В свою очередь, это означает, что все ненулевые элементы, кроме единичного, имеют максимальный мультипликативный порядок, равный 7.

Элемент α поля $\mathbf{GF}(q)$, имеющий максимальный мультипликативный порядок $d_\alpha q - 1$, называется *примитивным элементом поля*.

В любом конечном поле $\mathbf{GF}(q)$ всегда существует хотя бы один примитивный элемент α . Особенностью такого элемента является то, что все его $q - 1$ последовательных степеней $\alpha^1 = \alpha; \alpha^2, \dots, \alpha^{q-2}, \alpha^{q-1} = 1$ различны и соответствуют всем ненулевым элементам поля $\mathbf{GF}(q)$. Таким образом, в поле $\mathbf{GF}(2^3)$ все ненулевые элементы поля, за исключением единицы, примитивны, поскольку число ненулевых элементов является простым и, следовательно, мультипликативный порядок для этих элементов максимальен и равен 7.

Расширенное поле, построенное как множество полиномов по модулю некоторого неприводимого полинома $f(x)$, всегда содержит в качестве элемента поля полином x . Если окажется, что x — примитивный элемент, то соответствующий неприводимый полином $f(x)$ называется *примитивным полиномом*. Примитивный элемент α , степени которого образуют конечное поле, является корнем неприводимого полинома $f(x)$:

$$f(\alpha) = 0 \quad (19.37)$$

и, как будет показано ниже, удовлетворяет условию $\alpha^{q-1} = 1$.

Примитивные полиномы произвольной степени m определены над любым конечным полем. Они наиболее удобны для построения расширенных полей, поскольку позволяют реализовать простой вариант таблицы умножения элементов поля. Действительно, любые ненулевые элементы a и b расширенного поля $\mathbf{GF}(2^m)$ могут быть выражены как некоторые l -я и s -я степени примитивного элемента α : $a = \alpha^l, b = \alpha^s$. Тогда $a \cdot b = \alpha^{l+s}$ и, значит, таблица умножения двух элементов поля a и b представляет собой не что иное, как перечисление всех ненулевых степеней примитивного элемента α .

Построение расширенного поля

Построение расширенного поля $\mathbf{GF}(p^m)$ в виде степеней примитивного элемента является достаточно простой процедурой и предполагает следующий алгоритм действий.

На первом этапе выбирается порождающий поле примитивный полином степени m над простым полем $\mathbf{GF}(p)$: $f(x)x^m + f_{m-1}x^{m-1} + \dots + f_1x + f_0$. Тогда m -я степень x по модулю $f(x)$ определится как

$$x^m = -f_{m-1}x^{m-1} - f_{m-2}x^{m-2} - \dots - f_1x - f_0.$$

Предположение о примитивности x позволяет использовать для его обозначения α , в результате чего имеем равенство

$$\alpha^m = -f_{m-1}\alpha^{m-1} - f_{m-2}\alpha^{m-2} - \dots - f_1\alpha - f_0.$$

Отсюда

$$\alpha^{m+1} = \alpha \cdot \alpha^m = -f_{m-1}\alpha^m - f_{m-2}\alpha^{m-1} - \dots - f_0\alpha. \quad (19.38)$$

Если основным является поле с характеристикой $p = 2$, то в силу свойств двоичной арифметики поля $\mathbf{GF}(2)\alpha = -\alpha$, и выражение (19.38) можно записать в виде

$$\alpha^{m+1} = \alpha \cdot \alpha^m f_{m-1}\alpha^m + f_{m-2}\alpha^{m-1} + \dots + f_0\alpha, f_i \in \mathbf{GF}(2). \quad (19.39)$$

Последовательное возвведение в степень α позволяет, таким образом, определить все ненулевые элементы расширенного поля $\mathbf{GF}(p^m)$ в виде линейной комбинации первых m степеней α : $\alpha^0, \alpha^1, \dots, \alpha^{m-1}$ с коэффициентами из основного поля $\mathbf{GF}(p)$.

Рассмотрим пример построения поля $\mathbf{GF}(2^4)$ на основе примитивного полинома $f(\alpha) = \alpha^4 + \alpha + 1$, при этом на основании (19.37) справедливо выражение $\alpha^4 = \alpha + 1$.

Тогда последовательное возвведение в степень примитивного элемента α дает все ненулевые элементы поля $\mathbf{GF}(2^4)$. Результаты вычислений представлены в табл. 19.5.

Некоторые свойства расширенных конечных полей

Если $\mathbf{GF}(q)$ — конечное поле характеристики 2, то для любых элементов $\alpha, \beta \in \mathbf{GF}(q)$, $q = 2^m$, будет выполняться соотношение $(\alpha + \beta)^2 = \alpha^2 + \beta^2$.

Элементы поля $\mathbf{GF}(2^m)$ вида $\alpha^2, \alpha^{2^2}, \dots, \alpha^{2^l}$ называются *2-сопряженными* по степени двойки с элементом α . Вследствие конечности поля $\mathbf{GF}(2^m)$ последовательность, составленная из α и сопряженных с ним элементов, имеет ограниченный набор отличающихся друг от друга элементов. Так, если все элементы вида $\alpha^2, \alpha^{2^2}, \dots, \alpha^{2^{l-1}}$ различны, а продолжение последовательности приводит к повторению какого-либо уже содержащегося в ней элемента, то $\alpha^{2^l} = \alpha$. Длина l любой последовательности (цикла) сопряженных по степени 2 элементов поля $\mathbf{GF}(2^m)$ всегда делит степень расширения m .

Степени сопряженных элементов вида $\alpha^i, \alpha^{2i}, \dots, \alpha^{2^{l-1}i}$ образуют непересекающиеся циклические множества — циклотомические классы $C_i = \{i, 2i, 4i, \dots, 2^{l-1}i\}$. К примеру, в поле $\mathbf{GF}(2^4)$ можно выделить следующие циклические множества

$$C_1 = \{1, 2, 4, 8\} : \quad \alpha^1, \alpha^2, \alpha^4, \alpha^8, \alpha^{16} = \alpha^1, \alpha^{17} = \alpha^{17-15} = \alpha^2 \text{ и т. д.};$$

$$C_3 = \{3, 6, 12, 9\} : \quad \alpha^3, \alpha^6, \alpha^{12}, \alpha^{24} = \alpha^{24-15} = \alpha^9, \alpha^{18} = \alpha^{18-15} = \alpha^3 \text{ и т. д.};$$

$$C_5 = \{5, 10\} : \quad \alpha^5, \alpha^{10}, \alpha^{20} = \alpha^{20-15} = \alpha^5 \text{ и т. д.};$$

$$C_7 = \{7, 14, 13, 11\} : \quad \alpha^7, \alpha^{14}, \alpha^{28} = \alpha^{28-15} = \alpha^{13}, \alpha^{26} = \alpha^{26-15} = \alpha^{11} \text{ и т. д.}$$

Обозначение C_s выбрано таким образом, что s указывает на наименьшую степень соответствующего элемента поля, при этом в составе каждого множества содержится l_s элементов. Индексы $\{s\}$ называются *представителями классов* по модулю $(q - 1)$.

Одним из постулатов высшей алгебры является положение, что любой многочлен $f(x)$ степени m с действительными или комплексными коэффициентами всегда имеет ровно m действительных или комплексных корней x_1, x_2, \dots, x_m , что означает возможность его представления в виде

$$f(x) = \prod_{i=1}^m (x - x_i).$$

Аналогичное свойство имеет место и в случае произвольных полиномов над конечными полями.

Если α^i является корнем многочлена $\phi(x)$ над полем $\mathbf{GF}(2)$, то и все сопряженные с ним по степени два элементы $\alpha^{i \cdot 2^j}$, $j = 1, 2, \dots, l_s$ также являются

Таблица 19.5. Пример построения поля $\mathbf{GF}(2^4)$

Элемент поля	Элемент поля в векторной форме	Двоичный символ			
		a_3	a_2	a_1	a_0
0	0	0	0	0	0
α^0	1	0	0	0	1
α^1	α	0	0	1	0
α^2	α^2	0	1	0	0
α^3	α^3	1	0	0	0
α^4	$\alpha^4 = \alpha + 1$	0	0	1	1
α^5	$\alpha^5 = \alpha \cdot \alpha^4 = \alpha(\alpha + 1) = \alpha^2 + \alpha$	0	1	1	0
α^6	$\alpha^6 = \alpha \cdot \alpha^5 = \alpha(\alpha^2 + \alpha) = \alpha^3 + \alpha^2$	1	1	0	0
α^7	$\alpha^7 = \alpha \cdot \alpha^6 = \alpha(\alpha^3 + \alpha^2) = \alpha^4 + \alpha^3 = \alpha^3 + \alpha + 1$	1	0	1	0
α^8	$\alpha^8 = \alpha \cdot \alpha^7 = \alpha(\alpha^3 + \alpha + 1) = \alpha^4 + \alpha^2 + \alpha = \alpha + 1 + \alpha^2 + \alpha = \alpha^2 + 1$	0	1	0	1
α^9	$\alpha^9 = \alpha \cdot \alpha^8 = \alpha(\alpha^2 + 1) = \alpha^3 + \alpha$	1	0	1	0
α^{10}	$\alpha^{10} = \alpha \cdot \alpha^9 = \alpha(\alpha^3 + \alpha) = \alpha^4 + \alpha^2 = \alpha^2 + \alpha + 1$	0	1	1	1
α^{11}	$\alpha^{11} = \alpha \cdot \alpha^{10} = \alpha(\alpha^2 + \alpha + 1) = \alpha^3 + \alpha^2 + \alpha$	1	1	1	0
α^{12}	$\alpha^{12} = \alpha \cdot \alpha^{11} = \alpha(\alpha^3 + \alpha^2 + \alpha) = \alpha^4 + \alpha^3 + \alpha^2 = \alpha^3 + \alpha^2 + \alpha + 1$	1	1	1	1
α^{13}	$\alpha^{13} = \alpha \cdot \alpha^{12} = \alpha(\alpha^3 + \alpha^2 + \alpha + 1) = \alpha^4 + \alpha^3 + \alpha^2 + \alpha = \alpha^3 + \alpha^2 + 1$	1	1	0	1
α^{14}	$\alpha^{14} = \alpha \cdot \alpha^{13} = \alpha(\alpha^3 + \alpha^2 + 1) = \alpha^4 + \alpha^3 + \alpha = \alpha^3 + 1$	1	0	0	1
α^{15}	$\alpha^{15} = \alpha \cdot \alpha^{14} = \alpha(\alpha^3 + 1) = \alpha^4 + \alpha = \alpha + 1 + \alpha = 1 = \alpha^0$	0	0	0	1
α^{16}	$\alpha^{16} = \alpha \cdot \alpha^{15} = \alpha = \alpha^1$	0	0	1	0
...	...				

корнями полинома $\phi(x)$. Если α^i является некоторым ненулевым элементом поля $\mathbf{GF}(2^m)$, тогда неприводимый полином $\phi(x)$ наименьшей степени над $\mathbf{GF}(2)$, для которого $\phi(\alpha^i) = 0$, называется *минимальным многочленом* α^i над $\mathbf{GF}(2)$. В [7.7] поясняется, что минимальные многочлены для элементов α^i и $\alpha^{p \cdot i}$ равны

$$\phi_i(x) = \phi_{p \cdot i}(x). \quad (19.40)$$

Для минимального многочлена $\phi(x)$ элемента α^i также справедливо следующее выражение

$$\phi_s(x) = \prod_{j=s}^{l_s} (x - \alpha^j), \quad (19.41)$$

где l_s — длина цикла сопряженных с α^i по степени 2 элементов.

Таким образом, минимальный многочлен элемента $\alpha^i \in \mathbf{GF}(q)$ может быть представлен в виде

$$\phi_s(x) = (x - \alpha^i)(x - \alpha^{2i}) \dots (x - \alpha^{i2^{l_s-1}}).$$

В соответствии с теоремой Ферма для конечных полей [7.7], каждый элемент β поля $\mathbf{GF}(q)$, qp^m , удовлетворяет уравнению $\beta^q - \beta = 0$, или, что то же самое, является корнем уравнения $x^q - x$. В силу этой теоремы справедливо выражение

$$x^q - x = x^{q-1} - 1 = \prod_{\beta \in GF(q)}^{(x-\beta)}. \quad (19.42)$$

Из уравнений (19.41) и (19.42) следует, что

$$x^{q-1} - 1 = \prod_{j \in C_s}^{(x-\alpha^j)}, \quad (19.43)$$

где s принимает все значения из множества представителей классов по модулю $(q-1)$.

Выражение (19.42) иллюстрирует разложение бинома $x^{q-1} - 1$ на неприводимые сомножители в поле $\mathbf{GF}(q)$ и, согласно (19.22), определяет способ нахождения порождающих полиномов циклических кодов. Этот способ состоит в следующем:

- поиск всех циклотомических классов по модулю $(2^m - 1)$;
- для каждого циклотомического класса C_s вычислить минимальный полином $\phi_s(x)$ [7.7]; найденный многочлен $\phi_s(x)$, согласно (19.22), может быть применен в качестве порождающего полинома циклического кода.

В качестве примера можно решить задачу определения минимального многочлена $\phi_s(x)$ для поля $\mathbf{GF}(2^4)$. Используя результаты рассмотренного выше примера, определим минимальный полином для элемента α^5 (циклотомический класс C_5). Тогда на основании (19.41)

$$\begin{aligned} \phi_5(x) &= (x + \alpha^5)(x + \alpha^{10}) = x^2 + x(\alpha^5 + \alpha^{10}) + \alpha^{15} = \\ &= x^2 + x(\underbrace{\alpha^2 + \alpha}_{\alpha^5} + \underbrace{\alpha^2 + \alpha + 1}_{\alpha^{10}}) + 1 = x^2 + x + 1, \end{aligned}$$

где арифметические операции сложения и вычитания эквивалентны, а сумма элементов поля характеристики 2 одинаковых степеней равна 0 в силу свойств двоичной арифметики.

Можно получить выражения для остальных минимальных полиномов поля $\mathbf{GF}(2^4)$:

$$\begin{aligned} \phi_0(x) &= x + 1; \\ \phi_1(x) &= x^4 + x + 1; \\ \phi_3(x) &= x^4 + x^3 + x^2 + x + 1; \\ \phi_7(x) &= x^4 + x^3 + 1. \end{aligned}$$

19.3.8. Циклические коды Хэмминга

Циклические коды Хэмминга образует важное семейство циклических (n, k) -кодов. Свойства кодов Хэмминга как подмножества линейных кодов были подробно рассмотрены выше. При определении методов построения циклических кодов Хэмминга используется понятие примитивного многочлена.

Неприводимый многочлен $p(x)$ степени t называется *примитивным*, если наименьшая степень n , при которой многочлен $x^n + 1$ делится на $p(x)$ без остатка, равна $n = 2^m - 1$.

В табл. 19.5 приведены некоторые примитивные многочлены с коэффициентами из $\mathbf{GF}(2)$. Примитивные многочлены играют важную роль в технике передачи сообщений. Например, примитивный многочлен степени $n = 23$ используется



в устройствах перемешивания символов в сетях ISDN и xDSL. Примитивные многочлены являются также основой для построения порождающих многочленов псевдослучайных последовательностей, с помощью которых производится адресная передача сообщений в системах мобильной связи.

Любой циклический код Хэмминга длины $n = 2^m - 1$ с $m > 3$ может быть построен с помощью некоторого примитивного многочлена $p(x)$ степени m . Используя в качестве порождающего полинома $p(x)$, можно строить порождающие и проверочные матрицы, а также схемы декодеров любых циклических кодов Хэмминга (табл. 19.6).

Таблица 19.6. Порождающие и проверочные матрицы циклических кодов Хэмминга

m	$p(x)$	m	$p(x)$
3	$1 + x + x^3$	11	$1 + x^4 + x^6 + x^{12}$
4	$1 + x + x^4$	12	$1 + x^3 + x^4 + x^{13}$
5	$1 + x^2 + x^5$	13	$1 + x^2 + x^6 + x^{10} + x^{14}$
6	$1 + x + x^6$	14	$1 + x + x^{15}$
7	$1 + x^3 + x^7$	15	$1 + x + x^2 + x^3 + x^4 + x^{12} + x^{13} + x^{14} + x^{15}$
8	$1 + x^2 + x^3 + x^4 + x^8$	23	$1 + x + x^{23}$
9	$1 + x^4 + x^9$	32	$1 + x + x^2 + x^3 + x^4 + x^5 + x^7 + x^8 + x^{10}$
10	$1 + x^3 + x^{10}$		$+ x^{11} + x^{12} + x^{16} + x^{22} + x^{23} + x^{26} + x^{32}$
11	$1 + x^2 + x^{11}$		

19.3.9. CRC-коды

Примером использования семейства циклических кодов является контроль ошибок с помощью циклического избыточного кода, т. е. CRC-кода (Cyclic Redundancy Check), называемого также кодом Абрамсона. При передаче данных в пакетных режимах эти коды используются для определения целостности блоков данных (Frame Checking Sequence — FCS). Примером систем с FCS являются стандарты передачи данных X.25 (HDSL), ISDN, DECT и LAN. Коды CRC представляют собой расширения циклических кодов Хэмминга.

Пусть $p(x)$ — примитивный многочлен степени m , тогда порождающий многочлен CRC-кода $G_m(x)$ можно записать в виде произведения:

$$G_m(x) = (1 + x)p(x). \quad (19.44)$$

С помощью порождающего многочлена $G_m(x)$ может быть построен циклический CRC (n, k) -код с параметрами $n = 2^m - 1$, $k = 2^m - m - 2$, имеющий $m + 1$ проверочных символов и $d_{\min} = 4$. CRC-коды обладают следующими важными свойствами:

- все ошибки кратности 3 или меньше обнаруживаются;
- все ошибки нечетной кратности обнаруживаются;
- все пакеты ошибок длины $m + 1$ или меньше обнаруживаются;
- доля необнаружимых пакетов ошибок длины $m + 2$ составляет 2^{-m} ;
- доля необнаружимых пакетов ошибок длины, большей $m + 2$, составляет $2^{-(m-1)}$.

На практике часто используются укороченные CRC-коды. В табл. 19.4 приведены наиболее употребляемые порождающие многочлены CRC-кодов, а также указаны области их применения.

19.3.10. Двоичные БЧХ-коды.

Коды БЧХ представляют собой класс линейных циклических кодов, исправляющих кратные ошибки. Они являются обобщением ранее описанных кодов Хэмминга. Сокращение БЧХ связано с именами Боуза Р.К., Рей-Чоудхури Д.К. и Хоквингема А., одновременно описавшими в 1959–60 гг. эти коды.

Циклический код длины n над полем $\mathbf{GF}(q)$ называется кодом БЧХ с конструктивным расстоянием δ , если для некоторого целого числа $b \geq 0$ его порождающий многочлен равен

$$G(x) = \text{НОК}\{\phi_b(x), \phi_{b+1}(x), \phi_{b+2}(x), \dots, \phi_{b+\delta-2}(x)\}, \quad (19.45)$$

т. е. $G(x)$ — полином над полем $\mathbf{GF}(q)$ наименьшей степени такой, что элементы поля последовательных степеней $\alpha^b, \alpha^{b+1}, \alpha^{b+2}, \dots, \alpha^{b+\delta-2}$ являются его корнями. Это значит, что вектор $C(x)$ будет кодовым вектором БЧХ только тогда, когда

$$C(\alpha^b) = C(\alpha^{b+1}) = \dots = C(\alpha^{b+\delta-2}) = 0.$$

Двоичные БЧХ-коды строятся на основе расширенного поля $\mathbf{GF}(2^m)$. Если порождающий полином $G(x)$ имеет в расширенном поле $\mathbf{GF}(2^m)$ корни $\alpha, \alpha^2, \dots, \alpha^s$, где α — примитивный элемент поля $\mathbf{GF}(2^m)$, тогда циклический код с порождающим полиномом $G(x)$ обладает минимальным расстоянием d_{\min} , не меньшим $s+1$.

Таким образом, двоичный БЧХ-код с кодовым расстоянием $d_{\min} \geq 2t+1 = s+1$ является циклическим кодом, если его порождающий полином имеет $2t$ последовательных корней для элементов полей $\alpha^b, \alpha^{b+1}, \alpha^{b+2}, \dots, \alpha^{b+2t-1}$. Коды при $b=1$ называются кодами БЧХ в узком смысле, при $n=2^m-1$ коды называются *примитивными*. Размерность информационной части кода равна

$$k = n - \deg(G(x)).$$

Общий алгоритм построения БЧХ-кодов состоит в следующем: для значения m , обеспечивающего необходимую длину кода $n = 2^m-1$, из таблиц примитивных полиномов выбирается полином степени m и строится расширенное поле $\mathbf{GF}(2^m)$. В построенном поле выбирается последовательность $\alpha, \alpha^2, \dots, \alpha^s$ степеней примитивного элемента α , определяются все циклотомические классы элементов последовательности и для каждого циклотомического класса C_i вычисляется минимальный полином $\phi_i(x)$ [7.7].

Если α выбран в качестве корня порождающего полинома $G(x)$, то, в соответствии с (19.41), корнями будут и все сопряженные с α по степени 2 элементы $\alpha^2, \alpha^4, \dots$, которые вместе с тем являются и корнями соответствующего минимального многочлена

$$\phi_1(x) = (x - \alpha)(x - \alpha^2) \dots (x - \alpha^{2l_1-1}),$$

где l_1 — размер циклотомического класса элемента α , который вследствие примитивности α равен l_1 .

Следующим элементом, претендующим на роль корня $G(x)$, является α^2 , который уже учтен в сомножителе $\phi_1(x)$, поскольку сопряжен с α по степени 2.

Если минимальное расстояние конструируемого кода больше 2 ($s > 2$), то следующим корнем порождающего полинома будет α^3 , а также все 2-сопряженные с ним элементы $(\alpha^3)^2, (\alpha^3)^4, \dots$, которые являются корнями минимального для α^3 полинома

$$\phi_3(x) = (x - \alpha^3)(x - (\alpha^3)^2) \dots (x - (\alpha^3)^{2^{l_3}-1}),$$

где l_3 — длина цикла 2-сопряженных с α^3 элементов.

Данная операция построения порождающего полинома $G(x)$ продолжается до полного исчерпывания множества $\alpha, \alpha^2, \dots, \alpha^s$, т. е. до тех пор, пока каждый из них не войдет в число корней какого-нибудь минимального многочлена. В итоге порождающий полином $G(x)$ определяется как произведение всех получаемых минимальных многочленов

$$G(x) = \phi_1(z)\phi_3(z) \dots$$

Построенный таким образом циклический код может исправлять более $t = d_{\min} - 1/2 = s/2$ ошибок, в связи с этим величина $d^* = 2t + 1 = s + 1$ носит название *конструктивного расстояния кода*, поскольку истинное минимальное расстояние кода d_{\min} может быть больше конструктивного.

В общем случае двоичный БЧХ-код характеризуется следующими параметрами:

$$n = 2^m - 1, \quad d \geq 2t + 1, \quad k \geq n - mt.$$

Таблица 19.7. Элементы поля БЧХ-кодов

Элемент поля	Элемент поля в векторной форме	Двоичный символ		
		a_2	a_1	a_0
0	0	0	0	0
α^0	1	0	0	1
α^1	α	0	1	0
α^2	α^2	1	0	0
α^3	$\alpha^3 = \alpha + 1$	0	1	1
α^4	$\alpha^4 = \alpha \cdot \alpha^3 = \alpha(\alpha + 1) = \alpha^2 + \alpha$	1	1	0
α^5	$\alpha^5 = \alpha \cdot \alpha^4 = \alpha(\alpha^2 + \alpha) = \alpha^2 + \alpha + 1$	1	1	1
α^6	$\alpha^6 = \alpha \cdot \alpha^5 = \alpha(\alpha^2 + \alpha + 1) = \alpha^2 + 1$	1	0	1
α^7	$\alpha^7 = \alpha \cdot \alpha^6 = \alpha(\alpha^2 + 1) = \alpha^3 + \alpha = \alpha + 1 + \alpha = 1$	0	0	1
α^8	$\alpha^8 = \alpha \cdot \alpha^7 = \alpha$	0	0	1
...	...			

В качестве примера рассмотрим построение порождающего полинома $G(x)$ БЧХ-кода длины $n = 2^3 - 1$, исправляющего любую однократную ошибку ($t = 1$, $s = 2t = 2$). Расширенное поле $\mathbf{GF}(2^3)$, необходимое для нахождения $G(x)$, приведено в табл. 19.7. Поскольку $s = 2$, то корнями порождающего многочлена должны быть элементы α и α^2 .

Минимальный многочлен, отвечающий элементу α , имеет вид:

$g_1(x) = (x + \alpha)(x + \alpha^2)(x + \alpha^4) = (x^3 + (\alpha + \alpha^2 + \alpha^4)x^2 + (\alpha\alpha^2 + \alpha\alpha^4 + \alpha^2\alpha^4)x + \alpha\alpha^2\alpha^4)$,
где $\alpha\alpha^2\alpha^4 = \alpha^7 = 1$; $\alpha + \alpha^2 + \alpha^4 = \alpha + \alpha^2 + \alpha + \alpha^2 = 0$;

$$\alpha\alpha^2 + \alpha\alpha^4 + \alpha^2\alpha^4 = \alpha^3 + \alpha^5 + \alpha^6 = \alpha + 1 + \alpha^2 + \alpha + 1 + \alpha^2 + 1 = 1.$$

И поэтому $\phi_1(x) = x^3 + x + 1$. Элемент α^2 , являющийся корнем $G(x)$, уже учтен в $\phi_1(x)$, так что $G(x) = \phi_1(x) = x^3 + x + 1$.

Поскольку $\deg(G(x)) = 3$, число информационных символов k , содержащихся в каждом слове, будет $k = n - \deg(G(x)) = 4$, что соответствует циклическому коду Хэмминга $(n, k) = (7, 4)$, исправляющему любую однократную ошибку.

В настоящее время разработчикам телекоммуникационных систем нет необходимости осуществлять весь объем работы по поиску БЧХ-кода, поскольку подобные исследования были выполнены много лет назад, результатом чего явились детальные таблицы, содержащие готовые к использованию порождающие полиномы БЧХ-кодов [7.1, 7.7, 7.8].

Асимптотические оценки зависимости вероятности ошибки в кодовом блоке P_B , q -ичном символе P_S и бите P_b при декодировании кодов БЧХ по максимуму правдоподобия имеют вид [7.1]:

$$\begin{aligned} P_B &= \sum_{i=t+1}^n C_n^t P_q^i (1 - P_q)^{n-i}, \\ P_S &\leq \frac{1}{n} \sum_{i=t+1}^n (i+t) C_n^i P_q^i (1 - P_q)^{n-i}, \\ P_b &\approx \frac{q/2}{q-1} P_S. \end{aligned} \quad (19.46)$$

Для некоторых (n, k) -кодов БЧХ с $rk/n \approx 1/2$ на рис. 19.13 представлены зависимости вероятности ошибки на символ P_b от отношения сигнал/шум на бит E_b/N_0 , на рис. 19.14 показаны аналогичные зависимости для кодовых скоростей r , близких к $1/3$ и $2/3$. Как следует из сопоставления представленных графиков, энергетическая эффективность кодов БЧХ с уменьшением кодовой скорости понижается.

Как видно из этих рисунков, коды БЧХ обладают существенно лучшей эффективностью по сравнению с кодами Хэмминга, однако эти характеристики все еще очень далеки от предельных значений. Кроме того, сложность декодирования данных кодов при большом значении n очень велика. Поэтому коды БЧХ небольшой длины, так же как и коды Хэмминга, в основном применяются в качестве составляющих элементов более эффективных каскадных кодов.

19.3.11. Коды Рида–Соломона

Коды Рида–Соломона находят чрезвычайно широкое распространение в системах связи, хранения данных и компьютерных технологиях. В двоичном варианте эти коды характеризуются относительно невысокой эффективностью по отношению к независимым битовым ошибкам, однако являются чрезвычайно мощными как по отношению к пакетам ошибок, так и к обнаружению ошибок, находящихся на расстоянии, превышающем исправляющую способность кода. Это обуславливает широкое использование кодов Рида–Соломона в беспроводных системах связи с многолучевым распространением, в устройствах хранения информации, подверженных механическим повреждениям (CD, DVD и т. п.).

Коды Рида–Соломона относятся к недвоичным кодам БЧХ, значения кодовых слов которых взяты из поля $\mathbf{GF}(2^m)$. Если при построении порождающего

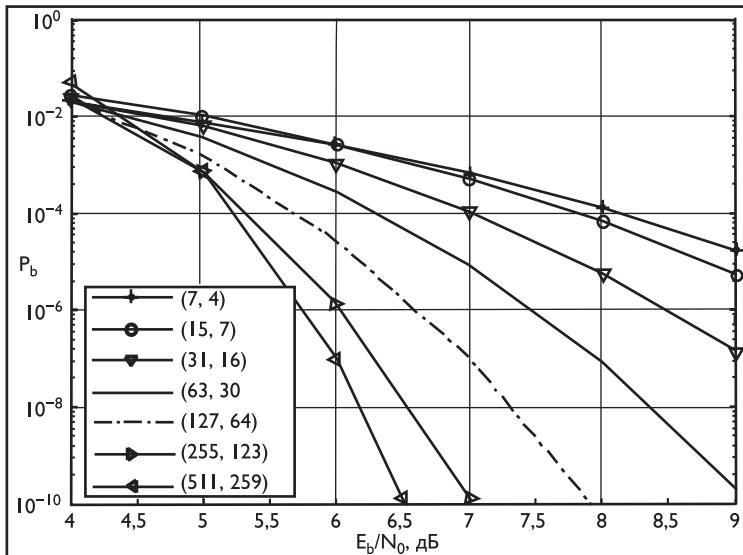


Рис. 19.13. Оценка характеристик различных кодов БЧХ в ДСК для кодовых скоростей, близких к $1/2$

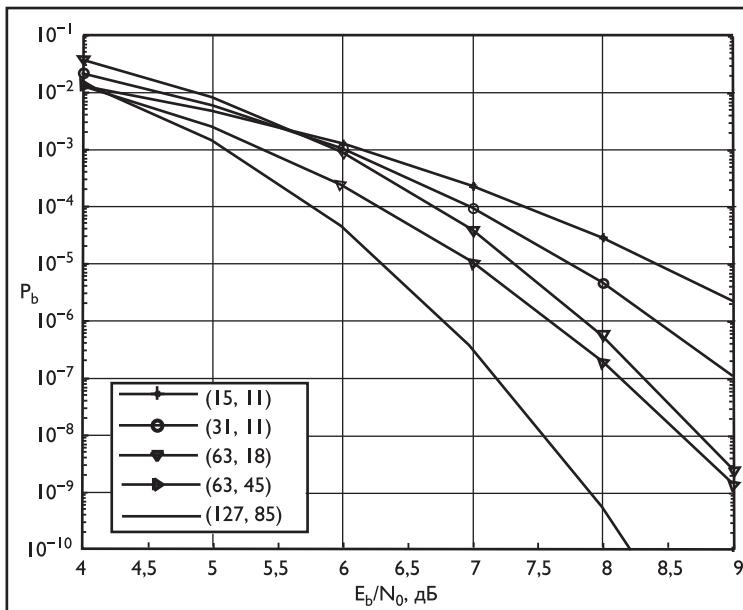


Рис. 19.14. Оценка характеристик различных кодов БЧХ в ДСК для кодовых скоростей, близких к $1/3$ и $2/3$

полинома составлять его как произведение биномов вида $x - \alpha^i$, $i = 1, 2, \dots, s$, то получающийся полином окажется недвоичным. Вследствие этого и код, порождаемый им, также окажется недвоичным, поскольку символы кода будут принадлежать расширенному полю $\mathbf{GF}(q)$, $q = 2^m$.

Таким образом, q -ичный ($q = 2^m$) код с порождающим полиномом вида

$$G(x) = \prod_{i=1}^s (x - \alpha^i), \quad (19.47)$$

называется кодом Рида–Соломона, или просто РС-кодом, и обладает кодовым расстоянием, равным $d_{\min} \geq s + 1$. Длина указанного кода составляет $nq - 1 = 2^m - 1$ q -ичных символов, а поскольку степень порождающего полинома равна s , то число информационных q -ичных символов будет равно $kn - s$, и следовательно, $d_{\min} \geq n - k + 1$.

Коды Рида–Соломона являются *оптимальными* с точки зрения величины кодового расстояния d_{\min} среди всех q -ичных кодов с аналогичными значениями длины и скорости кодирования.

Очевидно, что любой q -ичный ($q = 2^m$) символ может быть представлен в виде двоичного блока размерности m . Тогда в результате получаем двоичный код длины $n_b m \cdot n_t \cdot (2^m - 1)$ с числом информационных двоичных символов $k_b m \cdot k$ и скоростью $Rk_b/n_b k/n$. Этот код чрезвычайно эффективен в борьбе с так называемыми «пакетами ошибок»: если в результате ошибки будут искажены $t = (d - 1)/2$ последовательных 2^m -ичных символов, то будет повреждено в t раз большее число двоичных разрядов. Однако благодаря своей исправляющей способности РС-код может исправить любую ошибку кратности до t включительно, а значит, его двоичный эквивалент способен исправлять любой пакет ошибок длиной до $t \cdot t$ двоичных символов.

Для декодирования кодов Рида–Соломона существуют достаточно эффективные алгоритмы декодирования жестких решений, что позволяет использовать относительно длинные коды во многих практических приложениях, особенно в системах с q -ичной модуляцией.

Так как коды Рида–Соломона являются одной из разновидностей кодов БЧХ, то для оценки их эффективности можно воспользоваться формулами (19.46). Для примера на рис. 19.15 представлены зависимости оценки вероятности символической ошибки P_s от отношения сигнал/шум E_b/N_0 на информационный битов для различных значений q и кодовых скоростей r .

19.4. Сверточные коды

Сверточное (convolutional) кодирование можно пояснить, рассматривая действие кодирующего устройства [7.10, 7.11]. Структурная схема одного из вариантов построения такого устройства представлена на рис. 19.16. Оно состоит из $(M - 1)$ -разрядного сдвигового регистра, N сумматоров по модулю два и коммутатора K .

Кодируемые биты источника b_i в дискретные моменты времени $i = 1, 2, \dots$ последовательно поступают на вход кодера. Последние $M - 1$ битов источника запоминаются в регистре сдвига. Вместе с поступившим в дискретный момент времени i битом b_i они подаются на N сумматоров по модулю два. При этом в каждый дискретный момент времени i блок из M бит источника линейно преобразуется в N выходных двоичных кодовых символов сверточного кода $u_i^1, u_i^2, \dots, u_i^N$, которые с помощью коммутатора K занимают временной ин-

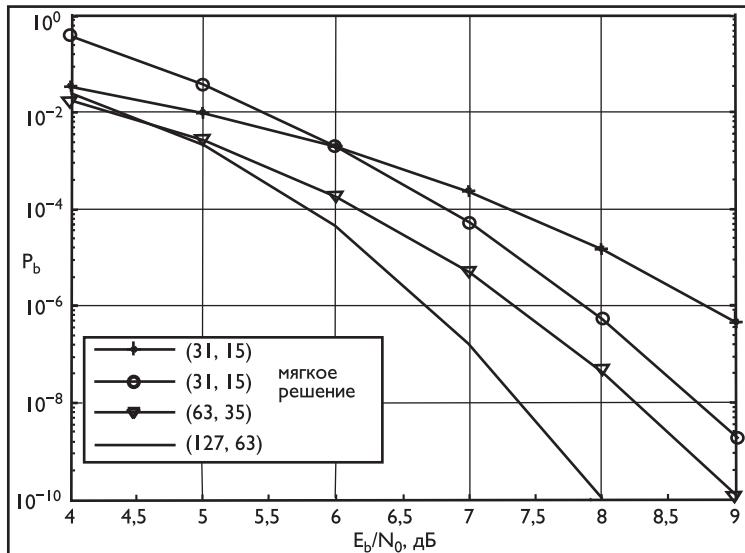


Рис. 19.15. Оценка характеристик кодов Рида-Соломона

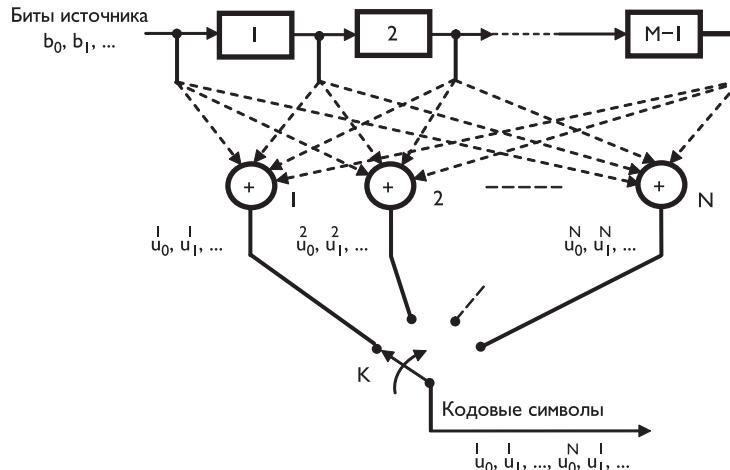


Рис. 19.16. Структурная схема сверточного кодера

тервал, равный длительности одного бита источника b_i . Опрашивая поочередно выходы N сумматоров в течение битового интервала τ_b , коммутатор преобразует параллельное представление кодовых символов в последовательное. С приходом очередного бита содержимое регистра сдвигается на одну ячейку вправо. Происходит частичное обновление кодируемого блока — в него водится поступивший от источника бит с одновременным исключением самого «старого» бита предыдущего блока. В результате вновь формируется блок из M бит источника, запаздывающий на один бит относительно предыдущего и содержащий $M - 1$

прежних битов и один новый. Этот блок кодируется по тому же правилу, что и предыдущий. Подобные шаги непрерывно повторяются один за другим с вовлечением каждый раз одного нового бита взамен последнего бита предыдущего блока. В результате на выходе кодера формируется поток символов сверточного кода: $u_0^1, u_0^2, \dots, u_0^N, u_1^1, u_1^2, \dots, u_1^N, \dots$ На каждый новый бит источника, обработанный по схеме рис. 19.16, кодер откликается N кодовыми символами, поэтому скорость кода в битах на кодовый символ равна $R = 1/N$.

Рассмотренная процедура кодирования для случая $M = 4, N = 2$ проиллюстрирована на рис. 19.17. Кодер как бы просматривает битовый поток источника сквозь скользящее окно ширины в M битовых интервалов, и кодирует все биты, попадающие в это окно на текущем шаге, в N кодовых символов. После каждого шага окно смещается вправо на один бит источника, и выполняется следующий шаг. Размер окна M или число бит источника, влияющих на кодовые символы на одном шаге, называется *конструктивным* или *кодовым ограничением*.

Использование пунктирных линий при обозначении связей между ячейками сдвигового регистра и сумматорами по модулю два на схеме рис. 19.16 означает возможность присутствия или отсутствия этих связей в конкретной схеме кодера и определяется видом образующих (порождающих, производящих) полиномов

$$g_n(x) = g_0^n + g_1^n x + \dots + g_j^n x^j + \dots + g_{M-1}^n x^{M-1}, \quad n = 1, 2, \dots, N, \quad (19.48)$$

где x — формальная переменная.

Значения коэффициентов $g_j^n \in \{0, 1\}$ полинома указывают на наличие или отсутствие связи между j -м разрядом регистра и n -м сумматором: $g_j^n = 1$ — определяет наличие связи, $g_j^n = 0$ — отсутствие. Выбор связей между сумматорами и разрядами регистра определяет правило кодирования и влияет на эффективность кода. Задача выбора связей, дающая оптимальные корректирующие свойства сверточного кода, сложна и в общем случае не решается. Однако для кодового ограничения $M < 20$ найдены образующие полиномы, обеспечивающие хорошую помехоустойчивость [7.11, 7.12–7.14].

Для примера на рис. 19.18 приведена схема сверточного кодера с длиной кодового ограничения $M = 3$, скоростью кодирования $R = 1/2$ и образующими полиномами $g_1(x)x^2 + x + 1$, $g_2(x)x^2 + 1$. Входные биты b_0, b_1, \dots преобразуются в два потока кодовых символов u_0^1, u_0^2, \dots и u_1^1, u_1^2, \dots , которые затем мультиплексируются так, чтобы u_i^1 и u_i^2 занимали соответственно четные и нечетные позиции в общем кодовом потоке.

Например, битовый поток $\{b_i\} = 10100100\dots$ порождает последовательности $\{u_i^1\} = 11011111\dots$ и $\{u_i^2\} = 10001101\dots$, которые после мультиплексирования образуют кодовый поток $\{u_i^1, u_i^2\} = 1110001011111011\dots$

Для краткости описания кодирующего устройства вместо двоичных полиномов обычно указываются их коэффициенты, которые объединяют в двоичное слово и представляют в восьмеричной системе счисления. Например, для кодера на рис. 19.18 коэффициенты образующих полиномов $g_1(x)$, $g_2(x)$ составляют двоичные числа $(111)_2$ и $(101)_2$, что соответствует паре восьмеричных чисел $(7, 5)_8$.

19.4.1. Систематические сверточные коды

Формируемые структурой по схеме на рис. 19.16 коды являются несистематическими — в них отсутствуют информационные символы. При $g_1(x) = 1$ сверточный

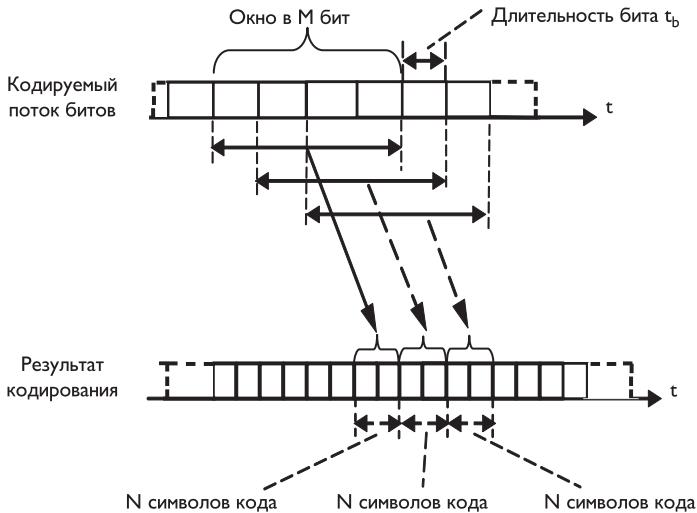


Рис. 19.17. Временные диаграммы, иллюстрирующие работу сверточного кодера при $M = 4, N = 2$

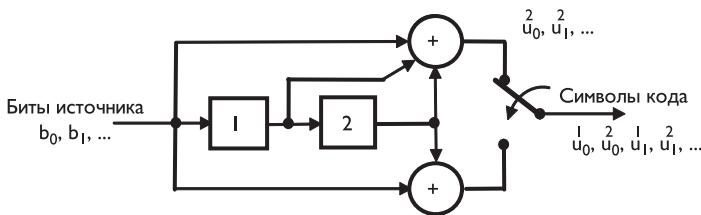


Рис. 19.18. Сверточный кодер со скоростью $R = 1/2$ на основе образующих полиномов $g_1(x)x^2 + x + 1, g_2(x)x^2 + 1$

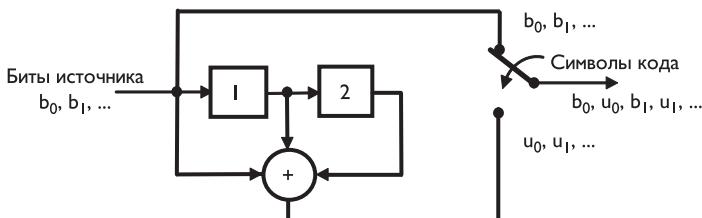


Рис. 19.19. Схема кодера систематического нерекурсивного сверточного кода со скоростью кодирования $R = 1/2$

код становится *систематическим*. В нем биты источника явно присутствуют на определенных позициях, чередуясь с проверочными. Для получения систематических сверточных кодов можно построить как нерекурсивные так и рекурсивные кодеры. Для примера на рис. 19.19 представлена схема кодера систематического нерекурсивного сверточного кода.

Однако в рамках структуры рис. 19.16 систематические коды, как правило, оказываются не лучшими в отношении корректирующей способности [7.15].

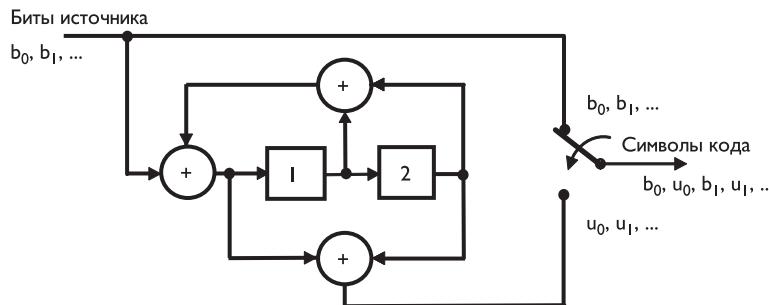


Рис. 19.20. Схема систематического рекурсивного сверточного кодера с порождающими полиномами $g_1(x)x^2 + x + 1$ и $g_2(x)x^2 + 1$

Поэтому в обстоятельствах, когда систематичность кода критически важна, сверточный кодер модифицируют в структуру с обратной связью [7.16]. Такие кодеры называются *рекурсивными*.

Для перевода несистематического сверточного кода со скоростью $R = 1/2$ в систематическую форму вычисляется остаток $r(x)b(x)/g_1(x)$ от деления полинома $b(x)$, описывающего кодируемую последовательность, на производящий полином $g_1(x)$. Определяются полиномы $u_1(x)b(x)$ и $u_2(x)r(x)/g_2(x)$, описывающие систематический выход и проверочные символы. Остаток $r(x)$ можно получить с использованием регистра сдвига с обратными связями, поэтому полином $g_1(x)$ называется полиномом обратной связи, а полином $g_2(x)$ — выходным полиномом. Формируемые таким образом коды называются рекурсивными сверточными кодами RSC (Recursive Systematic Convolutional). Пример RSC-кодера, полученного из кодера на рис. 19.18, показан на рис. 19.20.

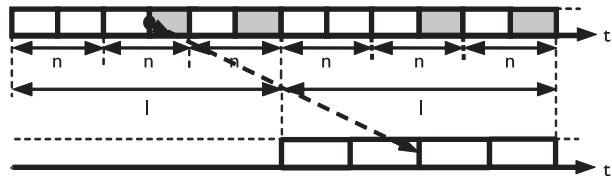
19.4.2. Управление скоростью кодирования

При использовании сверточных кодов со скоростью $R = 1/N$ наибольшая кодовая скорость равна $1/2$. Во многих практических случаях желательно работать на более высоких скоростях (например, $2/3$ или $3/4$). Одним из путей повышения кодовой скорости является переход к скоростям RS/N , где N и $S < N$ — натуральные числа. Такие скорости могут быть получены, если скользящее окно сдвигать каждый раз не на один, а на S символов, и формировать N кодовых символов на интервале, занимаемом S информационными символами, а не одним. Однако в настоящее время такой вид кодеров не находит широкого применения вследствие того, что сложность декодера экспоненциально растет с увеличением S . Поэтому чаще применяют другой способ повышения кодовой скорости, называемый *выкашиванием* (puncturing).

Под выкашиванием понимается удаление из кода некоторых символов по правилу, согласованному между передающей и приемной сторонами. Количество удаляемых битов определяет получающуюся кодовую скорость. Например, если с выхода декодера со скоростью $R = 1/2$ удалять каждый четвертый выходной бит, то получится код со скоростью $R = 2/3$. Основной особенностью выкашивания является то, что с помощью простого изменения числа удаляемых битов на базе одного и того же кодера могут быть построены коды с широким диапазоном кодовых скоростей.

При использовании выкалывания необходимо определить правило, по которому определяется местоположение удаляемых битов. В [7.10] рассматривается правило, согласно которому сверточный код со скоростью $R = 1/n$ делится на блоки, состоящие из l групп по n символов. В каждом блоке удаляют по одному кодовому символу в $g = n - g$ -символьных группах. Описанный алгоритм поясняется представленным на рис. 19.21 примером с параметрами $n = 2$, $l = 3$, $g = 2$, где удаляемые символы окрашены в серый цвет.

Рис. 19.21. Повышение скорости кодирования с помощью выкалывания при $n = 2$, $l = 3$, $g = 2$



Формируемый по рассмотренному правилу код с выкалыванием содержит l исходных битов сверточного кода в каждом блоке, длина которого после удаления g символов составляет $(l-g) \cdot n + g \cdot (n-1) = l \cdot n - g$ символов, так что в результате получается код со скоростью

$$R = \frac{l}{ln - g}.$$

Изменением параметров l и g при фиксированном n можно регулировать величину скорости в значительном диапазоне, добиваясь ее желаемого значения. Так, в рассмотренном примере $R = 3/4$.

Более компактно операцию выкалывания можно задать вектором выкалывания \mathbf{P} , число элементов которого определяет период выкалывания. Передаваемые биты в этом векторе отмечаются «1», а удаляемые — «0». В частности, для получения кодов со скоростями $R = 2/3$ и $R = 3/4$ из кода со скоростью $R = 1/2$ можно воспользоваться векторами

$$\mathbf{P} = [1110]^T \text{ и } \mathbf{P} = [110110]^T \text{ соответственно.}$$

Умножение \mathbf{P}^T на векторы кодовых символов сверточного кода дает соответствующие коды с выкалыванием.

Выкалывание может существенно снизить корректирующие способности сверточного кода, поэтому для оптимизации кодов с выкалыванием произведен машинный поиск, в результате которого получен целый ряд кодов с хорошими характеристиками.

19.4.3. Представление сверточных кодеров в виде цифровых фильтров

При описании сверточных кодеров и процесса кодирования используют разные подходы. Сверточные кодеры можно рассматривать как фильтры с конечной или бесконечной импульсной характеристикой (КИХ, БИХ), описывая связи в схеме фильтра с помощью образующих полиномов. Их можно представить в виде конечных автоматов, описывая процесс кодирования с помощью диаграмм состояний и кодовых решеток, позволяющих лучше понять алгоритмы декодирования.

Действительно, устройство, выполненное по схеме рис. 19.16, есть не что иное, как фильтр с конечной импульсной характеристикой (КИХ), реакцией которого на входной битовый поток является свертка последнего с импульсным откликом фильтра. Этот факт объясняет происхождение названия рассматриваемых кодов, а также лежит в основе удобного способа формального описания сверточного кодера. Свертка, связывающая кодовый символ u_i^n (т. е. появляющийся на выходе n -го сумматора при поступлении бита источника b_i) с входным битовым потоком:

$$u_i^n = \sum_{j=0}^{M-1} b_{i-j} g_j^n, \quad i = 0, 1, \dots, \quad n = 1, 2, \dots, N,$$

где, g_j^n — коэффициенты *порождающих* полиномов (19.48), определяющих вид схемы кодера.

Представленную на рис. 19.20 структуру кодера можно рассматривать как фильтр с бесконечной импульсной характеристикой (БИХ-фильтр).

19.4.4. Диаграммы состояний и кодовые решетки

Сверточный кодер можно рассматривать как конечный автомат, т. е. устройство с конечным числом состояний, и описывать с помощью диаграммы состояний.

Для сверточного кода со скоростью кодирования $R = 1/N$ состояние определяется содержимым $M - 1$ разрядов сдвигового регистра (рис. 19.16). Знание состояния и очередного бита данных на входе является необходимым и достаточным условием для определения данных на выходе.

Одним из способов представления кодирующих устройств является диаграмма состояния, включающая множество соответствующих возможным состояниям кодера узлов, связанных между собой направленными дугами (ветвями, путями), обозначающими переходы из одного состояния в другое за шаг кодирования. На ветвях диаграммы проставляют группы формируемых за переход выходных битов. При обозначении ветви используются сплошные и пунктирные линии. Вид ветви определяется значением входного бита, вызвавшим рассматриваемый переход.

Для кодера, схема которого представлена на рис. 19.19, диаграмма состояний показана на рис. 19.22. Кодер может принимать состояния 00, 10, 01 и 11 (первый символ отражает состояние крайней левой ячейки памяти), представляющие собой содержимое $M = 2$ разрядов сдвигового регистра. Существует всего два исходящих из каждого состояния перехода, соответствующие двум возможным значениям входного бита: переход, выполняемый при поступлении нуля, обозначен сплошной дугой, при поступлении единицы выполняется переход, обозначенный пунктирной линией. Например, если состояние кодера — 00, при поступлении очередного бита возможны только состояния 00 или 10. Для каждого пути между состояниями записано формируемое на выходе кодовое слово. Так, например, кодер, находящийся в состоянии 00, вырабатывает кодовую комбинацию 11 при поступлении на вход нулевого бита. Поэтому ветвь, отражающая переход 00 → 10, помечена символами 11 и т. д.

Двузначный характер переходов между состояниями объясняется поступлением на вход кодера только одного бита информации за один такт работы ($R = 1/N$). При скорости RS/N , когда скользящее окно сдвигается на S битов за один шаг кодирования, число возможных переходов из текущего состояния равно 2^S .

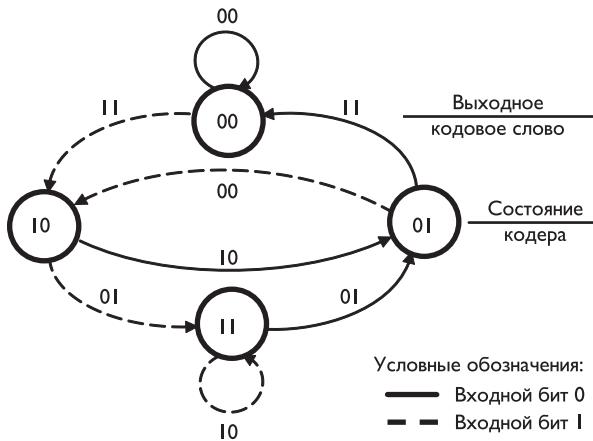


Рис. 19.22. Диаграмма состояний кодера, выполненного по схеме на рис. 19.18

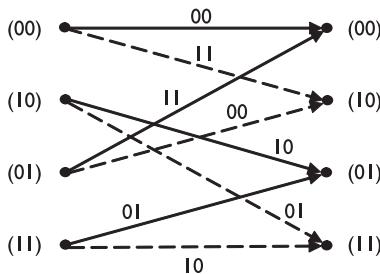


Рис. 19.23. Граф, описывающий состояния кодера, выполненного по схеме на рис. 19.18

Продолжая оставаться в рамках рассматриваемого примера (рис. 19.18), диаграмму состояний можно изобразить в виде графа, представленного на рис. 19.23. Он является аналогичной графической иллюстрацией состояния кодера для двух последовательных временных интервалов. Как и ранее, текущее состояние изменяется на последующее в зависимости от значения поступающего входного бита.

Эта версия диаграммы состояний указывает путь к построению модели, описывающей весь процесс кодирования во времени.

Пусть кодер находится в начальном состоянии 00. В этой ситуации, согласно представленному на рис. 19.23 графу, на первом такте работы кодер переходит или в состояние 00, или в состояние 10. Если его состояние остается прежним, т. е. 00, то на втором такте он снова переходит в одно из двух ранее указанных состояний. Если же на первом такте работы кодер оказывается в состоянии 10, то на втором такте возможен переход или в состояние 01, или в состояние 11. Если на втором такте работы кодер оказывается в состоянии 01, то на третьем такте возможен переход или в состояние 00, или в состояние 10. Если же на втором такте кодер оказывается в состоянии 11, то на третьем такте возможен переход или в состояние 01, или в состояние 11. После третьего такта работа кодера становится устойчивой, повторяя ранее описанные переходы, пока не прекратится поток входных битов. При отсутствии входной информации кодеру потребуется еще $M - 1$ тактов для обнуления, в течение которых будет продолжено формирование кодовых символов.

Модель поведения кодера, построенная на основе вышеприведенных рассуждений, представлена на рис. 19.24 и называется *решетчатой диаграммой* или просто *решеткой*. Любой путь по решетке представляет собой *кодовое слово* свер-

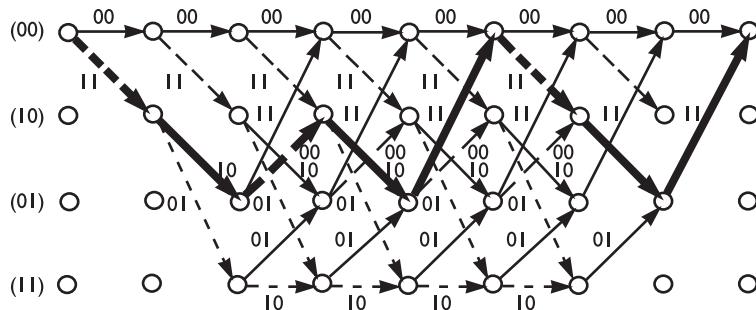


Рис. 19.24. Решетчатая диаграмма кодера, выполненного по схеме на рис. 19.18

точного кода. Например, входная информационная последовательность $\{b_i\} = 10100100$ отображается в кодовое слово $\{u_i^1, u_i^2\} = 1110001011111011$. Соответствующий этому кодовому слову путь выделен на рис. 19.24 жирной синей линией.

Сверточный код можно интерпретировать как достаточно длинный блоковый. Слова такого блокового кода есть просто различные пути на решетчатой диаграмме, так что минимальное расстояние Хэмминга между парами этих путей является кодовым расстоянием.

Для определения исправляющей способности сверточного кода необходимо установить минимальное расстояние Хэмминга между различными путями решетки. В случае сверточных кодов минимальное расстояние Хэмминга часто называют *свободным расстоянием* и обозначают как d_f . Данное название объясняется тем фактом, что вследствие линейности кода свободное расстояние есть не что иное, как минимальный вес ненулевого пути, т. е. пути, не совпадающего с верхней траекторией решетчатой диаграммы. Как отмечается в [7.10], для нахождения минимального расстояния Хэмминга в сверточном коде тестируются пути, которые отклоняются от нулевого в начале решетчатой диаграммы и после первого слияния с нулевым путем более от него не отвечаются. Среди всех путей с единственным отклонением от нулевого на рис. 19.23 кодовое слово $\{u_i^1, u_i^2\} = 11101100\dots$, кодирующее битовый поток $\{b_i\} = 100\dots$, имеет наименьший вес, величина которого равна пяти, так что $d_f = 5$. Свободное расстояние d_f гарантирует исправление любых $(d_f - 1)/2$ битовых ошибок, однако обычно исправляются и многие конфигурации большего числа ошибок. Существует лишь несколько примеров эффективных алгебраических правил сверточного кодирования. В основном же известные сверточные коды с хорошей корректирующей способностью были найдены с помощью компьютерного поиска.

19.4.5. Декодирование сверточных кодов

Рекуррентный характер процесса сверточного кодирования открывает путь к также рекуррентному и эффективному с вычислительной точки зрения алгоритму декодирования, получившему наименование алгоритма Витерби. Свое название этот алгоритм получил по имени автора, предложившего его в 1967 году. Алгоритм Витерби является оптимальным. Он обеспечивает максимально правдоподобное решение (см. 19.4.4), и сводится к нахождению кодового слова, наиболее близкого к принятому наблюдению с точки зрения выбранного критерия.

Жесткое декодирование

Для случая жестких решений, т. е. для двоичного симметричного канала, в качестве критерия близости выбрано расстояние Хэмминга. Наблюдение $\mathbf{Y} = \{y_i^1, y_i^2\}$ декодируется в такое кодовое слово $\tilde{\mathbf{U}} = \{\tilde{u}_i^1, \tilde{u}_i^2\}$, которое находится на наименьшем расстоянии Хэмминга относительно принятого \mathbf{Y} . Основная идея декодирования по Витерби заключается в поэтапном сравнении всех путей решетчатой диаграммы, которые представляют собой кодовые слова $\mathbf{U} = \{u_i^1, u_i^2\}$, с наблюдением \mathbf{Y} и отбрасывании тех из них, которые находятся на большем расстоянии от \mathbf{Y} , чем другие. Нетрудно убедиться в том, что если два пути, входящих в один и тот же узел, характеризуются различными расстояниями от наблюдения до данного узла, то у пути, который обладает большим кодовым расстоянием, отсутствует возможность стать впоследствии более близким к наблюдению. Он останется на большем расстоянии при любом продолжении обоих путей. Следовательно, из двух входящих в узел путей более удаленный от наблюдения может быть исключен из поиска ближайшего пути, что позволяет существенно сократить вычислительные затраты по сравнению с методом перебора всех возможных путей. Более подробно процедура декодирования может быть описана следующим образом.

На i -м шаге (этапе) декодирования, в течение которого принимается i -я n -символьная кодовая группа наблюдения \mathbf{Y} , выполняются следующие операции:

- определяются Хэмминговы расстояния между принятой n -символьной кодовой группой и каждой из ветвей решетчатой диаграммы; поскольку из каждого из 2^{M-1} узлов выходят две ветви, всего вычисляется 2^M расстояний;
- рассматриваются две ветви, идущие из разных предшествующих состояний к каждому из 2^{M-1} узлов;
- отвечающие указанным ветвям расстояния Хэмминга прибавляются к накапленным до i -го шага расстояниям Хэмминга двух соответствующих путей для получения новых значений расстояний; указанное накапливаемое расстояние пути называется *метрикой*;
- сравниваются метрики двух соревнующихся путей, идущих в одно и то же состояние; путь, находящийся на большем расстоянии от наблюдения, чем другой, отбрасывается и больше не учитывается в процедуре декодирования; оставшийся путь называется *выжившим* путем;
- для всех 2^{M-1} выживших путей запоминаются значения их метрик и декодер готов к переходу на $(i + 1)$ -й шаг процедуры.

Из приведенного описания следует, что ресурсосбережение алгоритма заключается в отбрасывании на каждом шаге половины из 2^M возможных путей, ведущих в 2^{M-1} узлов решетки. В результате число выживших путей остается постоянным и равным 2^{M-1} независимо от длины соревнующихся путей (кодовых слов), число которых удваивается на каждом шаге алгоритма декодирования.

Однако следует заметить, что всегда существует вероятность равенства текущих Хэмминговых расстояний двух путей, ведущих в один узел. Возможны различные стратегии разрешения подобных коллизий. Одна из них предполагает случайный равновероятный выбор выжившего пути. Это нарушает оптимальность декодирования, хотя сопутствующие энергетические потери, как правило,

пренебрежимо малы. Как альтернативный вариант можно посчитать оба конкурирующих пути выжившими и хранить их до тех пор, пока коллизия не разрешится на последующих шагах. При такой стратегии декодирование останется строго оптимальным, за что придется заплатить большим объемом памяти декодера.

Для лучшего понимания алгоритма Витерби рассмотрим в подробностях его функционирование на конкретном *примере* декодирования. Пусть сверточный код формируется кодером, выполненным по схеме на рис. 19.18. На приемной стороне наблюдается последовательность $\mathbf{Y} = 0100110000000000$.

Процесс декодирования этой последовательности проиллюстрирован решетчатыми диаграммами (рис. 19.25), на которых метрики узлов помечены непосредственно рядом с узлами, а принимаемые на текущем шаге пары символов наблюдения заключены в рамки.

Декодер инициализируется в предположении нулевого (т. е. 00) начального состояния регистра кодера. Два первых ($M - 1 = 2$) шага соответствуют переходному режиму кодера, когда лишь единственное ребро входит в каждое состояние (см. рис. 19.24), и поэтому все пути выживают. На первом шаге декодер сравнивает первую группу из $n = 2$ символов наблюдения \mathbf{Y} с двумя ребрами, выходящими из состояния (00). В соответствии с их расстоянием Хэмминга от наблюдаемых символов 01 как сплошное, так и пунктирное ребра решетки получают одинаковую, равную единице метрику, оцифровывающую каждое из ребер. Таким образом, метрики двух узлов, в которые входят ребра, равны единице. На следующем шаге измеряются расстояния между второй группой наблюдаемых символов 00 и двумя парами ветвей, исходящих из узлов (00) и (10). Полученные в результате этих измерений метрики указаны на соответствующих ветвях. Прибавленные к метрикам узлов предшествующего шага, они обновляют метрики узлов (00) и (10), а также инициализируют метрики еще двух узлов (01) и (11). Начиная с третьего шага в любой узел решетчатой диаграммы рис. 19.25 входит по два ребра, означая, что декодер должен решать, какое из них принадлежит выжившему пути. Чтобы не перегружать рисунок, метрики ребер на нем с этого шага не показаны. Как видно, на третьем шаге имеются два пути, ведущие к узлу (00). Их расстояния от наблюдаемых символов 010011 равны 3 и 2 соответственно. Первый из них не выживает и отбрасывается декодером вместе с его метрикой, поэтому на рисунке он зачеркнут, а на диаграмме, отвечающей следующему шагу, отсутствует. Второй путь выживает и запоминается со своей метрикой до следующего шага. Точно так же декодер выделяет выжившие пути и для остальных узлов. На последующих шагах декодер действует аналогично, сохраняя в памяти только $2^{M-1} = 4$ выживших путей, и каждая очередная диаграмма рис. 19.25 содержит только пути, выжившие на предшествующем шаге.

На 7-м шаге декодер впервые сталкивается с неоднозначностью: два пути приходят в узел (01), как и в узел (11), с равными расстояниями. Тот выбор выжившего пути, который показан на рисунке, сделан с помощью механизма равновероятного выбора, аналогичного бросанию монеты. Подобные события имеют место и на 8-м и 9-м шагах. Нетрудно убедиться, что любое альтернативное разрешение коллизий не повлияет на окончательный результат декодирования, за исключением, быть может, номера шага, на котором возможно первое считывание декодированных битов данных.

По завершении 9-го шага возникает критически важная ситуация: все выжившие пути совпадают друг с другом вплоть до седьмой группы наблюдаемых символов. Что бы ни произошло впоследствии, эта часть всех слившихся путей останется общей навсегда, означая, что соответствующие ей биты данных можно сразу выдать на выход как декодированные. Принимая во внимание, что ветви пути, показанные пунктиром, соответствуют нулю на входе кодера, а сплошной линией — единице, по выжившему пути (рис. 19.25) нетрудно видеть, что результатом декодирования будет последовательность $B = 1000000$. Сравнивая кодовое слово $U = 11101100000000$, соответствующее указанной последовательности B , с наблюдением $Y = 01001100000000$, можно видеть, что расстояние Хэмминга между ними равно двум, и если выданное декодером слово было передано в действительности, то имело место исправление двух символьных ошибок в полном согласии со свободным расстоянием $d_f = 5$. Подобные ситуации будут возникать и в дальнейшем, позволяя декодеру выдавать декодированные биты по ходу обработки потока символов наблюдения.

Выдача результатов декодирования в случайные моменты слияния выживших путей, как в рассмотренном примере, непрактична. Более предпочтительна процедура регулярного характера. С помощью многократных и практически исчерпывающих компьютерных тестов и имитационных экспериментов было установлено, что к i -му шагу декодирования слившийся участок всех выживших путей почти всегда заканчивается после $(i - 5M)$ битов данных, так что решение о любом бите может регулярно выдаваться на выход с задержкой в $5M$ [7.1].

Мягкое декодирование

Замечательное свойство сверточных кодов, придающее им еще большую привлекательность, — сравнительная простота осуществления мягкого декодирования.

Замена в алгоритме Витерби расстояния Хэмминга евклидовым преобразует декодирование в мягкое, оптимальное для АБГШ-канала. При этом метрики ребер и узлов становятся евклидовыми расстояниями. Такая модификация не лишит алгоритм Витерби присущих ему реализационных преимуществ. Метрики узлов вычисляются также рекурсивно, пошаговым аккумулированием метрик ребер, и на каждом шаге тот из двух путей, входящих в узел, который имеет худшую метрику, будет отброшен как невыживший.

Наиболее рациональна цифровая реализация декодеров, подразумевающая квантование входного наблюдения. Общепринято относить к числу мягких любые декодеры двоичных кодов с квантованием наблюдений более чем на два уровня. Исследования показывают, что в большинстве случаев 3-битовое (8-уровневое) квантование достаточно для достижения почти потенциальной (теоретически ожидаемой при непрерывной обработке) достоверности декодирования.

Выбор между жестким и мягким декодированием зависит от конкретных условий, поскольку более простое в реализации жесткое декодирование оплачивается энергетическими потерями, составляющими для АБГШ-канала 2 дБ и более, что считается довольно значимой цифрой.

В настоящее время известно много эффективных сверточных кодов, диапазон реального и потенциального применения которых в телекоммуникациях чрезвычайно широк. В частности, в 2G- и 3G-стандартах cdmaOne и WCDMA использу-

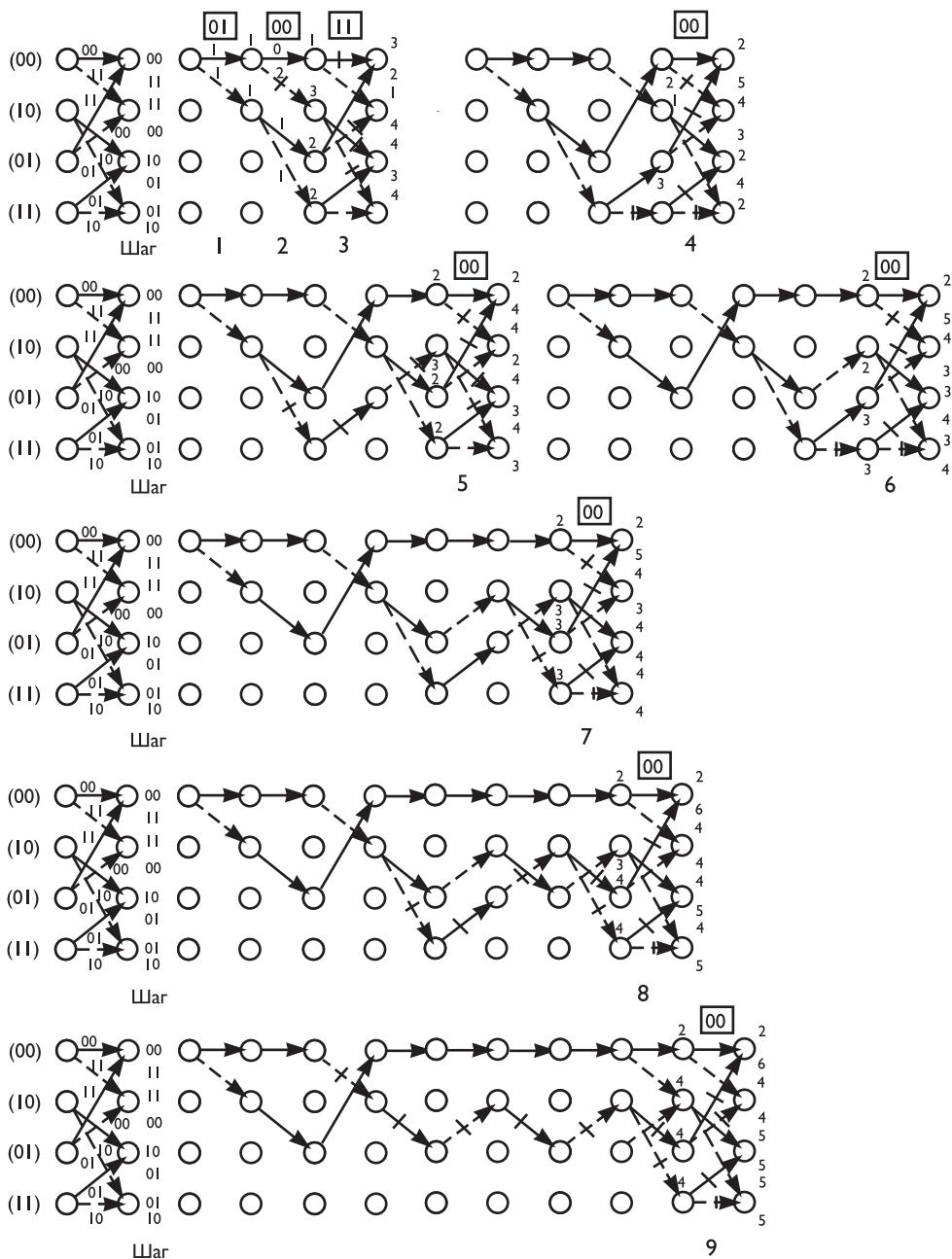


Рис. 19.25. Динамика декодирования сверточного кода по алгоритму Витерби

ются коды с длиной кодового ограничения $M = 9$ и скоростями $R = 1/2$, $R = 1/3$, обеспечивающие асимптотический выигрыш от кодирования около 7,8 дБ [7.10]. В 3G стандарте cdma2000 в дополнение к упомянутым выше предусмотрены

с параметрами $M = 9$, $R = 1/4$. Сверточные коды, представляя весьма значительную собственную ценность, служат также основой для построения других кодов, например, турбокодов, кратко обсуждаемых в следующем разделе и позволяющих надежно передавать данные со скоростями, близкими к границе Шеннона.

19.5. Турбокодирование

19.5.1. Основные положения

Турбокодирование является мощным средством повышения надежности и эффективности систем связи, обеспечивающим системам связи характеристики, близкие к теоретическим пределам. В настоящее время турбокодирование широко используется в системах космической, спутниковой, сотовой, проводной связи. Популярность турбокодов обусловлена их уникальной способностью обеспечивать помехоустойчивость передачи информации, близкую к теоретически достижимой (так называемый предел Шеннона), при умеренной сложности алгоритмов кодирования и декодирования.

Определение турбокодов впервые было дано в работе [7.17]. Турбокоды представляли собой отдельный класс двух параллельно каскадированных рекурсивных систематических сверточных кодов RSC с кодовой скоростью $R = 1/2$, конструктивной длиной $K = 5$ и псевдослучайным перемежителем длиной 65 536 битов.

Потенциал турбокодирования можно оценить по представленным результатам моделирования: с помощью данного кода авторам работы удалось получить вероятность ошибки на бит $P_b = 10^{-5}$ для BPSK модуляции в канале с аддитивным белым гауссовским шумом при отношении сигнал/шум 0,7 дБ после 18 итераций декодирования, что всего лишь на 0,5 дБ выше теоретически достижимой границы Шеннона для подобных параметров кодирования и канала. Этот результат был существенно лучше того, что позволяли получить все другие известные в то время коды.

С тех пор идея турбокодирования стремительно развивалась, и термин «турбокоды» был расширен на разные схемы каскадирования. Основными направлениями развития турбокодов считаются:

- сверточные турбокоды ТСС (*Turbo Convolutional Codes*), в которых используется параллельное каскадирование двух или большего числа сверточных кодеров;
- блочные турбокоды, или турбо коды произведения ТРС (*Turbo Product Codes*), формируемые путем последовательного или параллельного каскадирования двух или большего числа блочных кодеров.

Турбокодеки являются достаточно сложными устройствами, и их характеристики зависят от многих факторов. В данном разделе рассматриваются основные принципы турбокодирования и основные характеристики турбокодов.

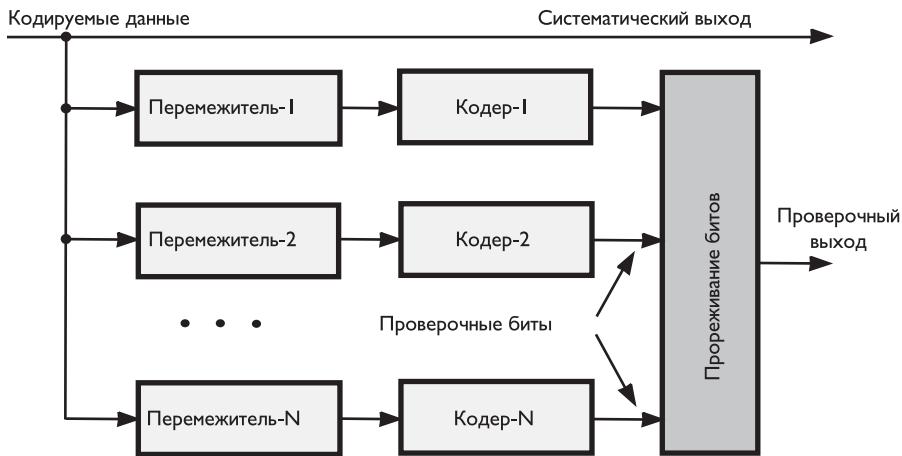


Рис. 19.26. Обобщенная схема турбокодера ТСС

19.5.2. Обобщенная схема турбокодера ТСС с параллельным каскадированием

Обобщенная схема турбокодера ТСС приведена на рис. 19.26. Из рисунка видно, что в кодере параллельно работают N кодеров.

Турбокодер формирует систематический код, в котором проверочная группа образуется из проверочных битов, генерируемых N кодерами RSC. Для регулирования общей скорости турбокода применяется схема прореживания проверочных битов. В канал связи передаются исходный информационный блок и прореженные проверочные биты. Общая кодовая скорость турбокода при использовании составляющих кодов со скоростью $1/2$ равна: $r = 1/(N + 1)$, где N — число кодеров турбокода. На практике часто ограничиваются использованием двух составляющих кодеров без перемежителя в первой ветви.

Таким образом, турбокодер в общем случае кодирует одну и ту же информацию как минимум дважды, но в разном порядке. Чем сильнее будут отличаться информационные блоки, подаваемые на турбокодеры с перемежителями, тем более «некоррелированным» будет информационный обмен между декодерами. В этом заключается ключевая идея турбокодирования, гарантирующая непрерывное улучшение корректирующей способности кода при итерационном процессе декодирования.

19.5.3. Сверточные турбокоды

Исследования показали [7.18], что рекурсивные систематические сверточные коды RSC являются более эффективными по сравнению с несистематическими сверточными кодами, поэтому сверточные турбокодеры строятся путем параллельного соединения двух или нескольких RSC-кодеров с небольшой длиной кодового ограничения ($K = 3 \dots 5$). Схема кодера такого турбокодера приведена на рис. 19.27.

Принцип работы турбокодера заключается в следующем [7.19]. На вход первого RSC-кодера по очереди поступают биты u_i информационного блока. Значе-

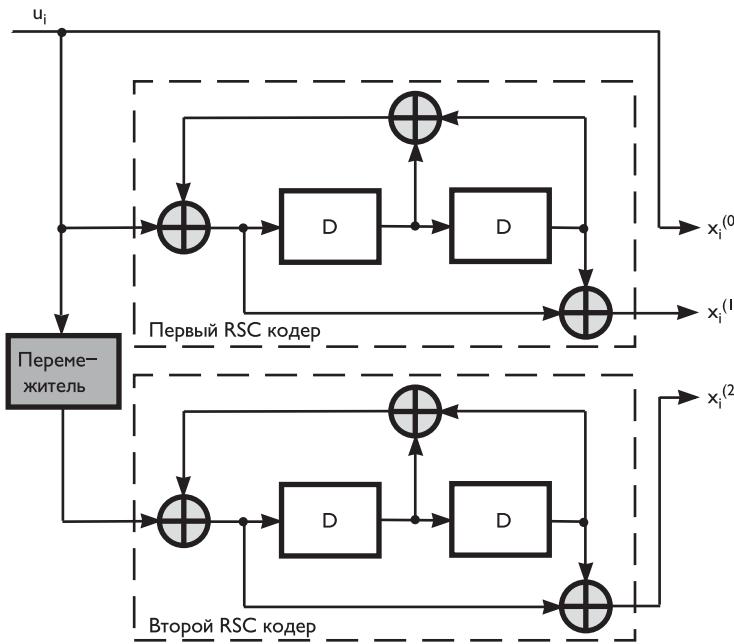


Рис. 19.27. Схема турбокодера

ния битов $x_i^{(0)}$ на систематическом выходе турбокодера совпадают со значениями входных битов. На проверочном выходе первого RSC-кодера формируются первые проверочные биты $x_i^{(1)}$. На вход второго RSC-кодера поступают информационные биты, номера которых определяются используемым перемежителем. В результате на проверочном выходе получается вторая группа проверочных битов $x_i^{(2)}$.

Перемежитель является важным элементом сверточного турбокодера. Его задача состоит в том, чтобы внести декорреляцию во входные информационные блоки первого и второго кодеров. Процедура перемежения состоит в считываии по некоторому псевдослучайному закону информационных битов из подготовленного ранее блока входной последовательности.

После кодирования всего информационного блока на вход одного или всех составляющих кодеров могут подаваться несколько дополнительных «хвостовых» битов (tail bits), обеспечивающих завершение работы кодера в нулевом состоянии. С выхода турбокодера на модулятор поступают информационный бит $x_i^{(0)}$ с систематического выхода первого RSC-кодера и два проверочных бита. В результате кодовая скорость r турбокода оказывается равной 1/3. Для получения других скоростей кодирования r применяется либо прореживание проверочных битов, либо использование по очереди проверочных битов то первого, то второго RSC-кодера.

Разумеется, и сверточные коды, и перемежители могут быть разными. Исследования показали, что структура перемежителя слабо влияет на характеристики турбокодов. В то же время с увеличением длины кодового ограничения сверточ-

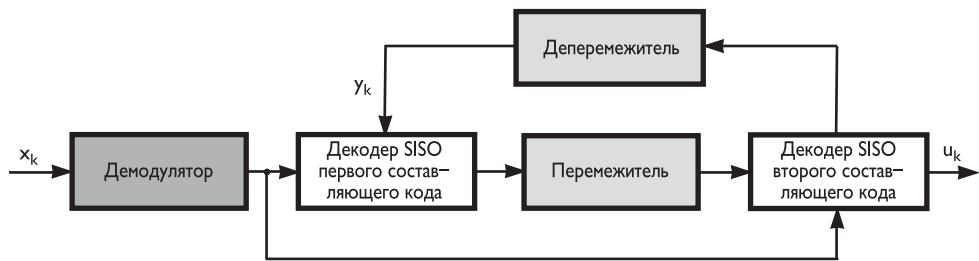


Рис. 19.28. Схема декодера турбокода

ного кода, как и глубины перемежения, эффективность турбокодов повышается [7.19].

19.5.4. Декодирование турбокодов

Декодирование кодовых слов турбокода осуществляется на основе разделения потока битов на несколько кодовых блоков (для кодера рис. 19.27 число кодовых блоков равно двум), причем информационные части этих блоков с точностью до перемежения идентичны. Это позволяет использовать параллельную схему декодирования, в которой два декодера производят декодирование своих кодовых блоков. Так как информационные части кодовых блоков идентичны, декодированную информацию первого (второго) декодера можно использовать в качестве априорной информации для второго (первого) декодера с целью многократного уточнения результатов декодирования [7.18]. В этом состоит принцип итерационного декодирования турбокода.

Вариант построения турбодекодера представлен на рис. 19.28. Декодер представляет собой каскадное соединение двух элементарных декодеров с «мягкими» входным и выходным сигналами (Soft-In Soft-Out — SISO). Каждый декодер выносит решение о значении декодируемого символа на основе критерия максимума апостериорной вероятности (Maximum A-Posteriori — MAP), обеспечивающего минимум вероятности ошибочного декодирования каждого бита и способного формировать мягкие решения для декодируемых битов [7.19] (апостериорная вероятность — условная вероятность случайного события при условии того, что известны апостериорные данные, т. е. данные, полученные после опыта). Мягкие решения обычно представляются с помощью логарифма отношения правдоподобия (Log Likelihood Ratio — LLR).

В статистике критерий отношения правдоподобия используется для сравнения двух моделей (нулевая и альтернативная модели). Отношение правдоподобия выражает, во сколько раз более вероятно, что данные соответствуют нулевой модели, а не альтернативной.

Каждый SISO-декодер имеет два входа:

- вход для сигнала, представляющего собой мягкое решение демодулятора;
- вход для сигнала апостериорной внешней информации от декодера второго элементарного кода.

Выходной сигнал декодеров для каждого информационного бита представляет собой величину, модуль которой пропорционален надежности приема дан-

ного информационного бита, а знак соответствует предполагаемому решению в виде 0 или 1. Существенным является то, что решение о каждом информационном бите вырабатывается каждым декодером с использованием сведений об информационных битах, содержащихся только в проверочной группе данного составного кода. Следовательно, апостериорная достоверность решения декодера оказывается некоррелированной с мягкими решениями демодулятора по каждому информационному биту, а также с информацией об информационных битах, содержащихся в проверочной группе другого составляющего кода. Поэтому оказывается возможным использовать внешнюю информацию каждого элементарного кода в качестве априорных сведений о передаваемых символах в процессе декодирования в другом составляющем декодере, осуществляя, таким образом, декодирование по методу MAP.

В процессе декодирования турбокода декодеры обмениваются друг с другом информацией, с каждой итерацией улучшая окончательное решение в смысле снижения вероятности ошибки на бит P_b в декодированной информационной последовательности. Однако, уже после первой итерации априорная информация, подаваемая на вход первого декодера с выхода второго, оказывается коррелированной с мягкими решениями демодулятора для проверочных символов первого составляющего кода. Поэтому улучшение окончательного решения с каждой итерацией становится меньше и величина P_b , достигаемая декодером таким методом, стремится к определенному пределу.

Декодирование заканчивается либо после выполнения заданного количества итерационных циклов, либо после достижения величиной поправки результата декодирования заданного порога. Окончательное (жесткое) решение о передаваемых информационных битах принимается после завершения последней итерации вторым декодером.

Методы построения SISO элементарных декодеров практически сводятся к использованию двух различных алгоритмов декодирования элементарных кодов, способных вырабатывать мягкие выходные решения о передаваемых информационных символах:

- алгоритм MAP, другое название BCJR (L. R. Bahl, J. Cocke, F. Jelinek, J. Raviv), и его упрощенные реализации log-MAP и max-log-MAP;
- алгоритм Витерби с мягким входным и выходным сигналом (Soft In Soft Out Viterbi Algorithm – SOVA).

Алгоритм MAP позволяет реализовывать потенциальные возможности турбокодов, однако он часто оказывается сложнее в реализации, чем метод SOVA, который, в свою очередь, может проигрывать алгоритму MAP в помехоустойчивости до 1 дБ при больших значениях глубины перемежения и кодового ограничения RSC.

Логарифм отношения правдоподобия

Мягкие решения в процессе декодирования представляются с помощью логарифма отношения правдоподобия LLR. Использование LLR упрощает процесс обмена информацией между декодерами составляющих кодов при итеративном декодировании турбокодов [7.19].

Значение LLR битов данных u_k обозначается $L(u_k)$ и определяется как логарифм отношения вероятностей двух возможных значений бита:

$$L(u_k) = \ln \left[\frac{P(u_k = +1)}{P(u_k = -1)} \right]. \quad (19.49)$$

Знак $L(u_k)$ показывает, какое из значений бита u_k (± 1) более правдоподобно, а амплитуда определяет надежность этого значения. В случае если $L(u_k) \approx 0$, то $P(u_k = +1) \approx P(u_k = -1) \approx 0,5$, следовательно, надежность данного решения минимальна. При $L(u_k) \gg 0$ имеется определенность в принимаемом решении.

Зная $L(u_k)$, можно вычислить вероятность того, что $u_k = +1$ или $u_k = -1$. Для этого, учитывая, что $P(u_k = -1) = 1 - P(u_k = +1)$, выражение (19.49) примет следующий вид:

$$e^{L(u_k)} = \frac{P(u_k = +1)}{1 - P(u_k = -1)},$$

откуда

$$P(u_k = \pm 1) = \left[\frac{e^{-L(u_k)/2}}{1 + e^{-L(u_k)}} \right] e^{\pm L(u_k)/2}. \quad (19.50)$$

Рассмотренный выше логарифм отношения правдоподобия LLR $L(u_k)$ выражается через априорную безусловную вероятность приема символа u_k .

Аналогичным образом определяется условный (апостериорный) LLR $L(u_k | \bar{y})$, основанный на апостериорной вероятности $P(u_k = \pm 1 | \bar{y})$ того, что выходной бит декодера будет принимать значения $u_k \pm 1$ при наблюдении на выходе канала передачи данных последовательности \bar{y} :

$$L(u_k | \bar{y}) = \ln \left[\frac{P(u_k = +1 | \bar{y})}{P(u_k = -1 | \bar{y})} \right]. \quad (19.51)$$

Искажающее влияние канала связи оценивается логарифмом отношения правдоподобия LLR $L(y_k | x_k)$, который связан с вероятностью приема из канала связи символа y_k при передаче $x_k \pm 1$ и определяется следующим образом:

$$L(y_k | x_k) = \ln \left[\frac{P(y_k | x_k = +1)}{P(y_k | x_k = -1)} \right]. \quad (19.52)$$

Если предположить, что передаваемый бит x_k передается через канал с аддитивным гауссовским шумом при использовании BPSK-модуляции, то условная вероятность приема символа y_k определяется так:

$$P(y_k | x_k = \pm 1) = \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{E_b}{2\sigma^2} (y_k \pm a)^2}, \quad (19.53)$$

где E_b — энергия, приходящаяся на один передаваемый бит, σ^2 — дисперсия шума и a — коэффициент затухания (при $a = 1$ получается канал с аддитивным белым гауссовским шумом без затухания).

Таким образом, для случая передачи по каналу с гауссовским шумом с модуляцией BPSK выражение (19.52) можно привести к виду:

$$\begin{aligned} L(y_k | x_k) &= \ln \left[\frac{P(y_k | x_k = +1)}{P(y_k | x_k = -1)} \right] = \ln \left[\frac{e^{-\frac{E_b}{2\sigma^2} (y_k - a)^2}}{e^{-\frac{E_b}{2\sigma^2} (y_k + a)^2}} \right] = \\ &= \frac{E_b}{2\sigma^2} (y_k + a)^2 - \frac{E_b}{2\sigma^2} (y_k - a)^2 = \frac{E_b}{2\sigma^2} 4y_k a = L_c y_k, \end{aligned} \quad (19.54)$$

где $L_c = \frac{E_b}{2\sigma^2} 4a$ определяет надежность канала связи и зависит только от отношения сигнал/шум и коэффициента затухания канала. Отсюда следует, что для канала с гауссовским шумом и двоичной фазовой модуляцией условный LLR $L(u_k | x_k)$ будет определяться произведением принятого символа y_k на надежность канала L_c .

MAP-алгоритм декодирования двоичных сверточных турбокодов

Алгоритм декодирования MAP был предложен в 1974 году для оценки апостериорной вероятности состояний и переходов марковского источника. Этот алгоритм также известен под названием BCJR, данным ему по первым буквам фамилий авторов (L.R. Bahl, J. Cocke, F. Jelinek, J. Raviv) [7.18]. Они в своих работах показали, как с помощью данного алгоритма можно декодировать блоковые и сверточные коды. В отличие от алгоритма Витерби SOVA, который минимизирует вероятность ошибки последовательности символов, MAP-алгоритм оптимален в смысле минимизации вероятности ошибки на бит. Однако MAP-алгоритм при работе обрабатывает все возможные пути кодовой решетки, что оказывается неприемлемо сложным для применения в реальных системах. Поэтому широкого распространения до открытия турбокодов он не получил.

Отличительной особенностью MAP алгоритма является то, что он кроме оценки значения бита также вычисляет вероятность правильности своего решения. Данное свойство очень важно для итеративного декодирования турбокодов, и поэтому в ряде исследовательских работ используется именно этот метод для декодирования составляющих кодов.

Основываясь на принятой последовательности y_k MAP алгоритм для каждого декодируемого бита x_k вычисляет вероятность того, что этот бит равен $+1$ или -1 . В соответствии с (3) данная операция эквивалентна определению апостериорного LLR $L(u_k | \bar{y})$. Применяя теорему умножения вероятностей $P(a \wedge b) = P(a|b)P(b)$, определяющую вероятность $P(a \wedge b)$ совместного появления зависимых событий a и b , выражение (19.51) приводится к виду:

$$L(u_k | \bar{y}) = \ln \left[\frac{P(u_k = +1 \wedge \bar{y})}{P(u_k = -1 \wedge \bar{y})} \right]. \quad (19.55)$$

Как показано в [7.19], значение апостериорного LLR $L(u_k | \bar{y})$ может быть представлено в следующем виде:

$$L(u_k | \bar{y}) = L(u_k) + L_c y_{ks} + L_e(u_k). \quad (19.56)$$

Первое слагаемое $L(u_k)$ является априорной информацией о значении бита u_k . В начале итерационного процесса различные значения u_k , как правило, равновероятны и $L(u_k) = 0$. В процессе итеративного декодирования турбокода декодер каждого составляющего кода обеспечивает другие декодеры оценкой априорного LLR $L(u_k)$.

Второе слагаемое $L_c y_{ks}$ является мягким выходом канала связи для систематического бита u_k , значение которого после передачи по каналу равно y_{ks} . При большом отношении сигнал/шум надежность канала L_c также велика, и текущий систематический бит будет оказывать значительное влияние на апостериорный LLR $L(u_k | \bar{y})$. Для канала связи с низким отношением сигнал/шум L_c невелико и, соответственно, значение систематического бита не будет сильно воздействовать на $L(u_k | \bar{y})$, вычисляемый MAP-декодером.

Последнее слагаемое $L_e(u_k)$ вычисляется на основе взаимосвязи закодированных битов с учетом априорной информационной последовательности $L(u_n)$ и принятой из канала последовательности \bar{y} , за исключением принятого систематического бита y_{ks} и априорной информации $L(u_k)$. Поэтому $L_e(u_k)$ называется внешней информацией для бита u_k . Выражение (19.56) показывает, что внешняя информация $L_e(u_k)$ от МАР декодера может быть получена из его мягкого решения $L(u_k | \bar{y})$ вычитанием априорной информации $L(u_k)$ и мягкого выхода демодулятора L_{cyks} для систематического бита u_k . Внешняя информация $L_e(u_k)$ зависит от априорной информации $L(u_n)$, $n \neq k$, и принятых систематических битов и не зависит от априорной информации $L(u_k)$ и систематического бита y_{ks} .

С учетом особенностей рассмотренного выше итеративного принципа работы МАР-алгоритма структурная схема декодера турбокода может быть реализована в виде, изображенном на рис. 19.29. Первый декодер на первой итерации принимает канальную последовательность $L_c \bar{y}^{(1)}$, включающую принятые версии систематических L_{cyks} и проверочных L_{cyki} битов первого кодера. В случае если в передатчике для повышения кодовой скорости применялось прореживание битов, на место удаленных битов вставляются нули.

Затем декодер первого составляющего кода на основе последовательности $L_c \bar{y}^{(1)}$ формирует оценку условного LLR $L_{11}(u_k | \bar{y})$ для каждого бита данных u_k . Индекс 11 показывает, что $L(u_k | \bar{y})$ вычислялся на первой итерации первым декодером. На первой итерации отсутствует априорная информация о значениях систематических битов, поэтому значения $L(u_k)$ в выражении (19.56) должны быть равны нулю. Затем начинает работать второй декодер. Он принимает канальную последовательность $L_c \bar{y}^{(2)}$, включающую перемеженную версию систематических битов L_{cyks} и проверочные L_{cyki} биты второго кодера. Кроме того, теперь турбодекодер может использовать условный LLR $L_{11}(u_k | \bar{y})$ при формировании априорной информации $L(u_k)$ для второго декодера. Как видно из рис. 19.29, внешняя информация первого декодера используется в качестве априорной для второго после перемежения, необходимого для организации последовательности в том же порядке, что и поступающие на второй декодер систематические биты. Таким образом, второй декодер формирует оценку условного LLR $L_{12}(u_k | \bar{y})$ с использованием канальной последовательности $L_c \bar{y}^{(2)}$ и внешней информации $L_e(u_k)$ первого декодера. На этом заканчивается первая итерация.

На второй итерации первый декодер для получения условного LLR $L_{21}(u_k | \bar{y})$ снова обрабатывает канальную последовательность $L_c \bar{y}^{(1)}$, но при этом используется априорная информация $L(u_k)$, полученная из апостериорного LLR $L_{12}(u_k | \bar{y})$, вычисленного вторым декодером на первой итерации. Затем снова работает второй декодер, формируя улучшенную оценку LLR $L_{22}(u_k | \bar{y})$, и т. д. При этом средняя вероятность ошибки на информационный бит постепенно уменьшается. Однако с каждой новой итерацией величина улучшения становится все меньше. Поэтому для снижения сложности декодера итеративный процесс прекращается после выполнения некоторого небольшого (порядка 8–10) количества итераций. Иногда для завершения декодирования используется какой-либо критерий, оценивающий возможное улучшение на следующих итерациях. Это позволяет значительно сократить среднее число итераций и сложность декодера при незначительных потерях в эффективности.

После выполнения заданного количества итераций или срабатывания условия завершения из деперемеженной оценки LLR $L_{i2}(u_k | \bar{y})$ второго декодера форми-

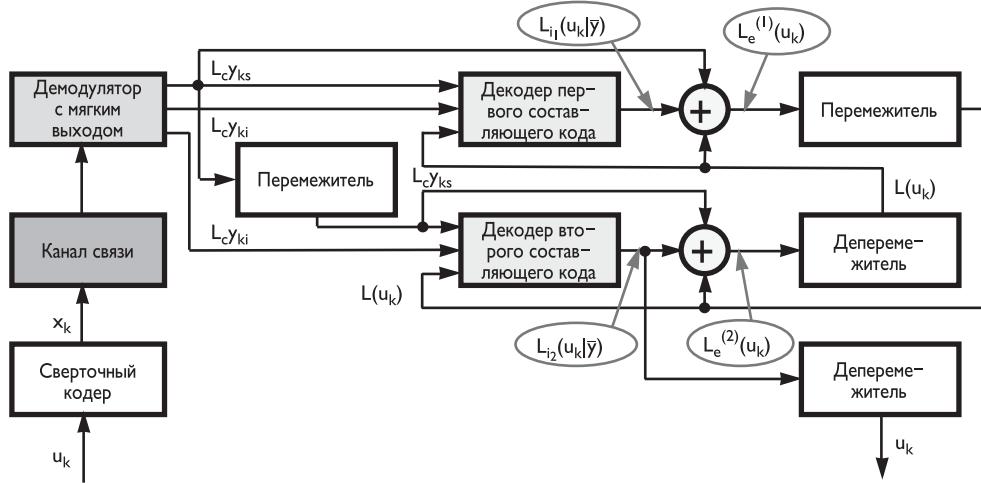


Рис. 19.29. Схема МАР-декодера турбокода

руется выход турбодекодера в виде блока значений апостериорных LLR. Знак этих значений определяет жесткое решение относительно значения декодируемого бита u_k , а модуль — надежность данного решения.

Как было отмечено выше, алгоритм МАР в своем непосредственном виде характеризуется существенной сложностью и большими вычислительными затратами. В связи с этим на основе МАР-алгоритма были разработаны алгоритмы декодирования с мягким решением — max-log-MAP и log-MAP, которые используются в декодерах составляющих кодов при итеративном декодировании и сложность которых значительно меньше сложности МАР-алгоритма.

19.5.5. Характеристики помехоустойчивости сврточных турбокодов ТСС

Исследованию помехоустойчивости турбокодов посвящено значительное число работ. Рассмотрим основные результаты, впервые полученные в работе [7.17], для сврточного турбокода с длиной кодового ограничения $K = 5$, скоростью $r = 1/2$, образующими полиномами $G_0 = 37$ и $G_1 = 21$, перемежителем длиной $256 \times 256 = 65\,536$ битов в канале связи с аддитивным белым гауссовским шумом и BPSK модуляцией. Результаты моделирования в виде графиков зависимостей вероятности ошибки на бит P_b от отношения сигнал/шум E_b/N_0 приведены на рис. 19.30.

Из графиков видно, что с увеличением числа итераций для отношений сигнал/шум $E_b/N_0 > 0,5$ дБ вероятность ошибки снижается. При малом числе итераций снижение вероятности ошибки больше, при большом числе итераций — меньше. При числе итераций больше 18 дальнейшее увеличение их числа для данного кода к снижению вероятности ошибки не приводит. При 18 итерациях вероятность ошибки $P_b = 10^{-5}$ достигается при отношении сигнал/шум $E_b/N_0 = 0,7$ дБ, что лишь на 0,5 дБ выше теоретической границы для данного канала.

Из графиков рис. 19.30 также видно, что при большом числе итераций и вероятностях ошибки $P_{\text{оп}} < (10^{-5} \dots 10^{-6})$ наблюдается эффект насыщения (error

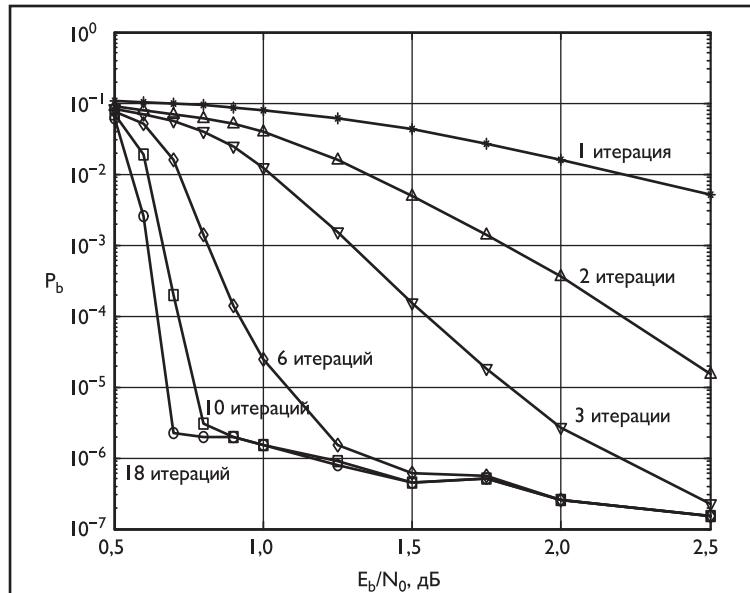


Рис. 19.30. Графики зависимости вероятности ошибки на бит P_b для сверточного турбокода со скоростью $r = 1/2$, длиной кодового ограничения $K = 5$, глубиной перемежения 65 536 битов в канале с белым шумом от отношения сигнал/шум E_b/N_0 при разном числе итераций [7.22]

floor) — резкое снижение крутизны кривых помехоустойчивости для заданного турбокода. Исследования показали, что одним из факторов, влияющих на точку перегиба, является длина перемежителя. Увеличение длины перемежителя в 4–5 раз понижает точку перегиба по вероятности ошибки примерно в 2 раза. Но есть и другие факторы, влияющие на положение точки перегиба и крутизну кривых помехоустойчивости после перегиба [7.19].

Следует отметить также влияние составляющих кодов на характеристики турбокодов. Важными параметрами, оказывающими влияние на эффективность турбокода, являются длина конструктивного ограничения составляющих кодов и их образующие полиномы [7.19]. Так, для турбокода с длиной конструктивного ограничения $K = 3$ оптимальными порождающими полиномами в смысле максимизации свободного расстояния являются полиномы $G_0 = 7$ и $G_1 = 5$ в восьмеричном представлении. Для длины конструктивного ограничения $K = 4$ оптимальными образующими полиномами составляющих кодов являются $G_0 = 15$ и $G_1 = 17$ в восьмеричном представлении. Эффективность данного турбокода на 0,25 дБ лучше, чем турбокода с $K = 3$. Для турбокода с конструктивной длиной $K = 5$ используются образующие полиномы $G_0 = 37$ и $G_1 = 21$. Увеличение длины конструктивного ограничения K улучшает характеристики турбокода, однако при увеличении K на единицу сложность декодера удваивается, поэтому на практике часто ограничиваются применением турбокодов с $K = 3$.

19.5.6. Блочные турбокоды

Блочные турбокоды ТРС, как и сверточные турбокоды ТСС, имеют высокие показатели помехоустойчивости. Исследования показали несколько более высокую эффективность блочных турбокодов по сравнению со сверточными при кодовых скоростях $r > 0,7$ [7.21]. Блочные турбокоды могут формироваться как с последовательным, так и с параллельным каскадированием двух или нескольких блочных кодов [7.22]. Считается, что блочные турбокоды с последовательным каскадированием более эффективны, чем с параллельным.

Так, блоковый турбокод $(64, 57) \times (64, 57)$ со скоростью $r = 0,793$ при вероятности битовой ошибки $P_b = 10^{-6}$ имеет на 1,5 дБ большую эффективность, чем каскадное соединение кода Рида–Соломона и сверточного кода с $r = 0,789$. Это результат всего лишь на 1,1 дБ хуже теоретически достижимой границы Шеннона для подобных параметров кодирования и канала [7.23, 7.24].

Особенностью ТРС является использование в качестве составных блоковых систематических кодов, таких как расширенные коды Хэмминга, коды с контролем четности (parity check codes), а также коды повторения (repetition codes). Применение простых составных кодов в совокупности с итеративным алгоритмом декодирования позволяет достичь достаточно хороших результатов как по вычислительным затратам, так и по эффективности [7.22].

Двумерный блочный турбокод условно изображается в виде прямоугольника (рис. 19.31) и основывается на двух систематических кодах: горизонтальном C_x и вертикальном C_y .

Входной поток битов, поступающих на кодер, построчно записывается в матрицу размерности $k_x \times k_y$. Сначала кодируется k_y строк, затем n_x столбцов, в результате чего получается кодированная матрица, содержащая как информационные, так и проверочные символы. Каждая строка матрицы представляет собой кодовое слово $C_x = (n_x, k_x)$ и состоит из k_x информационных символов и $n_x - k_x$ проверочных. Каждый столбец, соответственно, представляет собой кодовое слово C_y и состоит из k_y информационных символов и $n_y - k_y$ проверочных. Как правило, передача битов из кодированной матрицы в последующие цепи осуществляется построчно.

Характеристики блочных турбокодов не зависят от способа построения перекрестителя, так как строки и столбцы каскадных блочных кодов независимы. Случайное перемешивание битов данных не дает существенного улучшения.

В качестве достоинств блочных турбокодов можно отметить следующие [7.23].

- Блочные турбокоды имеют большое минимальное расстояние Хемминга. Минимальное расстояние Хемминга для ТРС равно произведению минимальных расстояний Хемминга для C_x и C_y составляющих кодов, из-за чего улучшаются характеристики кода при низких вероятностях ошибки. Поэтому коды, построенные по принципу ТРС, не имеют эффекта насыщения *error floor* в отличие от сверточных турбокодов.
- Блочные турбокоды имеют меньшую задержку по сравнению со сверточными турбокодами. Это означает, что для заданного времени декодирования ТРС могут использовать большее число итераций, что может обеспечить лучшие характеристики.

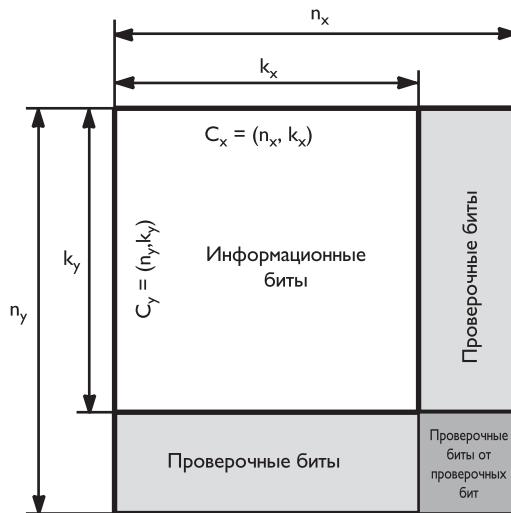


Рис. 19.31. Структура блочного турбокода

- Кодеки ТРС широко доступны, поскольку их реализация менее сложна, чем сверточных турбокодов.

К недостаткам подобных схем кодирования можно отнести необходимость использования различных структур кодов для обеспечения разных скоростей кодирования.

Итерационный процесс декодирования ТРС может быть построен на алгоритмах MAP и log-MAP. В качестве примера ниже приводятся результаты декодирования ТРС (табл. 19.8), рекомендованных стандартом IEEE 802.16 (WiMAX), по алгоритму log-MAP [7.23].

Таблица 19.8. Результаты декодирования ТРС

Код	Скорость, r	Примерная скорость, R
$(32, 26) \times (16, 11)$	0,5586	1/2
$(32, 26) \times (16, 15)$	0,762	3/4
$(32, 26) \times (32, 26)$	0,66	3/5
$(64, 63) \times (32, 26)$	0,8	2/3

С примером декодирования простейшего двумерного блочного турбокода с одним символом проверки на четность можно ознакомиться в работе [7.25].

На рис. 19.32 приведены зависимости вероятности ошибки на бит P_b для блочных турбокодов в канале с белым шумом.

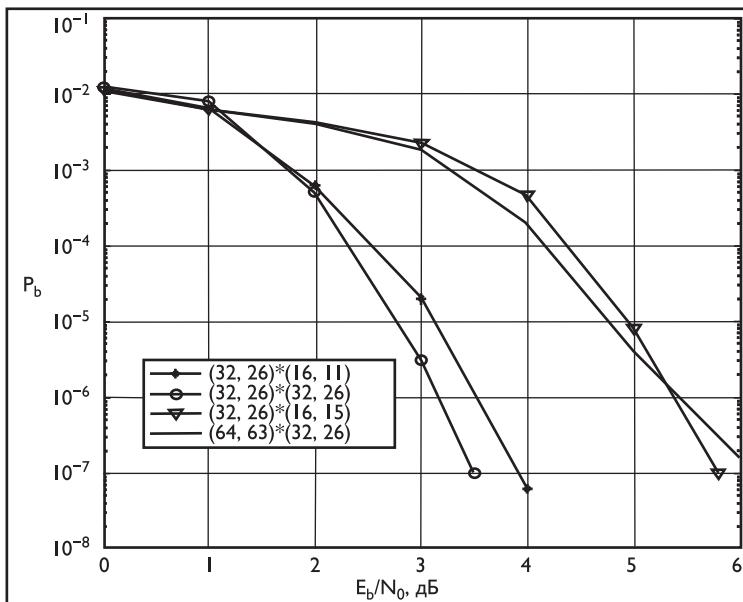


Рис. 19.32. Графики зависимости вероятности ошибки на бит P_b для блочных турбокодов в канале с белым шумом от отношения сигнал/шум E_b/N_0 при 30 итерациях [7.24]

19.6. Низкоплотностные коды

19.6.1. Основные положения

Коды с низкой плотностью проверок на четность (LDPC) — это класс линейных блоковых кодов, позволяющих получить превосходную эффективность с относительно малыми вычислительными затратами на их декодирование. Эти коды были предложены Р. Галлагером [7.25] еще в 1963 г., однако были забыты почти на сорок лет в связи со сложностью реализации алгоритмов их декодирования. Но развитие цифровой техники позволило преодолеть многие проблемы и в конце XX столетия исследования в области LDPC-кодов получили новый импульс. В 1981 г. Р.М. Таннером [7.26] было предложено использовать двудольные неориентированные графы, впоследствии названные графами Таннера, для описания структуры итеративно декодируемых кодов. В принципе любой блоковый код размерности (M, N) , где N — число битов, M — число проверок в кодовом слове, можно представить в виде двудольного графа Таннера [7.26], например, на рис. 19.33 изображен такой график для кода Хэмминга $(7, 4)$, проверочная матрица которого имеет вид:

$$H = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}.$$

Вершины графа называются проверочными (check nodes — CN) и битовыми узлами (variable nodes — VN), они обозначены на рис. 19.33 как m_i и n_i соответственно.

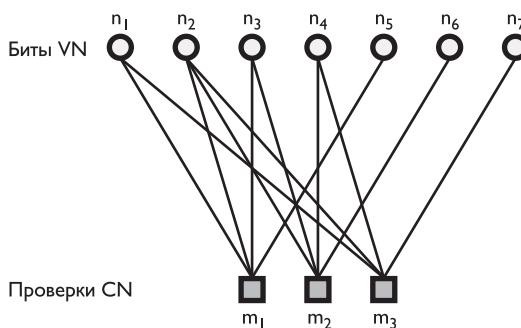


Рис. 19.33. Граф Таннера для кода Хэмминга (7, 4)

При помощи графа Таннера большинство алгоритмов декодирования LDPC-кодов можно представить в виде процесса последовательного обмена сообщениями между соединенными ребрами вершинами. Для проверочных и битовых узлов графа вводится понятия степени — величины, показывающей число ребер входящих в рассматриваемый узел. Степени битовых и проверочных узлов обозначаются d_c и d_r соответственно. Если d_c и d_r фиксированы для всех узлов, код называют регулярным, а если хотя бы один из этих параметров изменяется от узла к узлу — нерегулярным. Для описания нерегулярных кодов вводится ряд распределения степеней, показывающий долю узлов, имеющих конкретную степень.

В 1996 г. вышла в свет первая после Р. Галлагера работа, посвященная использованию LDPC-кодов в качестве кодов, способных вплотную приблизиться к границе Шеннона при достаточно большой длине кодового слова [7.27]. Появление этой статьи породило целую волну исследований, посвященных поиску новых, более эффективных структур LDPC-кодов, а также альтернативных алгоритмов их декодирования с различными соотношениями эффективность/производительность.

В последнее время LDPC-коды получили широкое распространение благодаря своей превосходной эффективности. Использование LDPC-кодов предусматривает большинство современных стандартов передачи данных (например, стандарты IEEE 802.11, IEEE 802.16), стандартов цифрового вещания (например, стандарты DVB-S2, DVB-T2, DVB-C2).

19.6.2. Классификация LDPC-кодов

По определению, данному Р. Галлагером [7.25], низкоплотностный код — это линейный код, проверочная матрица H которого размерности $M \times N$ содержит $d_c \ll M$ единиц в каждом столбце и $d_r \ll N$ единиц в каждой строке. Причем распределение единиц по столбцам и строкам в общем случае случайно.

На практике случайное распределение единиц крайне неудобно — для кодирования и декодирования приходиться хранить проверочные и генераторные матрицы, что достаточно накладно, особенно при больших длинах кодов.

Очевидным средством борьбы с этой проблемой является переход к низкоплотностным кодам, проверочная матрица которых обладает какой-то структурой. Простейший вариант структуризации проверочной матрицы — использование циклических кодов [7.28].

Формально проверочная матрица такого кода представляет собой циркулянтную матрицу размерности $N \times N$, в которой каждая строка получается циклическим сдвигом вправо предыдущей строки. Значение влияния циклическости проверочной матрицы на сложность декодера LDPC-кода сложно переоценить, поскольку каждая из строк матрицы однозначно определяется предыдущей строкой, в связи с чем реализация декодера может быть существенно упрощена по сравнению со случайной структурой проверочной матрицы. Как известно, кодер циклического кода достаточно просто реализовать с использованием сдвигового регистра и набора сумматоров [7.28].

К недостаткам циклических кодов можно отнести фиксированный, для всех скоростей кодирования, размер проверочной матрицы $N \times N$, что подразумевает более сложный декодер, а также высокий Хэммингов вес строк, что усложняет структуру декодера. Дополнительно стоит заметить, что циклический код — всегда регулярный.

К достоинствам помимо упрощения кодирования/декодирования следует отнести большое минимальное расстояние и очень низкий порог при итеративном декодировании [7.28].

Желание преодолеть недостатки циклических LDPC-кодов привело к появлению квазициклических LDPC-кодов. Квазициклические коды также имеют хорошую структуру, позволяющую упростить кодер и декодер. В дополнение к этому они позволяют более гибко подойти к разработке кода, в частности позволяют синтезировать нерегулярные коды. Проверочная матрица такого кода представляет собой не что иное, как набор циркулянтных подматриц:

$$H = \begin{bmatrix} A_{11} & \cdots & A_{1N} \\ \vdots & & \vdots \\ A_{M1} & \cdots & A_{MN} \end{bmatrix}.$$

Очевидно, что для получения низкоплотностного кода, циркулянтные матрицы должны быть разреженными, что на практике означает использование в качестве циркулянтов единичных матриц. Для того чтобы получить нерегулярный код, какие-то подматрицы просто объявляются нулевыми.

19.6.3. Методы построения проверочных матриц

Методы построения LDPC-кодов также можно разбить на классы. К первому классу относятся все алгоритмические способы и способы, использующие вычислительную технику. А ко второму — способы, основанные на теории графов, математике конечных полей, алгебре и комбинаторике.

При этом стоит заметить, что первый класс методов позволяет получать как случайные, так и структурированные LDPC-коды, в то время как второй нацелен на получение только структурированных LDPC-кодов, хотя бывают и исключения.

В отличие от других линейных блоковых кодов, таких как БЧХ или кодов Рида–Соломона, имеющих строгий алгоритм синтеза кодов с заданными параметрами, для LDPC-кодов существует множество способов построения кодов.

Здесь мы рассмотрим способы построения LDPC-кодов, предложенные Галлагером и МакКеем. А также подробно рассмотрим более сложный алгоритм построения достаточно эффективных кодов повторения накопления, используемый в некоторых стандартах.

Коды Галлагера

Р. Галлагер предложил следующий алгоритм построения низкоплотностных кодов. Пусть проверочная матрица кода имеет вид:

$$H = \begin{bmatrix} H_1 \\ H_2 \\ \vdots \\ H_{d_c} \end{bmatrix},$$

где подматрицы H_a , $a = 1, \dots, d_c$ имеют структуру, которая может быть описана следующим образом.

Для любых двух целых μ и d_r , больших 1, каждая подматрица H_a имеет размерность $\mu \times \mu \cdot d_r$, при этом веса строк этой подматрицы — d_r , а столбцов — 1. Подматрица H_1 имеет специфическую форму: для $i = 0, 1, \dots, \mu - 1$ i -я строка содержит все d_r единиц на позициях с i от d_r до $(i + r) \cdot r - 1$. Оставшиеся подматрицы получаются перестановкой столбцов матрицы H_1 . Очевидно, что результирующая матрица H — регулярная матрица размерности $\mu \cdot d_c \times \mu \cdot d_r$ с весами строк и столбцов d_r и d_c соответственно.

Важной характеристикой матрицы LDPC-кода является отсутствие циклов определенного размера (кратности). Под циклом кратности 4 понимается наличие в двух разных столбцах проверочной матрицы ненулевых элементов на совпадающих позициях. Отсутствие цикла кратности 4 определяется вычислением скалярного произведения столбцов матрицы: если всевозможные скалярные произведения всех столбцов матрицы не превосходят 1, то это означает отсутствие в матрице циклов кратности 4. Цикл кратности 4 является минимально возможным и встречается существенно чаще циклов большей длины (6, 8, 10 и т. д.). Присутствие в матрице LDPC-кода циклов любой кратности свидетельствует о заложенной в структуру матрицы избыточности, не приводящей к улучшению помехоустойчивых свойств кода. Пример циклов кратности 4 приведен на рис. 19.34.

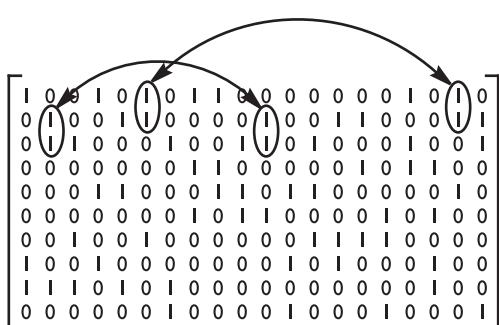


Рис. 19.34. Циклы кратности 4 в матрице LDPC-кода

Рассмотренный выше алгоритм не гарантирует отсутствие циклов кратности 4, однако они могут быть удалены впоследствии. Галлагер показал [7.25], что ансамбль таких кодов обладает прекрасными свойствами. Также была показана возможность реализации достаточно простых кодеров, поскольку проверочные биты такого кода могут быть найдены по проверочной матрице кода как функция информационных узлов.

Пример. проверочной матрицы кода Галлагера $(20, 5)$ $d_c = 3$, $d_r = 4$, $\mu = 5$ имеет следующий вид:

$$H = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Коды МакКей

По прошествии тридцати пяти лет МакКей, будучи незнакомым с работой Галлагера, повторно открыл преимущества кодов с разреженными матрицами, и был первым, кто при помощи компьютерного моделирования показал возможность этих кодов вплотную приблизиться к границе Шеннона как для двоичного симметричного канала, так и для канала с аддитивным белым гауссовским шумом. МакКей предложил несколько компьютерных алгоритмов построения проверочных матриц низкоплотностных кодов [7.27]. Приведем некоторые из них в порядке увеличения сложности реализации.

Проверочная матрица H синтезируется путем случайного генерирования столбцов веса d_c и, насколько это возможно, равномерным распределением весов строк d_r .

Проверочная матрица H синтезируется путем случайного генерирования столбцов веса d_c и строк веса d_r , с дополнительной проверкой на отсутствие циклов кратности 4.

Проверочная матрица H синтезируется по алгоритму 2, с дополнительным удалением циклов кратности 4.

Проверочная матрица H синтезируется по алгоритму 3, с дополнительным условием, что проверочная матрица имеет вид $H = [H_1 H_2]$, где H_2 — обратимая матрица.

Недостатком алгоритмов МакКея является отсутствие какой-либо структуры в проверочных матрицах, что усложняет процесс кодирования.

Кодирование осуществляется приведением матрицы H к виду $H = [P^T I]$ (или умножением H на H_2^{-1} для алгоритма 4), из которого можно получить генераторную матрицу в систематической форме $G = [PI]$. Проблема при кодировании по матрице G заключается в том, что подматрица P , в общем случае, не является разреженной. То есть для кодов, представляющих интерес, сложность кодирования оказывается достаточно высокой.

Коды повторения накопления

Выше были рассмотрены достаточно простые алгоритмы построения проверочных матриц LDPC-кодов, появившиеся одними из первых. С тех пор было разработано достаточно большое количество разнообразных алгоритмов, позволяющих получать коды, обладающие лучшей эффективностью и хорошо структурированной проверочной матрицей.

Рассмотрим один из таких алгоритмов на примере кода стандарта DVB-T2. В этом стандарте используется систематический нерегулярный код повторения-накопления (irregular repeat accumulate — IRA) [7.29]. IRA-коды — это класс LDPC-кодов, разработанный на основе кодов повторения-накопления (repeat accumulate — RA). Отличительными особенностями этого класса кодов является возможность использования алгоритмов кодирования по проверочной матрице, а также возможность использования компактных форм хранения самих проверочных матриц.

Граф Таннера для IRA-кода, как правило, изображают в несколько необычной форме, для этих кодов битовые узлы удобно разбить на информационные узлы (information nodes — IN) и узлы четности (parity nodes — PN). На рис. 19.35 изображен граф Таннера для кода повторения-накопления, битовые узлы обозначены кругами, а проверочные — квадратами; битовые узлы, расположенные слева — информационные, поскольку соответствуют исходным битам, требующим кодирования, а расположенные справа — узлы четности PN, они соответствуют полученным в результате кодирования проверочным битам.

Код DVB-T2 является нерегулярным — степени символьных вершин переменные, в то время как степени кодовых вершин одинаковы (исключение составляет только самый первый кодовый узел, степень которого на единицу меньше).

Для описания распределения степеней информационных вершин IRA-кода вводится ряд распределения (f_1, \dots, f_J) , где f_i обозначает долю информационных узлов, соединенных с i проверочными узлами, $f_i \geq 0$, $\sum_i f_i = 1$. Каждый проверочный узел соединен с a информационными узлами, и полная запись параметров IRA-кода имеет вид $(f_1, \dots, f_J; a)$.

Для графа Таннера систематического IRA-кода, изображенного на рис. 19.35, параметры записываются как $(f_2, \dots, f_J; a)$. Слева расположены K информационных узлов, по середине и справа — M проверочных узлов и узлов четности соответственно. Каждый проверочный узел соединяется ровно с a информационными узлами, это соединение может быть описано при помощи матрицы случайных перестановок $M \times a$ ребер. Соединение узлов четности с проверочными изображено зигзагообразной линией (см. рис. 19.35).

Из структуры рассмотренного графа Таннера видно, что проверочную матрицу кода можно представить в виде двух подматриц $H = [H_u H_p]$.

Структура проверочных матриц LDPC-кода имеет следующий вид, представленный на следующей странице.

Подматрица H_u размерности $M \times K$ — разреженная матрица, обозначенная на рис. 19.35 как матрица перестановок. А подматрица H_p , размерности $M \times M$ — ступенчатая матрица, которой на рис. 19.35 соответствует ломаная линия, соединяющая проверочные узлы с узлами четности.

$$H = \begin{bmatrix} h_{11} & h_{12} & h_{13} & \dots & \dots & \dots & \dots & h_{1K} \\ h_{21} & h_{22} & h_{23} & & & & & \\ h_{31} & h_{32} & h_{33} & & & & & \\ \vdots & & & & H_u & & & \\ \vdots & & & & & & & \\ h_{M1} & & & & & & & h_{MK} \end{bmatrix} \begin{bmatrix} 1 \\ 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{array}{c} \uparrow \\ M \end{array}$$

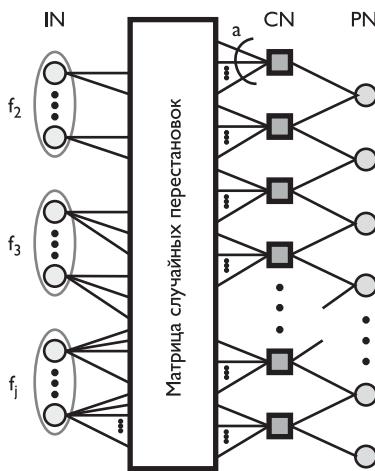


Рис. 19.35. Граф Таниера для IRA LDPC-кода

Как известно из теории, кодирование линейных блоковых кодов осуществляется при помощи генераторной матрицы, которую, в свою очередь, можно получить по проверочной. Генераторная матрица, соответствующая данной проверочной матрице H , имеет вид:

$$G = (I \ H_u^T \ H_p^{-T}).$$

При этом получение матрицы H_u^T не представляет особого труда, а подматрица H_p^{-T} генераторной матрицы G для IRA-кода является верхней треугольной матрицей:

$$H_p^{-T} = \begin{bmatrix} 1 & 1 & \dots & 1 \\ 1 & 1 & \dots & 1 \\ \ddots & & \vdots & \\ 1 & 1 & & \\ 1 & & & \end{bmatrix}. \quad (19.57)$$

Можно показать [7.28], что операция умножения на матрицу (19.57) эквивалента накоплению результата в простейшем аккумуляторе. Таким образом, в общем случае IRA-кодер, состоит из матричного умножителя и аккумулятора [7.28] (рис. 19.36), кодовое слово при этом состоит из информационной части u (входа матричного умножителя) и проверочной части p (выхода аккумулятора).

Заметим также, что получение генераторной матрицы в классическом смысле не требуется — преобразование матрицы перестановок и синтез аккумулятора тривиальны, что позволяет использовать проверочную матрицу как для кодирования, так и для декодирования.

В стандарте DVB-T2 используется аналогичная схема кодирования, однако она имеет ряд особенностей, связанных с тем, что проверочные матрицы хранятся в сжатом виде [7.30].

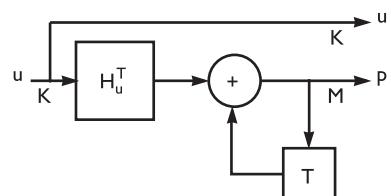


Рис. 19.36. Кодер IRA-кода

Используемые проверочные матрицы LDPC-кода, помимо того, что они описывают IRA-код, имеют некоторую периодичность в структуре, что позволяет существенно упростить их хранение, а также позволяет получить эффективную архитектуру декодеров таких кодов [7.31].

Способ построения матрицы H_u заключается в делении информационных узлов на группы, по Q узлов в каждой, причем все узлы группы должны иметь одинаковую степень. При этом для того чтобы однозначно определить схему соединения всех Q информационных узлов с проверочными узлами, достаточно указать только те проверочные узлы, которые соединены с первым информационным узлом в рассматриваемой группе.

Обозначим как c_1, c_2, \dots, c_{w_l} индексы проверочных узлов, соединенных с первым информационным узлом в группе, тогда индексы проверочных узлов любого i -го информационного узла в группе можно получить, воспользовавшись формулами (19.58):

$$\begin{aligned} [c_1 + (i-1)q] \mod (n-k), \\ [c_2 + (i-1)q] \mod (n-k), \\ \vdots \\ [c_{w_l} + (i-1)q] \mod (n-k), \end{aligned} \tag{19.58}$$

где $q = (n-k)/Q$.

Таким образом, для полного описания структуры матрицы используемого кода необходимо указать проверочные узлы, соединенные с первым информационным узлом в каждой из групп. Код стандарта DVB-T2 предусматривает использование фиксированного размера групп для всех скоростей и размеров блоков — это $Q = 360$ битовых узлов, при этом очевидно, что число групп в матрице будет отличаться для различных скоростей и размеров блоков. В стандарте приводятся таблицы, в которых перечисляются проверочные узлы для первого битового узла каждой из групп для всех скоростей кодирования и используемых блоков передачи данных.

Пример такой таблицы стандарта DVB-T2 для короткого блока ($N = 16\,200$) со скоростью кодирования $2/3$ представляет собой запись матрицы размерности $10\,800 \times 16\,200$ и имеет следующий вид:

0	2084	1613	1548	1286	1460	3196	4297	2481	3369	3451	4620	2622
1	122	1516	3448	2880	1407	1847	3799	3529	373	971	4358	3108
2	259	3399	929	2650	864	3996	3833	107	5287	164	3125	2350
3	342	3529								1	2583	1180
4	4198	2147								2	1542	509
5	1880	4836								3	4418	1005
6	3864	4910								4	5212	5117
7	243	1542								5	2155	2922
8	3011	1436								6	347	2696
9	2167	2512								7	226	4296
11	2835	705								9	3926	1640

10 4606 1003	8 1560 487
12 3426 2365	10 149 2928
13 3848 2474	11 2364 563
14 1360 1743	12 635 688
0 163 2536	13 231 1684
	14 1129 3894

19.6.4. Алгоритмы декодирования низкоплотностных кодов

Рассмотрим несколько алгоритмов декодирования LDPC-кодов — классические, предложенные еще Р. Галлагером, а также еще два более новых алгоритма.

Р. Галлагер в своей работе [7.25] предложил два итеративных алгоритма декодирования низкоплотностных кодов. Первый — алгоритм с распространением доверия (belief propagation, BP). Этот алгоритм обладает максимальной эффективностью среди всех известных алгоритмов декодирования. Доказано, что алгоритм BP достигает максимума правдоподобия, при условии, что проверочная матрица кода не содержит циклов [7.26]. Расплата за столь высокую эффективность — максимальная вычислительная сложность среди всех известных алгоритмов.

Второй алгоритм, предложенный Р. Галлагером, — алгоритм с инверсией бита (bit flip algorithm, BF). Он очень прост в реализации и использует жесткие решения модема относительно принятых битов. Расплата за простоту — низкая эффективность этого алгоритма. Оба алгоритма в настоящее время хорошо исследованы и для них разработано множество модификаций. Также существует и множество альтернативных оригинальных алгоритмов декодирования LDPC-кодов.

Начнем рассмотрение с алгоритма BF [7.32]. Его можно представить в виде основных шагов, выполняемых итеративно:

- инициализация,
- обновление проверочных узлов,
- обновление битовых узлов,
- инверсия битов.

Первый шаг алгоритма — инициализация — выполняется только один раз для каждого кодового слова. Суть ее заключается в вынесении жестких решений относительно принятых битов.

$$x_n = \text{sign}(y_n), \quad n = 1, \dots, N, \quad (19.59)$$

где x_n — жесткие решения модема относительно принятых битов y_n .

Второй шаг — обновление проверочных узлов:

$$\sigma_m = \sum_{i=1}^{d_r} x_{n_i} [\text{mod}2]. \quad (19.60)$$

где x_{n_i} — входящие в проверочный узел m сообщения, σ_m — исходящее из этого проверочного узла сообщение, d_r — количество смежных битовых узлов.

На рис. 19.37а изображено обновление проверочных битов с использованием графа Таннера. Здесь x_n — по существу не что иное, как знак принятого из канала значения, полученный на этапе инициализации. В случае алгоритма с инверсией бита исходящее сообщение σ_m может принимать всего два значения — 0 или 1, что, в свою очередь, показывает, выполняется соответствующая проверка или нет.

Третий шаг алгоритма — обновление битовых узлов:

$$Z_n = \sum_{i=1}^{d_c} \sigma_{m_i}, \quad (19.61)$$

где Z_n — исходящее из битового узла n сообщение, σ_{m_i} — входящие в этот битовый узел сообщения, d_c — количество смежных проверочных узлов.

Графическая интерпретация этого шага представлена на рис. 19.37б. В случае алгоритма с инверсией бита значение исходящего сообщения Z_n равно числу невыполненных проверок для битового узла n . После того как найдены значения Z_n для всех битовых узлов, находится и инвертируется бит, для которого Z_n максимально.

Можно заметить, что алгоритм является очень простым, а следовательно, очень быстрым, однако эффективность этого алгоритма существенно ниже алгоритма ВР. Другим существенным недостатком этого алгоритма является его медленная сходимость, особенно для блоков большой длины. В силу указанных недостатков этот алгоритм практически не используется. Поэтому обратимся к алгоритму с распространением доверия.

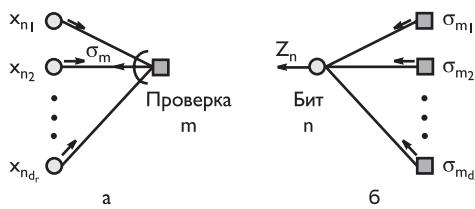


Рис. 19.37. Алгоритм с инверсией бита: а — обновление проверочных узлов, б — обновление битовых узлов

Оригинальный алгоритм распространения доверия, предложенный Галлагером, имеет ряд существенных недостатков, самый существенный из которых — высокая сложность реализации, в связи с чем в таком виде он практически не применяется, поэтому рассмотрим упрощенный, но достаточно близкий по смыслу алгоритм, рекомендованный к использованию стандартом DVB для декодирования используемых в нем кодов.

Алгоритм ВР в трактовке, предложенной в [7.31], можно представить в виде следующих шагов:

- инициализация,
- обновление проверочных узлов,
- обновление битовых узлов,
- вынесение жестких решений.

Последние три шага выполняются итеративно.

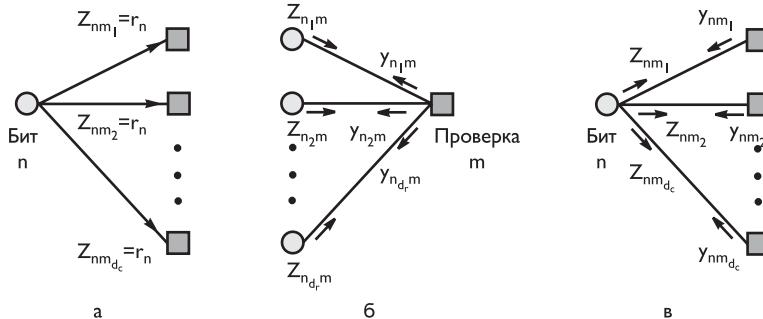


Рис. 19.38. Алгоритм с распространением доверия: а) инициализация, б) обновление проверочных узлов, в) обновление битовых узлов

На рис. 19.38а изображена инициализация алгоритма. Z_{nm_i} — сообщения от битового узла n к соответствующему проверочному узлу m_i , r_n — канальное логарифмическое отношение правдоподобия (log likelihood ratio, LLR) для битового узла n .

Второй шаг алгоритма — обновление проверочных узлов — изображен на рис. 19.38б. $Z_{n_i m}$ — входящие сообщения для проверочного узла m от d_r смежных битовых узлов, $y_{n_i m}$ — исходящие сообщения от узла m к битовым узлам n_i . Исходящие сообщения рассчитываются по следующей формуле:

$$y_{n_i m} = g(Z_{n_1 m}, \dots, Z_{n_{i-1} m}, Z_{n_{i+1} m}, \dots, Z_{n_{d_r} m}), \quad (19.62)$$

где

$$g(a, b) = \text{sign}(a) \times \text{sign}(b) \times \min(|a|, |b|) + \text{LUT}_g(a, b), \quad (19.63)$$

$$\text{LUT}_g(a, b) = \log(1 + e^{-|a+b|}) - \log(1 + e^{-|a-b|}). \quad (19.64)$$

Выражение (19.64) — наиболее трудоемкая операция во всем алгоритме, поэтому обычно функцию $\log(1 + e^{-|x|})$ представляют таблицей для нескольких значений x , при этом вычисление (19.64) сводится к двум обращениям к таблице и нахождению разности полученных значений.

Третий шаг — обновление битовых узлов — изображен на рис. 19.38в. На рисунке y_{nm_i} — входящие сообщения для битового узла n от d_c смежных проверочных узлов, Z_{nm_i} — исходящие сообщения для каждого проверочного узла m_i .

$$Z_{nm_i} = |r_n| - \sum_{j \neq i}^{d_c} y_{nm_j}. \quad (19.65)$$

После обновления всех битовых узлов находятся жесткие оценки принятых битов, которые представляют собой не что иное, как знак Z_n :

$$Z_n = Z_{nm_i} + y_{nm_i}. \quad (19.66)$$

Если все проверки выполняются при подстановке в них жестких оценок принятых битов, это значит, что в результате декодирования найдено допустимое кодовое слово. При этом вычисления прекращаются, а найденное кодовое слово считается результатом декодирования. В противном случае выполняется еще одна итерация обновления проверочных и битовых узлов. В случае если достигну-

то максимальное заданное число итераций декодирования, а допустимое кодовое слово не найдено, вычисления прекращаются, а текущее кодовое слово считается результатом декодирования.

К недостаткам алгоритма ВР относятся высокая сложность реализации, а также необходимость оценки отношения сигнал/шум на входе приемника для расчета канальных логарифмических отношений правдоподобия.

Естественной целью исследователей стал поиск алгоритма декодирования, обладающего столь же высокой эффективностью, как и у алгоритма ВР, но лишенного недостатков последнего. Следствием этого стало появление большого числа субоптимальных алгоритмов декодирования, основанных на алгоритме ВР и являющихся его упрощенными модификациями.

Рассмотрим две распространенные модификации алгоритма ВР — UMP-ВР и UMP-APP [7.33]. Префикс UMP означает критерий оценки достоверности принятых битов — равномерно наиболее мощный (uniformly most powerful). Отличительной особенностью этих алгоритмов является отсутствие необходимости в вычислении отношения сигнал/шум на входе приемника, что позволяет существенно упростить декодирование.

Рассмотрим алгоритм UMP-ВР. Как было сказано выше, этот алгоритм является модификацией алгоритма ВР и отличается от него только шагом обновления проверочных узлов. Вычисление исходящих после проверки сообщений осуществляется по формуле (19.62), однако:

$$g(a, b) = \text{sign}(a) \times \text{sign}(b) \times \min(|a|, |b|), \quad (19.67)$$

т. е. используется только первое слагаемое (19.63), что позволяет вычислять $y_{n i m}$ как минимальный элемент проверки.

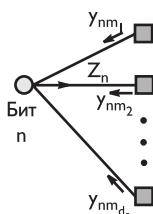


Рис. 19.39. Обновление битовых узлов в алгоритме UMP-APP

Алгоритм UMP-APP можно рассматривать как дальнейшее упрощение алгоритма UMP-ВР. Этот алгоритм отличается от алгоритма UMP-ВР шагом обновления битовых узлов. Здесь вычисляется только одно исходящее сообщение от битового узла n (рис. 19.39), вместо d_c , как это было в ВР и UMP-ВР алгоритмах.

$$Z_n = |r_n| - \sum_{i=1}^{d_c} y_{n i m_j}. \quad (19.68)$$

На рис. 19.40 изображены зависимости вероятности битовой ошибки от отношения сигнал/шум для рассмотренных алгоритмов декодирования. Результаты приведены для кода скорости 3/4 и размерности 64 800 стандарта DVB-T2, число итераций декодирования для всех алгоритмов равно 20.

19.6.5. Оценка сложности алгоритмов декодирования

В качестве оценки вычислительной сложности алгоритма будем рассматривать количество элементарных арифметических операций в алгоритме декодирования. Под элементарными операциями будем понимать сложение (по модулю 2, поскольку поле двоичное), сравнение, умножение, обращение к памяти, а также операцию взятия знака.

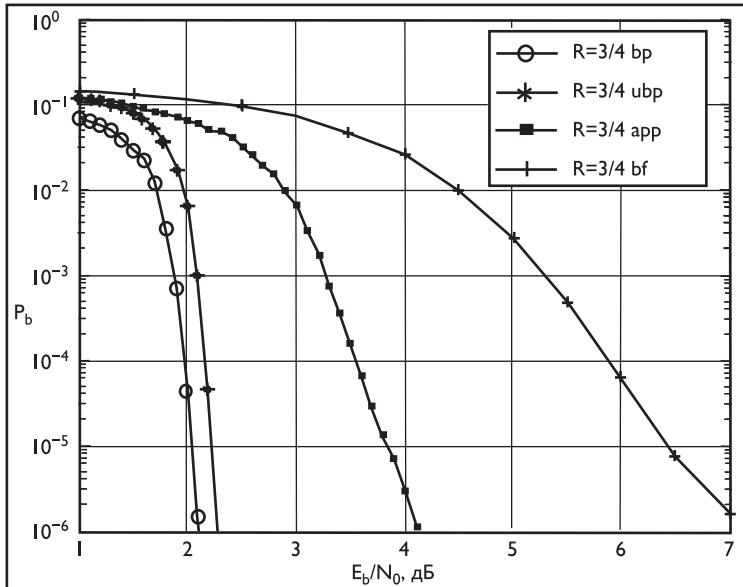


Рис. 19.40. Зависимости вероятности битовой ошибки от отношения сигнал/шум для рассмотренных алгоритмов декодирования кода стандарта DVB-T2 размерности 64 800 и скорости кодирования 3/4

Начнем с алгоритма BF. Вычисление одной проверки требует сложить значения всех d_c проверок для данного бита (выполнить $d_c - 1$ операцию сложения), т. е. для N битов понадобится $N \cdot (d_c - 1)$ сложений. Обновление проверочных узлов требует сложить все d_r битов участвующих в проверке, что для M проверок потребует $M \cdot (d_r - 1)$ операций сложения. Поиск максимального Z_n требует сравнения всех N элементов, что требует $(N - 1)$ операций.

Итого для выполнения алгоритма BF требуется $M(d_r - 1) + Nd_c - 1$ операций для одной итерации обработки блока из N битов.

При анализе сложности алгоритма ВР будем считать, что инициализация выполняется модемом, т. е. на вход декодера поступает последовательность мягких оценок. Тогда начнем со второго шага алгоритма. Вычисление $y_{n,m}$ для i -го бита представляет собой вычисление $d_r - 1$ раз функции $g(a, b)$, которая требует выполнения двух операций взятия знака, двух операций умножения, одну операцию сравнения и две операции обращения к памяти, всего 7 операций, итого $7M(d_r - 1)$. При этом для каждой проверки находится d_r значений $y_{n,m}$, итого для M проверок имеем $7Md_r(d_r - 1)$ операций.

Обновление битов требует нахождения значения Z_{nm_i} для каждой из d_c проверок, входящих в битовый узел n . Вычисление Z_{nm_i} требует выполнения $d_c - 1$ вычислительных операций, итого для N битов $Nd_c(d_c - 1)$ операций.

Вычисление жесткого решения Z_n потребует одной операции сложения и одной операции взятия знака для каждого бита кодового слова, т. е. $2N$ операций.

Вычисления можно несколько упростить, если объединить вычисления на шагах 3 и 4. И сначала вычислять сумму Z_n , что потребует d_c операций для

одного бита, т. е. Nd_c операций для всего кодового слова. А суммы Z_{nm_i} можно находить вычитанием из Z_n соответствующего значения y_{nm_i} , что потребует одной операции сложения для каждой проверки бита n , т. е. d_c операций. Для всего кодового слова получим Nd_c операций. То есть затраты на 3-й и 4-й шаг составят $Nd_c + Nd_c + N = N(2d_c + 1)$, где дополнительные N операций — взятие знака для получения жесткого решения.

На каждой итерации вычисляется синдром, что требует, как было показано выше, $M(d_r - 1)$ операций.

Всего для выполнения алгоритма BP требуется $M(d_r - 1) + 7Md_r(d_r - 1) + N(2d_c + 1)$ операций.

Рассмотрим теперь алгоритм UMP-BP. Ключевое отличие здесь — это второй шаг алгоритма, который позволяет существенно снизить вычислительные затраты на декодирование. Для нахождения $y_{n_i m}$ для i -го бита по прежнему требуется $d_r - 1$ раз найти значения функции $g(a, b)$. Однако за счет отказа от вычисления $LUT_g(a, b)$ нахождение модуля $y_{n_i m}$ сводится к поиску минимального значения среди входящих в проверку битов за исключением i -го. А знак находится как произведение всех знаков битов, входящих в проверку, за исключением i -го. Более того, как нетрудно показать, для расчета всех $y_{n_i m}$ достаточно найти два значения $y_{n_i m}$ — для бита, обладающего минимальным значением Z_{nm_i} , модули всех остальных $y_{n_j m, j \neq i}$ будут одинаковыми, различаться будут только их знаки [7.33]. Можно показать, что поиск двух минимальных элементов в массиве может быть выполнен за $d_r + \lceil \log_2 d_r \rceil - 2$ операций сложения [7.33]. Поиск знака может осуществляться следующим образом: находится произведение знаков всех битов, входящих в проверку, что требует $d_r - 1$ операций, а затем для каждого $y_{n_i m}$ выполняется умножение на знак Z_{nm_i} , т. е. d_r операций для каждого бита, всего получаем $N(d_r + \lceil \log_2 d_r \rceil - 2 + d_r - 1 + d_r) = N(3d_r + \lceil \log_2 d_r \rceil - 3)$ операций.

Таким образом, для выполнения алгоритма UMP-BP требуется $M(d_r - 1) + N(3d_r + \lceil \log_2 d_r \rceil - 3) + N(d_c + 1)$ операций.

Алгоритм UMP-APP дополнительно упрощает и обновление битов, которое, по сути, аналогично нахождению жестких решений в алгоритме BP, что требует Nd_c операций, и еще N операций на взятие знака. То есть для выполнения алгоритма UMP-APP потребуется $M(d_r - 1) + N(3d_r + \lceil \log_2 d_r \rceil - 3) + N(d_c + 1)$ операций. Вычислительная сложность всех рассмотренных алгоритмов сведена в табл. 19.9.

Таблица 19.9. Вычислительные затраты на выполнение одной итерации декодирования блока из N бит для различных алгоритмов

Алгоритм	Число вычислительных операций
BF	$M(d_r - 1) + Nd_c - 1$
UMP-APP	$M(d_r - 1) + N(3d_r + \lceil \log_2 d_r \rceil - 3) + N(d_c + 1)$
UMP-BP	$M(d_r - 1) + N(3d_r + \lceil \log_2 d_r \rceil - 3) + N(2d_c + 1)$
BP	$M(d_r - 1) + 7Md_r(d_r - 1) + N(2d_c + 1)$

ГЛАВА 20

СИСТЕМЫ МОДУЛЯЦИИ И СИГНАЛЬНОГО КОДИРОВАНИЯ ЦИФРОВОЙ ИНФОРМАЦИИ

Основной функцией процесса модуляции дискретной цифровой информации является преобразование этой информации в непрерывный аналоговый сигнал, согласованный с характеристиками канала его передачи. При цифровой полосовой модуляции синусоидальное колебание дискретно с интервалом T_c , называемым цифровым символом, может изменяться по амплитуде, частоте или фазе (или их комбинаций).

Основные методы полосовой модуляции можно разделить на следующие виды:

- амплитудная манипуляция (ASK — **A**mplitude **S**hift **K**eying);
- частотная манипуляция (FSK — **F**requency **S**hift **K**eying);
- фазовая манипуляция (PSK — **P**hase-**S**hift **K**eying);
- квадратурная амплитудная модуляция (QAM — **Q**uadrature **A**mplitude **M**odulation).

Способы приема информации разделяют на два вида:

- когерентный прием, при котором демодулятор использует информацию о фазе несущей;
- некогерентный прием, когда демодулятор работает без знания абсолютной величины фазы входного сигнала.

При идеальном когерентном приеме демодулятор содержит прототипы каждого возможного сигнала, дублирующие весь алфавит передаваемых сигналов по всем параметрам. Такой приемник автоматически подстраивается под фазу входного сигнала, перемножает и интегрирует входной сигнал с каждым прототипом, т. е. реализует корреляцию. Некогерентные системы демодуляции более просты, но обладают существенным недостатком — большей вероятностью ошибки приема информации. Некогерентный прием радиосигнала с фазовой манипуляцией возможен, например, когда используется дифференциальная фазовая манипуляция (DPSK — **D**ifferential **P**SK), при которой в процессе вычисления текущего символа в качестве опорной фазы применяется фаза сигнала предыдущего символа.

В практике передачи цифровой информации чаще всего используется фазовая манипуляция и квадратурная амплитудная модуляция.

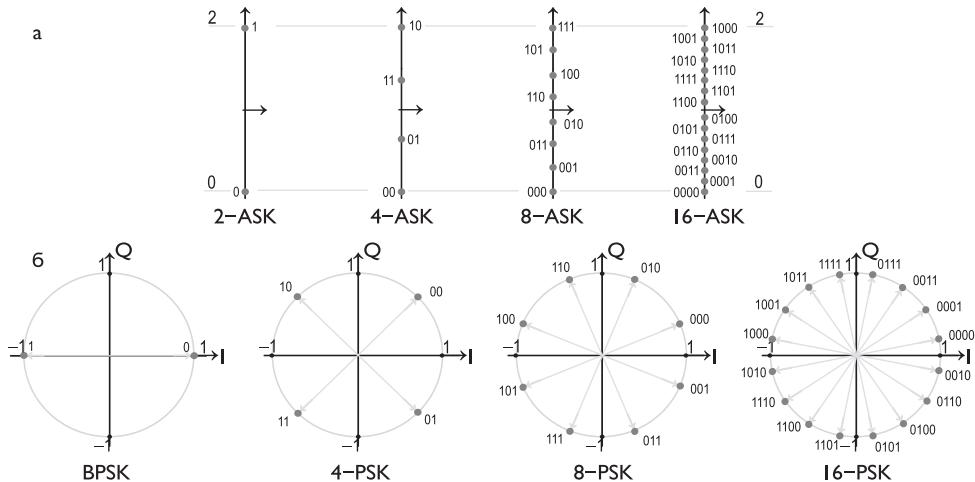


Рис. 20.1. Сигнальные созвездия при амплитудной (а) и фазовой (б) манипуляции

20.1. Созвездия дискретной модуляции

В общем виде аналоговый сигнал, промодулированный дискретной цифровой информацией, можно представить следующей формулой:

$$u(t) = U(t) \cos[2\pi f(t)t + \varphi(t)] = U(t) \operatorname{Re}\{\exp[j(2\pi f(t)t + \varphi(t))]\}. \quad (20.1)$$

В соответствующий момент времени он принимает одно из $K = 2^k$ дискретных значений, множество которых называется сигнальным алфавитом, а k — число битов, которыми может быть определен весь сигнальный алфавит [7.34–7.36].

20.1.1. Одномерные и двумерные созвездия

На рис. 20.1а приведены варианты одномерных созвездий при амплитудной манипуляции (ASK) для двух, четырех, восьми и шестнадцати значений амплитуды $U(k)$ и неизменном значении частоты $\omega_0 = 2\pi f_0$ и фазы $\phi(t) = \phi_0$. Аналогичную структуру одномерных созвездий можно представить при частотной манипуляции (FSK) в радиосигнале.

Рис. 20.1б иллюстрирует варианты двумерных созвездий при фазовой манипуляции (PSK) для двух (BPSK — Binary Phase-Shift Keying), четырех, восьми и шестнадцати значений фазы $\phi(k)$ и неизменном значении амплитуды $U(t) = U_0$.

При построении двумерных созвездий используются две составляющие (син-фазная и квадратурная):

$$I = U_0 \cos[\varphi(k)] \text{ и } Q = U_0 \sin[\varphi(k)], \quad (20.2)$$

поскольку $u(k) = U_0 \cos[\omega_0 t + \varphi(k)] = I \cos \omega_0 t - Q \sin \omega_0 t$.

Сигнальный алфавит характеризуется двумя основными параметрами — скоростью и помехоустойчивостью. Скорость сигнального алфавита определяется в битах и равна $Rk/2$.

Помехоустойчивость ансамбля сигналов обычно характеризуют относительным минимальным расстоянием между двумя ближайшими точками созвездия,

которое определяется отношением:

$$d_{\text{опт}} = d_{\min}/\sigma, \quad (20.3)$$

где для ASK (FSK) и PSK величины d_{\min} соответственно равны

$$d_{\min}^{\text{ASK}} = d_0/(2^k - 1), \quad d_{\min}^{\text{PSK}} = d_0 \sin(\pi/2^k),$$

$d_0 = 2$ — нормированное расстояние между двумя точками созвездий при ASK-2 или PSK-2, а σ — среднеквадратический уровень созвездия, равный

$$\sigma = \sqrt{\frac{1}{N} \sum_{n=0}^N C_n^2},$$

C_n — длина вектора n -й точки созвездия.

Более эффективной является система модуляции, при которой одновременно используются алгоритмы дискретного цифрового преобразования и амплитуды синусоидального сигнала и его фазы.

20.1.2. Сигнальные созвездия квадратурно-амплитудной модуляции

Чаще всего используется одновременная передача двух отдельных k -битовых информационных блоков на двух несущих, находящихся в квадратуре ($\cos \omega_0 t$ и $\sin \omega_0 t$), называемая квадратурно-амплитудной модуляцией (QAM):

$$u(k) = I(k) \cos \omega_0 t - Q(k) \sin \omega_0 t = \text{Re} \{ [I(k) + jQ(k)] \exp(j\omega_0 t) \} = \\ = U(k) \cos[\omega_0 t + \varphi(k)], \quad (20.4)$$

где $U(k) = \sqrt{I^2(k) + Q^2(k)}$, $\varphi(k) = \arctg \frac{Q(k)}{I(k)}$.

На рис. 20.2 приведены сигнальные созвездия квадратурно-амплитудной модуляции — от QAM-4, обычно обозначаемой как QPSK (Quadrature Phase Shift Keying) или 4-PSK, до QAM-64. При QPSK используется два бита на отсчет, при QAM-8 — 3 бита (старшие два бита — в прямоугольниках, а младший бит — в кружках), при QAM-16 — 4 бита (старшие два бита — в прямоугольниках, а два младших — в кружках), при QAM-32 — 5 бит (старшие два бита — в больших прямоугольниках, следующий на ними бит — во вложенных меньших прямоугольниках, а два младших бита — в кружках) и при QAM-64 — 6 бит (старшие два бита — в серединах больших квадратов, следующие за ними два бита — в серединах меньших квадратов, а два младших бита — в кружках).

В табл. 20.1 приведены результаты расчетов минимального расстояния между точками созвездий, среднеквадратического уровня сигналов и межточечного расстояния, нормированного к этому уровню для различных созвездий ASK, PSK и QAM.

Как следует из этой таблицы, при равном количестве точек созвездий системы фазочастотной манипуляции более эффективны, чем системы с амплитудной манипуляцией. Но наибольшей эффективностью обладают системы с квадратурно-амплитудной модуляцией.

Радиосигнал с квадратурно-амплитудной модуляцией демодулируется путем умножения на два несущих колебания, сдвинутых по фазе относительно друг друга на 90° , а результаты умножения фильтруются с использованием ФНЧ.

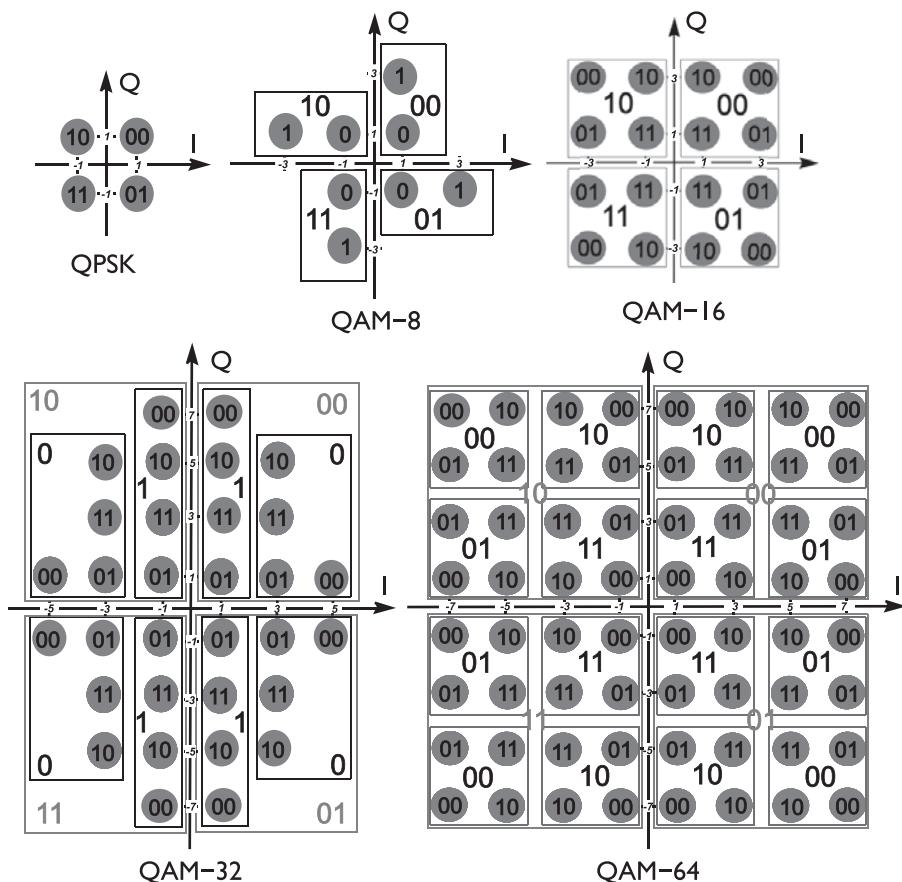


Рис. 20.2. Варианты сигнальных созвездий: QPSK, QAM-8, QAM-16, QAM-32 и QAM-64

Поскольку

$$\begin{aligned}
 u_I(t) &= u_{QAM}(t) \cos \omega_0 t = \{I(t) \cos \omega_0 t - Q(t) \sin \omega_0 t\} \cdot \{\cos \omega_0 t\} = \\
 &= \frac{1}{2} I(t) + \frac{1}{2} I(t)(-\sin 2\omega_0 t) - \frac{1}{2} Q(t) \sin 2\omega_0 t, \\
 u_Q(t) &= u_{QAM}(t) \sin \omega_0 t = \{I(t) \cos \omega_0 t - Q(t) \sin \omega_0 t\} \cdot \{-\sin \omega_0 t\} = \\
 &= \frac{1}{2} Q(t) - \frac{1}{2} I(t) \sin 2\omega_0 t - \frac{1}{2} Q(t) \cos 2\omega_0 t,
 \end{aligned} \tag{20.5}$$

на выходах ФНЧ выделяются сигналы, пропорциональные огибающим синфазной $I(t)$ и квадратурной $Q(t)$ составляющих (рис. 20.3).

Серьезной проблемой является выбор нумерации точек сигнальных созвездий при фазовой и квадратурно-амплитудной модуляциях. Чаще всего используется нумерация сигналов с использованием кода Грея, при котором ближайшие по евклидову расстоянию сигналы отличаются номерами с расстоянием Хемминга, равным 1 (см. рис. 20.1 и 20.2). Вообще говоря, наименьшим евклидовым расстоянием будет между точками сигналов, располагаемых в созвездиях изображенной

Таблица 20.1. Минимальное расстояние между точками созвездий

Вид манипуляции	Минимальное расстояние между точками созвездия d_{\min}	Среднеквадратический уровень созвездия σ	Нормированное расстояние $d_{\text{нн}} = d_{\min}/\sigma$
ASK-4	$\frac{2}{3} \approx 0,6667$	$\sqrt{\frac{14}{3}} \approx 1,24721$	$\sqrt{\frac{2}{7}} \approx 0,5345$
ASK-8	$\frac{2}{7} \approx 0,2857$	$\sqrt{\frac{10}{7}} \approx 1,1952$	$\sqrt{\frac{2}{35}} \approx 0,2390$
ASK-16	$\frac{2}{15} \approx 0,13334$	$\sqrt{\frac{62}{45}} \approx 1,1738$	$\sqrt{\frac{2}{155}} \approx 0,1136$
ASK-32	$\frac{2}{31} \approx 0,0645$	$\sqrt{\frac{42}{31}} \approx 1,1640$	$\sqrt{\frac{2}{651}} \approx 0,0554$
ASK-64	$\frac{2}{63} \approx 0,0317$	$\sqrt{\frac{254}{189}} \approx 1,1593$	$\sqrt{\frac{2}{2667}} \approx 0,0274$
QPSK	$2 \sin \frac{\pi}{4} \approx 1,4142$	1	$2 \sin \frac{\pi}{4} \approx 1,4142$
PSK-8	$2 \sin \frac{\pi}{8} \approx 0,7654$	1	$2 \sin \frac{\pi}{8} \approx 0,7654$
PSK-16	$2 \sin \frac{\pi}{16} \approx 0,3902$	1	$2 \sin \frac{\pi}{16} \approx 0,3902$
PSK-32	$2 \sin \frac{\pi}{32} \approx 0,1960$	1	$2 \sin \frac{\pi}{32} \approx 0,1960$
PSK-64	$2 \sin \frac{\pi}{64} \approx 0,0981$	1	$2 \sin \frac{\pi}{64} \approx 0,0981$
QAM-8	$\sqrt{\frac{2}{5}} \approx 0,6325$	$\sqrt{\frac{7}{10}} \approx 0,8367$	$\frac{2}{\sqrt{7}} \approx 0,7559$
QAM-16	$\frac{\sqrt{2}}{3} \approx 0,4714$	$\frac{\sqrt{5}}{3} \approx 0,7454$	$\sqrt{\frac{2}{5}} \approx 0,6325$
QAM-32	$\frac{\sqrt{2}}{5} \approx 0,2828$	$\frac{\sqrt{11}}{5} \approx 0,6633$	$\sqrt{\frac{2}{11}} \approx 0,4264$
QAM-64	$\frac{\sqrt{2}}{7} \approx 0,2020$	$\sqrt{\frac{3}{7}} \approx 0,6547$	$\sqrt{\frac{2}{21}} \approx 0,3086$

на рис. 20.4 гексагональной решетки. Однако этот алгоритм построения решетки практически не применяется, поскольку в этом случае невозможно пронумеровать точки этих сигналов кодом Грея.

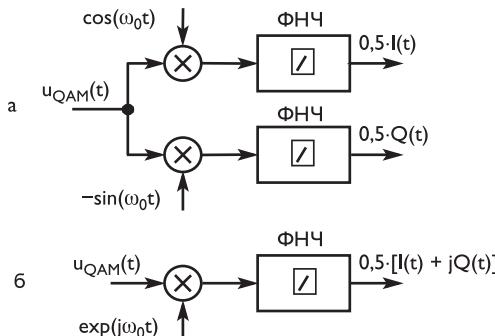


Рис. 20.3. Демодуляция QAM радиосигналов при вещественной (а) и комплексной (б) обработке

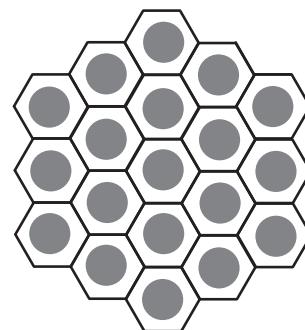


Рис. 20.4. Пример сигнального созвездия с расположением точек в структуре гексагональной решетки

Важнейшим параметром системы цифровой передачи информации является отношение средней мощности сигнала к эффективной мощности шума, обозначаемое обычно как сигнал/шум, S/N или SNR . Одним из критериев, характеризующих качество цифровой передачи, является нормированная версия SNR , определяемая отношением энергии бита информации E_b (равной мощности сигнала S , умноженной на интервал передачи бита T_b — $E_b = S \cdot T_b$) к спектраль-

ной плотности мощности шума N_0 (оцениваемой отношением мощности шума N и ширины полосы используемого радиоканала $W - N_0 = N/W$). Поскольку интервал передачи бита обратно пропорционален скорости передачи информации $T_b = 1/R$, безразмерная величина E_b/N_0 характеризуется отношением S/N , нормированным на ширину полосы и скорость передачи битов [7.11]:

$$\frac{E_b}{N_0} = \frac{S}{N} \frac{W}{R} \quad (20.6)$$

Эта величина определяет качественную меру производительности цифровой связи, чем меньше величина E_b/N_0 , тем эффективнее процесс демодуляции информации при заданной вероятности ошибки. E_b/N_0 — эффективная метрика цифровых систем, позволяющая сравнивать сигналы, обеспечивающие передачу как однобитовой, так и многобитовой цифровой информации.

Если битовая скорость передачи равна пропускной способности канала RC , то, учитывая соотношение Шеннона [7.2] при наличии в канале аддитивного белого гауссова шума

$$C = W \log_2 \left(1 + \frac{S}{N} \right), \quad (20.7)$$

можно модифицировать выражение (20.6) следующим образом:

$$\frac{E_b}{N_0} = \frac{W}{C} \left(2^{\frac{C}{W}} - 1 \right). \quad (20.8)$$

Раскрыв неопределенность этого выражения при $C/W \rightarrow 0$, получим нижнее предельное значение E_b/N_0 :

$$\frac{E_b}{N_0} = \frac{1}{\log_2 e} \approx 0,693 \quad \text{или в децибелах} \quad -\frac{E_b}{N_0} = -1,59 \text{ дБ.} \quad (20.9)$$

С помощью соотношений (20.8) и (20.9) можно получить зависимость C/W от E_b/N_0 , изображенную кривой линией на графике зависимости относительной пропускной способности канала передачи цифровой информации R/W (в бит/с/Гц) от выраженной в децибелах величины E_b/N_0 (рис. 20.5) [7.2]. Этот график характеризует эффективность использования полосы частот канала цифровой связи.

Граница пропускной способности цифровых систем разделяет плоскость на две части, одна из которых характеризует область возможной реализации различных средств связи, когда $R < C$, а вторая — недоступную область при пропускной способности меньшей требуемой битовой скорости передачи $C < R$. Верхняя часть графика при $R/W > 1$ и $R < C$ определяет область ограниченной полосы, при использовании которой могут быть реализованы эффективные системы цифровой манипуляции несущих с передачей на интервале одного символа $T = 1/\Delta W$ (ΔW — полоса пропускания канала) одного или более битов информации. Нижняя часть графика при $R/W < 1$ и $R < C$, обозначенная как область ограниченной мощности, позволяет создавать системы цифровой манипуляции, обеспечивающие передачу на заданном интервале менее одного бита информации (к таким системам, не получившим широкого применения, относятся системы с частотной манипуляцией FSK).

На этом рисунке обозначены точки, характеризующие фазовую манипуляцию несущих при вероятности ошибки $P_b = 10^{-5}$. Очевидно, что эффективность использования полосы частот (в идеальном случае при использовании пря-

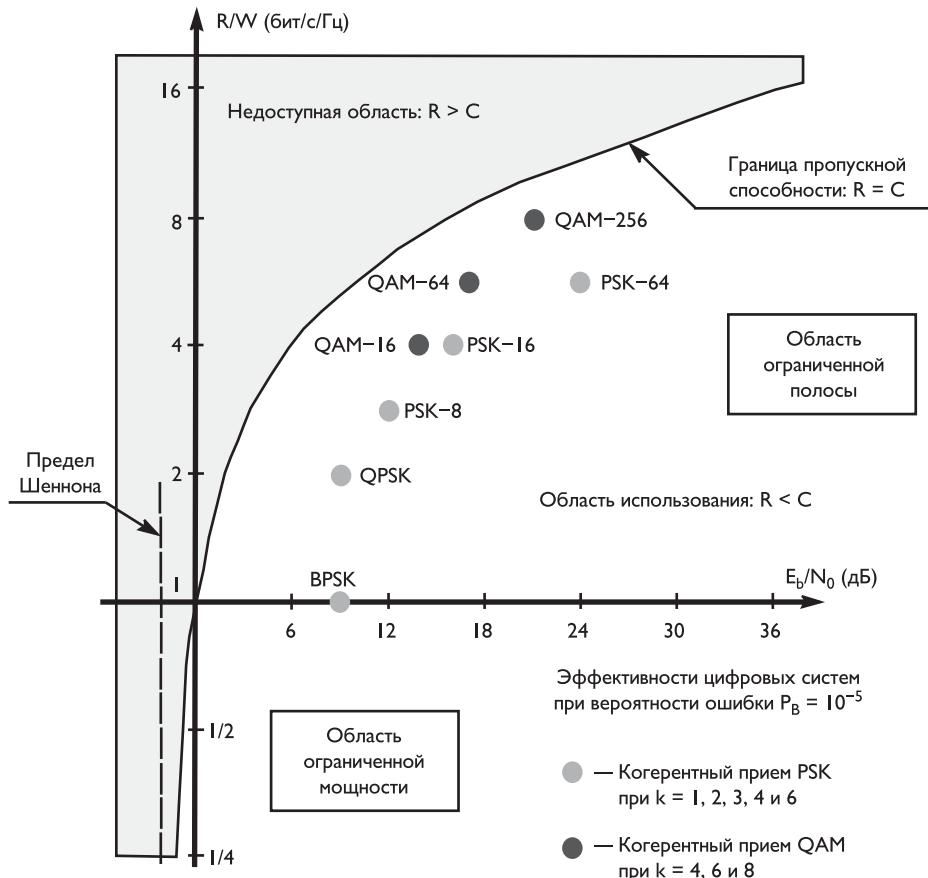


Рис. 20.5. Плоскость, характеризующая эффективность использования полосы частот

угольного фильтра с граничной частотой Котельникова–Найквиста) определяется соотношением

$$R/W = \log_2 K = k \text{ (бит/с/Гц).} \quad (20.10)$$

где K — размер набора символов.

Для реальных каналов эта эффективность несколько снижается в связи с требованием увеличения полосы пропускания для создания условий применения реализуемых фильтров.

Отметим, что положение точек фазовой манипуляции при BPSK ($K = 2$) и при QPSK ($K = 4$) соответствует одинаковым значениям E_b/N_0 . Следовательно, при одинаковом значении E_b/N_0 для схемы QPSK эффективность использования полосы частот равна 2 бит/с/Гц, а для схемы BPSK — только 1 бит/с/Гц. Эта уникальная особенность — следствие того, что QPSK является эффективной комбинацией двух сигналов манипуляции BPSK, передаваемых на ортогональных составляющих несущей.

Кроме того, на рис. 20.5 приведены рабочие точки фазовой манипуляции PSK при $k = 3, 4$ и 6 , а также при квадратурно-амплитудной модуляции QAM при

$k = 4, 6$ и 8 . Видно, что схемы QAM наиболее эффективно используют полосу радиочастот.

Ввиду того, что в цифровой системе передачи за время символьного интервала T_S передается k битов информации, ширина полосы пропускания радиоканала W Гц связана со скоростью передачи символов P_S , скоростью передачи битов R и эффективным временем передачи каждого бита T_b следующими соотношениями:

$$W = \frac{1}{T_s} = R_s = \frac{R}{k} = \frac{1}{kT_b}. \quad (20.11)$$

Для любого цифрового метода манипуляции параметров несущих частот отношение мощности принимаемого сигнала P_r к спектральной плотности мощности шума N_0 можно записать следующим образом:

$$\frac{P_r}{N_0} = \frac{E_b}{N_0} R = \frac{E_s}{N_0} R_s, \quad (20.12)$$

где $E_s = kE_b$.

При демодуляции радиосигнала с фазовой манипуляцией несущей вероятность возникновения символьной ошибки при гауссовском шуме в канале связи при $k > 1$ с достаточной степенью точности описывается следующим приближенным выражением:

$$P_E = 2Q \left[\sqrt{2k \frac{E_b}{N_0}} \sin \left(\frac{\pi}{2^k} \right) \right], \quad (20.13)$$

где $Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^{\infty} \exp(-u^2/2) du$.

В табл. 20.2 приведены результаты расчета требуемого отношения энергии бита информации E_b к спектральной плотности мощности шума N_0 (E_b/N_0) при различных вероятностях возникновения символьных ошибок P_E в случае демодуляции радиосигнала с фазовой манипуляцией, а на рис. 20.6 изображены соответствующие кривые зависимости P_E от E_b/N_0 в дБ.

Таблица 20.2. Результаты расчета отношения энергии бита информации к спектральной плотности шума

Вид манипуляции	$\frac{P_E}{k}$	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}
		4,323	6,790	8,398	9,588	10,530	11,309	11,972	12,550
QPSK	2	7,895	10,362	11,970	13,160	14,102	14,881	15,544	16,122
PSK-8	3	12,500	14,964	16,573	17,763	18,705	19,483	20,147	20,724
PSK-16	4	17,508	19,974	21,583	22,772	23,714	24,493	25,156	25,734
PSK-32	5	22,726	25,192	26,801	27,990	28,932	29,911	30,375	30,952
PSK-64	6	28,074	30,541	32,149	33,339	34,281	35,060	35,723	36,301
PSK-128	7	33,514	35,981	37,589	38,779	39,721	40,500	41,163	41,741
PSK-256	8								

Как следует из приведенных данных, например, при использовании модуляции QPSK в случае, если $E_b/N_0 \approx 4,5$ дБ вероятность появления символьной ошибки равна $P_E = 10^{-2}$, а при E_b/N_0 менее 13 дБ эта вероятность составляет $P_E = 10^{-9}$. При удвоении числа точек звездной диаграммы (увеличении k на единицу) требуемые значения отношений E_b/N_0 увеличиваются примерно на 3,5 дБ.

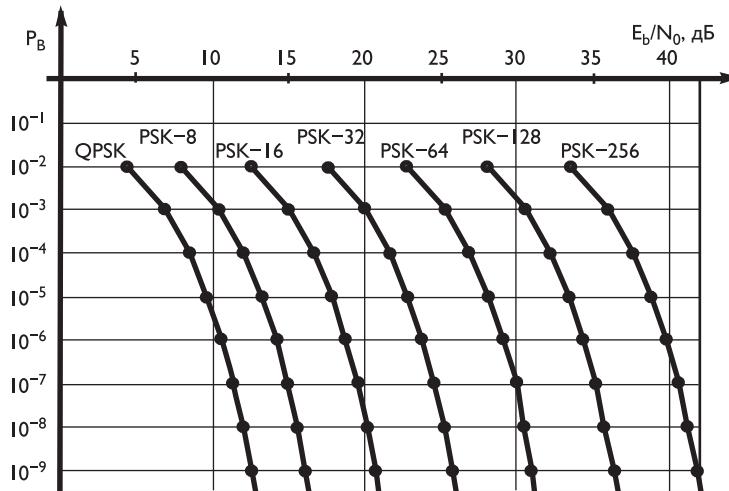


Рис. 20.6. Зависимость вероятности появления символической ошибки P_E от эффективной метрики цифровых систем с фазовой манипуляцией (E_b/N_0 в дБ)

Как следует из рис. 20.6, зависимости $P_E F_k(E_b/N_0)$ характеризуются практически параллельными кривыми при изменении k .

Квадратурно-амплитудная модуляция является логическим продолжением QPSK (см. рис. 20.2), так как сигнал QAM состоит из двух независимых составляющих, сдвинутых по фазе на 90° . Каждый блок исходной цифровой информации из k битов (k — четное чисто) разделяется на два подблока из $k/2$ битов, обеспечивающих модуляцию этих двух синусоидальных составляющих. Эти составляющие в приемном устройстве разделяются с помощью согласующей системы и обрабатываются независимо.

Вероятность битовой ошибки при модуляции QAM, использовании кода Грея, Гауссового канала и приема с помощью согласованных фильтров определяется выражением [7.11]:

$$P_B = \frac{2(k-1)}{k \log_2 k} Q \left[\sqrt{2 \left(\frac{3 \log_2 k}{k^2 - 1} \right) \frac{E_b}{N_0}} \right], \quad (20.14)$$

где $Q(x)$ определено в формуле (20.13).

В табл. 20.3 приведены результаты расчета требуемого отношения E_b/N_0 при различных вероятностях возникновения символических ошибок P_E в случае демодуляции радиосигнала с квадратурно-амплитудной модуляцией, а на рис. 20.7 изображены соответствующие кривые зависимости P_E от E_b/N_0 в дБ.

В этой же таблице повторены результаты расчета при использовании модуляции QPSK. Из сравнения данных табл. 20.2 и табл. 20.3 очевидна существенно большая эффективность модуляции QAM по сравнению с PSK, причем чем больше используемых точек звездной диаграммы, тем больше различие в эффективности этих систем. Так, если при $k = 4$ различие величин E_b/N_0 составляет около 4 дБ, то при $k = 8$ это различие увеличивается более чем на 20 дБ.

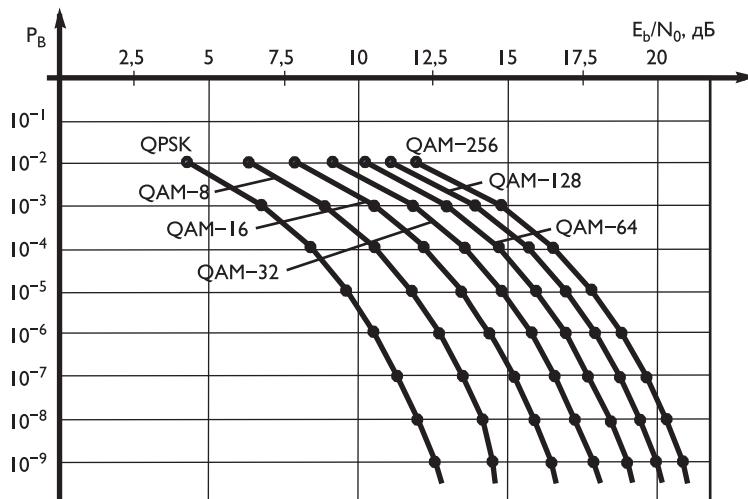


Рис. 20.7. Зависимости эффективной метрики цифровых систем с квадратурно-амплитудной модуляцией (E_b/N_0 в дБ) от вероятности появления символической ошибки P_b

Таблица 20.3. Результаты расчета отношения энергии бита информации к спектральной плотности шума при различных вероятностях возникновения символьных ошибок

Вид манипуляции	$\frac{P_E}{k}$	10 ⁻²	10 ⁻³	10 ⁻⁴	10 ⁻⁵	10 ⁻⁶	10 ⁻⁷	10 ⁻⁸	10 ⁻⁹
		4,323	6,790	8,398	9,588	10,530	11,309	11,972	12,550
QPSK	2	4,323	6,790	8,398	9,588	10,530	11,309	11,972	12,550
QAM-8	3	6,334	8,901	10,555	11,768	12,725	13,514	14,185	14,768
QAM-16	4	7,882	10,552	12,205	13,435	14,402	15,198	15,874	16,461
QAM-32	5	9,144	11,840	13,546	14,788	15,763	16,564	17,244	17,834
QAM-64	6	10,211	12,953	14,677	15,929	16,910	17,715	18,398	18,990
QAM-128	7	11,136	13,917	15,656	16,916	17,902	18,711	19,396	19,990
QAM-256	8	11,954	14,768	16,520	17,787	18,777	19,589	20,276	20,872

Что же касается кривых, изображенных на рис. 20.7, то следует заметить, что они по форме весьма близки, с ростом k они сближаются.

20.1.3. Фильтрация сигналов в передающих и приемных устройствах

В модуляторе практически любого передающего устройства осуществляется фильтрация сигнала с целью ограничения его спектра и исключения влияния составляющих сигнала на иные системы связи в условиях ограниченного частотного ресурса. Фильтрация сигнала применяется также и в приемном устройстве для устранения влияния внеполосных помех и максимизации отношения сигнал/шум.

В системах цифровой связи так называемые baseband-фильтры реализуются в цифровом виде.

Чем компактнее спектр передачи сигналов, тем выше скорость передачи данных. Максимальное сужение требуемой полосы теоретически реализуется с ис-

пользованием фильтра, граничная частота которого соответствует частоте Котельникова–Найквиста, равной $f_{\text{гр}} = 1/2T$, где T — интервал следования отсчетов сигнала. С одной стороны, такой фильтр практически нереализуем, а с другой — использование фильтра с характеристикой, близкой к идеальной, приводит к тому, что длительность импульсных откликов достаточно больших амплитуд оказывается очень большой. Эти «хвосты» импульсных откликов

$$h(t) \approx \text{sinc}(\pi t/T)$$

не вносят межсимвольную интерференцию только в том случае, если выборка производится точно в соответствующий момент времени через интервал T . Даже небольшие ошибки в синхронизации могут привести к межсимвольной интерференции.

Таким образом, применение подобных фильтров хотя и реализует компактный спектр сигнала и позволяет наиболее эффективно использовать полосу частот, но при этом воспроизведение импульсной последовательности оказывается весьма чувствительным к ошибкам синхронизации, приводящим к межсимвольной интерференции.

Эти дефекты фильтрации существенно могут быть уменьшены при использовании класса фильтров, передаточная функция которых, называемая *«проподнятым косинусом»*, определяется следующим выражением:

$$H(f) = \begin{cases} 1, & |f/f_{\text{гр}}| \leq 1 - \alpha, \\ \cos^2 \frac{\pi}{4} \frac{|f/f_{\text{гр}}| - 1 + \alpha}{\alpha}, & 1 - \alpha < |f/f_{\text{гр}}| \leq 1 + \alpha, \\ 0, & |f/f_{\text{гр}}| > 1 + \alpha, \end{cases} \quad (20.15)$$

где $f_{\text{гр}} = 1/2T$, α — коэффициент увеличения полосы фильтра и сглаживания его частотной характеристики.

На рис. 20.8а изображены характеристики фильтров для пяти значений коэффициентов α . Чем меньше коэффициент α , тем меньше полоса пропускания фильтра, но больше дрожание фронтов сигнала (джиттер), что приводит трудностям синхронизации. При $\alpha = 0$ форма характеристики соответствует идеальному фильтру с минимальной шириной полосы пропускания по Котельникову–Найквисту. Следует отметить, что при $\alpha = 1$ избыток полосы увеличивается на 100% и граничная частота фильтра становится равной $2f_{\text{гр}}$.

Нормированный импульсный отклик, соответствующий функции (20.15), определяется как:

$$h(t) = \text{sinc}(\pi t/T) \cdot \frac{\cos(\pi \alpha t/T)}{1 - (2\alpha t/T)^2}. \quad (20.16)$$

Эти импульсные отклики для пяти значений коэффициента приведены на рис. 20.8б.

Как следует из этого графика, при $\alpha = 0,5$ выбросы импульсного отклика практически ограничены на интервале до $t/T < 3$, а при $\alpha > 0,75$ — до $t/T < 2$. На практике часто используется значение $\alpha = 0,35$.

Таким образом, сглаживание спектра передаваемых цифровых сигналов обеспечивает снижение восприимчивости к ошибкам синхронизации принимаемой информации за счет увеличения полосы пропускания канала связи.

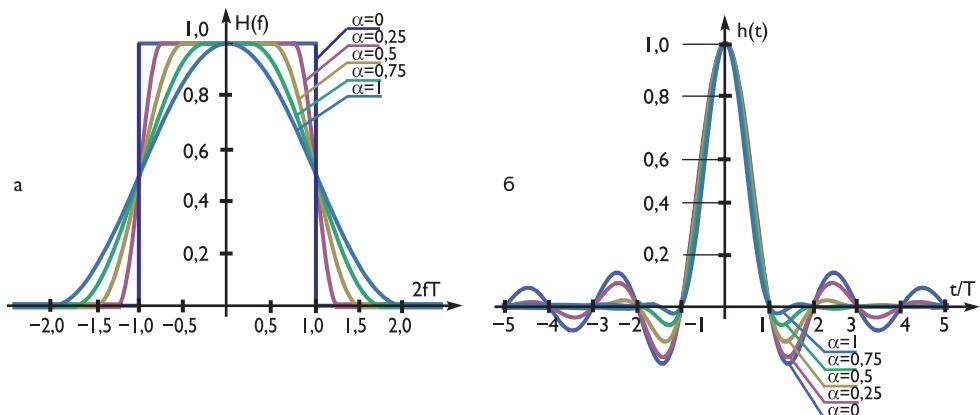


Рис. 20.8. Частотные характеристики (а) и импульсные отклики (б) фильтров типа «поднятого косинуса»

20.2. Система многочастотной модуляции

Одним из эффективных способов передачи цифровой информации является использование модуляции цифровой информации с применением частотно разделенных ортогональных несущих. Этот многочастотный метод передачи называется OFDM (Orthogonal Frequency Division Multiplex — ортогональное частотное разделение мультиплекса).

Этот способ позволяет эффективно использовать спектр выделенной полосы частот, а также бороться с неблагоприятными условиями воздействия помех в канале связи, к которым наряду с гауссовским шумом относятся замирание сигнала, интерференция, многолучевое распространение сигналов.

OFDM-сигнал может рассматриваться как множество медленно модулируемых узкополосных сигналов, а не как один быстро модулируемый широкополосный сигнал. Низкая символьная скорость делает возможным использование защитного интервала между символами, что позволяет справляться с временным рассеянием и устранять межсимвольные искажения.

Принцип OFDM-модуляции заключается в следующем [7.37–7.39]. В полосе канала связи передается множество несущих, каждая из которых модулируется, например, с использованием QAM-модуляции, частью общего цифрового потока. До преобразования спектра такого сигнала цифровой поток разбивается на последовательности, каждая из которых соответствует передаче kR_a битов информации, где R_a — число активных несущих, k — коэффициент используемой QAM-модуляции (или число битов информации, передаваемой на каждой активной несущей). Длительность T_0 интервала, на котором передаются все указанные kR_a битов информации, определяет минимальную частоту несущей $f_U = 1/T_U$ и интервал между несущими, т. е. частотный интервал $(R_a + R_n)f_U$, где R_n — число пассивных несущих, характеризует групповой спектр мощности радиосигнала.

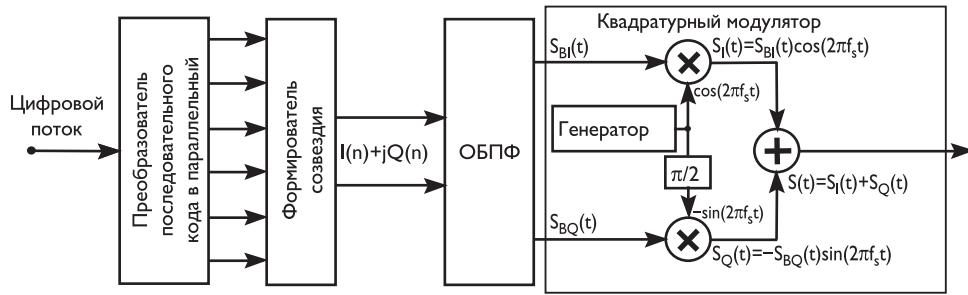


Рис. 20.9. Структурная схема идеального OFDM-модулятора

20.2.1. Структурные схемы OFDM-модуляторов и демодуляторов

На рис. 20.9 приведена структурная схема идеального OFDM-модулятора. Цифровой поток поступает на вход преобразователя последовательной информации в параллельную. На выходе этого преобразователя формируется код, состоящий из k битов и соответствующий используемой QAM-модуляции несущих ($k = 2$ при QPSK, $k = 4$ при QAM-16, $k = 6$ при QAM-64 и т. д.).

Последовательно каждые k битов преобразуются в параллельный код, который подается на формирователь созвездия, преобразующий этот код в значения соответствующих векторов звездной диаграммы, как показано, например, на рис. 20.2. Поток битов трансформируется в формирователе созвездия в поток I и Q сигналов:

$$S(n) = I(n) + jQ(n), \quad 1 \leq n \leq 2N + 1, \quad (20.17)$$

где $2N + 1 = R_a + R_n$.

Блок обратного быстрого преобразования Фурье (ОБПФ) преобразует последовательности (20.17) во временную комплексную функцию:

$$S_B(t) = \sum_{n=1}^{2N+1} [I(n) + jQ(n)] \exp(2\pi j n f_U t) = S_{BI}(t) + jS_{BQ}(t), \quad (20.18)$$

где

$$S_{BI}(t) = \sum_{n=1}^{2N+1} [I(n) \cdot \cos(2\pi n f_U t) - Q(n) \cdot \sin(2\pi n f_U t)] \quad \text{—синфазная составляющая сигнала,}$$

$$S_{BQ}(t) = \sum_{n=1}^{2N+1} [I(n) \cdot \sin(2\pi n f_U t) + Q(n) \cdot \cos(2\pi n f_U t)] \quad \text{—квадратурная составляющая сигнала}$$

(преобразование синфазной составляющей по Гильберту).

Связь между частотой f_U и количеством несущих частот $2N + 1$ сигнала (20.18) в используемой полосе частот можно представить графиком, приведенным на рис. 20.10а, на котором изображены амплитуды отсчетов составляющих частот nf_0 ($n = 1, 2, \dots, 2N + 1$).

В соответствии со схемой квадратурного модулятора эти два сигнала $S_{BI}(t)$ и $S_{BQ}(t)$ перемножаются соответственно на сдвинутые по фазе на 90° синусоиды

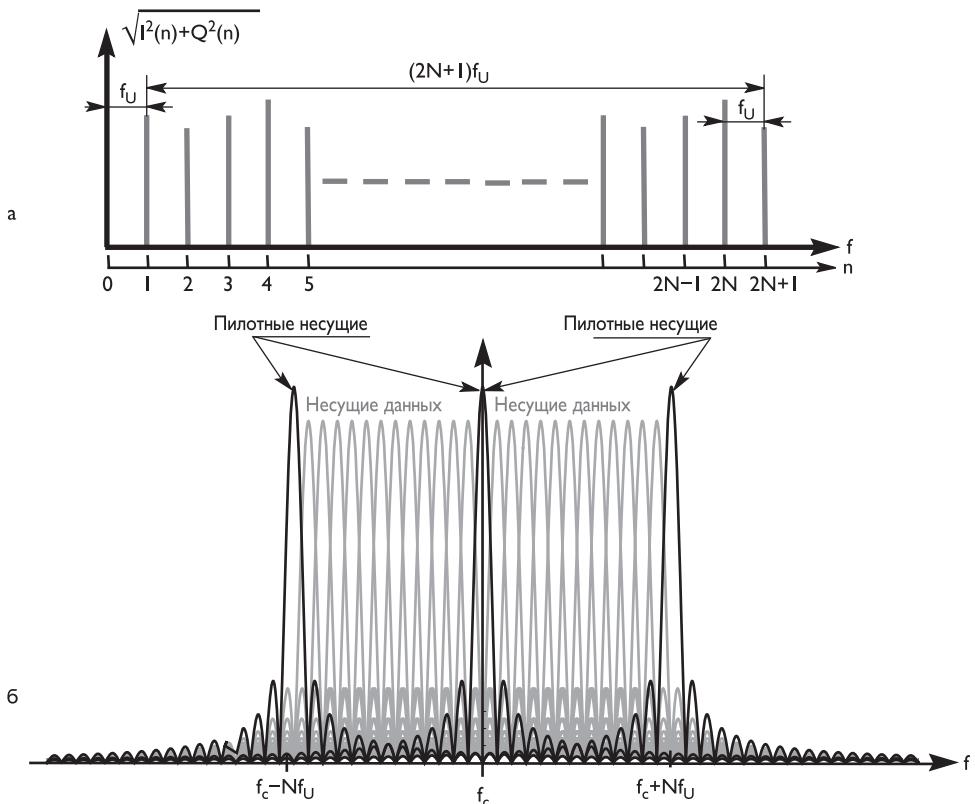


Рис. 20.10. График спектральных составляющих ОБПФ (а) и групповой спектр несущих частот (б)

дальние сигналы. На выходах перемножителей формируются две составляющие:

$$S_I(t) = S_{BI}(t) \cdot \cos(2\pi f_s t) = \sum_{n=1}^{2N+1} \left\{ \frac{1}{2} I(n) \cdot [\cos 2\pi(nf_U + f_s) + \cos 2\pi(nf_U - f_s)] - \frac{1}{2} Q(n) \cdot [\sin 2\pi(nf_U + f_s) + \sin 2\pi(nf_U - f_s)] \right\},$$

$$S_Q(t) = -S_{BQ}(t) \cdot \sin(2\pi f_s t) = \sum_{n=1}^{2N+1} \left\{ \frac{1}{2} I(n) \cdot [\cos 2\pi(nf_U + f_s) - \cos 2\pi(nf_U - f_s)] - \frac{1}{2} Q(n) \cdot [\sin 2\pi(nf_U + f_s) - \sin 2\pi(nf_U - f_s)] \right\}.$$

В результате суммирования этих двух составляющих окончательно формируется выходной сигнал OFDM-модулятора. При этом разностные частоты $(nf_0 - f_s)$ взаимно исключаются.

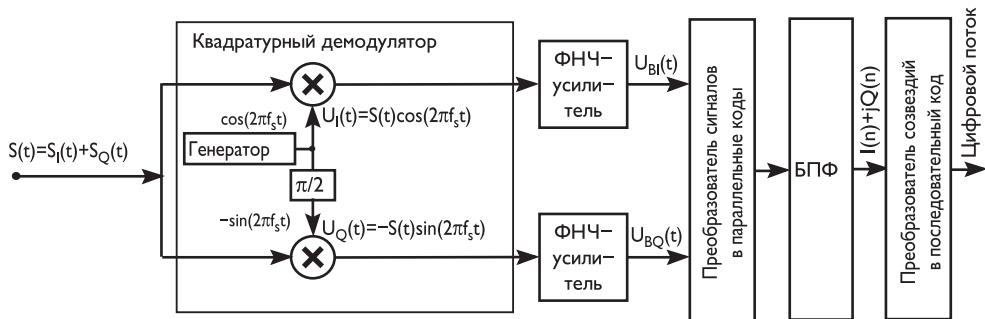


Рис. 20.11. Структурная схема идеального демодулятора

Если частота генератора f_s выбрана равной $f_s = f_c + (N + 1) \cdot f_U$, где f_c — центральная частота радиоканала, то выходной сигнал определяется следующим соотношением:

$$S(t) = \sum_{n=1}^{2N+1} \{I(n) \cdot \cos[2\pi(f_s + nf_U)t] - Q(n) \cdot \sin[2\pi(f_s + nf_U)t]\} = \\ = \sum_{n=-N}^N \{I(n) \cdot \cos[2\pi(f_c + nf_U)t] - Q(n) \cdot \sin[2\pi(f_c + nf_U)t]\}. \quad (20.19)$$

На рис. 20.10б изображена структура группового спектра несущих частот OFDM-сигнала. Здесь условно показаны составляющие синусоидальные сигналы, промодулированные соответствующими данными созвездий дискретной QAM-модуляции. В составе сигнала имеются также специальные пилотные несущие, на которых передается информация о параметрах системы. Эти пилотные несущие используются также для обеспечения устойчивой синхронизации и коррекции характеристик в демодуляторе.

Таким образом, в системах с OFDM модуляцией передаваемая цифровая информация разделена на большое число низкоскоростных подканалов, длительность тактового интервала передачи каждой несущей весьма велика. Такое построение системы преобразует широкополосный канал с одной несущей в большое число независимых узкополосных каналов с частотным разделением, что упрощает коррекцию параметров затухающего сигнала. Более того, ряд групп несущих может быть полностью подавлен при приеме, если дополнительно вводится корректирующее кодирование данных в сочетании с временным и частотным перемежением. При таком построении системы модуляцию часто называют COFDM (Coded Orthogonal Frequency Division Multiplexing — кодированное ортогональное частотное мультиплексирование, разновидность технологии OFDM, сочетающая канальное кодирование и OFDM).

На рис. 20.11 приведена структурная схема идеального демодулятора, содержащего квадратурный демодулятор и ряд преобразователей, формирующих передаваемую цифровую информацию.

В квадратурном демодуляторе входной сигнал $S(t)$ перемножается с двумя сдвинутыми на 90° синусоидальными сигналами опорной частоты f_s . На двух

выходах выделяются сигналы:

$$U_I(t) = S(t) \cdot \cos(2\pi f_s t) = \sum_{n=1}^{2N+1} \left\{ \frac{1}{2} I(n) \cdot [\cos 2\pi n f_U t + \cos 2\pi(2f_s + n f_U) t] - \frac{1}{2} Q(n) \cdot [\sin 2\pi n f_U t + \sin 2\pi(2f_s + n f_U) t] \right\},$$

$$U_Q(t) = S(t) \cdot [-\sin(2\pi f_s t)] = \sum_{n=1}^{2N+1} \left\{ \left(\frac{1}{2} I(n) \cdot [\sin 2\pi n f_U t - \sin 2\pi(2f_s + n f_U) t] + \frac{1}{2} Q(n) \cdot [\cos 2\pi n f_U t - \cos 2\pi(2f_s + n f_U) t] \right) \right\}.$$

Эти два сигнала поступают соответственно на два фильтра нижних частот и усиливаются в два раза. На входы преобразователя в параллельные коды при этом подаются два сигнала:

$$U_{BI}(t) = \sum_{n=1}^{2N+1} [I(n) \cdot \cos(2\pi n f_U t) - Q(n) \cdot \sin(2\pi n f_U t)], \quad (20.20)$$

$$U_{BQ}(t) = \sum_{n=1}^{2N+1} [I(n) \cdot \sin(2\pi n f_U t) + Q(n) \cdot \cos(2\pi n f_U t)]. \quad (20.21)$$

Таким образом, на вход схемы быстрого преобразования Фурье подается сигнал, который может быть представлен в следующем виде:

$$U_B(t) = U_{BI}(t) + jU_{BQ}(t) = \sum_{n=1}^{2N+1} [I(n) + jQ(n)] \exp(2\pi j n f_U t). \quad (20.22)$$

Очевидно: после преобразования Фурье формируется последовательность, определяющая векторы звездной диаграммы:

$$S(n) = I(n) + jQ(n), \quad 1 \leq n \leq 2N + 1, \quad (20.23)$$

которые выходным преобразователем преобразуются в передаваемую цифровую последовательность.

20.2.2. Защитный интервал, устранение интерференции между несущими частотами

Ортогональность субканалов при разделении их в демодуляторе с применением БПФ может быть реализована только при отсутствии интерференции между несущими частотами, что на практике не выполняется из-за ограничения спектра и многолучевого распространения сигналов. Устранение влияния этих искажений осуществляется путем дополнения активного интервала сигнала так называемым защитным интервалом, как показано на рис. 20.12а.

Активный интервал непрерывного символа циклически расширяется во временной области путем введения защитного интервала, содержащего часть сигнала активного интервала. Учитывая, что на интервале T_U располагается целое число периодов синусоидальных колебаний частот $n f_U$, где $n = 1, 2, \dots, 2N + 1$,

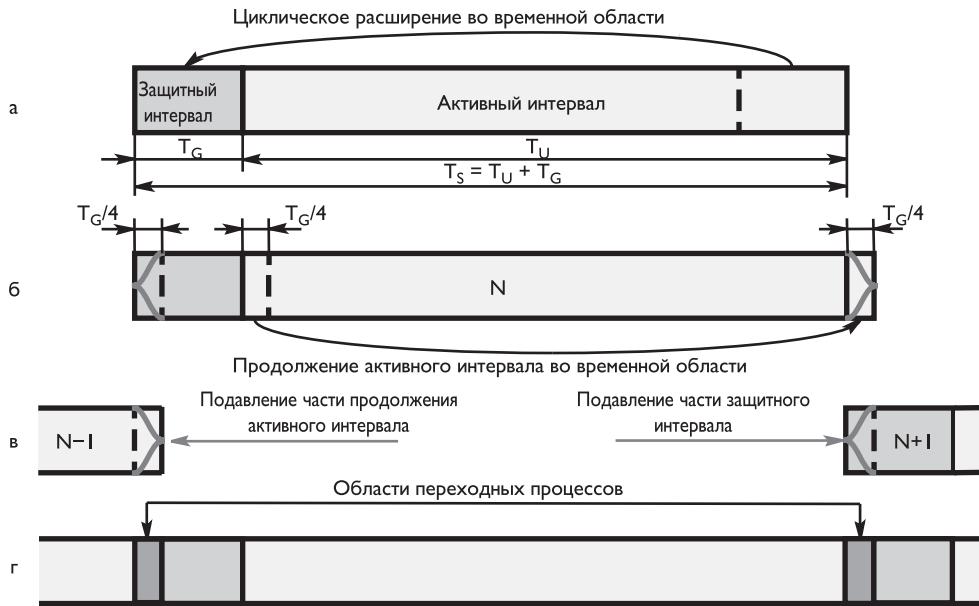


Рис. 20.12. К формированию символов сигнала OFDM

переход от защитного интервала длительностью T_G к активному интервалу не нарушает непрерывности сигнала. Общая длительность символа сигнала OFDM оказывается равной $T_S = T_U + T_G$.

При формировании последовательности таких символов сигнала OFDM на переходах от активного интервала предыдущего символа к защитному интервалу последующего символа возникают разрывы функций, которые в ограниченном по полосе пропускания канале преобразуются в переходные процессы, искажающие сигналы защитных интервалов. Для того чтобы уменьшить влияние этих переходных процессов, формируется продолжение активного интервала во временной области на некоторый интервал, по длительности равный части защитного интервала, например величине $T_G/4$, как показано на рис. 20.12б.

Затем участки сформированных продолжений активных интервалов перемножаются на функции $u(t_1) = \cos^2 \frac{\pi}{2} \frac{t_1}{T_G/4}$, где $t_1 = 0$ соответствует окончанию активного интервала, $t_1 \leq T_G/4$, а начальные участки защитных интервалов перемножаются на функции $u(t_2) = \cos^2 \frac{\pi}{2} \left(1 - \frac{t_2}{T_G/4}\right)$, где $t_2 = 0$ соответствует началу защитного интервала, $t_2 \leq T_G/4$, как отмечено на рис. 20.12б и в. Непрерывный сигнал OFDM формируется путем суммирования символов рис. 20.12б и в. Приведенные на рис. 20.12г области переходных процессов реализуют сигналы, плавно переходящие от активных интервалов предыдущих символов к защитным интервалам текущих символов OFDM-сигналов.

Рациональный выбор длительности защитного интервала позволяет устранить помехи, вызываемые эхосигналами, если их задержка относительно основного сигнала не превышает, скажем, половины длительности защитного интервала.

На рис. 20.13а–в изображены основной сигнал и два эхо-сигнала, задержка каждого из которых не превышает длительности защитного интервала. Сум-

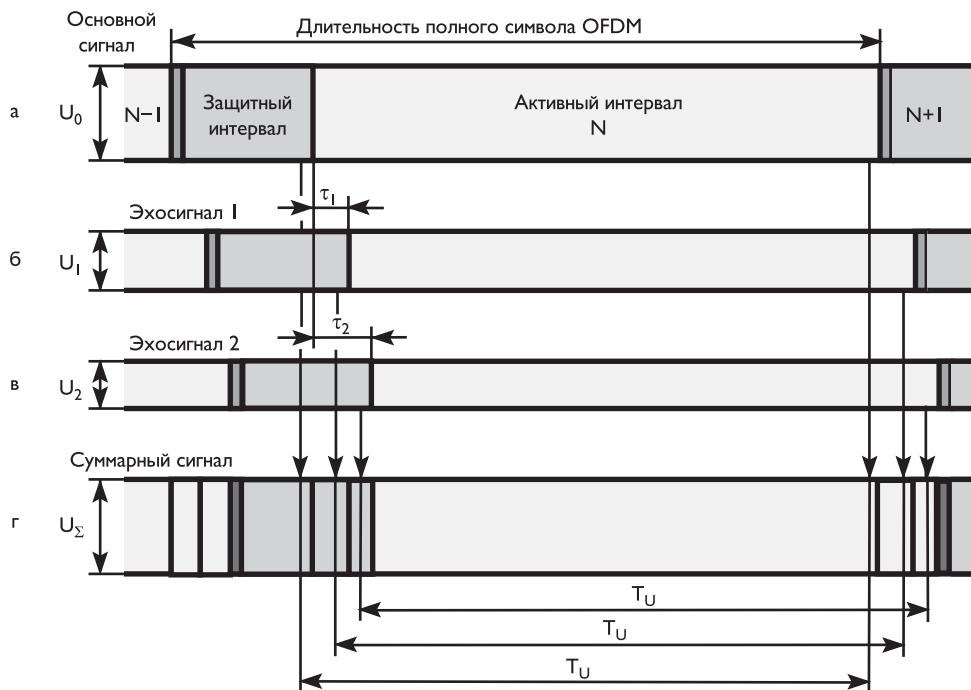


Рис. 20.13. К влиянию эхо-сигналов при демодуляции OFDM сигналов

марный сигнал рис. 20.13г, в котором время T_G выбрано большим максимально возможного времени задержки распространения эхосигналов, межсимвольная интерференция не возникает. Ввиду того, что введение защитного интервала снижает пропускную способность канала связи, желательно, чтобы его длительность была выбрана наименее возможной.

Поскольку сигнал OFDM представляет собой последовательность множества независимых синусоидальных сигналов, согласно центральной предельной теореме теории вероятностей распределение его уровней приближается к закону Гаусса. При этом вероятность того, что превышение пиковой мощности над эффективной мощностью составляет 9,6 dB, равна 0,1%, а что это превышение равно 12 dB — менее 0,01%.

20.2.3. Иерархические режимы OFDM-модуляции

Важным фактором повышения эффективности использования OFDM-сигналов является использование так называемых *иерархических режимов модуляции*. Для обеспечения приоритетного декодирования наиболее важной информации используют как обычные равномерные, так и неравномерные сигнальные созвездия, чаще всего с различными расстояниями между квадрантами. Это позволяет реализовывать модуляцию составляющих OFDM-сигнала различными транспортными потоками, имеющими различные приоритеты.

Примером иерархической модуляции, обозначаемой MR16-QAM [7.41], является равномерное сигнальное созвездие, изображенное на рис. 20.14а. В интерва-

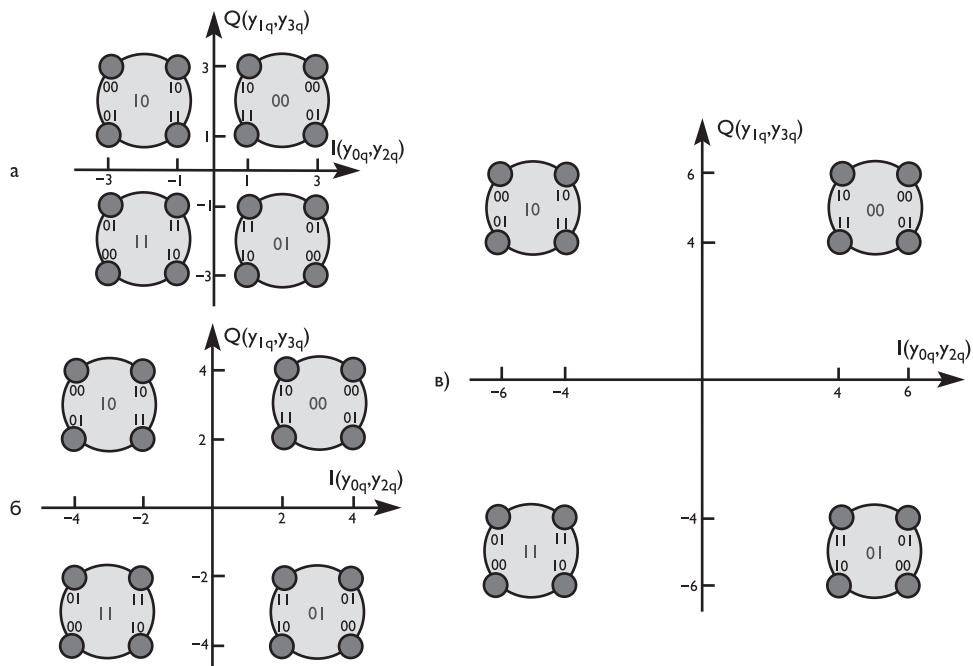


Рис. 20.14. Сигнальные созвездия MR16-QAM с иерархической модуляцией при: $\alpha = 1$ (а), $\alpha = 2$ (б), $\alpha = 4$ (в)

ле q -го активного символа четыре бита информации разделяются на две последовательности: y_{0q} , y_{2q} и y_{1q} , y_{3q} . Первая из этих последовательностей поступает на синфазный, а вторая — на квадратурный каналы модулятора. Биты y_{0q} и y_{1q} характеризуют наиболее важную информацию и определяют номер квадранта, их значение, обозначенное двоичными кодами в центрах кругов, одинаково для всех точек квадранта. Менее приоритетные биты — y_{2q} и y_{3q} определяют точки в квадранте, и их разделение оказывается менее помехозащищенным.

Еще более защищенной от различного рода ошибок является передача приоритетной информации при использовании неравномерных сигнальных созвездий, как показано на рис. 20.14б и в. Неравномерность сигнального созвездия обозначается индексом α :

- $\alpha = 1$ определяет иерархическую модуляцию с равномерным сигнальным созвездием;
- $\alpha = 2$ характеризует неравномерное сигнальное созвездие, при котором расстояние между двумя ближайшими точками созвездий в смежных квадрантах в два раза больше расстояний между точками в каждом из квадрантов;
- $\alpha = 4$ определяет неравномерное сигнальное созвездие, при котором расстояние между двумя ближайшими точками созвездий в смежных квадрантах в четыре раза больше расстояний между точками в каждом из квадрантов.

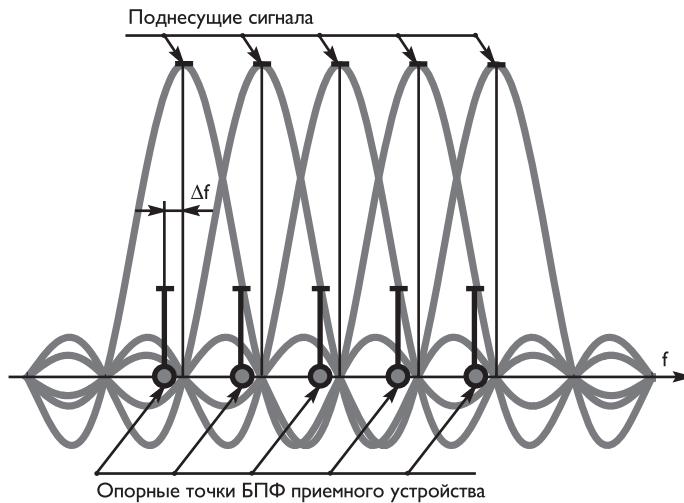


Рис. 20.15. Несогласованное дискретное преобразование

20.3. Алгоритмы синхронизации OFDM-сигналов в приемных устройствах

Важнейшей проблемой реализации систем с использованием OFDM-модуляции является слежение за подстройкой частоты дискретизации сигналов при выполнении прямого быстрого преобразования Фурье (БПФ) в демодуляторе.

В силу ряда дестабилизирующих факторов номинальные частоты спектральных составляющих OFDM-сигнала могут не совпадать с соответствующими частотами гармоник БПФ, как показано на рис. 20.15.

Дискретизация принимаемого сигнала с частотой, отличающейся от номинальной величины, является причиной снижения точности восстановления модулированных символов, передаваемых на каждой поднесущей, и, кроме того, вызывает снижение отношения сигнал-шум.

Оценка величины частотного смещения Δf осуществляется с применением двух основных способов:

- автокорреляционной (некогерентной) демодуляции с использованием заложенной в OFDM-сигнал избыточности в виде составляющих защитного интервала;
- когерентной демодуляции с применением эталонного сигнала, содержащего только пилотные составляющие (см. рис. 20.10).

20.3.1. Методы демодуляции OFDM-сигнала

Автокорреляционный метод анализа основан на реализации согласованного фильтра, использующего совпадающие структуры (циклический префикс) сигнала, расположенные в защитном и активном интервалах передаваемого символа.

Простейший вариант построения такого фильтра приведен на рис. 20.16. Для этой цели используется скользящее окно, состоящее из двух активных областей

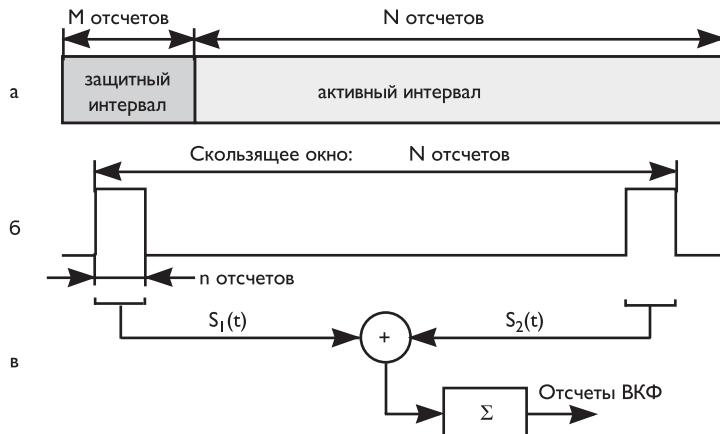


Рис. 20.16. Построение простейшего согласованного фильтра для идентификации циклического префикса

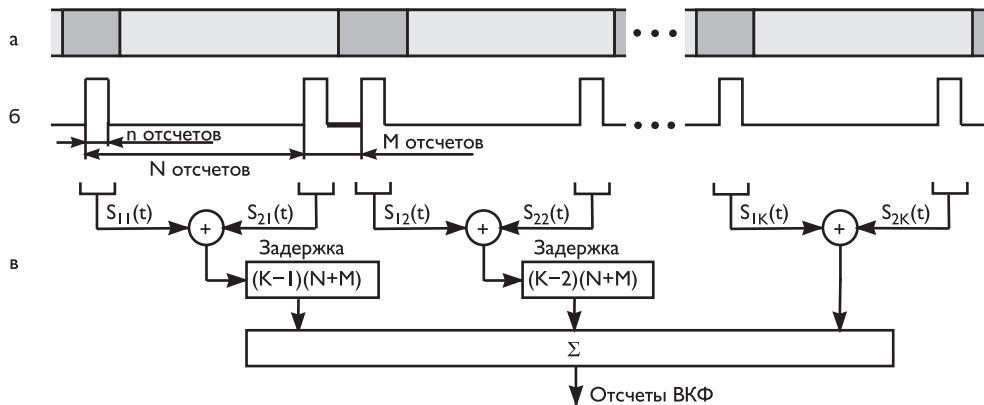


Рис. 20.17. Взвешенный корреляционный фильтр, согласованный с несколькими смежными OFDM-символами

(стробов), смещенных на длительность активной части символа, равного N дискретных отсчетов (рис. 20.15а, б). Длительность каждого строба (n отсчетов) скользящего окна выбирается несколько меньше длительности активного интервала (M отсчетов).

Значение коэффициента корреляции, очевидно, будет наибольшим в случае максимальной схожести сигналов $S_1(n)$ и $S_2(n)$, расположенных в областях двух стробов скользящего окна (см. рис. 20.16в):

$$R = \sum_{i=1}^n S_1(n) \cdot S_2(n) = \begin{cases} R_{\max} & \text{при } S_1(n) \equiv S_2(n); \\ r \cdot R_{\max} & \text{при } S_1(n) \neq S_2(n), r \ll 1. \end{cases} \quad (20.24)$$

Для повышения точности оценки величины R_{\max} часто применяют фильтр, согласованный с несколькими смежными OFDM-символами, как показано на рис. 20.17.

В данном случае сначала вычисляются произведения в каждом из скользящих окон — $S_{1k}(n) \cdot S_{2k}(n)$, $1 \leq k \leq K$, затем после необходимой задержки

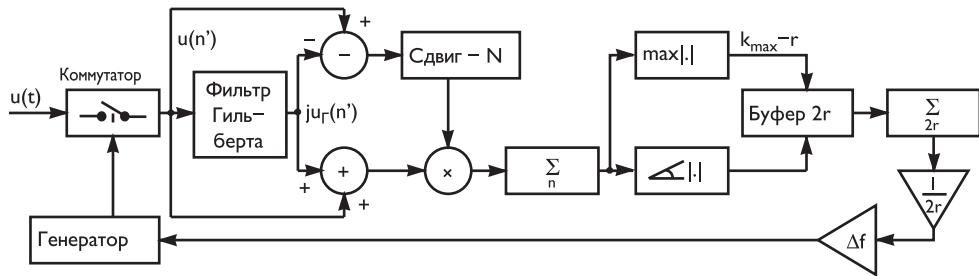


Рис. 20.18. Структурная схема системы подстройки частоты в OFDM-демодуляторе

до совмещения по времени вычисленных отсчетов произведения суммируются. Использование двух и более символов для оценки значения R_{\max} приводит к существенному уменьшению величины r и, следовательно, к уменьшению величины шумовой составляющей.

Аналитический способ оценки частотного смещения основывается на вычислении корреляционной функции $G(k)$ как сечения функции неопределенности в плоскости искомого частотного сдвига в соответствии со структурной схемой рис. 20.18.

Пусть на вход схемы поступает сигнал

$$u(t) = \sum_{m=1}^M a_m \cos(2\pi m f_U t + \varphi_m).$$

После дискретизации сигнала с частотой, не кратной опорной частоте f_U и равной $f = N f_U + \Delta f$, его отсчеты определяются соотношением:

$$\begin{aligned} u(n') &= \sum_{m=1}^M a_m \cos \left[2\pi m \frac{n'}{N(1 + \frac{\Delta f}{N f_U})} + \varphi_m \right] \cong \\ &\cong \sum_{m=1}^M a_m \cos \left[2\pi m \frac{n'}{N} \left(1 - \frac{\Delta f}{N f_U} \right) + \varphi_m \right]. \end{aligned} \quad (20.25)$$

Результирующая формула отсчетов сигнала справедлива, поскольку $N f_U \gg \Delta f$. На выходе фильтра Гильберта, сдвигающего все фазы синусоидальных сигналов на 90° , формируется сигнал, отсчеты которого равны

$$u_\Gamma(n') = \sum_{m=1}^M a_m \sin \left[2\pi m \frac{n'}{N} \left(1 - \frac{\Delta f}{N f_U} \right) + \varphi_m \right]. \quad (20.26)$$

Таким образом, на выходе вычитателя формируется комплексный сигнал с отсчетами, расположенными в области первого по времени строба скользящего окна, —

$$S_1(n') = \sum_{m=1}^M a_m \exp \left\{ -j \left[2\pi m \frac{n'}{N} \left(1 - \frac{\Delta f}{N f_U} \right) + \varphi_m \right] \right\}, \quad (20.27)$$

а на выходе сумматора выделяется комплексно сопряженный сигнал, отсчеты

которого в области второго строба скользящего окна равны

$$S_2(n') = \sum_{m=1}^M a_m \exp \left\{ j \left[2\pi m \frac{n' - N}{N} \left(1 - \frac{\Delta f}{N f_U} \right) + \varphi_m \right] \right\}. \quad (20.28)$$

После перемножения и суммирования в интервале строба, выделяющего n отсчетов, вычисляется корреляционная функция:

$$G(k) = G_{Re}(k) + jG_{Im}(t) = \sum_{n''=1}^n \sum_{m'=1}^M \sum_{m''=1}^M a_{m'} a_{m''} \exp [-j(\varphi_{m'} - \varphi_{m''})] \times \\ \times \exp \left[-2\pi j m' \frac{k + n''}{N} \left(1 - \frac{\Delta f}{N f_U} \right) \right] \cdot \exp \left[2\pi j m'' \frac{k + n'' - N}{N} \left(1 - \frac{\Delta f}{N f_U} \right) \right].$$

Далее осуществляется поиск максимума модуля этой функции, положение которого соответствует максимуму функции корреляции между составляющими защитного и активного интервалов OFDM-сигнала. В случае, когда

$$\left| G(k) = \sqrt{G_{Re}^2(k) + G_{Im}^2(t)} \right| \Rightarrow \max(k = k_{\max}), \quad (20.29)$$

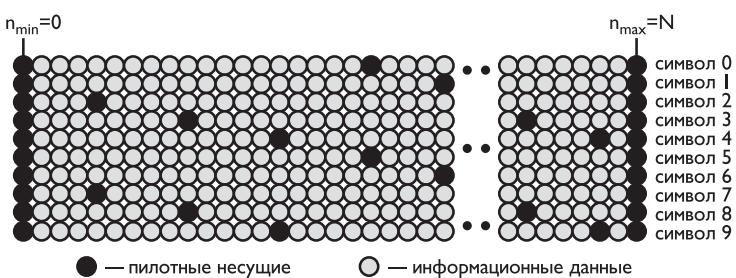
сдвиг фазы комплексного сигнала оказывается равным

$$\psi(k_{\max}) = \arctg \frac{G_{Im}}{G_{Re}} \cong 2\pi n \frac{\Delta f}{N f_U}. \quad (20.30)$$

Это соотношение справедливо лишь в идеальном случае, когда помехи в канале отсутствуют. Для уменьшения их влияния на расчет частоты сдвига гетеродина приемного устройства в буферном блоке (рис. 20.18) запоминается $2r$ значений вычисленных фаз комплексного сигнала в области максимума модуля функции (20.29) и производится усреднение значений фазы (20.30). Далее осуществляется вычисление сдвига частоты, который используется для коррекции частоты гетеродина.

20.3.2. Оценка частотного смещения при приеме OFDM-сигнала

Рис. 20.19. Кадровая структура OFDM-сигнала



Оценка частотного смещения на основе эталонного сигнала осуществляется с применением повторяющихся и рассеянных пилотных составляющих в OFDM-сигнале. Повторяющиеся пилотные несущие вводятся во все без исключения OFDM-символы. Рассеянные пилотные несущие меняют свое положение от одного OFDM-символа к другому и обычно представляются в виде диаграммы кадровой структуры, изображенной на рис. 20.19.

Принцип выделения величины частотного смещения аналогичен приведенному выше [7.42, 7.43]. Структурная схема обработки сигнала подобна приведенной на рис. 20.18. Однако на вход схемы подается сигнал, состоящий только из pilotных несущих, выделяемых с использованием схемы БПФ и преобразуемых в последовательность дискретных отсчетов путем ОБПФ.

Литература к части VII

- 7.1. *Кларк Дж., Кейн Дж.* Кодирование с исправлением ошибок в системах цифровой связи / Пер. с англ. под ред. Б.С. Цыбакова. М.: Радио и связь, 1987. 392 с.
- 7.2. *К. Шеннон.* Работы по теории информации и кибернетики/ Под ред. Р.Л. Добрушина, О.Б. Лупанова. М.: Иностранный литература, 1963. 829 с.
- 7.3. *Финк Л.М.* Теория передачи дискретных сообщений. 1970.
- 7.4. *Галлагер Р.* Теория информации и надежная связь. М.: Наука, 1974.
- 7.5. *Золотарев В.В., Овечкин Г.В.* Помехоустойчивое кодирование. Методы и алгоритмы: Справочник. М.: Горячая линия-Телеком, 2004. 126 с.
- 7.6. *Вернер М.* Основы кодирования: Учебник для вузов. М.: Техносфера, 2006. 288 с.
- 7.7. *Мак-Вильямс Ф.Дж., Слоен Н.Дж.А.* Теория кодов, исправляющих ошибки. М.: Связь, 1979. 744 с.
- 7.8. *Блейхут Р.* Теория и практика кодов, контролирующих ошибки. М.: Мир, 1986.
- 7.9. *Морелос-Сарагоса Р.* Искусство помехоустойчивого кодирования. Методы, алгоритмы, применение. М.: Техносфера, 2006. 320 с.
- 7.10. *Ипатов В.* Широкополосные системы и кодовое разделение сигналов.
- 7.11. *Склляр Б.* Цифровая связь. Теоретические основы и практическое применение. 2-е изд./ Пер. с англ. М.: Издательский дом «Вильямс», 2003. 1104 с.
- 7.12. *Odenwalder J.P.* Optimal Decoding of Convolutional Codes. Ph. D. dissertation, University of California. Los Angeles, 1970.
- 7.13. *Curry S.J.* Selection of Convolutional Codes Having Large Free Distince. Ph. D. dissertation, University of California. Los Angeles, 1971.
- 7.14. *Larsen K.J.* Short Convolutional Codes With Maximum Free Distince for Rates 1/2, 1/3, and 1/4// IEEE Inf. Theory. Vol. IT19. 1973. N 3. 1973, p. 371–372.
- 7.15. *Viterbi A.J. and Omura J.K.* Principles of Digital Communication and Coding. McGrow-Hill Book Company, New York, 1979.
- 7.16. *Шлома А.М., Бакулин М.Г., Крейндельин В.Б., Шумов А.П.* Новые технологии в системах мобильной Радиосвязи/ Под ред. Шломы А.М. 2005. 455 с.
- 7.17. *Berrou C., Glavieux A. and Thitimajshima P.* «Near Shannon limit error-correcting coding and decoding: turbo-codes». ICC 1993., Geneva, Switzerland, p. 1064-1070. May, 1993.
- 7.18. *Bahl L.R., J.Cocke J., F. Jelinek F., J. Raviv J.* Optimal Decoding of Linear Codes for Minimizing Symbol Error Rate// IEEE Transactions on Information Theory. Vol. 20, p. 284-287. March, 1974.

- 7.19. Шлома А.М., Бакулин М.Г., Крейнделин В.Б., Шумов А.П. Новые алгоритмы формирования и обработки сигналов системах подвижной связи. М.: Горячая линия — Телеком, 2008. 344 с.
- 7.20. Benedetto S., G. Montorsi G. Serial concatenation of block and convolutional codes// Electronic Letters. Vol. 32, p. 887–888. May 9th, 1996.
- 7.21. Valenti M.C. «Iterative Detection and Decoding for Wireless Communications»: PhD Dissertation. Univ. of Virginia. July, 1999.
- 7.22. Hagenauer Joachim, Elke Offer Elke, and Lutz Papke Lutz. «Iterative Decoding of Binary Block and Convolutional Codes»// IEEE Transactions on Information Theory. Vol. 42. N. 2. March, 1996.
- 7.23. AHA Application Note. Primer: Turbo Product Codes. <http://www.aha.com>
- 7.24. David Williams. Turbo Product Code FEC Contribution: Advanced Hardware Architectures. Project // IEEE 802.16 Broadband Wireless Access Working Group. June, 2000.
- 7.25. Gallager R.G. «Low-density parity-check codes». Cambridge, MA: M.I.T. Press, 1963.
- 7.26. Tanner R.M. «A recursive approach to low complexity codes»// IEEE Trans. Info. Theory. Vol. IT-27, N.5, p. 533-547. Sept., 1981.
- 7.27. MacKay D.J.C., Neal R.M. «Near Shannon limit performance of low density parity check codes». Electronics Letters, 32(18):1645-1646. August, 1996.
- 7.28. Ryan W.E, Lin S. «Channel codes. Classical and modern». Cambridge, University Press, 2009.
- 7.29. Jin H., Khandecar A. and McEliece R. Irregular repeat-accumulate codes. Proc. 2nd. Int. Sump. On Turbo Codes and Related Topics, Brest, France, Sept. 2000. p. 1–8.
- 7.30. Frame structure channel coding and modulation for a second generation digital terrestrial television broadcasting system (DVB-T2), DVB Document A122, 2008.
- 7.31. Kienle F., Brack T., Wehn N. «A synthesizable IP Core for DVB-S2 LDPC Code Decoding»// IEEE Conference on Design Automation and Test Europe (DATE). 2005.
- 7.32. Zhang J., Fossorier Marc P.C. «A modified weighted bit-flip decoding of low-density parity-check codes»// IEEE Communication Letters. Vol. 8. N. 3. March, 2004.
- 7.33. Fossorier M.P.C., Mihaljevic M., H.Imai H. «Reduced complexity iterative decoding of low density parity check codes based on belief propagation»// IEEE Trans. on Comm. Vol. 47. N. 5, p. 673-680. May, 1999.
- 7.34. Зубарев Ю.Б., Кривошеев М.И., Красносельский И.Н. Цифровое телевизионное вещание. Основы, методы, системы. М.: НИИР, 2001.
- 7.35. Волков Л.Н., Немировский М.С., Шинаков Ю.С. Системы цифровой радиосвязи: базовые методы и характеристики: Учеб. пособие. М.: Эко-Трендз, 2005.

- 7.36. Вишневский В.М., Портной С.Л., Шахнович И.В. Энциклопедия WiMAX. Путь к 4G. М.: Техносфера, 2009.
- 7.37. Цимер Р., Петерсон Р. Цифровая связь/ Пер. с англ. М.: Техносфера, 1989.
- 7.38. <http://allfpga.com/wiki/OFDM>
- 7.39. <http://allfpga.com/wiki/> Математическое описание OFDM
- 7.40. Ahmad R.S. Bahai, Burton R. Salzberg, Mustafa Ergen. Multi-carrier digital communications: theory and applications of OFDM.
<http://www.shelfari.com/books/1386824/Multi-carrier-Digital-Communications-Theory-And-Applications-Of->
- 7.41. ITU-R Recommedation BT.1306-1. Error Correction, Data Framing Modulation and Emission Metods for Digital Terrestrial Television Broadcasting
- 7.42. Landström D. Synchronization in OFDM Systems. Lund University/ Department of Applied Electronics, 1999.
- 7.43. Hanzo L., Keller T. OFDM and MC-CDMA. A Primer// IEEE Press. Communication Society, Sponsor John&Sons, Ltd, 2006.

ЧАСТЬ VIII

ОСНОВНЫЕ СТАНДАРТЫ СИСТЕМ ЦИФРОВОГО ТЕЛЕВИЗИОННОГО ВЕЩАНИЯ

Введение

Основополагающим фактором разработки и внедрения видеинформационных систем является их стандартизация, необходимая для производителей соответствующей аппаратуры, разработчиков сетей вещания, формирователей контента, провайдеров и конечных пользователей.

Стандартизацией телекоммуникационных систем занимается работающий под эгидой ООН Международный союз электросвязи МСЭ (ITU — International Telecommunications Union), являющийся межправительственной организацией и состоящий из трех секторов:

- радиосвязи (ITU-R, ITU Radiocommunication Sector), играет важнейшую роль в глобальном управлении использованием радиочастотного спектра и орбит спутников;
- стандартизации телекоммуникаций (ITU-T, ITU Telecommunication Standardization Sector), разрабатывает Рекомендации, являющиеся определяющими элементами инфраструктуры информационно-коммуникационных технологий и охватывающие все вопросы — от функциональных возможностей базовой сети и широкополосной связи до услуг последующих поколений, например IPTV;

- развития (ITU-D, ITU Telecommunications Development Sector) — новый сектор, созданный в 1989 году на базе Бюро телекоммуникационных исследований, рассматривает многочисленные аспекты расширения доступа к информационно-коммуникационным технологиям (ИКТ) и преодоления «цифрового разрыва».

Важную роль в утверждении международных стандартов играет Международная организация по стандартизации (ISO — International Organization for Standardization) — неправительственный орган, включающий сеть институтов стандартизации из 148 стран. ISO успешно работает в области стандартизации телекоммуникационных систем при плотном сотрудничестве с Международной электротехнической комиссией МЭК (IEC — International Electrotechnical Commission). ISO и IEC используют единую систему нумерации стандартов, и в обозначениях ряда стандартов зачастую фигурируют названия обеих организаций.

Следует отметить, что технической проработкой будущих стандартов в области телекоммуникаций занимаются национальные и межнациональные организации:

- Европейский институт стандартизации по телекоммуникациям (ETSI — European Telecommunications Standards Institute);
- Европейская конференция почтовых и телекоммуникационных ведомств (Conference of European Postal and Telecommunication Administrations — CEPT);
- Американский национальный институт стандартов (ANSI — American National Standards Institute);
- Японская ассоциация стандартов (JESA — Japanese Engineering Standards Association);
- Министерство почт и телекоммуникаций Великобритании (MPT — Ministry of Post and Telecommunications);
- Ассоциация электронной промышленности США (EIA — Electronics Industries Association);
- Институт инженеров по электротехнике и электронике США (IEEE — Institute of Electrical and Electronics Engineers) и др.

Следует отметить, что членами IEEE являются ANSI и ISO. Как правило, разработанные IEEE стандарты утверждаются ISO либо ITU.

В России Федеральное агентство по техническому регулированию и метрологии (РОССТАНДАРТ) организует экспертизу проектов национальных стандартов, разрабатываемых отраслевыми институтами. По мере интеграции с мировым телекоммуникационным сообществом отечественные стандарты должны играть все большую роль в повышении эффективности систем и сетей вещания и связи.

Наиболее удачной попыткой стандартизовать протоколы обмена информацией является так называемая модель взаимодействия открытых систем (МВОС, OSI — Open System Interconnection), разработанная и утвержденная ISO. В соответствии с Рекомендацией ITU-T X.200 эта модель подразделяет протоколы взаимодействия систем на семь уровней — физический, канальный, сетевой, транспортный, сеансовый, представительский и прикладной [8.1–8.3].

Физический уровень (Physical, Layer 1) определяет электрические и механические характеристики подключения к физическим каналам связи и процедуры передачи потоков битов между его двумя узлами.

Основными функциями физического уровня являются:

- создание и прекращение работы каналов связи;
- эффективное распределение коммуникационных ресурсов между пользователями;
- модуляции или иные преобразования цифровых данных и соответствующих сигналов, передаваемых по каналам связи.

Канальный уровень, или уровень звена данных (Data Link, Layer 2), отвечает за упаковку информации в кадры определенной длины, формирование контрольных сумм и проверку содержимого кадров после их приема, а также возможную повторную передачу неподтвержденных кадров. Этот уровень реализует функциональные и процедурные операции, обеспечивающие передачу данных между сетевыми организациями в зависимости от архитектуры используемой сети.

Сетевой, или пакетный, уровень (Network, Layer 3) формирует сетевые адреса пакетов, управляет потоками, адресацией, маршрутизацией, организацией и поддержанием транспортных соединений. Сетевой уровень обеспечивает функциональные и процедурные средства передачи последовательностей данных из узла источника информации в одной сети к узлу назначения в другой сети, сохраняя при этом качество услуг. Сетевой уровень выполняет функцию маршрутизации передаваемой информации, а также может выполнять фрагментацию и повторную ее сборку.

Транспортный уровень (Transport, Layer 4) предназначен для трансляции потоков данных от передатчика (одного порта) через сколь угодно сложную среду передачи к приемнику (другому порту, где практически завершаются операции транспортировки данных и начинаются вычислительные процессы). Он обеспечивает прозрачную и надежную передачу данных между конечными пользователями, реализуя надежную передачу данных услуг, контроль достоверности данной связи через управление потоками данных. Транспортный уровень также обеспечивает подтверждение успешной передачи данных и отправляет следующие данные, если нет ошибки.

Сеансовый уровень (Session, Layer 5) обеспечивает организацию, поддержание и окончание логической связи между прикладными процессами. Он устанавливает и завершает соединение между локальным и удаленным приложениями, реализует полнодуплексную, полудуплексную или симплексную систему связи, а также определяет необходимость перезапуска процедур.

Уровень представления (Presentation, Layer 6) преобразует форматы данных и команд в форму, удобную для используемой прикладной программы. Этот слой обеспечивает независимость от формы представления данных (например, шифрования) путем преобразования сетевых форматов.

Прикладной уровень (Application, Layer 7) характеризует процесс обработки информации, определяя тем самым работу прикладной программы. На этом уровне пользователь непосредственно использует программные приложения, они выходят за рамки модели OSI.

ГЛАВА 21

СТАНДАРТЫ ЦИФРОВОГО ТЕЛЕВИЗИОННОГО ВЕЩАНИЯ

Одной из важных и сложных задач стандартизации цифрового телевизионного вещания являлось использование эффективных методов передачи телевизионных программ по наземным, спутниковым и кабельным системам связи.

Принципиальной особенностью разработки таких стандартов является учет технологических особенностей различных участков телевизионного канала, жестких международных норм на параметры каналов передачи телевизионных сигналов, принципов планирования сетей вещания.

Цифровое сжатие сигналов динамических изображений и звукового сопровождения телевидения в сочетании с цифровым канальным кодированием и модуляцией позволяют существенно повысить эффективность использования частотного ресурса по сравнению с аналоговыми телевизионными системами [8.4].

В табл. 21.1. приведен перечень основных стандартов цифрового телевизионного вещания, годы принятия стандартов и основные их параметры.

На рис. 21.1 приведено изображение картины мирового использования стандартов цифрового эфирного (наземного) телевизионного вещания.

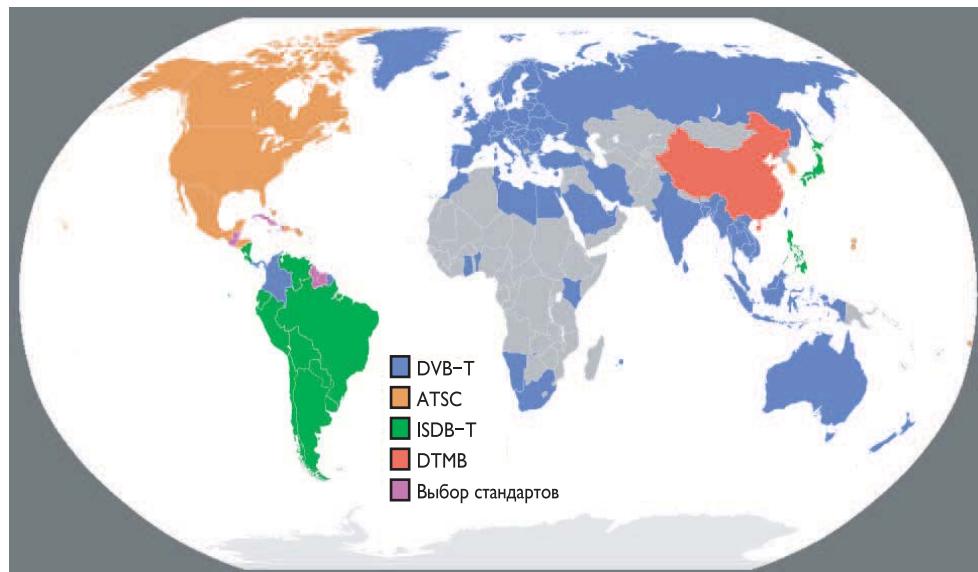


Рис. 21.1. Стандарты цифрового эфирного телевизионного вещания

Таблица 21.1. Стандарты цифрового телевизионного вещания

Обозначение	Стандарт, рекомендация	Назначение телевещания	Диапазон частот, ширина спектра, цифровой поток	Год утверждения стандарта
ATSC	Doc. A/53. ATSC		VHF, UHF 6 МГц	1996
		наземное	19,39 Мбит/с	
		кабельное	38, 78 Мбит/с	
DVB-T	ETSI EN 300 744	наземное	VHF, UHF 6, 7, 8 МГц 4,98... 31,67 Мбит/с	1997
DVB-T2	ETSI EN 302 755		VHF, UHF 1,7, 6, 7, 8, 10 МГц 7,49... 50,32 Мбит/с	2008
DVB-S	ETSI EN 300 421	спутниковое	C- и К-диапазоны	1997
DVB-S2	ETSI EN 302 307		C-, К- и Ка-диапазоны	2004
DVB-C	ETSI EN 300 429	кабельное	VHF, UHF 6, 8 МГц 25,00... 50,20 Мбит/с	1998
DVB-C2	ETSI EN 302 769		VHF, UHF 6, 8 МГц 25,10... 84,8 Мбит/с	2009
ISDB-T	ITU-R BT.1306-1	наземное	VHF, UHF 6, 7, 8 МГц 3,65... 30,98 Мбит/с	1999
ISDB-S	ITU-R BO.1408	спутниковое	C- и К-диапазоны 13,6... 54,4 Мбит/с	1999
ISDB-C	ITU-T J.83, J.183	кабельное	VHF, UHF 6, 8 МГц 25,00... 50,20 Мбит/с	2004
DTMB	GB 20600-2006	наземное	VHF, UHF 6, 8 МГц 5,41... 28,8 Мбит/с	2006

Стандарт американского цифрового телевизионного вещания, утвержденный Федеральной комиссией по связи США (FCC — Federal Communications Commission) в 1996 году, был ориентирован на переход к телевидению высокой четкости.

Семейство стандартов цифрового телевидения, разработанное консорциумом DVB (Digital Video Broadcasting), стандартизировано Европейским институтом телекоммуникационных стандартов (ETSI).

Эти стандарты охватывают все уровни модели взаимодействия открытых систем OSI с разной степенью детализации для различных способов передачи цифрового сигнала: наземного (фиксированного и мобильного), спутникового, кабельного (как классического, так и IPTV). На более высоких уровнях OSI стандартизируются системы условного доступа, способы организации информации для передачи в среде IP, различные метаданные и др.

Стандарты DVB, как и остальные стандарты цифрового телевидения, в высокой степени связаны со стандартами MPEG-2 и MPEG-4, которые определяют тип используемого транспорта и способы кодирования изображений и звука в цифровом телевидении.



Стандарты DVB, которые приняты ETSI, подразделяются на несколько групп: EN — обязательные стандарты, TS — технические спецификации (неполный предварительный стандарт), TR — рекомендации (технические отчеты), необязательные для исполнения.

Стандарты цифрового телевидения DVB делятся на группы по сфере применения. Каждая группа имеет сокращенное название с префиксом DVB-, например, DVB-DATA — группа стандартов, посвященная передаче данных по сетям цифрового телевидения.

21.1. Стандарт цифрового телевидения ATSC

Стандарт американского цифрового телевидения ATSC, разработанный Комитетом перспективных систем телевидения (Advanced Television Systems Committee), предусматривал замену системы аналогового телевизионного вещания NTSC [8.5–8.9]. В документе ITU-R [8.10] этот стандарт обозначен как «система А».

Главным направлением цифрового вещания в США был полный переход к телевидению высокой четкости (HDTV, High Definition Television) с новым качеством изображений на экране широкоформатного телевизора, отношением ширины экрана к высоте 16:9 и числом активных строк 1080. При этом HDTV-программы сопровождаются высококачественной звуковой системой Dolby AC-3 5.1 (пять основных звуковых каналов и один дополнительный низкочастотный канал).

Однако стандарт ATSC предусматривает возможность передачи цифровых программ в различных форматах с использованием как прогрессивной (p, Progressive), так и чересстрочной (i, Interlaced) «американских» разверток (табл. 21.2).

Таблица 21.2. Форматы изображений стандарта ATSC для частот развертки: 60(59,94)/30(29,97)/24(23,976) Гц

Число строк	Число пикселов в строке	Ширина к высоте экрана	Частота развертки
1080	1920	16 : 9	30i, 30p, 24p
720	1280	16 : 9	60p, 30p, 24p
480	704	16 : 9 и 4 : 3	60p, 30i, 30p, 24p
480	640	4 : 3	60p, 30i, 30p, 24p

Стандартом также предусмотрены форматы изображений для «европейских» частот развертки (табл. 21.3).

Таблица 21.3. Форматы изображений стандарта ATSC для частот развертки 50/25 Гц

Число строк	Число пикселов в строке	Ширина к высоте экрана	Частота развертки
1080	1920	16 : 9	50i, 25p
720	1280	16 : 9	50p, 25p
576	720	16 : 9 и 4 : 3	50p, 25i, 25p
	544		
	480		25i, 25p
	352		
288	352	16 : 9 и 4 : 3	25p

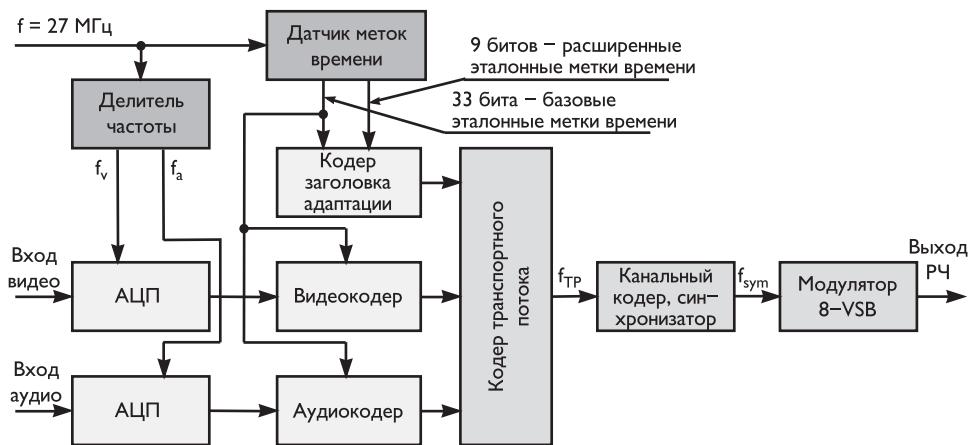


Рис. 21.2. Структурная схема системы 8-VSB ATSC

Основной технической задачей цифрового телевидения ATSC была передача HDTV-программы с заданной помехоустойчивостью в наземном радиоканале с ограниченной полосой частот 6 МГц. В стандарте применен запатентованный фирмой Zenith [8.11] метод многоуровневой модуляции с частично подавленной боковой полосой частот (VSB, Vestigal Side-Band). Для цифрового наземного ТВ используется 8-уровневая модуляция с треллисным (Trellis) или решетчатым кодированием, и по этой причине режим называется 8-VSB [8.11–8.14].

Стандарт ATSC предусматривает и более высокоскоростную версию для кабельного телевидения — режим 16-VSB без треллисного кодирования.

На рис. 21.2 приведена структурная схема системы 8-VSB ATSC. При преобразовании аналоговой информации применяется набор частот, формируемый из сигналов эталонной частоты 27 МГц. С использованием этих сигналов формируются два типа меток эталонов времени — базовые эталоны программы (33 бита) и расширенные эталоны (9 битов). Расширенные эталоны соответствуют отсчетам частоты 90 кГц, формируемой путем деления частоты 27 МГц, и применяются при формировании меток времени отображения и декодирования информации видео и аудиокодеров. Кодер транспортного потока выполняет роль мультиплексора, объединяющего информационные потоки и сервисную информацию в стандартный поток MPEG-2 TS.

В систему адаптации потока MPEG-2 к наземному каналу вещания входят канальный кодер и модулятор. Обобщенная структурная схема преобразования потока MPEG-2 изображена на рис. 21.3.

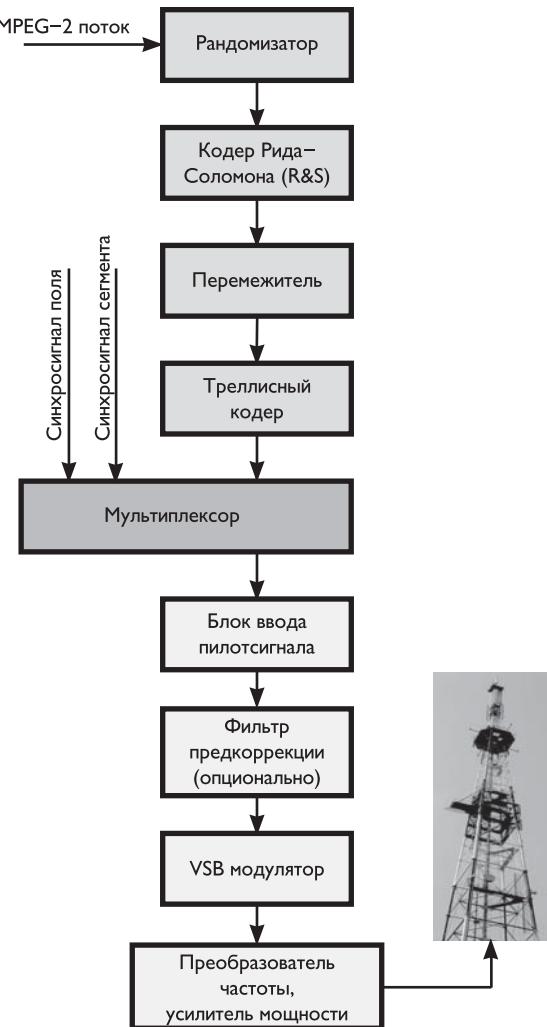
На вход поступает последовательность пакетов стандартного транспортного потока MPEG-2, в которых передаются видеосигнал, сигнал звукового сопровождения и различные дополнительные данные.

Первой операцией обработки данных является рандомизация, задача которой — преобразование сигнала в квазислучайную последовательность. Рандомизация выполняется путем сложения потока данных с псевдослучайной последовательностью, формируемой генератором, выполненным на базе 16-разрядного

регистра сдвига с обратной связью (рис. 21.4). Выходные сигналы восьми триггеров регистра образуют байты псевдослучайной последовательности (D_0, D_1, \dots, D_7), которые поразрядно складываются по модулю 2 (с использованием элементов «исключающее ИЛИ», XOR) с байтами передаваемых данных.

Инициализация генератора псевдослучайной последовательности осуществляется путем загрузки в регистр шестнадцатеричного числа F18016 (в триггер регистра, с выхода которого снимается сигнал обратной связи, записывается старший бит числа начальной загрузки). Инициализация производится в начале поля данных во время импульса синхронизации первого сегмента данных.

Рис. 21.3. Схема формирования радиосигнала системы 8-VSB ATSC



Затем данные разбиваются на сегменты, состоящие из 832 символов (рис. 21.5), следующих с частотой $f_s = 10,762$ МГц.

Длина сегмента соответствует форматам пакетов данных потока MPEG-2, состоящих из 188 байтов каждый. Из пакета исключается байт синхронизации, а к

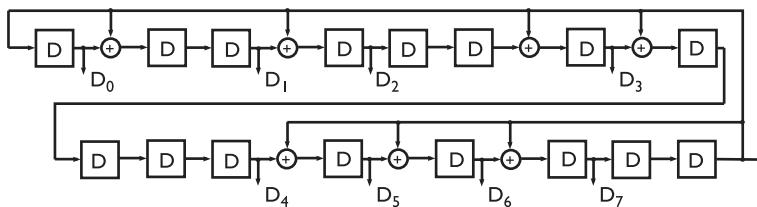


Рис. 21.4. Формирователь псевдослучайной последовательности рандомизатора данных

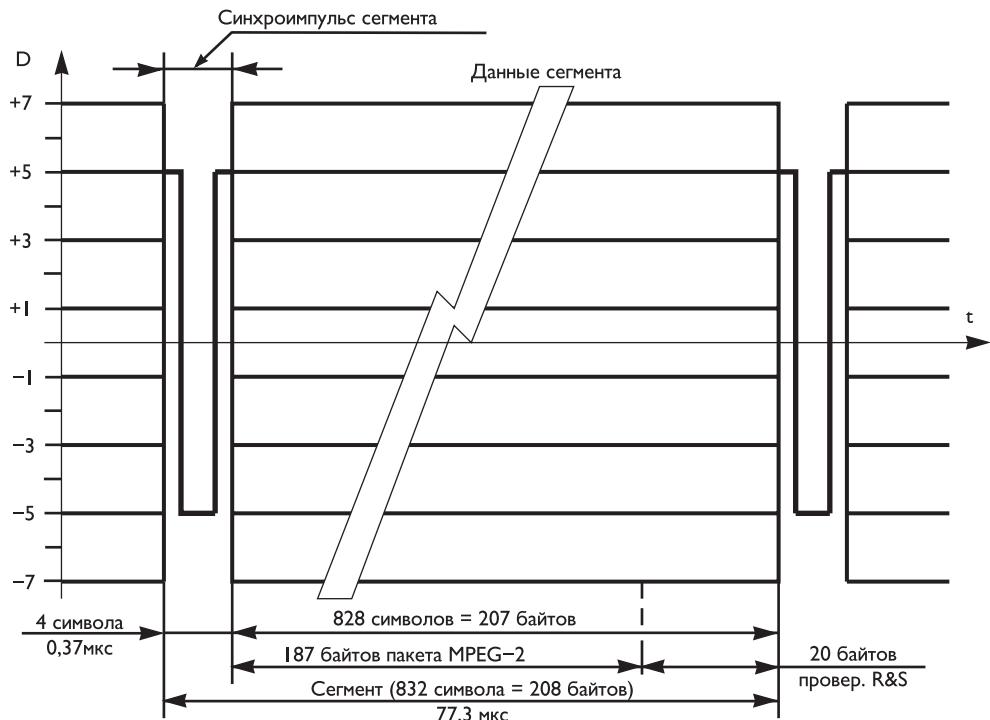


Рис. 21.5. Уровни и форма модулирующего сигнала в системе 8-VSB ATSC

оставшимся 187 байтам данных добавляются двадцать проверочных байтов кода Рида-Соломона, реализующего функции внешнего кода. Структура образованного кодового слова (207, 187) позволяет исправлять до $t = 10$ ошибочных байтов в каждом слове. Проверочные байты RS передаются в конце каждого сегмента. Кодер RS реализуется на основе регистра сдвига с 2 t -байтовыми элементами памяти и обратными связями. Процесс кодирования сводится к операциям сложения и умножения по модулю 256.

Максимально допустимая длительность импульсной помехи для RS оценивается соотношением:

$$T_{RS} = \frac{2n \cdot t}{f_s} = \frac{2 \cdot 2 \cdot 10}{10,762(\text{МГц})} = 3,71 \text{ мкс},$$

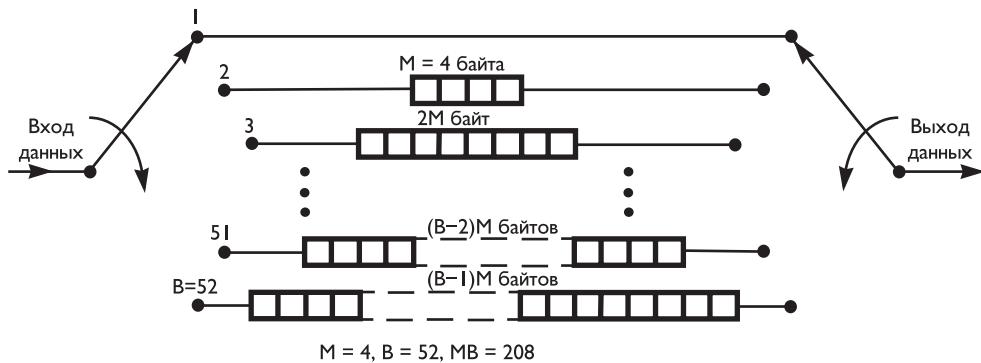


Рис. 21.6. Межсегментное перемежение данных в системе 8-VSB ATSC

где n — число информационных битов, приходящихся на символ (для режима 8-VSB $n = 2$). При большей длительности число ошибочных байтов внутри сегмента превышает корректирующую способность кода RS, и сегмент будет принят с ошибками.

Далее следует межсегментное перемежение данных, выполняемое на коммутируемых регистрах, в которых каждая ячейка содержит один байт данных (рис. 21.6). Благодаря перемежению в сегмент попадают байты из разных транспортных пакетов и разных кодовых слов кода Рида–Соломона. Межсегментное перемежение перемешивает байты данных в пределах 52 сегментов данных одного поля, что соответствует $1/6$ части поля или приблизительно 4 мс во времени. Перемежению подвергаются только байты данных. Начало процесса перемежения привязано к первому байту данных поля.

Межсегментное перемежение обеспечивает повышение эффективности исправления больших пакетов ошибок. После перемежения внутри каждого сегмента соседними оказываются байты, находящиеся на расстоянии 52 байтов в исходном сегменте. Таким образом, в ATSC комбинация алгоритмов RS и перемежения допускает длительность пакета ошибок $B \cdot T_{RS} = 52 \cdot 3,71$ мкс = 193 мкс.

В системе 8-VSB используется еще система внутреннего кодирования для исправления ошибок, реализующая решетчатую сигнально-кодовую конструкцию на основе сверточного кода со скоростью $R = 2/3$.

На рис. 21.7а приведена структурная схема системы внутреннего кодирования, в которой сверточный кодер совмещен со схемами решетчатого кодирования и 8-уровневым устройством отображения (маппером) кодов символов данных на модуляционные символы. Вместе с решетчатым кодированием осуществляется внутрисегментное перемежение. Для этого используется двенадцать одинаковых кодеров и предварительных кодеров режекции (рис. 21.7б). На входной коммутатор поступают байты с устройства межсегментного перемежения, причем каждый байт обрабатывается целиком одним сверточным кодером. Биты с нечетными номерами (7, 5, 3, 1) каждого байта подвергаются предварительному кодированию, а биты с четными номерами (6, 4, 2, 0) — сверточному кодированию. Внутрисегментное перемежение достигается путем кодирования символов группами: символы 0, 12, 24, 36, ... образуют символы первой группы, символы 1, 13,

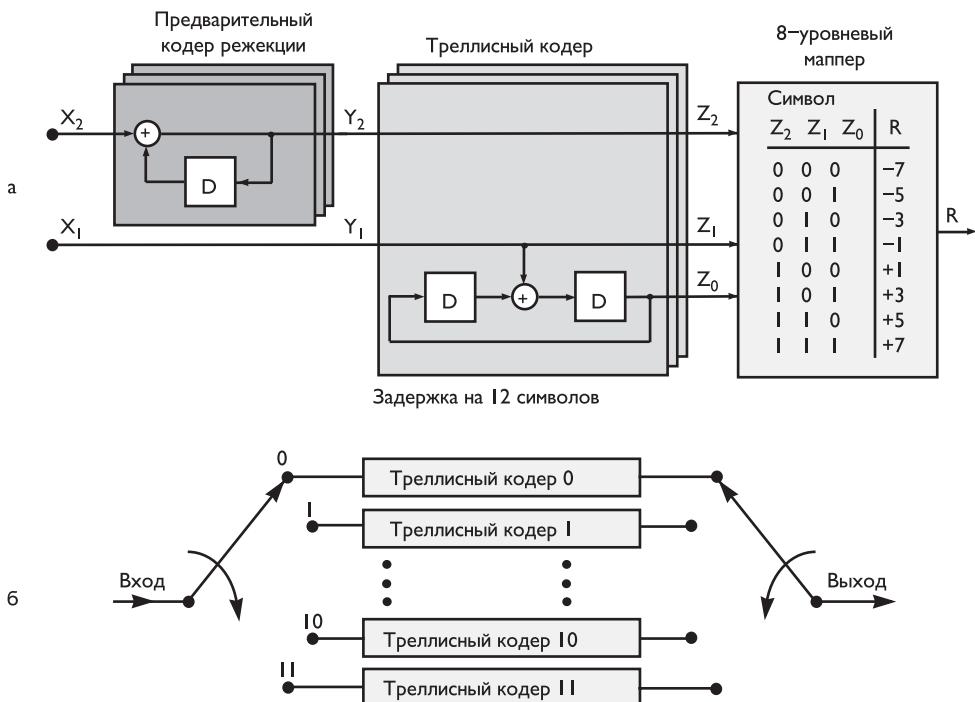


Рис. 21.7. Формирование 8-позиционного сигнала (а) и внутрисегментное перемежение (б) при треллисном кодировании

25, 37, ... — второй, и так далее, всего образуется 12 групп. Данные, получаемые с выходного мультиплексора, переключаются от кодера номер 0 до кодера номер 11 в первом сегменте поля данных, но во втором сегменте порядок следования меняется, и символычитываются сначала от кодеров 4–11, а затем от 0–3. В третьем сегменте символычитываются с кодеров 8–11, а затем с 0–7. Структура перемежения в трех сегментах повторяется в 312 сегментах поля. После того как вставлены синхросигналы сегментов данных, порядок следования символов данных таков, что символы с одного кодера отстоят друг от друга на расстоянии 12 символов.

Структура сформированного кадра, состоящего из двух полей данных, показана на рис. 21.8. Поле и кадр данных определяют структуру цифрового потока, содержащего информацию о компрессированных изображении и звуке.

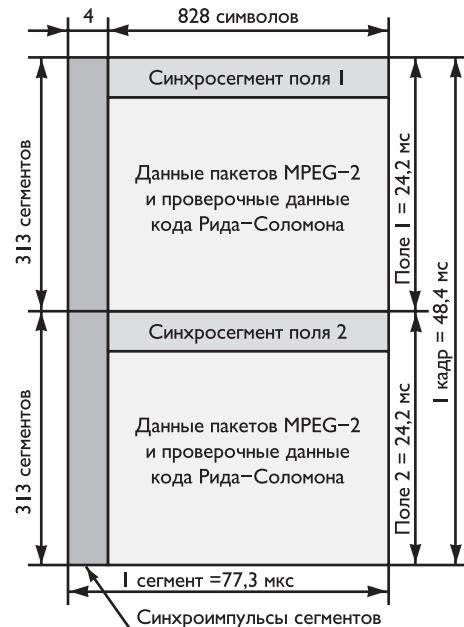
Объем дополнительных данных, добавляемых к полезным для синхронизации и исправления ошибок, сравнительно невелик и составляет 11%, поскольку $(207/187) \times (313/312) = 1,11$ во всех режимах, а скорость потока данных системы 8-VSB с выхода формирователя транспортного потока и доставляемого на вход декодера транспортного потока в приемнике равна $T_r = 2f_s \times (188/208) \times (312/313) = 19,39$ Мбитов/с.

Чтобы найти объем полезных передаваемых данных, надо рассчитанную скорость уменьшить в соответствии со структурой кадра:

$$2f_s \times (187/207) \times (312/313) \times (828/832) = 19,29 \text{ Мбит/с.}$$

Коэффициент $(187/207)$ учитывает потери, связанные с добавлением проверочных данных при кодировании с целью обнаружения и исправления ошибок, коэффициент $(828/832)$ позволяет учесть наличие сигнала синхронизации сегмента, а $(312/313)$ — включение синхросегмента, отвечающего за синхронизацию кадров данных.

Рис. 21.8. Кадр данных VSB системы ATSC



Окончательное формирование сигнала для модулятора происходит в мультиплексоре путем вставки синхросигналов, добавления пилот-сигнала и осуществления, при необходимости, предкоррекции сформированных сигналов (рис. 21.3).

Группа из 312 сегментов, дополненная синхросегментом, образует поле данных, а два поля по 313 сегментов — кадр данных. Синхросегмент (рис. 21.9) имеет длительность 832 символа (как и сегмент данных). Он включает в себя синхросигнал сегмента и на большей части своей длительности принимает всего два уровня ($+5$ и -5). Синхросегмент обеспечивает кадровую синхронизацию данных при отношении сигнал/шум до 0 дБ. Он используется также в качестве опорного сигнала при автоматической настройке корректоров приемника.

Пакет опорного сигнала PN511, состоящий из 511 двоичных символов, используется для настройки корректоров (трансверсальных цифровых фильтров), компенсирующих искажения канала на больших интервалах времени. Для настройки корректоров «малого времени» используются три опорные последовательности PN63. Средняя из этих трех последовательностей инвертируется от поля к полю для идентификации полей. Перед началом каждого поля данных производится инициализация генераторов последовательностей, заключающаяся

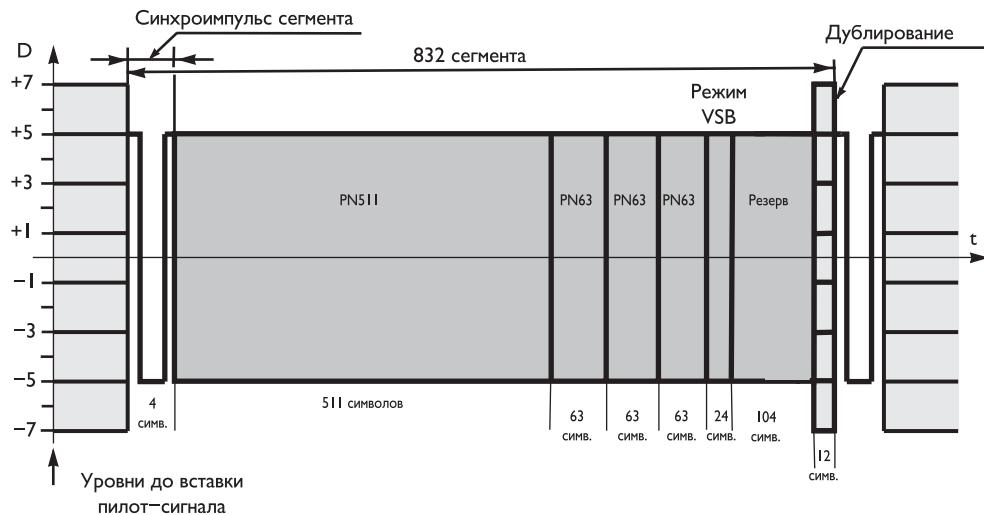


Рис. 21.9. Двухуровневый синхросегмент поля (для режима 8-VSB последние 12 символов предыдущего сегмента дублируются последними резервными символами синхросегмента поля)

в загрузке в регистры заданных двоичных чисел (100111 — в регистр, используемый в генераторе PN63, и 010000000 — в регистр PN511).

Дополнительно в синхросегменте используется 24 символа для сигнализации о режиме передачи (8-VSB или 16-VSB) и 104 резервных символа, применяемых для передачи сервисной информации и сигнализации. После резервных интервалов передаются 12 символов, представляющих собой повторение 12 последних символов предшествующего сегмента. Эти повторяющиеся символы, включаемые только в системе 8-VSB, используются для настройки устройств приемника.

Благодаря рандомизации данных уровней в сформированном сигнале равновероятны и средний уровень двухполярного сигнала практически равен нулю. Введение постоянного смещения величиной в 1,25 единицы к сигналу данных обеспечивает формирование пилот-сигнала в виде несущей радиосигнала постоянного уровня.

Величина несущей невелика и добавляет всего 0,3 дБ к общей мощности радиосигнала, но этого оказывается достаточно для восстановления несущей в приемнике в целях синхронного детектирования. Схемы восстановления несущей и синхросигналов действуют независимо от количества позиций многоуровневого модулирующего сигнала и работают при отношении сигнал/шум до 0 дБ. Уровень пилот-сигнала оказывается ниже средней мощности излучаемого сигнала на 11,3 дБ.

До подачи полностью сформированной последовательности на модулятор производится предкоррекция сигнала с помощью специальных фильтров, устраняющих возможные изменения его формы. Структура этих фильтров зависит от характеристик применяемых схем последующих преобразований сигнала.

Поступающий на вход модулятора сигнал подвергается амплитудной модуляции с последующим ограничением одной боковой полосы. Форма нормированной

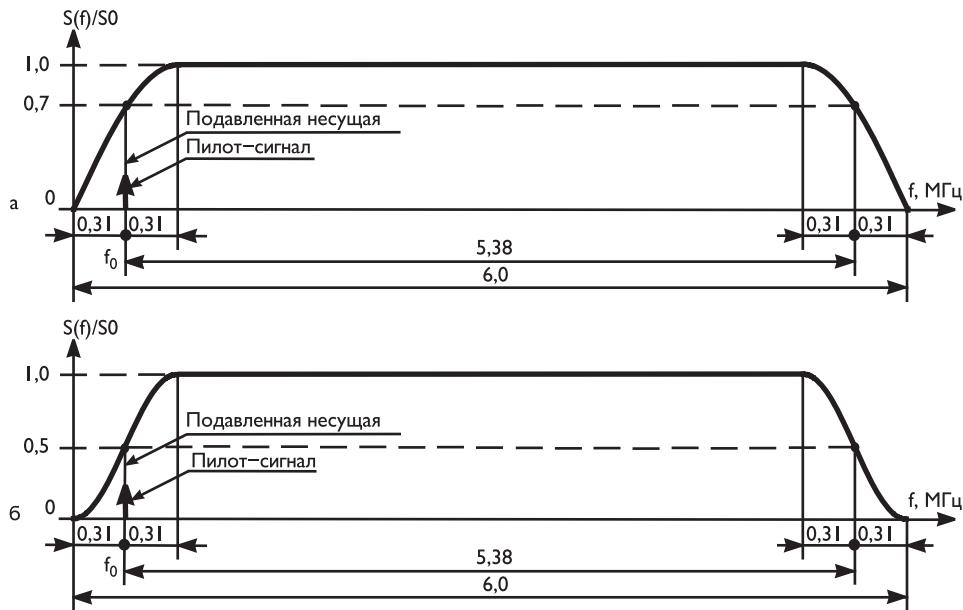


Рис. 21.10. Нормированная амплитудно-частотная характеристика передатчика системы ATSC (а) и эквивалентная амплитудно-частотная характеристика канала передатчик-приемник (б)

амплитудно-частотной характеристики передатчика приведена на рис. 21.10а. Слоны этой характеристики имеют форму косинусоидальной функции, и ее форма определяется следующим соотношением:

$$K(f) = \begin{cases} \cos \frac{\pi}{2} \cdot \frac{f-f_0-\alpha f_c/2}{\alpha f_c}, & f_0 - \alpha f_c/2 \leq f \leq f_0 + \alpha f_c/2, \\ 1, & f_0 + \alpha f_c/2 \leq f \leq f_0 + f_c - \alpha f_c/2, \\ \cos \frac{\pi}{2} \cdot \frac{f-f_0-f_c+\alpha f_c/2}{\alpha f_c}, & f_0 + f_c - \alpha f_c/2 \leq f \leq f_0 + f_c + \alpha f_c/2, \end{cases} \quad (21.1)$$

где f_0 — несущая частота передатчика; $f_c = f_s/2$ — половина частоты следования символов системы 8-VSB ATSC, равная ширине полосы нормированной АЧХ, отсчитанной по уровню $1/\sqrt{2} \approx 0,707$; $\alpha = 0,1152$.

Точное значение этой частоты следования цифровых символов равно $f_s = 4,5 \times (684/286)$ МГц = 10,762 МГц; 4,5 МГц — это частотный интервал между несущими видео и звука, который в системе аналогового телевидения NTSC равен 286-й гармонике строчной частоты (4,5 МГц = $286 \times f_H$) и, следовательно, $f_s = 684 \times f_H$.

Ширина спектра радиосигнала, измеренная на уровне 0,707 нормированной АЧХ фильтра передатчика, с одной стороны, равна половине частоты следования символов системы ATSC ($f_c f_s/2 = 5,381$ МГц), а с другой — 342 гармонике строчной частоты системы NTSC ($342 \times f_H = 5,381$ МГц). Полная ширина спектра сигнала ATSC, равная 6 МГц, определяется соотношением $(381 + 1/3) \times f_H$ и лишь на 11,5% больше расчетной полосы спектра символов системы, что характеризует высокую эффективность использования полосы частот канала связи.

Сформированный на выходе схемы VSB-модулятора аналоговый сигнал в полосе 6 МГц через преобразователь частоты и усилитель мощности передается в антенну и излучается, обеспечивая доставку программы ТВ-приемнику (рис. 21.3).

При приеме такого сигнала неподавленная несущая используется для фазовой и частотной синхронизации местного гетеродина с целью восстановления исходного суммарного сигнала. АЧХ приемного устройства аналогична АЧХ передатчика, и следовательно, с учетом сглаживающих фильтров передатчика и приемника измеренная на уровне 0,5 эквивалентной суммарной АЧХ системы ATSC ширина спектра сигнала равна половине частоты следования символов $f_c f_s / 2$ (рис. 21.106). Следовательно, эквивалентная АЧХ канала передатчик–приемник равна

$$K_{\text{пп}}(f) = K^2(f). \quad (21.2)$$

Несколько слов следует сказать о фильтрации сигнала NTSC в приемном устройстве системы ATSC, поскольку возможно нарушение приема цифрового сигнала в случае параллельной работы системы аналогового телевидения. Форма спектра радиосигнала аналогового телевидения NTSC приведена на рис. 21.11а. С целью уменьшения влияния этого сигнала в приемнике ATSC применяется гребенчатый фильтр, модуль эквивалентной амплитудно-частотной характеристики которого изображен на рис. 21.11б.

Длительность задержки в гребенчатом фильтре при этом подбирается равной $\tau = 12/f_s = 1/(57 \cdot f_H)$, а расстояние между соседними нулями модуля АЧХ равно $57 \cdot f_H$.

Точное смещение частоты пилот-сигнала системы ATSC относительно несущей видео NTSC устанавливается равным $(56+19/22) \times f_H$, поэтому спектр ATSC сдвинут на 45,8 кГц по отношению к номинальным границам радиоканала NTSC, достигая границы верхнего смежного канала на уровне примерно -40 дБ.

В стандарте ATSC в качестве критерия качества видеоизображения вводится понятие «порога видимости» (ПВ), определяемое пороговым значением отношения сигнал/шум на входе приемника. Так, для системы эфирного вещания 8-VSB теоретическим порогом определена величина С/Ш = 14,9 дБ, что примерно на 20 дБ ниже величины ПВ для аналогового телевидения.

Стандартом ATSC предусмотрен также высокоскоростной режим 16-VSB без треллисного кодирования для кабельного телевидения. В таком режиме обеспечивается передача потока цифровой информации в 38,78 Мбит/с и теоретическим порогом задается величина С/Ш = 28,3 дБ. Основные параметры двух режимов работы системы ATSC приведены в табл. 21.4.

21.2. Система цифрового наземного телевизионного вещания DVB-T

Стандарт DVB-T входит в группу европейских телевизионных стандартов DVB (Digital Video Broadcasting) [8.4, 8.12, 8.16, 8.17]. В документе ITU-R [8.10] этот стандарт обозначен как «система В». В нем принята чересстрочная развертка с частотой полукадров 50 Гц и разрешением 625 строк. Как опция предусматривается и телевидение высокой четкости (ТВЧ) с увеличенным разрешением

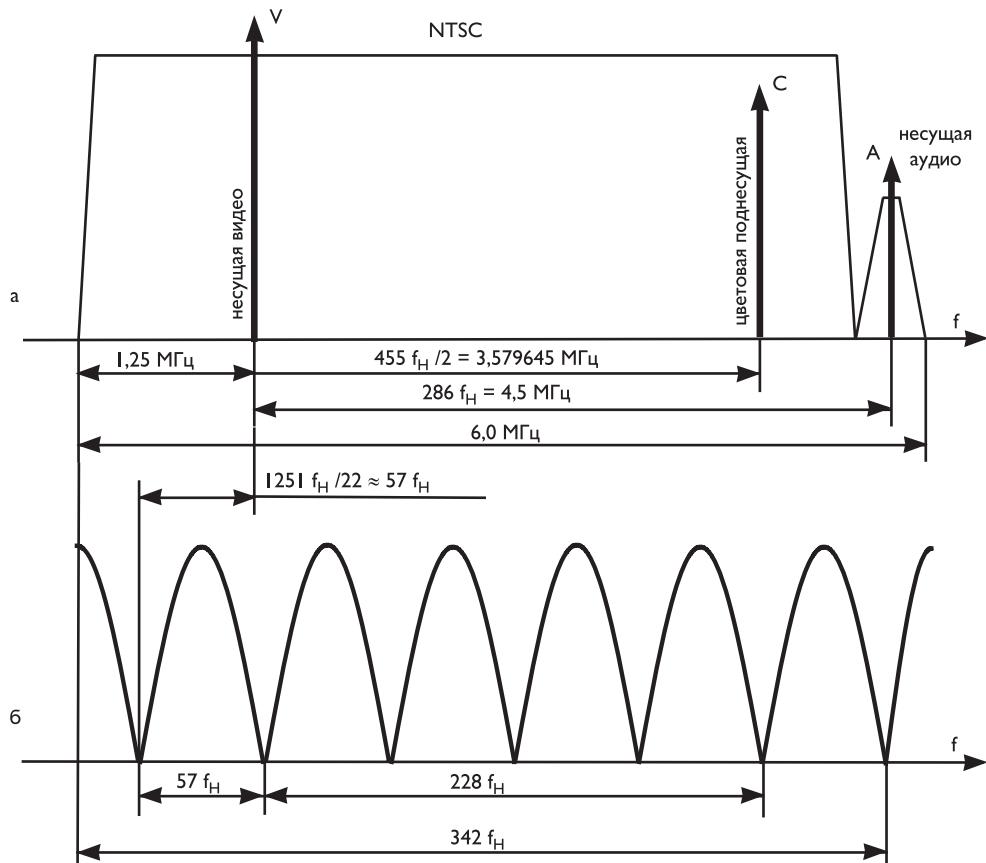


Рис. 21.11. Форма спектра радиосигнала системы аналогового телевидения NTSC (а) и АЧХ гребенчатого фильтра, используемого в приемниках системы цифрового телевидения ATSC (б)

по вертикали и горизонтали и использованием широкоформатного изображения 16 : 9. Для передачи аудио поддерживается несколько стандартов, в том числе MUSICAM и Dolby AC-3. Стандарт DVB-T предлагает три варианта ширины канала — 8, 7 и 6 МГц. В качестве системы модуляции используется схема COFDM, разработанная специально для борьбы с помехами от многолучевого приема. Для борьбы с переотраженными сигналами в DVB-T используется специальный алгоритм: информация посыпается с передатчика не непрерывным потоком, а разбивается на небольшие пакеты (символы) и передается с перерывами (защитными интервалами). Кроме того, стандарт DVB-T существенно уменьшает влияние помех, создаваемых аналоговым телевидением систем PAL/SECAM, что делает его особенно ценным для стран с большой территорией (в особенности для России).

Обобщенная структурная схема передающего комплекса системы DVB-T содержит две подсистемы: преобразования исходных данных в цифровой поток MPEG-2 и канальный эфирный модулятор (рис. 21.12).

Таблица 21.4. Параметры системы ATSC для наземного и кабельного телевидения

Параметр	Наземное телевидение 8-VSB	Кабельное телевидение 16-VSB
Ширина полосы частот канала, МГц	6	6
Превышение предельной ширины полосы частот, %	11,5	11,5
Частота следования символов, МГц	10,76	10,76
Количество битов на символ	3	4
Скорость решетчатого кодирования	2/3	—
Кодирование Рида–Соломона	(207, 187)	(207, 187)
Длина сегмента, количество символов	832	832
Синхросигнал сегмента, количество символов на один сегмент	4	4
Синхросегмент поля	1 на 313 сегментов	1 на 313 сегментов
Скорость передачи данных, Мбит/с	19,3	38,6
Фильтрация сигнала NTSC в приемнике	Гребенчатый фильтр	—
Относительная мощность пилот-сигнала, дБ	0,3	0,3
Пороговое отношение сигнал/шум C/N , дБ	14,9	28,3

В состав подсистемы формирования цифрового потока MPEG-2 входит набор кодирующих систем нескольких телевизионных программ, содержащих кодеры видео- и аудиоинформации и иных данных, цифровые потоки которых объединены программным мультиплексором. Суммарный поток цифровых данных MPEG-2 образуется на выходе транспортного мультиплексора, объединяющего потоки нескольких программных мультиплексоров.

При необходимости реализовать какой-либо вид масштабируемости динамических изображений или передавать ряд телевизионных программ с различным приоритетом в структуре передающей части системы предусматривается разделитель данных, поступающих на два тракта транспортных потоков с высоким и низким приоритетом.

Статистические кодеры, установленные в каждом из этих трактов, обеспечивают рандомизацию данных (скремблирование), реализуя квазиравномерное рассеивание энергии (рис. 21.13).

Входной транспортный поток MPEG-2 организован в виде пакетов, содержащих один синхробайт (47HEX) и 187 байтов мультиплексированных данных (рис. 21.14а).

Порядок обработки данных со стороны передачи всегда начинается со старшего бита (most significant bit, MSB) синхронизирующего байта (0100 0111, $MSB = 0$), который не скремблируется, но его значение инвертируется в каждом восьмом транспортном пакете.

Процесс дерандомизации цифровой последовательности в приемнике осуществляется по аналогичной схеме.

Порождающий генераторный полином для псевдослучайной двоичной последовательности (pseudo-random bit sequence, PRBS) имеет вид:

$$G(X) = 1 + X^{14} + X^{15}. \quad (21.3)$$

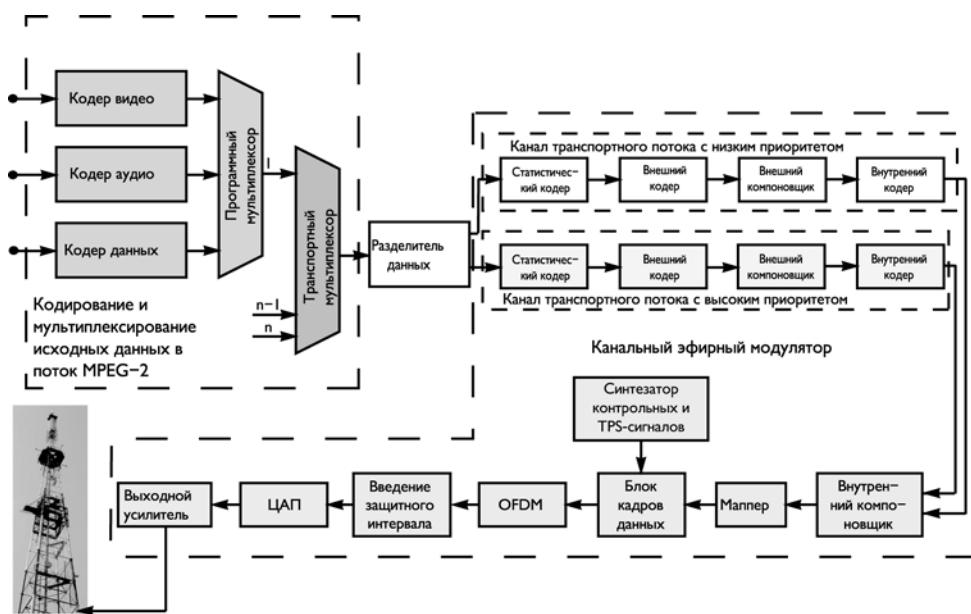


Рис. 21.12. Структурная схема передающей части системы DVB-T

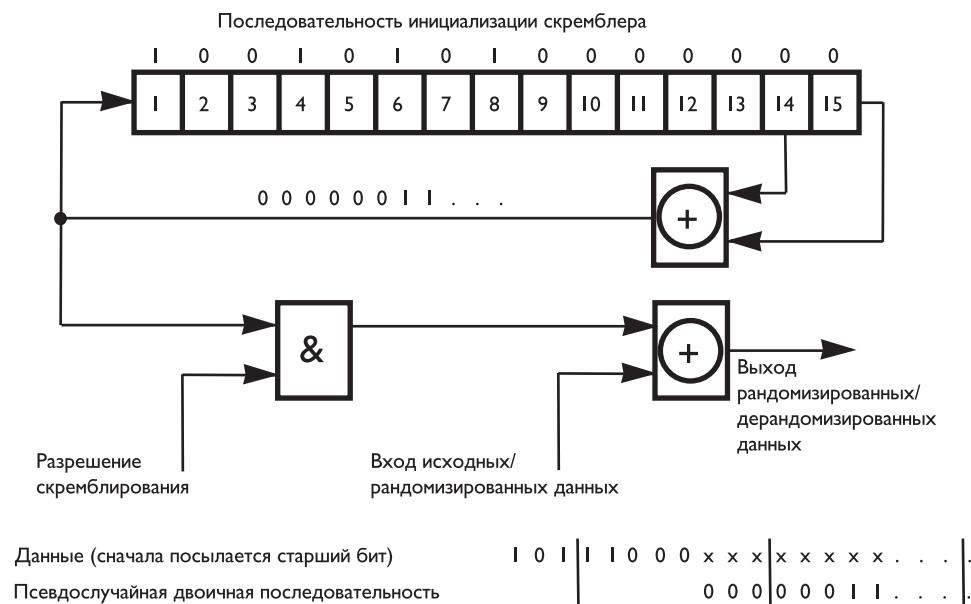


Рис. 21.13. Структурная схема статистического кодера системы DVB-T



Sinc1 — нерандомизированный инвертированный синхробайт
 Sincn — нерандомизированный синхробайт, $n = 2, 3, \dots, 8$

Рис. 21.14. Стадии процесса адаптации, распределения энергии, внешнего кодирования и компоновки: а) — пакет мультиплексированных данных MPEG-2; б) — рандомизированные байты данных; в) — рандомизированные байты данных, дополненные пакетами Рида–Соломона; г) — структура данных после внешней компоновки (степень перемешивания $I = 12$ байт)

Загрузка последовательности «100101010000000» в регистры статистического кодера, как указано на рис. 21.13, должна инициироваться при запуске каждого восьми транспортных пакетов. Для обеспечения сигнала инициализации декодера скремблирования выполняется побитовое инвертирование синхронизирующего байта MPEG-2 первого транспортного пакета в группе из восьми пакетов из 47_{HEX} (Sync) в B8_{HEX} (биты) (рис. 21.14б).

Первый бит на выходе генератора PRBS соответствует первому биту (т. е. MSB) первого байта, следующего за инвертированным синхробайтом. Для осуществления других функций синхронизации генерация PRBS продолжается на протяжении синхробайтов последующих 7 транспортных пакетов. Таким образом, последовательность PRBS должна иметь период 1503 байта.

Внешнее кодирование поступающих от статистических кодеров транспортных пакетов осуществляется укороченным кодом Рида–Соломона RS (204, 188, 8), который должен быть применен к каждому рандомизированному транспортному пакету (рис. 21.14б) для порождения пакета, защищенного от ошибок (рис. 21.14в). Кодирование Рида–Соломона должно быть также применено к синхронизирующему байту пакета, как к неинвертированному (47_{HEX}), так и к инвертированному (B8_{HEX}).

Порождающий полином кода:

$$g(x) = (x + \lambda^0)(x + \lambda^1) \dots (x + \lambda^{15}), \quad (21.4)$$

где $\lambda = 02_{HEX}$. Порождающий полином поля:

$$p(x) = x^8 + x^4 + x^3 + x^2 + 1. \quad (21.5)$$

Вслед за внешним кодированием схемой внешнего компоновщика (рис. 21.15) производится сверточная побайтовая компоновка со степенью перемешивания

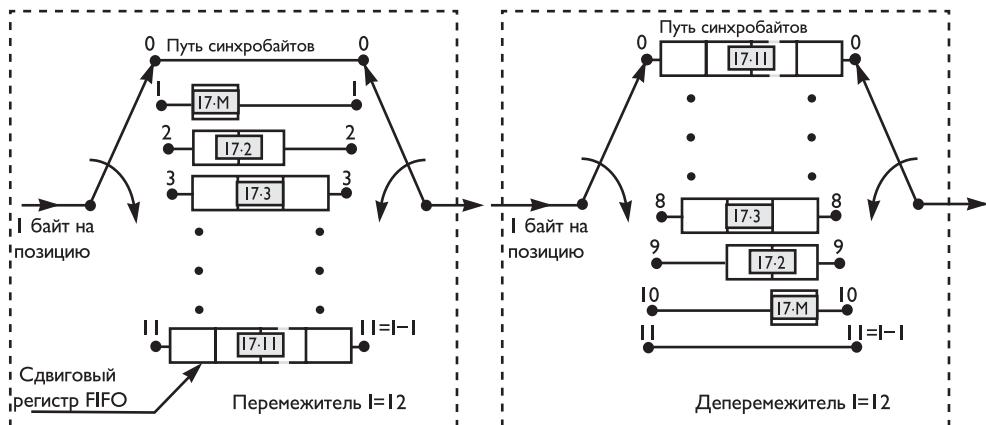


Рис. 21.15. Концептуальная диаграмма внешнего компоновщика и распаковщика $I = 12$, применяемая к пакетам, защищенным от ошибок (рис. 21.14в). В результате формируется скомпонованная структура данных (рис. 21.14г).

Байты данных, сформированные на выходе внешнего компоновщика, состоят из защищенных от ошибок пакетов и разграничены инвертированными или неинвертированными синхробайтами.

Каждое j -е звено компоновщика представляет собой регистр FIFO с размером ячеек $j \cdot M$, где $M = 17 = N/I$, $N = 204$. Ячейки FIFO содержат 1 байт, а работа входного и выходного коммутаторов синхронизирована.

Для обеспечения синхронизации Sync байты и $\overline{\text{Sync}}$ байты следует всегда адресовать в нулевое звено компоновщика (соответствующее нулевой задержке).

Распаковщик принципиально подобен компоновщику, но индексы его звеньев идут в обратном порядке (т. е. $j = 0$ соответствует наибольшей задержке).

Система *внутреннего кодирования* предусматривает сверточное кодирование с различной скоростью, основанное на материнском сверточном коде со скоростью $1/2$ с 64 состояниями решетчатой диаграммы и различных масках выкашивания. Это позволяет выбрать наиболее приемлемый уровень коррекции ошибок для конкретной сети вещания или наиболее приемлемую скорость потока данных как в неиерархическом, так и в иерархическом режиме передачи.

Порождающие полиномы материнского восьмеричного кода $G_1 = 171_{\text{ОСТ}}$ для выхода X и $G_2 = 133_{\text{ОСТ}}$ для выхода Y (рис. 21.16).

При использовании двухуровневой иерархической передачи сигналов с различным приоритетом каждое из двух параллельных кодирующих устройств канала может иметь собственную скорость кода. В дополнение к материнскому коду со скоростью $1/2$ система предусматривает скорости $2/3$, $3/4$, $5/6$ и $7/8$.

Маски выкашивания сверточного кода используются в соответствии с табл. 21.5. В этой таблице X и Y относятся к двум выходам сверточного кодирующего устройства.

Первый бит символа, к которому применено сверточное кодирование, всегда соответствует X_1 . Вслед за внутренними кодерами передающей части системы DVB-T включен внутренний компоновщик, поблочно обеспечивающий побитовую компоновку, за которой следует символная компоновка.

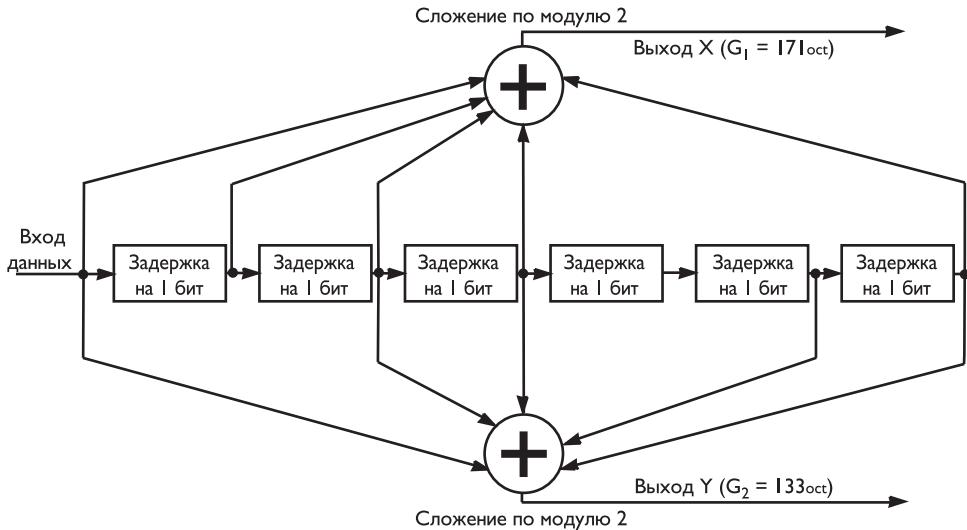


Рис. 21.16. Материнский сверточный кодер со скоростью 1/2

Таблица 21.5. Маски выкашивания для возможных скоростей кода

Скорости кода r	Маски выкашивания	Передаваемая последовательность (после параллельно-последовательного преобразования)
1/2	$X: 1$ $Y: 1$	$X_1 \quad Y_1$
2/3	$X: 1 \ 0$ $Y: 1 \ 1$	$X_1 \quad Y_1 \quad Y_2$
3/4	$X: 1 \ 0 \ 1$ $Y: 1 \ 1 \ 0$	$X_1 \quad Y_1 \quad Y_2 \quad X_3$
5/6	$X: 1 \ 0 \ 1 \ 0 \ 1$ $Y: 1 \ 1 \ 0 \ 1 \ 0$	$X_1 \quad Y_1 \quad Y_2 \quad X_3 \quad Y_4 \quad X_5$
7/8	$X: 1 \ 0 \ 0 \ 0 \ 1 \ 0 \ 1$ $Y: 1 \ 1 \ 1 \ 1 \ 0 \ 1 \ 0$	$X_1 \quad Y_1 \quad Y_2 \quad Y_3 \quad Y_4 \quad X_5 \quad Y_6 \quad X_7$

При побитовой компоновке на входы демультиплексоров поступает до двух битовых потоков, которые демультиплексируются на V подпотоков, где $V = 2$ для QPSK, $V = 4$ для 16-QAM и $V=6$ для 64-QAM. В неиерархическом режиме единый входной поток демультиплексируется на V подпотоков. В иерархическом режиме поток высокого приоритета демультиплексируется на 2 подпотока, а поток низкого приоритета демультиплексируется на $V - 2$ подпотока. Это применимо как для однородных, так и для неоднородных режимов QAM (рис. 21.17 и 21.18).

Демультиплексирование определяется как отображение входных битов x_{di} на выходные биты b_{edo} .

В неиерархическом режиме:

$$x_{di} = b_{[d_i \bmod V] \div (V/2) + 2[d_i \bmod (V/2)], d_i \div V}, \quad (21.6)$$

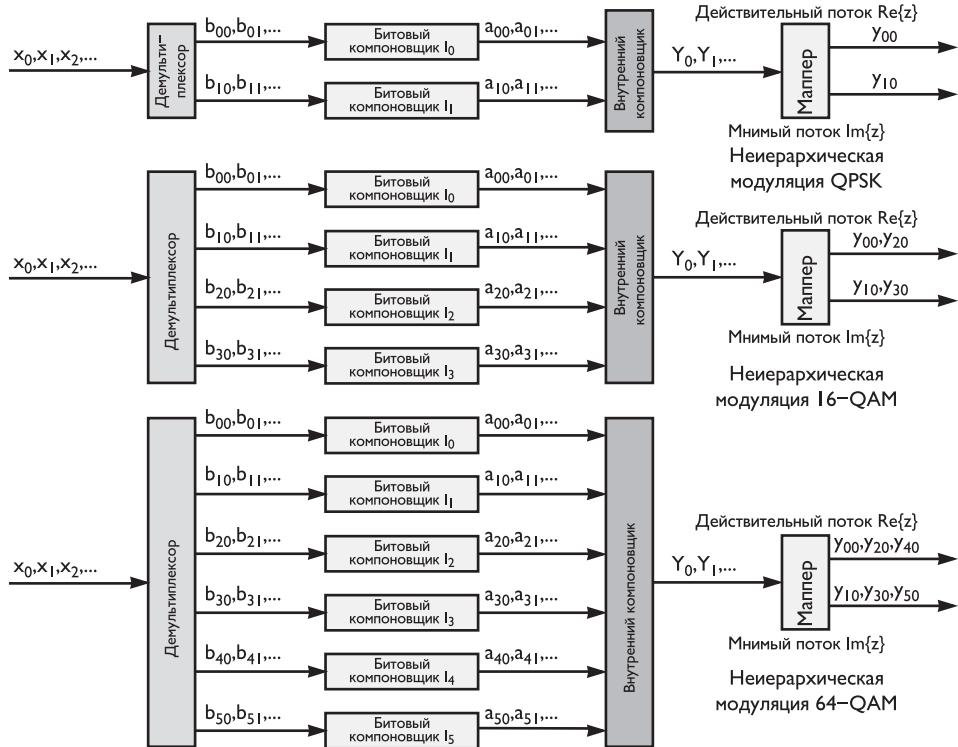


Рис. 21.17. Отображение входных битов на выходные символы модуляции при неиерархическом режиме передачи

а в иерархическом режиме:

$$\begin{aligned} x'_{d_i} &= b_{d_i \bmod 2}, d_i \bmod 2, \\ x''_{d_i} &= b_{[d_i \bmod (V-2)] \bmod ((V-2)/2)} + 2[d_i \bmod ((V-2)/2)] + 2, d_i \bmod (V-2), \end{aligned} \quad (21.7)$$

где x_{d_i} — данные, поступающие на вход демультиплексора в неиерархическом режиме, x'_{d_i} — входные данные демультиплексора высокого приоритета, x''_{d_i} — входные данные низкого приоритета в иерархическом режиме, d_i — номер входного бита, b_{ed_0} — данные на выходе демультиплексора, e — номер демультиплексированного битового потока ($0 \leq e < V$), d_0 — номер бита данного потока на выходе демультиплексора, \bmod — оператор получения остатка при целочисленном делении (деление по модулю), \bmod — оператор целочисленного деления. Результаты демультиплексирования входных данных приведены в табл. 21.6.

Битовая компоновка (см. рис. 21.17 и 21.18) осуществляется только для полезных данных. Каждый компоновщик имеет один и тот же размер блока, но последовательность компоновки в каждом случае различна. В блоке битового компоновщика помещается 126 битов. Следовательно, процесс компоновки блока повторяется двенадцать и сорок восемь раз на символ полезных данных в зависимости от используемых двух режимов OFDM-модуляции, обозначаемых как режим «2k» и режим «8k».

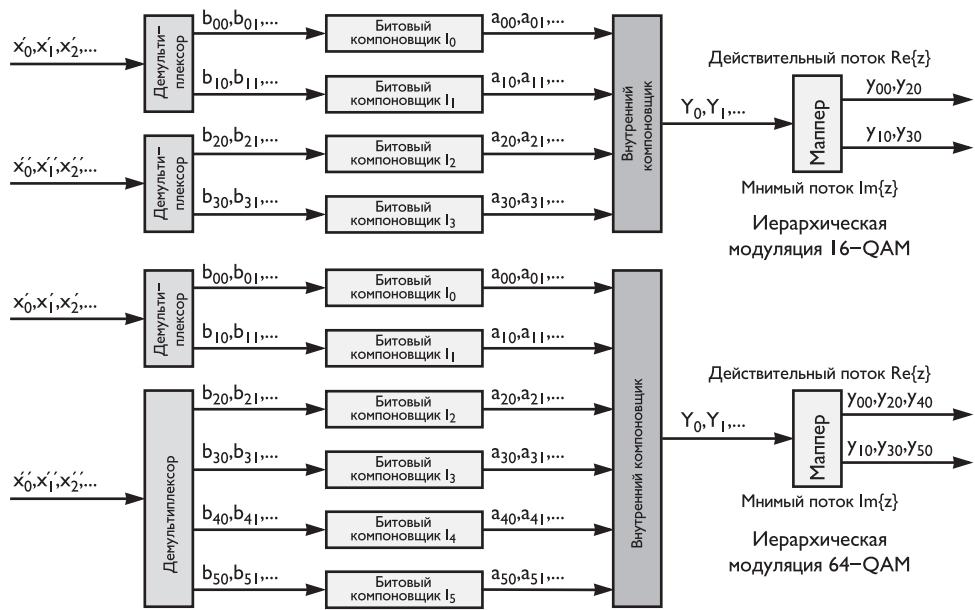


Рис. 21.18. Отображение входных битов на выходные символы модуляции при иерархическом режиме передачи

Таблица 21.6. Отображение демультиплексированных данных

Режим модуляции	Неиерархический режим		Иерархический режим	
	Входные данные	Выходные данные	Входные данные	Выходные данные
QPSK	x_0	b_{00}		
	x_1	b_{10}		
16-QAM	x_0	b_{00}	x'_0	b_{00}
	x_1	b_{20}	x'_1	b_{10}
	x_2	b_{10}	x''_0	b_{20}
	x_3	b_{30}	x''_1	b_{30}
64-QAM	x_0	b_{00}	x'_0	b_{00}
	x_1	b_{20}	x'_1	b_{10}
	x_2	b_{10}	x''_0	b_{20}
	x_3	b_{30}	x''_1	b_{40}
	x_4	b_{40}	x''_2	b_{30}
	x_5	b_{50}	x''_3	b_{50}

Для каждого битового компоновщика вектор данных входного бита определяется следующим образом:

$$B(e) = (b_{e0}, b_{e1}, b_{e2}, \dots, b_{e125}), \quad (21.8)$$

где e изменяется от 0 до $V - 1$

Скомпонованный выходной вектор данных

$$A(e) = (a_{e0}, a_{e1}, a_{e2}, \dots, a_{e125}) \quad (21.9)$$

определяется так:

$$a_{ew} = b_{eH_e(w)}, \quad w = 0, 1, 2, \dots, 125,$$

где $H_e(w)$ — функция перестановки, различная для каждого из компоновщиков:

$$\begin{aligned} I0 : H_0(w) &= w; \\ I1 : H_1(w) &= (w + 63) \bmod 126; \\ I2 : H_2(w) &= (w + 105) \bmod 126; \\ I3 : H_3(w) &= (w + 42) \bmod 126; \\ I4 : H_4(w) &= (w + 21) \bmod 126; \\ I5 : H_5(w) &= (w + 84) \bmod 126. \end{aligned}$$

Выходы V -битовых компоновщиков сгруппированы для формирования цифровых символов данных таким образом, что каждый символ из V битов состоит в точности из одного бита от каждого из V компоновщиков. Следовательно, побитовый компоновщик выдает на выход слово y' из V бит, причем старший бит y' поступает с выхода $I0$:

$$y'_w = (a_{0w}, a_{1w}, \dots, a_{(V-1)w}). \quad (21.10)$$

Выходной символьный компоновщик обеспечивает отображение V -битовых слов на $N_{\max} = 1512$ (режим $2k$) или $N_{\max} = 6048$ (режим $8k$) активных несущих в символе OFDM.

Скомпонованный вектор данных $Y = (y_0, y_1, y_2, \dots, y_{N_{\max}-1})$, содержащий 12 групп по 126 слов выходных данных в режиме $2k$ и 48 групп по 126 слов данных в режиме $8k$, определяется как:

$$\begin{aligned} y_{H(q)} &= y'_q \text{ для четных символов при } q = 0, \dots, N_{\max} - 1, \\ y_q &= y'_{H(q)} \text{ для нечетных символов при } q = 0, \dots, N_{\max} - 1, \end{aligned}$$

где функция перестановки $H(q)$ определяется при помощи следующего алгоритма:

```

 $q = 0;$ 
 $\text{for } (i = 0; i < M_{\max}; i = i + 1) \{$ 
 $H(q) = (i \bmod 2) \cdot 2^{N_r-1} + \sum_{j=0}^{N_r-2} R_i(j) \cdot 2^j;$ 
 $\text{if } (H(q) < N_{\max}) \text{ then } q = q + 1 \}.$ 

```

$M_{\max} = 2048$ в режиме $2k$ и $M_{\max} = 8192$ в режиме $8k$, а векторы R_i вычисляются из векторов R'_i с использованием битовых перестановок (табл. 21.7); при этом двоичные слова R'_i , состоящие из $(N_r - 1)$ битов, $N_r = \log_2 M_{\max}$, принимают следующие значения:

$$\begin{aligned} i = 0, 1 : \quad R'_i[N_r - 2, N_r - 3, \dots, 1, 0] &= 0, 0, \dots, 0, 0; \\ i = 2 : \quad R'_i[N_r - 2, N_r - 3, \dots, 1, 0] &= 0, 0, \dots, 0, 1; \\ 2 < i < M_{\max} : \quad R'_i[N_r - 3, N_r - 4, \dots, 1, 0] &= R'_{i-1}[N_r - 2, N_r - 3, \dots, 2, 1]; \\ &\text{в режиме } 2k : R'_i[9] = R'_{i-1}[0] \oplus R'_{i-1}[3]; \\ &\text{в режиме } 8k : R'_i[11] = R'_{i-1}[0] \oplus R'_{i-1}[1] \oplus R'_{i-1}[4] \oplus R'_{i-1}[6]. \end{aligned}$$

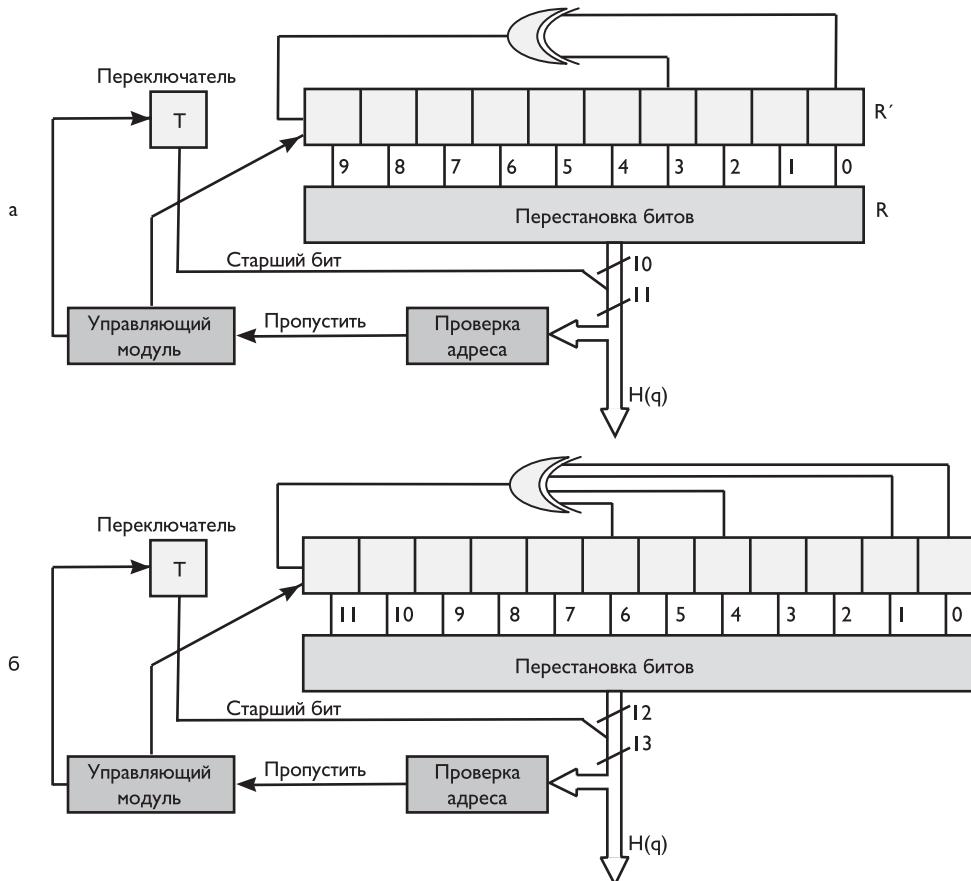


Рис. 21.19. Порождающие схемы внутренних символьных компоновщиков для режима $2k$ (а) и для режима $8k$ (б)

Таблица 21.7. Битовые перестановки в двух режимах системы DVB-T

Режим $2k$	Положение бита R'_i	9	8	7	6	5	4	3	2	1	0		
	Положение бита R_i	0	7	5	1	8	2	6	9	3	4		
Режим $8k$	Положение бита R'_i	11	10	9	8	7	6	5	4	3	2	1	0
	Положение бита R_i	5	11	3	0	10	8	6	9	2	4	1	7

Схематические блок-диаграммы алгоритмов, порождающих функции перестановки, представлены на рис. 21.19.

Выход внутреннего компоновщика подключен к мапперу, отображающему данные на созвездие сигнала OFDM (см. главу 20). При формировании кадра OFDM используется отображение Грея либо на однородные (QPSK, 16-QAM, 64-QAM), либо на неоднородные диаграммы (16-QAM, 64-QAM). Потоки данных, представляющие действительные $\text{Re}\{z\}$ и мнимые $\text{Im}\{z\}$ составляющие неиерархической комплексной модуляции, определяющие модуль и фазу каждой несущей OFDM-сигнала, обозначены на рис. 21.17, а для иерархической комплексной модуляции — на рис. 21.18.

В блоке кадров данных формируется последовательность, организованная в виде кадров, состоящая из 68 символов, используемых для формирования OFDM-сигнала. Четыре кадра составляют один суперкадр.

Каждый символ имеет длительность T_S и состоит из $K = 6817$ несущих в режиме $8k$ и из $K = 1705$ несущих в режиме $2k$. Он включает полезную часть длительности T_U и защитный интервал длительности T_G , который является циклическим продолжением полезной части и вводится перед ней (табл. 21.8).

Таблица 21.8. Числовые значения параметров OFDM для режимов $8k$ и $2k$

Параметр		Режим $8k$	Режим $2k$
Число несущих		6817	1705
Длительность T_U	Для каналов 8 МГц	896 мкс	224 мкс
	Для каналов 7 МГц	1024 мкс	256 мкс
	Для каналов 6 МГц	1194,667 мкс	298,6667 мкс
Интервал между несущими $1/T_U$	Для каналов 8 МГц	1116 Гц	4464 Гц
	Для каналов 7 МГц	976,563 Гц	3906,25 Гц
	Для каналов 6 МГц	837,054 Гц	334,8214 Гц
Интервал между максимальной и минимальной несущими $(K_{\max} - K_{\min})/T_U$	Для каналов 8 МГц	7,61 МГц	7,61 МГц
	Для каналов 7 МГц	6,66 МГц	6,66 МГц
	Для каналов 6 МГц	5,71 МГц	5,71 МГц

Примечание: Цифры, указанные наклонным шрифтом, — приближенные; $K_{\min} = 0$; $K_{\max} = 6816$ для режима $8k$ и $K_{\max} = 1704$ для режима $2k$.

Длительность защитного интервала может иметь четыре значения (табл. 21.9), длительность обычно определяется как количеством элементарных периодов T , так и в микросекундах. Элементарный период для канала 8 МГц равен $T = 7/64$ мкс, для канала 7 МГц — $T = 1/8$ мкс и для канала 6 МГц — $T = 7/48$ мкс.

Таблица 21.9. Длительность символьной части и защитных интервалов

Параметр		Режим $8k$				Режим $2k$			
T_G/T_U		1/4	1/8	1/16	1/32	1/4	1/8	1/16	1/32
T_U	Канал	$8192 \times T$				$2048 \times T$			
	8 МГц	896 мкс				224 мкс			
	7 МГц	1024 мкс				256 мкс			
	6 МГц	1194,667 мкс				298,6667 мкс			
T_G	Канал	$2048 \times T$	$1024 \times T$	$512 \times T$	$256 \times T$	$512 \times T$	$256 \times T$	$128 \times T$	$64 \times T$
	8 МГц	224 мкс	112 мкс	56 мкс	28 мкс	56 мкс	28 мкс	14 мкс	7 мкс
	7 МГц	256 мкс	128 мкс	64 мкс	32 мкс	64 мкс	32 мкс	16 мкс	8 мкс
	6 МГц	298,7 мкс	149,3 мкс	74,7 мкс	37,3 мкс	74,7 мкс	37,3 мкс	18,7 мкс	9,3 мкс
$T_S = T_U + T_G$	Канал	$10240 \times T$	$9216 \times T$	$8704 \times T$	$8448 \times T$	$2560 \times T$	$2304 \times T$	$2176 \times T$	$2112 \times T$
	8 МГц	1120 мкс	1008 мкс	952 мкс	924 мкс	280 мкс	252 мкс	238 мкс	231 мкс
	7 МГц	1280 мкс	1152 мкс	1088 мкс	1056 мкс	320 мкс	288 мкс	272 мкс	264 мкс
	6 МГц	1493 мкс	1344 мкс	1269 мкс	1232 мкс	373 мкс	336 мкс	317 мкс	308 мкс

Символы в кадре OFDM пронумерованы от 0 до 67 и содержат информацию о передаваемых данных и опорную информацию. Поскольку сигнал OFDM включает множество отдельно модулированных несущих, каждый символ можно рассматривать разделенным на ячейки, каждая из которых соответствует модуляции, выполненной на одной несущей в течение одного символа.

В дополнение к передаваемым данным кадр OFDM содержит специальную служебную информацию в виде

- рассеянных пилотных несущих (scattered pilots);
- повторяющихся пилотных несущих (continual pilots);
- несущих сигнализации о параметрах передачи (TPS).

Пилотные несущие используются для кадровой, частотной, временной синхронизации, оценки канала, идентификации режима передачи, а также для отслеживания фазового шума.

Излучаемый сигнал описывается следующим выражением:

$$S(t) = \operatorname{Re} \left\{ e^{j \cdot 2\pi \cdot f_c \cdot t} \cdot \sum_{m=0}^{\infty} \sum_{l=0}^{67} \sum_{k=K_{\min}}^{K_{\max}} c_{m,l,k} \cdot \Psi_{m,l,k}(t) \right\}, \quad (21.11)$$

где

$$\Psi_{m,l,k}(t) = \begin{cases} e^{j \cdot 2\pi \cdot \frac{k'}{T_U} \cdot (t - T_G - l \cdot T_s - 68 \cdot m \cdot T_s)} & \text{при } (l + 68 \cdot m) \cdot T_s \leq t \leq (l + 68 \cdot m + 1) \cdot T_s, \\ 0 & \text{иначе,} \end{cases}$$

k — номер несущей; l — номер символа OFDM; m — номер передаваемого кадра; $k' = k - (K_{\max} + K_{\min})/2$ — индекс несущей относительно центральной частоты; K — число переданных несущих; T_s — длительность символа; T_U — длительность активной части символа; T_G — длительность защитного интервала; f_c — центральная частота радиосигнала; $C_{m,l,k}$ — комплексный символ k -й несущей l -го символа в m -м кадре OFDM.

Величины $C_{m,l,k}$ — это нормализованные значения модуляции точки Z созвездия согласно применяемому алфавиту модуляции. Коэффициенты нормализации определяются соотношением $E[C \cdot C^*] = 1$ и приведены в табл. 21.10.

Таблица 21.10. Нормализующие коэффициенты для символов данных

Тип модуляции	Коэффициент неоднородности	Нормализующий коэффициент
QPSK		$C = Z/\sqrt{2}$
16-QAM	$\alpha = 1$	$C = Z/\sqrt{10}$
	$\alpha = 2$	$C = Z/\sqrt{20}$
	$\alpha = 4$	$C = Z/\sqrt{52}$
64-QAM	$\alpha = 1$	$C = Z/\sqrt{42}$
	$\alpha = 2$	$C = Z/\sqrt{60}$
	$\alpha = 4$	$C = Z/\sqrt{108}$

В рассеянных или повторяющихся пилотных ячейках опорная информация передается на «усиленном» уровне мощности, так что $E[C \cdot C^*] = 16/9$.

Период повторения положений рассеянных пилотных ячеек — четыре символа OFDM. Количество несущих, соответствующее передаче полезных данных, остается постоянным от символа к символу: в режиме $2k$ — 1512 несущих, и в режиме $8k$ — 6048 несущих.

Повторяющиеся и рассеянные пилотные ячейки модулируются на основе псевдослучайной двоичной последовательности (PRBS), w_k , в соответствии с индексом их несущей k . Порождающий полином для PRBS имеет вид (рис. 21.20):

$$X^{11} + X^2 + 1.$$

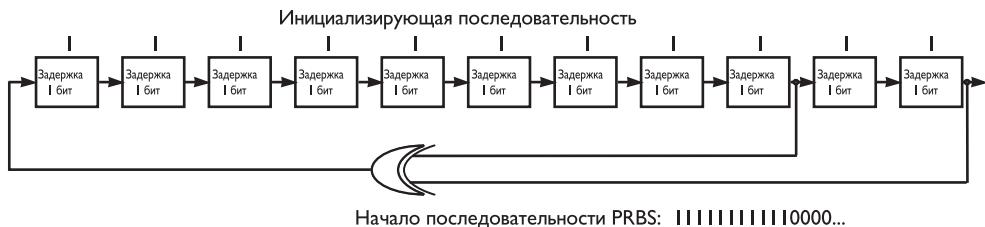


Рис. 21.20. Алгоритм порождения последовательности PRBS

Эта последовательность управляет также начальной фазой информации о параметрах передачи (TPS).

Опорная информация передается в каждом символе в пилотных ячейках. Модуляция рассеянных пилотных ячеек представима следующим образом:

$$\operatorname{Re}\{C_{m,l,k}\} = 4/3 \cdot 2(1/2 - w_k), \operatorname{Im}\{C_{m,l,k}\} = 0,$$

где m — номер кадра, l — номер символа, k — номер несущей.

Для символа номера $0 \leq l \leq 67$ несущие, для которых номер k , принадлежащий подмножеству $\{kK_{\min} + 3 \cdot (l \bmod 4) + 12p, p \text{ — неотрицательное целое число}\}$, принимающее все возможные значения, большие или равные нулю, при условии, что результирующее значение $k \in [K_{\min}, K_{\max}]\}$.

В дополнение к рассеянным пилотным ячейкам вводится 177 повторяющихся пилотных ячеек в режиме $8k$ и 45 — в режиме $2k$ (табл. 21.11).

Таблица 21.11. Номера повторяющихся пилотных несущих

Режим $2k$	Режим $8k$
0 48 54 87 141 156 192 201 255 279 282 333 432 450 483 525 531 618 636 714 759 65 780 804 873 888 918 939 942 969 984 1050 1101 1107 1110 1137 1140 1146 1206 1269 1323 1377 1491 1683 1704 1752 1758 1791 1845 1860 1896 1905 1959 1983 1986 2037 2136 2154 2187 2229 2235 2322 2340 2418 2463 2469 2484 2508 2577 2592 2622 2643 2646 2673 2688 2754 2805 2811 2814 2841 2844 2850 2910 2973 3027 3081 3195 3387 3408 3456 3462 3495 3549 3564 3600 3609 3663 3687 3690 3741 3840 3858 3891 3933 3939 4026 4044 4122 4167 4173 4188 4212 4281 4296 4326 4347 4350 4377 4392 4458 4509 4515 4518 4545 4548 4554 4614 4677 4731 4785 4899 5091 5112 5160 5166 5199 5253 5268 5304 5313 5367 5391 5394 5445 5544 5562 5595 5637 5643 5730 5748 5826 5671 5877 5892 5916 5985 6000 6030 6051 6054 6081 6096 6162 6213 6219 6222 6249 6252 6256 6318 6361 6435 6489 6603 6795 6816	

Параметры сигнализации TPS передаются параллельно на 17 несущих в режиме $2k$ и на 68 несущих в режиме $8k$. Каждая несущая TPS в одном и том же символе передает один и тот же дифференциальный кодированный информационный бит. Номера несущих TPS приведены в табл. 21.12.

Несущие TPS передают следующую информацию:

- тип модуляции, включая значение α созвездия QAM;
- информацию об иерархии;

- величину защитного интервала;
- скорость внутреннего кода;
- режим передачи ($2k$ или $8k$);
- номер кадра в суперкадре.

Таблица 21.12. Номера несущих TPS

Режим $2k$	Режим $8k$
34 50 209 346 413 569 595 688 790 901 1219 1262 1286 1469 1594 1687	34 50 209 346 413 569 595 688 790 901 1073 1219 1262 1286 1469 1594 1687 1738 1754 1913 2050 2117 2273 2299 2392 2494 2605 2777 2923 2966 2990 3173 3298 3391 3442 3458 3617 3754 3821 3977 4003 4096 4198 4309 4481 4627 4670 4694 4877 5002 5095 5146 5162 5321 5458 5525 5681 5707 5800 5902 6013 6185 6331 6374 6398 6581 6706 6799

Каждый символ OFDM передает один бит TPS. Каждый блок TPS, соответствующий одному кадру OFDM, содержит 68 битов, которые определены следующим образом:

- 1 бит инициализации;
- 16 синхронизирующих битов;
- 37 информационных битов;
- 14 избыточных битов для защиты от ошибок.

Из 37 информационных битов используются только 23, а оставшиеся 14 битов зарезервированы для использования в будущем и установлены в ноль.

Информация о параметрах передачи передается в соответствии со следующими правилами. Ячейки TPS передаются на «нормальном» уровне мощности, т. е. передаются с энергией, равной среднему значению по всем ячейкам данных.

Каждая несущая TPS модулирована дифференциальной BPSK (DBPSK) и передает одно и то же сообщение. DBPSK инициализируется в начале каждого блока TPS.

Следующее правило применимо к дифференциальной модуляции пилотных ячеек TPS на несущей k символа l ($l > 0$) в кадре m :

- если $s_l = 0$, то $\text{Re}\{C_{m,l,k}\} = \text{Re}\{C_{m,l-1,k}\}$; $\text{Im}\{C_{m,l,k}\} = 0$;
- если $s_l = 1$, то $\text{Re}\{C_{m,l,k}\} = -\text{Re}\{C_{m,l-1,k}\}$; $\text{Im}\{C_{m,l,k}\} = 0$.

Абсолютная модуляция несущих TPS в первом символе кадра вычисляется из опорной последовательности w_k следующим образом:

$$\text{Re}\{C_{m,l,k}\} = 2^{(1/2-w_k)},$$

$$\text{Im}\{C_{m,l,k}\} = 0.$$

Величины и назначение битов TPS s_0-s_{67} следующие:

- Первый бит s_0 используется для инициализации системы определения параметров передачи в соответствии с опорной последовательностью.
- Биты s_1-s_{16} характеризуют синхронизирующее слово, равное для первого и третьего блоков TPS 001101011101110 и для второго и четвертого блоков TPS 1100101000010001 в каждом суперкадре.

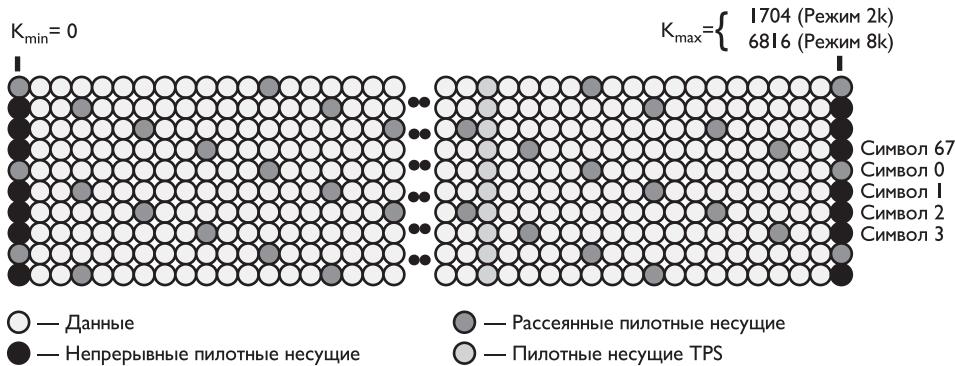


Рис. 21.21. Структура кадра системы DVB-T

- Биты $s_{17}—s_{22}$ определяют длину TPS в двоичном исчислении и в настоящее время имеют значение 010111.
- Биты $s_{23}—s_{24}$ обозначают номер кадра в каждом суперкадре: «00» — 1-й кадр; «01» — 2-й кадр; «10» — 3-й кадр и «11» — 4-й кадр.
- Биты $s_{25}—s_{26}$ определяют используемое созвездие OFDM: «00» — QPSK; «01» — 16-QAM; «10» — 64-QAM. Код «11» зарезервирован.
- Битами $s_{27}—s_{29}$ передается информация об иерархии: «000» — неиерархическая; если передача является иерархической, то «001» — $\alpha = 1$; «010» — $\alpha = 2$; «011» — $\alpha = 4$. Коды от «100» до «111» зарезервированы.
- Биты $s_{30}—s_{32}$ характеризуют скорость кода в потоке с высоким приоритетом (HP), а биты $s_{33}—s_{35}$ — с низким приоритетом (LP): «000» — скорость кода $1/2$; «001» — $2/3$; «010» — $3/4$; «011» — $5/6$; «100» — $7/8$. Коды от «101» до «111» зарезервированы.
- Биты $s_{36}—s_{37}$ определяют величину защитного интервала: «00» — $1/32$; «01» — $1/16$; «10» — $1/8$ и «11» — $1/4$.
- Биты $s_{38}—s_{39}$ характеризуют режим передачи: «00» — $2k$; «01» — $8k$. Коды «10» и «11» зарезервированы.
- Биты $s_{40}—s_{53}$ — нулевые, зарезервированы.
- Биты $s_{54}—s_{67}$ несут информацию о защите от ошибок, содержащую 14 битов четности сокращенного кода Бозе—Чодхури—Хокенхема BCH (67, 53, $t = 2$).

Порождающий полином кода:

$$h(x) = x^{14} + x^9 + x^8 + x^6 + x^5 + x^4 + x^2 + x + 1. \quad (21.12)$$

Сокращенный код BCH может быть получен с помощью добавления 60 обнуленных бит перед битовой информацией на входе кодера BCH (127, 113, $t = 2$). После кодирования BCH эти нулевые биты должны быть удалены, что приводит к длине кодового слова BCH в 67 битов.

Структура кадра системы DVB-T в составе информационных данных, непрерывных, рассеянных и TPS пилотных несущих приведена на рис. 21.21.

Структура кадра OFDM допускает передачу целого числа 204-байтовых пакетов Рида—Соломона в суперкадре OFDM, при этом не требуется передача какой-либо дополнительной информации (табл. 21.13).

В табл. 21.13 приведена промоделированная производительность системы DVB-T, предусматривающая оценку канала при отсутствии фазового шума для всех ви-

Таблица 21.13. Количество пакетов Рида–Соломона в суперкадре OFDM для всех комбинаций защитного интервала, скоростей кода и форм модуляции

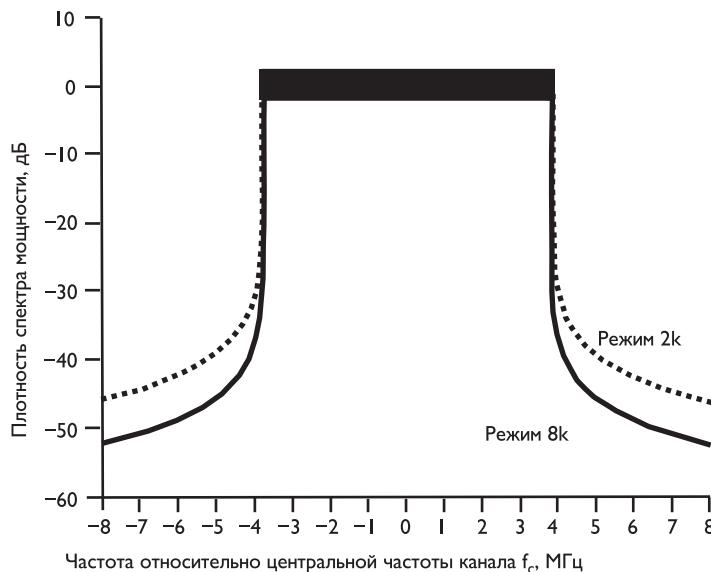
Скорость кода	QPSK		16-QAM		64-QAM	
	Режим 2k	Режим 8k	Режим 2k	Режим 8k	Режим 2k	Режим 8k
1/2	252	1008	504	2016	756	3024
2/3	336	1344	672	2688	1008	4032
3/4	378	1512	756	3024	1134	4536
5/6	420	1680	840	3360	1260	5040
7/8	441	1764	882	3528	1323	5292

дов неиерархических систем и каналов Гаусса, Райса и Релея, расчетные полезные скорости битового потока системы DVB-T (полоса канала 8 МГц). В этой таблице приведены требуемые отношения сигнал/шум (C/N), необходимые для того, чтобы BER при неиерархической передаче достиг значения 2×10^{-4} после декодера Витерби для всех комбинаций скоростей кодирования и типов модуляции, что определяет квазибезошибочный прием (QEF), означающий менее одного случая неисправимой ошибки в час ($BER=10^{-11}$ на входе демультиплексора MPEG-2). Соответствующие достижимые скорости передачи полезной информации (определенные после декодера Рида–Соломона) представлены в зависимости от величины защитного интервала.

Таблица 21.14. Отношение С/N, необходимое для квазибезошибочного приема информации сигналов DVB-T и полезная скорость битового потока

		Виды каналов			Скорость битового потока (Мбит/с)					
					Защитный интервал					
Модуляция	Скорость кода	Гаусса	Райса	Релея	1/4	1/8	1/16	1/32		
		1/2	3,1	3,6	5,4	4,98	5,53	5,65	6,03	
QPSK	2/3	4,9	5,7	8,4	6,64	7,37	7,81	6,04		
	3/4	5,9	6,6	10,7	7,46	8,29	8,78	9,05		
	5/6	6,9	6,0	13,1	8,29	9,22	9,76	10,05		
	7/8	7,7	8,7	16,3	8,71	9,68	10,25	10,56		
	16-QAM	1/2	8,8	9,6	11,2	9,95	11,06	11,71	12,06	
64-QAM		2/3	11,1	11,6	14,2	13,27	14,75	15,61	16,09	
		3/4	12,5	13,0	16,7	14,93	16,59	17,56	18,10	
		5/6	13,5	14,4	19,3	16,59	18,43	19,52	20,11	
		7/8	13,9	15,0	22,8	17,42	19,35	20,49	21,11	
64-QAM	1/2	14,4	14,7	16,0	14,93	16,59	17,56	18,10		
	2/3	16,5	17,1	19,3	19,91	22,12	23,42	24,13		
	3/4	18,0	18,6	21,7	22,39	24,88	26,35	27,14		
	5/6	19,3	20,0	25,3	24,88	27,65	29,27	30,16		
	7/8	20,1	21,0	27,9	26,13	29,03	30,74	31,67		

Теоретически вычисленный спектр сигнала DVB-передатчика для стандарта 8 МГц иллюстрируется на рис. 21.22 ($T_G T_U / 4$). Поскольку длительность символа OFDM больше, чем обратная величина интервала между несущими, главная доля мощности спектральной плотности каждой несущей уже, чем удвоенный интервал несущей. Таким образом, в пределах номинальной полосы 7,608258 МГц

Рис. 21.22. Мaska спектра системы DVB-T при $T_G T_U / 4$

для режима $8k$ или 7,611607 МГц для режима $2k$ спектральная плотность непостоянна.

В табл. 21.15 приведены основные параметры системы DVB-T для канала 8 МГц.

Таблица 21.15. Основные эксплуатационные параметры стандарта DVB-T

Параметр	Значение параметра	
	Режим 8k	Режим 2k
Число несущих в символе OFDM	6817	1705
Число несущих полезных данных в символе OFDM	6048	1512
Число рассеянных пилот-сигналов в кадре OFDM	524	131
Число повторяющихся пилот-сигналов в кадре OFDM	177	45
Число несущих сигнализации о параметрах передачи в кадре OFDM	68	17
Длительность полезной части символа OFDM, мкс	896	224
Разнос соседних несущих, Гц	1116	4464
Разнос между крайними несущими в символе OFDM, МГц	7,608258	7,611607
Частота следования символов данных, МГц	6,75	6,75
Ширина полосы частот канала, МГц	8 (6, 7)	
Число битов на символ	2, 4, 6	

Таблица 21.15 (окончание)

Кодирование кода Рида–Соломона		(204, 188, $t = 8$)						
Длительность псевдослучайной последовательности, байт		1503						
Скорость передачи полезных данных, Мбит/с		4,98... 31,67						
Скорость внутреннего кода		1/2, 2/3, 3/4, 5/6, 7/8						
Модуляция несущих		QPSK, 16-QAM, 64-QAM						
Относительный защитный интервал T_G/T_U	1/4	1/8	1/16	1/32	1/4	1/8	1/16	1/32
Длительность полезной части символа T_U , мкс	896				224			
Длительность защитного интервала T_G , мкс	224	112	56	28	56	28	14	7
Длительность символа $T_S = T_G + T_U$, мкс	1120	1008	952	924	280	252	238	231
Максимальный разнос между передатчиками в одночастотной сети (SFN), км	67,2	33,6	16,8	8,4	16,8	8,4	4,2	2,1

21.3. Усовершенствованная система цифрового наземного ТВ-вещания DVB-T2

В феврале 2006 года в рамках консорциума DVB был создан исследовательский комитет (Study Mission), который должен был оценить потенциал различных технологий и приступить к разработке стандарта DVB-T2, реализующего ряд коммерческих требований:

- обеспечение приема на существующие домашние антенны;
- переход на новый стандарт не должен требовать изменения инфраструктуры передающей сети;
- обеспечение, как минимум, 30–50%-го прироста пропускной способности каналов относительно DVB-T при идентичных условиях передачи;
- улучшение работы одночастотных сетей (SFN);
- реализация возможности сосуществования в одном РЧ-канале услуг, передаваемых с разной степенью помехоустойчивости;
- повышение гибкости использования полосы и частот;
- наличие механизма снижения отношения пиковой и средней мощности передаваемого сигнала и др.

Все эти рекомендации были выполнены при разработке стандарта DVB-T2. К середине 2010 г. решение о внедрении этого стандарта приняли администрации Австрии, Великобритании, Германии, Дании, Индии, Испании, Италии, Казахстана, Словакии, Украины, Финляндии, Чешской Республики, Швеции, Южной Африки.

Обобщенная модель системы DVB-T2 представлена на рис. 21.23. На входе(ах) системы могут быть один или множество транспортных потоков MPEG-2 (Trans-

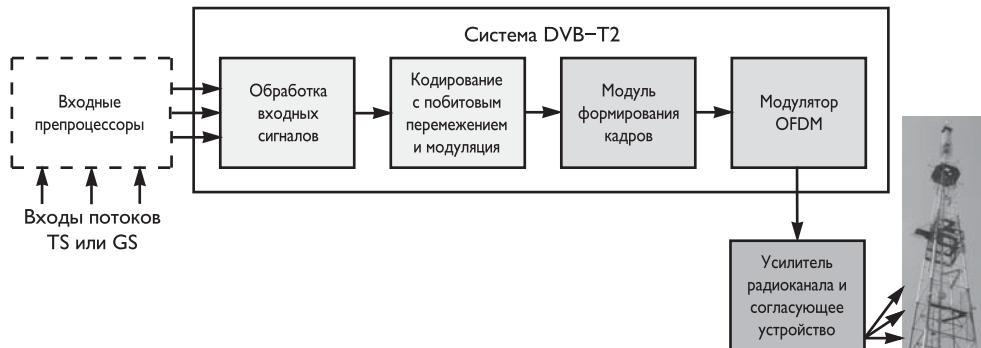


Рис. 21.23. Обобщенная структурная схема системы DVB-T2

port Stream — TS) и/или один или множество обобщенных потоков (Generic Stream — GS). Входной препроцессор, не являющийся частью системы, может включать в себя сплиттер входных сигналов или демультиплексор транспортных потоков для распределения сервисов на входы системы, представляющие собой один или множество логических потоков данных. Далее они передаются в индивидуальных магистралях физического уровня (Physical Layer Pipe — PLP).

На выход системы подается обычно один сигнал для передачи по одномуциальному радиоканалу. Однако дополнительно система может генерировать набор выходных сигналов для передачи на группу передающих антенн в так называемом режиме передачи MISO (Multiple Input, Single Output).

Входной препроцессор обеспечивает согласование входных сигналов с блоком их обработки в системе DVB-T2, содержащим один, если используется единичный входной поток SIS (режим типа «А»), или несколько модулей адаптации режима при наличии множества входных потоков MIS (режим типа «В»). Предусмотрено использование следующих форматов входных данных:

- транспортный поток (TS) [8.19];
- обобщенный инкапсулированный поток (Generic Encapsulated Stream — GSE) [8.20];
- обобщенный непрерывный поток (Generic Continuous Stream — GCS, поток пакетов переменной длины, в котором модулятор не обращает внимания на границы пакетов);
- обобщенный поток пакетов фиксированной длины (Generic Fixed-length Packets Stream — GFPS); данная форма сохранена для совместимости со стандартом спутникового телевидения DVB-S2 [8.21].

Подсистема входного интерфейса должна преобразовывать входной поток во внутренний логический битовый формат (рис. 21.24).

Входной интерфейс должен обеспечить чтение битов поля данных размера DFL (Data Field Length):

$$0 < DFL < (K_{bch} - 80), \quad (21.13)$$

где K_{bch} — число битов, поступающих на вход кодера БЧХ. 10-байтовый (80 битов) заголовок базовой полосы вставляется перед полем данных.

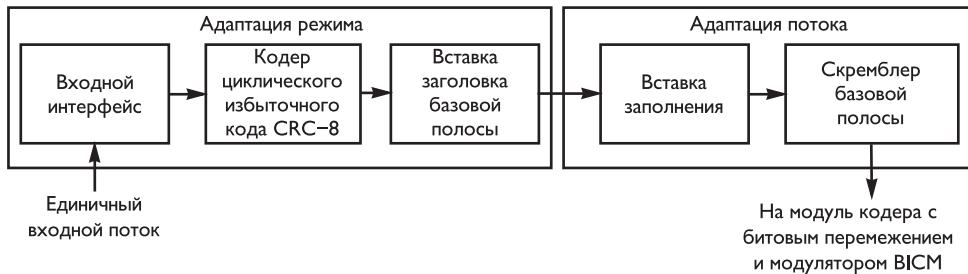


Рис. 21.24. Структура входного интерфейса стандарта DVB-T2

В случае если используется несколько входных потоков данных, в составе каждого канала адаптации режима вводятся блоки синхронизации входного потока и компенсации задержки. Для цели синхронизации вводится 2 или 3 байта в специальное поле ISSY (Input Stream Synchronisation), в котором передается значение счетчика, тактируемого сигналом тактовой частоты модулятора ($1/T$, где T — элементарный период, определяемый шириной полосы используемого радиоканала).

Требования, установленные стандартом, предусматривают постоянство скоростей битовых потоков на выходе мультиплексора передатчиков и на входе демультиплексора приемников, а также постоянство сквозной задержки. В некоторых входных сигналах транспортных потоков может присутствовать большая доля нулевых пакетов. Для устранения излишних накладных расходов передачи нулевые пакеты удаляются, но таким образом, чтобы они могли быть вставлены в приемнике в точности на то же самое место, где они находились первоначально. С этой целью после каждого переданного пользовательского пакета UP (User Packet) вводится поле размерностью в 1 байт — DNP (Deleted Null-Packets), характеризующее удаленные нулевые пакеты.

Кодирование CRC-8 применяется для детектирования ошибок только в обычном режиме для пакетированных потоков. Реализация проверки с помощью циклических избыточных CRC-кодов позволяет детектировать ошибки передачи на стороне приемника. С этой целью вычисленный CRC-8 код вставляется после пользовательского пакета UP.

Этот код формируется сдвиговым регистром и определяется полиномом:

$$G_8(x) = x^8 + x^7 + x^6 + x^4 + x^2 + 1. \quad (21.14)$$

Адаптация режима завершается вставкой перед полем данных 10-байтового заголовка базовой полосы фиксированной длины, определяющего формат поля данных. Предусмотрено два варианта заголовка базовой полосы:

- один (рис. 21.25а) применяется при обычных режимах, когда используются либо обобщенный поток пакетов фиксированной длины GFPS, либо транспортный поток TS, а также при обобщенном инкапсулированном потоке GSE или обобщенном непрерывном потоке GCS;
- другой (рис. 21.25б) используется при режимах с повышенной эффективностью для потоков TS или GSE; при этом CRC8 для пользовательских пакетов не вычисляется, единичное поле синхронизации ISSY вставляется

опционально в заголовок базовой полосы, величина длины пользовательского пакета UPL не передается.

Параметры поля типа режима адаптации (MATYPE) и формат заголовка базовой полосы приведены в табл. 21.16 и 21.17, соответственно.

Таблица 21.16. Параметры поля MATYPE заголовка базовой полосы

Параметры					
1-й байт					
TS/GS (2 бита)	SIS/MIS (1 бит)	CCM/ACM (1 бит)	ISSYI (1 бит)	DNP (1 бит)	EXT (2 бита)
00 — GFPS			0 — выключено	0 — выключено	зарезервировано
01 — GCS	0 — MIS	0 — ACM	1 — включено	1 — включено	
10 — GSE	1 — SIS	1 — CCM			
11 — TS					

2-й байт					
Если $SIS/MIS = 0$, то второй байт является идентификатором входных потоков (ISI), в противном случае второй байт зарезервирован для последующего использования.					

Примечания: CCM/ACM — неадаптивное и адаптивное кодирование и модуляция.

Если ISSYI = 1, должен быть активирован счетчик (22 бита), синхронизированный с частотой дискретизации модулятора $1/T$.

Адаптация потоков (рис. 21.24) обеспечивает:

- планирование для режима с множеством входных потоков MIS;
- заполнение кадра BBFRAME (Base Band Frame) до постоянной длительности (K_{bch} битов) и/или передачи внутриполосной сигнализации;
- скремблирование для рассеивания энергии.

Таблица 21.17. Параметры заголовка базовой полосы адаптации режима

Поле	Размер (байты)	Описание
MATYPE	2	См. табл. 21.16
UPL	2	Длина пользовательского пакета в битах, в пределах 0... 65535
DFL	2	Длина поля данных в битах, в пределах 0... 53760
SYNC	1	Копия синхробайта пользовательского пакета
SYNCD	2	Расстояние в битах от начала поля DATA FIELD до первого полного пользовательского пакета (UP) поля данных. $SYNCD = 0$ означает, что первый UP выровнен по началу поля данных, $SYNCD = 65535$ означает, что ни один из UP не начинается в поле DATA FIELD.
CRC-8 MODE	1	Операция НЕ-И-ИЛИ (XOR), примененная к полям CRC-8 (1 байт) и MODE (1 байт). CRC-8 — код обнаружения ошибок, применяемый к первым 9 байтам заголовка. MODE (8 битов) должно обозначать: 0 — обычный режим; 1 — режим с повышенной эффективностью; прочие значения — зарезервированы для последующего использования.

Выходной поток должен представлять собой кадр базовой полосы BBFRAME, показанный на рис. 21.26а. Для формирования информации о передаваемом множестве потоков используется динамическая сигнализация о принятом решении,

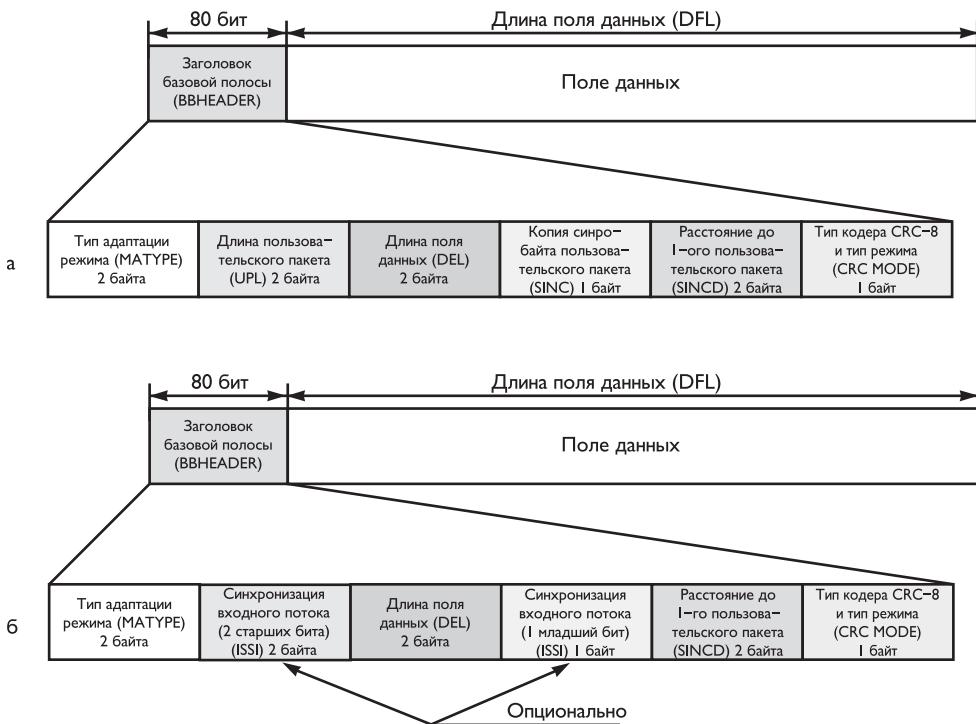


Рис. 21.25. Варианты заголовков базовой полосы стандарта DVB-T2

в каких ячейках итогового сигнала передаются данные, относящиеся к конкретным входным потокам, т. е. реализуется точная компоновка кадровой структуры сигнала DVB-T2.

В процессе адаптации потока специальный модуль отображения ячеек компонует ячейки каждого входного потока физического уровня PLP и соответствующую сигнализацию в массивы, соответствующие символам OFDM.

Значение K_{bch} , включающее заголовок базовой полосы, поле данных и поле заполнения, зависит от скорости используемой коррекции ошибок. Заполнение может применяться, когда предназначенных для передачи данных недостаточно для полного заполнения кадра, или если в кадре должно быть размещено целое число пакетов данных.

При этом нулевые биты (K_{bch} -DFL-80) добавляются после поля данных и итоговый кадр должен иметь постоянную длину K_{bch} битов.

Отображение поля заполнения для внутриполосной сигнализации при передаче типа «A» реализует следующие функции:

- фиксирует тип поля заполнения;
- указывает на количество следующих перед ним суперкадров;
- информирует о количестве ячеек OFDM от начала одного субслайса до начала следующего субслайса той же самой магистрали физического уровня PLP в том же самом канале соответствующего кадра;

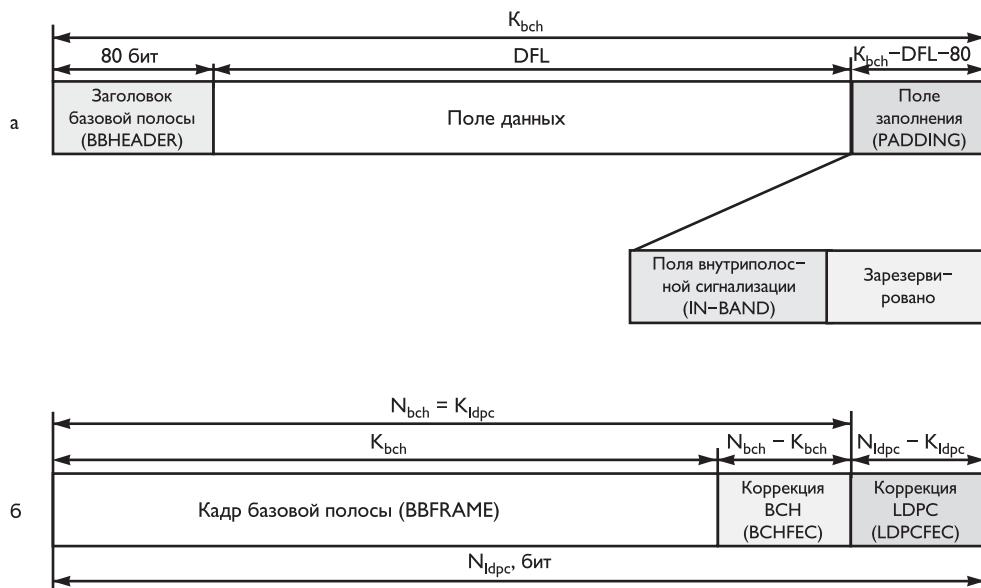


Рис. 21.26. Формат поля заполнения на выходе адаптера потока при передаче типа «А» (а) и формат данных до побитового перемежения (б)

- фиксирует параметры идентификатора (ID) стартовой последовательности кадра с частотно-временным разделением на слайсы (TFS);
- сигнализирует стартовую позицию текущей PLP в соответствующем кадре;
- задает количество блоков FEC для поля PLP_ID, содержащихся в кадре перемежения, который отображается на следующий кадр.

Рандомизация полного кадра BBFRAME осуществляется, начиная со старшего бита и заканчивая последним битом K_{bch} . Последовательность скремблирования осуществляется сдвиговым регистром, аналогичным регистру, используемому в стандарте DVB-T (рис. 21.13), в котором не используется схема «И» разрешения скремблирования. Порождающий полином соответствует соотношению (21.3).

На рис. 21.27 приведена структурная схема кодирования с побитовым перемежением и модуляции информации в системе DVB-T2 (BICM — Bit Interleaved Coding and Modulation).

Подсистема кодирования с опережающей коррекцией ошибок (FEC) выполняет внешнее кодирование (BCH), внутреннее кодирование (LDPC) и побитовое перемежение. Входной поток состоит из кадров BBFRAME, а выходной поток — из кадров FECFRAME.

Проверочные биты (BCHFEC) систематического внешнего кода BCH присоединяются после поля BBFRAME, а проверочные биты (LDPCFEC) внутреннего кодера LDPC присоединяются после поля BCHFEC, как показано на рис. 21.26б.

Используется два формата данных до побитового перемежения:

- для основных кадров FECFRAME — $N_{ldpc} = 64\ 800$ битов,
- для сокращенных кадров FECFRAME — $N_{ldpc} = 16\ 200$ битов.

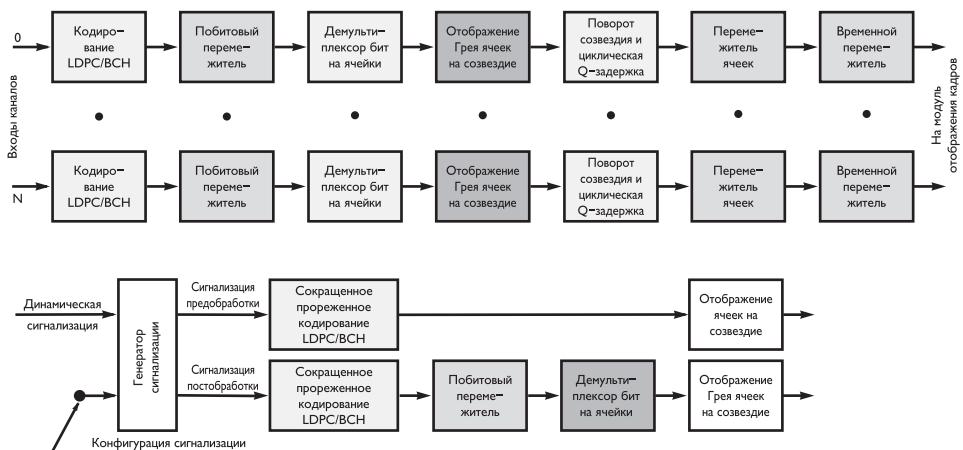


Рис. 21.27. Структурная схема кодирования с побитовым перемежением и модуляции

Параметры кодирования для обычных и сокращенных кадров приведены соответственно в табл. 21.18а и б.

Таблица 21.18а. Параметры кодирования для основных кадров при $N_{ldpc} = 64\,800$ битов

LDPC-код	Блок BCH-кода K_{bch} некодированный	Блок BCH-кода N_{bch} кодированный; блок LDPC-кода K_{ldpc} некодированный	Коррекция t ошибок BCH-кода	$N_{bch} - K_{bch}$	Блок LDPC-кода N_{ldpc} кодированный
1/2	32208	32400	12	192	64 800
3/5	38688	38880	12	192	64 800
2/3	43040	43200	10	160	64 800
3/4	48408	48600	12	192	64 800
4/5	51648	51840	12	192	64 800
5/6	53840	54000	10	160	64 800

Таблица 21.18б. Параметры кодирования для сокращенных кадров при $N_{ldpc} = 16\,200$ битов

Идентификатор LDPC-кода	Блок BCH-кода K_{bch} некодированный	Блок BCH-кода N_{bch} кодированный; блок LDPC-кода K_{ldpc} некодированный	Коррекция t ошибок BCH-кода	$N_{bch} - K_{bch}$	Эффективная скорость LDPC-кода $K_{ldpc}/16\,200$	Блок LDPC-кода N_{ldpc} кодированный
1/4*	3072	3240	12	168	1/5	16 200
1/2	7032	7200	12	168	4/9	16 200
3/5	9552	9720	12	168	3/5	16 200
2/3	10632	10800	12	168	2/3	16 200
3/4	11712	11880	12	168	11/15	16 200
4/5	12432	12600	12	168	7/9	16 200
5/6	13152	13320	12	168	37/45	16 200

* Эта кодовая скорость используется только для защиты сигнализации предобработки, но не для защиты данных.

Для генерации безошибочного пакета код внешнего кодирования BCH ($N_{\text{bch}}, K_{\text{bch}}$), корректирующий t ошибок, применяется к каждому кадру BBFRAME. Полиномы кода BCH для обычных и сокращенных кадров приведены в табл. 21.19. Для получения порождающего полинома кода коррекции t ошибок кодера BCH необходимо перемножить первые t полиномов, соответствующих размеру кадра N_{ldpc} .

$\mathbf{M} = (m_{K_{\text{bch}}-1}, m_{K_{\text{bch}}-2}, \dots, m_1, m_0)$ информационных битов кодируются в кодовое слово кода BCH следующим образом:

- полином сообщения $m(x) = m_{K_{\text{bch}}-1}x^{K_{\text{bch}}-1} + m_{K_{\text{bch}}-2}x^{K_{\text{bch}}-2} + \dots + m_1x + m_0$ умножается на $x^{N_{\text{bch}}-K_{\text{bch}}}$;
- $x^{N_{\text{bch}}-K_{\text{bch}}}m(x)$ делится на $g(x)$, порождающий полином. Остаток равен $d(x) = d_{N_{\text{bch}}-K_{\text{bch}}-1}x^{N_{\text{bch}}-K_{\text{bch}}-1} + \dots + d_1x + d_0$;
- создается выходное кодовое слово \mathbf{I} , которое формирует информационное слово \mathbf{I} для кодирования с помощью кодов LDPC в соответствии со следующим:

$$\begin{aligned} \mathbf{I} &= (i_0, i_1, \dots, i_{N_{\text{bch}}-1}) = \\ &= (m_{K_{\text{bch}}-1}, m_{K_{\text{bch}}-2}, \dots, m_1, m_0, d_{N_{\text{bch}}-K_{\text{bch}}-1}, d_{N_{\text{bch}}-K_{\text{bch}}-2}, \dots, d_1, d_0). \end{aligned}$$

Примечание: равнозначный полином кодового слова имеет вид

$$c(x) = x^{N_{\text{bch}}-K_{\text{bch}}}m(x) + d(x).$$

Таблица 21.19. Полиномы кода BCH для основных и сокращенных кадров FECFRAME

	Основные кадры $N_{\text{ldpc}} = 64\,800$	Сокращенные кадры $N_{\text{ldpc}} = 16\,200$
$g_1(x)$	$1 + x^2 + x^3 + x^5 + x^{16}$	$1 + x + x^3 + x^5 + x^{14}$
$g_2(x)$	$1 + x + x^4 + x^{5+6} + x^8 + x^{16}$	$1 + x^6 + x^8 + x^{11} + x^{14}$
$g_3(x)$	$1 + x^2 + x^3 + x^4 + x^5 + x^7 + x^8 + x^9 + x^{10} + x^{11} + x^{16}$	$1 + x + x^2 + x^6 + x^9 + x^{10} + x^{14}$
$g_4(x)$	$1 + x^2 + x^4 + x^6 + x^9 + x^{11} + x^{12} + x^{14} + x^{16}$	$1 + x^4 + x^7 + x^8 + x^{10} + x^{12} + x^{14}$
$g_5(x)$	$1 + x + x^2 + x^3 + x^5 + x^8 + x^9 + x^{10} + x^{11} + x^{12} + x^{16}$	$1 + x^2 + x^4 + x^6 + x^8 + x^9 + x^{11} + x^{13} + x^{14}$
$g_6(x)$	$1 + x^2 + x^4 + x^5 + x^7 + x^8 + x^9 + x^{10} + x^{12} + x^{13} + x^{14} + x^{15} + x^{16}$	$1 + x^3 + x^7 + x^8 + x^9 + x^{13} + x^{14}$
$g_7(x)$	$1 + x^2 + x^5 + x^6 + x^8 + x^9 + x^{10} + x^{11} + x^{13} + x^{15} + x^{16}$	$1 + x^2 + x^5 + x^6 + x^7 + x^{10} + x^{11} + x^{13} + x^{14}$
$g_8(x)$	$1 + x + x^2 + x^5 + x^6 + x^8 + x^9 + x^{12} + x^{13} + x^{14} + x^{16}$	$1 + x^5 + x^8 + x^9 + x^{10} + x^{11} + x^{14}$
$g_9(x)$	$1 + x^5 + x^7 + x^9 + x^{10} + x^{11} + x^{16}$	$1 + x + x^2 + x^3 + x^9 + x^{10} + x^{14}$
$g_{10}(x)$	$1 + x + x^2 + x^5 + x^7 + x^8 + x^{10} + x^{12} + x^{13} + x^{14} + x^{16}$	$1 + x^3 + x^6 + x^9 + x^{11} + x^{12} + x^{14}$
$g_{11}(x)$	$1 + x^2 + x^3 + x^5 + x^9 + x^{11} + x^{12} + x^{13} + x^{16}$	$1 + x^4 + x^{11} + x^{12} + x^{14}$
$g_{12}(x)$	$1 + x + x^5 + x^6 + x^7 + x^9 + x^{11} + x^{12} + x^{16}$	$1 + x + x^2 + x^3 + x^5 + x^6 + x^7 + x^8 + x^{10} + x^{13} + x^{14}$

Кодер внутреннего кодирования LDPC обрабатывает выход внешнего кодирования, $I = (i_0, i_1, \dots, i_{K_{\text{ldpc}}-1})$, как блок информации размером K_{ldpc} , и систематически кодирует его в кодовое слово Λ размером N_{ldpc} , где

$$\Lambda = (\lambda_0, \lambda_1, \lambda_2, \dots, \lambda_{N_{\text{ldpc}}}) = (i_0, i_1, \dots, i_{K_{\text{ldpc}}-1}, p_0, p_1, \dots, p_{N_{\text{ldpc}}-K_{\text{ldpc}}-1}) \quad (21.15)$$

Параметры кода LDPC ($N_{\text{ldpc}}, K_{\text{ldpc}}$) приведены в табл. 21.18. Задача внутреннего кодирования — определить $N_{\text{ldpc}}-K_{\text{ldpc}}$ проверочных битов ($p_0, p_1, \dots, p_{N_{\text{ldpc}}-K_{\text{ldpc}}-1}$) для каждого блока информационных битов $K_{\text{ldpc}}, (i_0, i_1, \dots, i_{K_{\text{ldpc}}-1})$.

При обработке основных кадров ($N_{\text{ldpc}} = 64\,800$) используются специальные таблицы А.1–А.6, приведенные в Приложении А стандарта DVB-T2 [8.17]. В качестве примера рассмотрим алгоритм обработки для скорости кода, равной 2/3. В этом случае используется таблица А.3, начальный участок которой приведен в табл. 21.20.

Таблица 21.20. Скорость кода 2/3, $N_{\text{ldpc}} = 64\,800$

0	10491	16043	506	12826	8065	8226	2767	240	18673	9279	10579	20928
1	17819	8313	6433	6224	5120	5824	12812	17187	9940	13447	13825	18483
2	17957	6024	8681	18628	12794	5915	14576	10970	12064	20437	4455	7151
3	19777	6183	9972	14536	8182	17749	11341	5556	4379	17434	15477	18532
4	4651	19689	1608	659	16707	14335	6143	3058	14618	17894	20684	5306
5	9778	2552	12096	12369	15198	16890	4851	3109	1700	18725	1997	15882
6	486	6111	13743	11537	5591	7433	15227	14145	1483	3887	17431	12430
7	20647	14311	11734	4180	8110	5525	12141	15761	18661	18441	10569	8192
8	3791	14759	15264	19918	10132	9062	10010	12786	10675	9682	19246	5454
9	19525	9485	7777	19999	8378	9209	3163	20232	6690	16518	716	7353
10	4588	6709	20202	10905	915	4317	11073	13576	16433	368	3508	21171
11	14072	4033	19959	12608	631	19494	14160	8249	10223	21504	12395	4322
12	13800	14161										
13	2948	9647										
14	14693	16027										
15	20506	11082										
.....												

Последовательность выполнения такого алгоритма обработки информации заключается в нижеследующем.

Инициализируется $p_0 = p_1 = p_2 = \dots = p_{N_{\text{ldpc}}-K_{\text{ldpc}}-1} = 0$;

Первый из информационных битов, i_0 , аккумулируется по адресам проверочных битов, определенным в первой строке табл. 21.20:

$$\begin{array}{lll}
 p_0 = p_0 \oplus i_0 & p_{12826} = p_{12826} \oplus i_0 & p_{240} = p_{240} \oplus i_0 \\
 p_{10491} = p_{10491} \oplus i_0 & p_{8065} = p_{8065} \oplus i_0 & p_{18673} = p_{18673} \oplus i_0 \\
 p_{16043} = p_{16043} \oplus i_0 & p_{8226} = p_{8226} \oplus i_0 & p_{9279} = p_{9279} \oplus i_0 \\
 p_{506} = p_{506} \oplus i_0 & p_{2767} = p_{2767} \oplus i_0 & p_{10579} = p_{10579} \oplus i_0 \\
 p_{20928} = p_{20928} \oplus i_0 & &
 \end{array}$$

Для следующих 359 информационных битов, i_m , $m = 1, 2, \dots, 359$, значения i_m аккумулируются по адресам проверочных битов $\{x + m \bmod 360 \times Q_{\text{ldpc}} \bmod (N_{\text{ldpc}} - K_{\text{ldpc}})$, где x обозначает адрес сумматора проверочных битов, соответствующего первому биту i_0 , а Q_{ldpc} — постоянная, зависящая от кодовой скорости, определенная в табл. 21.21. В данном случае, для скорости 2/3 $Q_{\text{ldpc}} = 60$

и для информационного бита i_1 , к примеру, выполняются следующие операции:

$$\begin{aligned}
 p_{60} &= p_{60} \oplus i_1 & p_{12886} &= p_{12886} \oplus i_1 & p_{300} &= p_{300} \oplus i_1 \\
 p_{10551} &= p_{10551} \oplus i_1 & p_{8125} &= p_{8125} \oplus i_1 & p_{18733} &= p_{18733} \oplus i_1 \\
 p_{16103} &= p_{16103} \oplus i_1 & p_{8286} &= p_{8286} \oplus i_1 & p_{9339} &= p_{9339} \oplus i_1 \\
 p_{566} &= p_{566} \oplus i_1 & p_{2827} &= p_{2827} \oplus i_1 & p_{10639} &= p_{10639} \oplus i_1 \\
 p_{20988} &= p_{20988} \oplus i_1
 \end{aligned}$$

Для 361-го информационного бита i_{360} адреса сумматоров проверочных битов приведены во второй строке табл. 21.20. Аналогично, адреса сумматоров проверочных битов следующих 359 информационных битов i_m , $m = 361, 362, \dots, 719$, получают с помощью формулы $\{x + (m \bmod 360) \times Q_{ldpc}\} \bmod (N_{ldpc} - K_{ldpc})$, где x обозначает адрес сумматора проверочных битов, соответствующего информационному биту i_{360} , т. е. элементам во второй строке табл. 21.20.

Аналогично, для каждой новой группы из 360 информационных битов используется новая строка табл. 21.20 для поиска адресов сумматоров проверочных битов.

После того как закончатся все информационные биты, итоговые проверочные биты формируют путем последовательного выполнения, начиная с $i = 1$, следующих операций:

$$p_i = p_i \oplus p_{i-1}, \quad i = 1, 2, \dots, N_{ldpc} - K_{ldpc} - 1.$$

Итоговое значение p_i , $i = 0, 1, \dots, N_{ldpc} - K_{ldpc} - 1$ равно проверочному биту p_i .

Таблица 21.21. Значения Q_{ldpc} для основных и сокращенных кадров

Скорость кода	Основные кадры	Сокращенные кадры
1/4	—	36
1/2	90	25
3/5	72	18
2/3	60	15
3/4	45	12
4/5	36	10
5/6	30	8

Внутреннее кодирование сокращенных кадров осуществляется аналогичным образом, но вместо таблиц, приведенных в Приложении А стандарта DVB-T2, используются таблицы, приведенные в Приложении В этого же стандарта.

Выход А (21.15) кодера LDPC подвергается побитовому перемежению, которое состоит из перемежения проверочных битов, за которым следует перемежение со сдвигом начала столбцов. Перемежение проверочных битов осуществляется в соответствии со следующим правилом (информационные биты не перемежаются):

$$\begin{aligned}
 u_i &= \lambda_i, \quad 0 \leq i < K_{ldpc}, \quad u_{K_{ldpc}+360t+s} = \lambda_{K_{ldpc}+Q_{ldpc}t+s}, \\
 0 &\leq s < K_{ldpc}, \quad 0 \leq t < Q_{ldpc},
 \end{aligned} \tag{21.16}$$

где Q_{ldpc} определено в табл. 21.21.

Конфигурация перемежения со сдвигом начала столбцов для каждого формата модуляции определена табл. 21.22.

Таблица 21.22. Структура побитового перемежителя

Модуляция	Строки N_r		Столбцы N_c
	$N_{ldpc} = 64\ 800$	$N_{ldpc} = 16\ 200$	
16-QAM	8100	2025	8
64-QAM	5400	1350	12
256-QAM	4050	—	16
	—	2025	8

При перемежении со сдвигом начала столбцов биты данных u_i с перемежителяя проверочных битов последовательно записываются в перемежитель со сдвигом начала столбцов по столбцам и последовательно считаются по строкам (старший бит заголовка считывается первым). Запись стартовой позиции каждого столбца сдвигается на величину t_c в соответствии с табл. 21.23 и рис. 21.28.

Работа перемежителя осуществляется следующим образом.

Входной бит u_i , $0 \leq i < N_{ldpc}$, записывается в столбец c_i , строку r_i устройства перемежения, где $c_i = i \bmod N_r$,

$$r_i = i + t_{c_i} \bmod N_r.$$

Выходной бит v_j , $0 \leq j < N_{ldpc}$, считывается из строки r_j , столбец c_j , где $r_j = j \bmod N_c$, $c_j = j \bmod N_r$.

Таблица 21.23. Параметр сдвига начала столбцов t_c

Модуляция	Столбцы N_c	N_{ldpc}	Параметр сдвига t_c по столбцам															
			0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
16-QAM	8	64 800	0	0	2	4	4	5	7	7	—	—	—	—	—	—	—	
		16 200	0	0	0	1	7	20	20	21	—	—	—	—	—	—	—	
64-QAM	12	64 800	0	0	2	2	3	4	4	5	5	7	8	9	—	—	—	
		16 200	0	0	0	2	2	2	3	3	3	6	7	7	—	—	—	
256-QAM	16	64 800	0	2	2	2	2	3	7	15	16	20	22	22	27	27	28	32
	8	16 200	0	0	0	1	7	20	20	21	—	—	—	—	—	—	—	

Таким образом, для 64-QAM и $N_{ldpc} = 64\ 800$, порядок битов на выходе перемежения со сдвигом по столбцам должен быть следующим:

$$(v_0, v_1, v_2, \dots, v_{64799}) = (u_0, u_{5400}, u_{16198}, \dots, u_{53992}, u_{59231}, u_{64790}).$$

Более длинный список индексов с правой стороны определяет все 12 столбцов:

$$0, 5400, 16198, 21598, 26997, 32396, 37796, 43195, 48595, 53993, 59392, 64791, \dots, 5399, 10799, 16197, 21597, 26996, 32395, 37795, 43194, 48594, 53992, 59391, 64790.$$

Каждый кадр FECFRAME преобразовывается в кодированный и модулированный FEC блок с опережающей коррекцией ошибок. Для этого входные биты сначала демультиплексируются на параллельные модулирующие значения ячеек, и затем эти модулирующие значения отображаются на значения созвездия. Количество ячеек выходных данных и эффективное количество битов на ячейку η_{mod} заданы табл. 21.24.

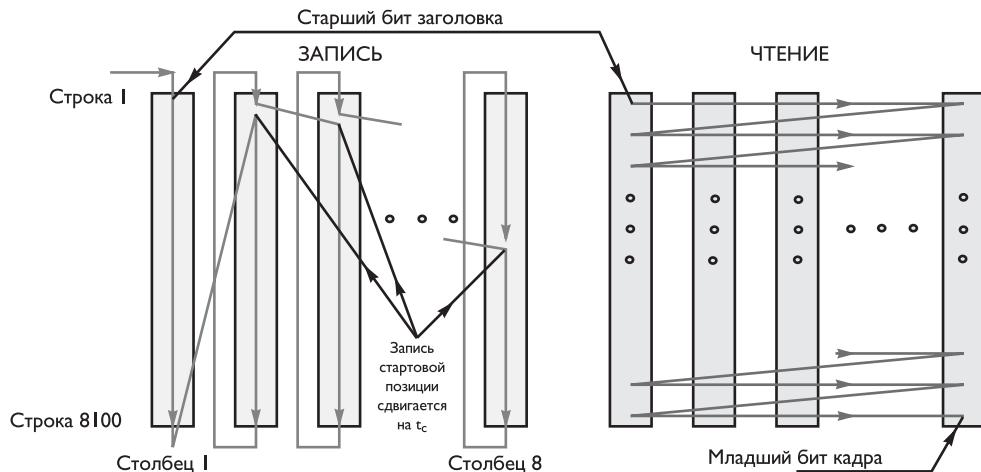


Рис. 21.28. Схема побитового перемежения для кадров FECFRAME основной длины и модуляции 16-QAM

Таблица 21.24. Параметры побитового отображения на созвездия

Длина блока LDPC-кода (N_{ldpc})	Режим модуляции	η_{mod}	Число выходных ячеек данных
64 800	256-QAM	8	8100
	64-QAM	6	10800
	16-QAM	4	16 200
	QPSK	2	32400
16 200	256-QAM	8	2025
	64-QAM	6	2700
	16-QAM	4	4050
	QPSK	2	8100

Битовый поток v_{di} от побитового перемежителя демультиплексируется на $N_{substreams}$ подпотоков (табл. 21.25).

Таблица 21.25. Количество подпотоков в демультиплексоре

Модуляция	N_{ldpc}	Количество подпотоков, $N_{substreams}$
QPSK	Любое	2
16-QAM	Любое	8
64-QAM	Любое	12
256-QAM	64 800	16
	16 200	8

Демультиплексирование определяется как отображение подвергнутых побитовому перемежению входных битов v_{di} на выходные биты $b_{e,do}$ на выходе демультиплексора (табл. 21.26).

Параметры демультиплексирования битов на подпотоки для кодовой скорости 3/5 приведены в таблице 12b стандарта DVB-T2.

За исключением модуляции QPSK ($N_{ldpc} = 64 800$ или $16 200$) и 256-QAM ($N_{ldpc} = 16 200$), слова длиной $N_{substreams}$ разбиваются на два модулирующих

значения ячеек длиной $\eta_{mod}N_{substreams}/2$ на выходе демультиплексора. Первые $\eta_{mod}N_{substreams}/2$ бита $[b_{0,do} \dots b_{N_{substreams}/2-1,do}]$ образуют первую пару выходных модулирующих значений ячеек $[y_{0,2do} \dots y_{\eta_{mod}-1,2do}]$, а оставшиеся выходные биты $[b_{N_{substreams}/2,do} \dots b_{N_{substreams}-1,do}]$ образуют второе выходное модулирующее значение ячеек $[y_{0,2do+1} \dots y_{\eta_{mod}-1,2do+1}]$, которое подается на модуль отображения созвездий.

Таблица 21.26. Параметры демультиплексирования битов на подпотоки для всех кодовых скоростей (за исключением 3/5)

Формат модуляции	Число входных/выходных битов	Параметры демультиплексирования																				
		Входные биты							Выходные биты													
QPSK	Входные биты	0	1																			
	Выходные биты	0	1																			
16-QAM	Входные биты	0	1	2	3	4	5	6	7													
	Выходные биты	7	1	4	2	5	3	6	0													
64-QAM	Входные биты	0	1	2	3	4	5	6	7	8	9	10	11									
	Выходные биты	11	7	3	10	6	2	9	5	1	8	4	0									
256-QAM, $N_{ldpc} = 64\ 800$	Входные биты	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15					
	Выходные биты	15	1	13	3	8	11	9	5	10	6	4	7	12	2	14	0					
256-QAM, $N_{ldpc} = 16\ 200$	Входные биты	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	7					
	Выходные биты	7	3	1	5	2	6	4	0													

$d_i \bmod N_{substreams}$ — число входных битов;

$do = di \bmod N_{substreams}$ — число битов в заданном потоке на выходе демультиплексора;

e — количество демультиплексированных битовых потоков ($0 \leq e < N_{substreams}$), зависящее от do .

В случае модуляции QPSK ($N_{ldpc} = 64\ 800$ или $16\ 200$) и 256-QAM ($N_{ldpc} = 16\ 200$), слова длиной $N_{substreams}$, поступающие от демультиплексора, образуют выходные модулирующие значения ячеек и поступают непосредственно на модуль отображения созвездий, таким образом:

$$[y_{0,do} \dots y_{\eta_{mod}-1,do}] = [b_{0,do} \dots b_{N_{substreams}-1,do}] . \quad (21.17)$$

Каждое модулирующее значение ячеек $[y_{0,q} \dots y_{\eta_{mod}-1,q}]$, поступающее от демультиплексора, отображается на созвездия одного из типов: QPSK, 16-QAM, 64-QAM или 256-QAM. Модуляция BPSK используется только для отображения сигнализации.

Точные значения действительных и мнимых компонентов $\operatorname{Re}(z_q)$ и $\operatorname{Im}(z_q)$ для каждой комбинации соответствующих входных битов $y_{e,q}$ для различных созвездий приведены в табл. 21.27.

Точки созвездия z_q для каждого входного модулирующего значения ячеек $[y_{0,q} \dots y_{\eta_{mod}-1,q}]$ нормализуются в соответствии с табл. 21.28 для получения точного комплексного значения ячейки f_q , которое будет использоваться.

В стандарте DVB-T2 используется новаторская техника поворота созвездия на определенный круговой угол, что существенно повышает устойчивость сигнала

при типичных проблемах эфира. За счет поворота диаграммы на точно подобранный угол каждая точка созвездия приобретает уникальные координаты, не повторяющие остальными точками. Принцип поворота диаграммы для системы 16-QAM приведен на рис. 21.29.

Таблица 21.27. Отображения входных битов на созвездия

Отображение созвездия для BPSK									
$y_{0,q}$	1	0							
$\text{Re}(z_q)$	-1	0							
$\text{Im}(z_q)$	0	0							
Отображение созвездия для QPSK									
$y_{0,q}$	1	0							
$\text{Re}(z_q)$	-1	1							
$y_{1,q}$	1	0							
$\text{Im}(z_q)$	-1	1							
Отображение созвездия для 16-QAM									
$y_{0,q}$	1	1	0	0					
$y_{2,q}$	0	1	1	0					
$\text{Re}(z_q)$	-3	-1	1	3					
$y_{1,q}$	1	1	0	0					
$y_{3,q}$	0	1	1	0					
$\text{Im}(z_q)$	-3	-1	1	3					
Отображение созвездия для 64-QAM									
$y_{0,q}$	1	1	1	1	0	0	0	0	
$y_{2,q}$	0	0	1	1	1	1	0	0	
$y_{4,q}$	0	1	1	0	0	1	1	0	
$\text{Re}(z_q)$	-7	-5	-3	-1	1	3	5	7	
$y_{1,q}$	1	1	1	1	0	0	0	0	
$y_{3,q}$	0	0	1	1	1	1	0	0	
$y_{5,q}$	0	1	1	0	0	1	1	0	
$\text{Im}(z_q)$	-7	-5	-3	-1	1	3	5	7	
Отображение созвездия для 256-QAM									
$y_{0,q}$	1	1	1	1	1	1	1	0	0
$y_{2,q}$	0	0	0	0	1	1	1	1	1
$y_{4,q}$	0	0	1	1	1	0	0	0	1
$y_{6,q}$	0	1	1	0	0	1	1	0	1
$\text{Re}(z_q)$	-15	-13	-11	-9	-7	-5	-3	-1	1
$y_{1,q}$	1	1	1	1	1	1	1	0	0
$y_{3,q}$	0	0	0	0	1	1	1	1	0
$y_{5,q}$	0	0	1	1	1	0	0	1	1
$y_{7,q}$	0	1	1	0	0	1	1	0	0
$\text{Im}(z_q)$	-15	-13	-11	-9	-7	-5	-3	-1	1

Если используется поворот созвездия, то нормализованные значения ячеек каждого FEC-блока $F = (f_0, f_1, f_{N_{\text{cells}}-1})$, поступающие от модуля отображения созвездий, вращаются в комплексной плоскости, а мнимая часть циклически задерживается на одну ячейку в пределах FEC-блока. N_{cells} — число ячеек на FEC-блок, заданное табл. 21.29.

Таблица 21.28. Нормирующие множители для ячеек данных

Модуляция	Нормирование
BPSK	$f_q = z_q$
QPSK	$f_q = \frac{z_q}{\sqrt{2}}$
16-QAM	$f_q = \frac{z_q}{\sqrt{10}}$
64-QAM	$f_q = \frac{z_q}{\sqrt{42}}$
256-QAM	$f_q = \frac{z_q}{\sqrt{170}}$

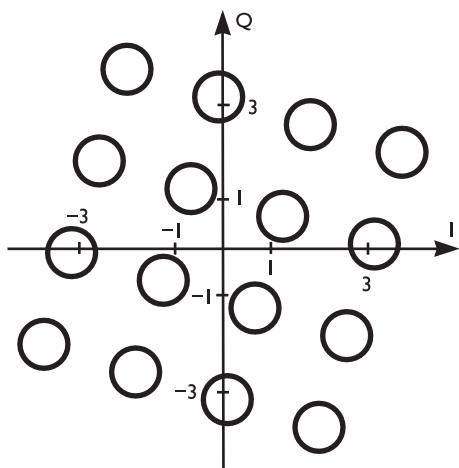


Рис. 21.29. Принцип поворота звездной диаграммы OFDM сигнала для системы 16-QAM

Выходные ячейки $G = (g_0, g_1, \dots, g_{N_{\text{cells}}-1})$ заданы следующими выражениями:

$$g_0 = \operatorname{Re}(e^{j\frac{2\pi\Phi}{360}} f_0) + j \cdot \operatorname{Im}(e^{j\frac{2\pi\Phi}{360}} f_{N_{\text{cells}}-1}), \quad (21.18)$$

$$g_q = \operatorname{Re}(e^{j\frac{2\pi\Phi}{360}} f_q) + j \cdot \operatorname{Im}(e^{j\frac{2\pi\Phi}{360}} f_{q-1}),$$

где $q = 1, 2, \dots, N_{\text{cells}} - 1$, Φ — угол поворота, зависящий от типа модуляции (табл. 21.30).

Поворот созвездий должен использоваться только для передаваемых данных и не должен использоваться для ячеек сигнализации. Если поворот созвездия не применяется, ячейки передаются на перемежитель ячеек неизмененными, т. е. $g_q f_q$.

Таблица 21.29. Параметры временного перемежителя

Длина блока LDPC-кода (N_{ldpc})	Режим модуляции	Число ячеек в блоке LDPC-кода (N_{cells})	Число строк N_r
64 800	256-QAM	8100	1620
	64-QAM	10800	2160
	16-QAM	16 200	3240
	QPSK	32400	6480
16 200	256-QAM	2025	405
	64-QAM	2700	540
	16-QAM	4050	810
	QPSK	8100	1620

Таблица 21.30. Угол поворота для каждого типа модуляции

Модуляция	QPSK	16-QAM	64-QAM	256-QAM
Ф(градусов)	29,0	16,8	8,6	$\text{atan}(1/16)$

Примечание: $\text{atan}(1/16)$ обозначает арктангенс $1/16$, выраженный в градусах.

Псевдослучайный перемежитель ячеек (рис. 21.27) равномерно распределяет ячейки в кодовом слове FEC-кода, обеспечивая в приемнике некоррелированный набор результатов канальных искажений. На выходе этого перемежителя, $G(r) = (g_{r,0}, g_{r,1}, g_{r,2}, \dots, g_{r,N_{\text{cells}}-1})$, должны быть ячейки данных $(g_0, g_1, g_2, \dots, g_{N_{\text{cells}}-1})$ FEC-блока с индексом r , генерированного с помощью поворота созвездия и циклической задержки.

Выход перемежителя представляет собой вектор

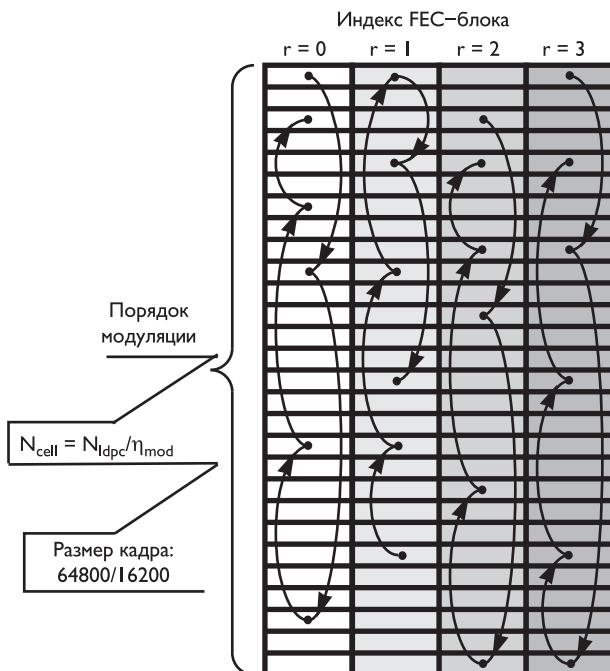
$$D(r) = (d_{r,0}, d_{r,1}, d_{r,2}, \dots, d_{r,N_{\text{cells}}-1}), \quad (21.19)$$

где $d_{r,L_r(q)}g_{r,q}$ для каждого $q = 0, 1, \dots, N_{\text{cells}} - 1$; N_{cells} — число выходных ячеек данных на FEC-блок, определенный в табл. 21.29, и $L_r(q)$ — заданная стандартом функция перестановки, зависящая от величины $P(r)$ сдвига, используемой в FEC-блоке r . Величина $L_r(q)$ задана следующим выражением:

$$L_r(q) = [L_0(q) + P(r)] \bmod N_{\text{cells}},$$

где $L_0(q)$ — базовая функция перестановки и $P(r)$ — величина сдвига, которая должна использоваться в FEC-блоке r (рис. 21.30).

Рис. 21.30. Пример схемы перемежения ячеек



Параметры временного перемежения в системе DVB-T2 могут быть различными для различных входных сигналов. FEC-блоки от перемежителя ячеек для

каждой входной информации группируются в кадры перемежения (которые отображаются на один или несколько кадров). Каждый кадр перемежения содержит динамически изменяющееся целое число FEC-блоков. Число FEC-блоков в кадре перемежения с индексом n обозначается как $N_{BLOCKS_IF}(n)$, может изменяться от минимального значения 0 до наибольшей величины 1023 и сигнализируется в динамической сигнализации.

Каждый кадр перемежения либо отображается непосредственно на один кадр, либо распространяется на несколько кадров. Каждый кадр перемежения также делится на один или несколько (N_{TI}) блоков, где TI — блок временного перемежения, соответствующий однократному использованию памяти временного перемежителя. Число FEC-блоков, содержащихся в блоках кадра перемежения, может отличаться. При этом временное перемежение каждой исходной информации может быть выполнено с помощью трех опций (при активации путем соответствующей сигнализации):

- каждый кадр перемежения содержит один блок и отображается в точности на один кадр, как показано на рис. 21.31а;
- каждый кадр перемежения содержит один блок и отображается на несколько кадров (на рис. 21.31б показан пример, в котором один кадр перемежения отображается на два кадра);
- каждый кадр перемежения отображается в точности на один кадр, и кадр перемежения делится на несколько блоков, как показано на рис. 21.31в.

Каждый кадр перемежения либо отображается в точности на один кадр OFDM, либо распространяется на несколько кадров. Длина периода временного перемежения T_P не должна быть больше одного суперкадра.

Временной перемежитель перемежает ячейки на протяжении одного TI -блока, содержащего динамически изменяющееся целое число FEC-блоков.

Максимальное число ячеек памяти временного перемежителя на один источник информации равно $M_{TI} = 2^{19} + 2^{15}$.

Временной перемежитель TI представляет собой строчно-столбцовый блоковый перемежитель: число строк N_r в перемежителе равно числу ячеек в FEC-блоке (N_{cells}), деленному на 5, и число столбцов $N_c = 5 \cdot N_{FEC}(n, s)$. Отсюда число заполненных столбцов изменяется от TI -блока к TI -блоку в зависимости от скорости ячеек.

В целях достижения максимальной гибкости в выборе характеристик временного перемежителя кадры перемежения на выходе временного перемежителя могут быть разбиты на несколько субслайсов. Разбиение на субслайсы может использоваться в сочетании с опцией временного перемежения, при этом кадр перемежения делится на несколько TI -блоков.

Передаваемая информация может мультиплексироваться различным образом, например, как показано на рис. 21.32.

Генерация, кодирование и модуляция сигнализации уровня 1 (L1) обеспечивает приемник средствами доступа к магистралям физического уровня (PLP) в пределах T2-кадров. На рис. 21.33 показана структура сигнализации L1, разделенная на три основных сектора: сигнализация P1, сигнализация предобработки L1 и сигнализация постобработки L1. Сигнализация P1 предназначена для определения типа передачи и основных параметров передачи. Остальная сигнализация

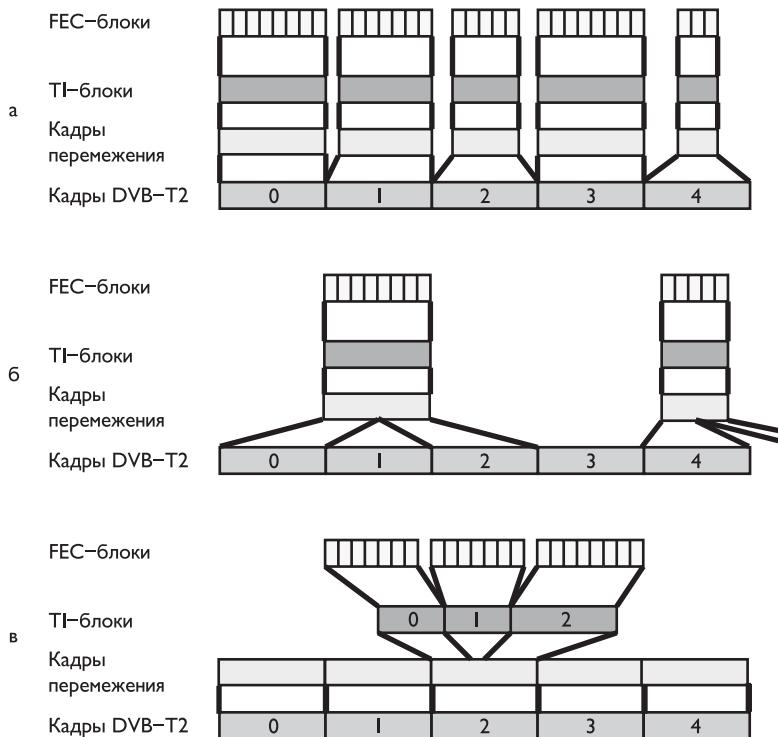
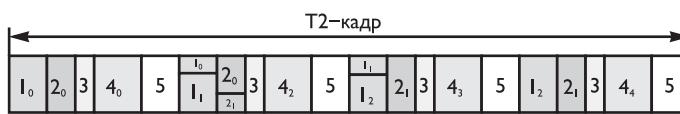


Рис. 21.31. Примеры временного перемежения информации



- PLP1 — Три блока временного перемежения
- PLP2 — Два блока временного перемежения
- PLP3 — Один блок временного перемежения
- PLP4 — Четыре блока временного перемежения
- PLP5 — Один блок временного перемежения

Рис. 21.32. Мультиплексная передача с временным разделением каналов (PLP) с различными периодами перемежения

передается в символе P2, где также могут передаваться данные. Сигнализация предобработки L1 позволяет принимать и декодировать сигнализацию постобработки L1, которая, в свою очередь, передает параметры, необходимые приемнику для доступа к магистралям физического уровня. В дальнейшем сигнализация постобработки L1 разбивается на две основные части: конфигурируемую и динамическую, после которых могут следовать optionalные поля расширения. Сигнализация постобработки L1 заканчивается кодом CRC и заполнением (если требуется).

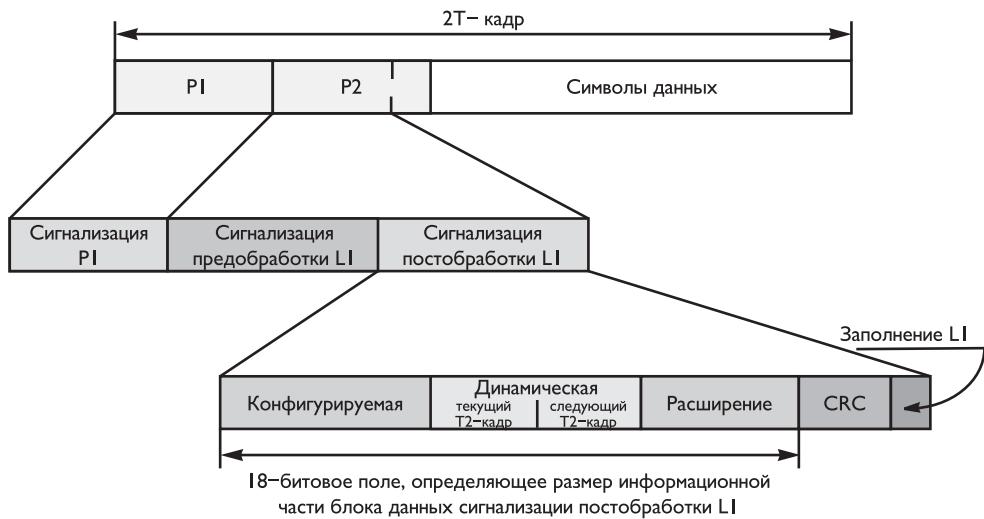


Рис. 21.33. Структура сигнализации L1

Передаваемая в поле P1 информация (7 битов) относится к двум типам: первый тип (биты S1) используется для различения формата начального поля (тип кадра — SISO или MISO, т. е. одна или несколько передающих, но одна приемная антенна); второй тип (биты двух полей S2) помогает приемнику быстро охарактеризовать базовые параметры передачи — длину БПФ и типы начального поля. Символ начального поля P1 предназначен для выполнения четырех задач. Во-первых, он используется при сканировании исходного сигнала для быстрого распознавания сигнала T2. Второе назначение символа P1 — идентификация собственно начального поля системы T2. Символ P1 устроен так, что он может быть использован для того, чтобы отличать его от других форматов, используемых в частях последующего расширения (FEF), существующих в том же суперкадре. Третья задача — сигнализация базовых параметров передатчиков, необходимых для декодирования остальной части начального поля во время процесса инициализации. Четвертая задача символа P1 — дать возможность приемнику детектировать и корректировать частотную и временную синхронизацию.

Сигнализация предобработки L1 состоит из следующих полей:

- 8-битовое поле, определяющее типы передаваемых входных потоков в текущем суперкадре (TS, GSE и др.);
- 3-битовое поле, характеризующее, сколько раз сигнал DVB-T2 был регенерирован;
- 1-битовый флаг, указывающий, осуществляется ли динамическая сигнализация постобработки L1 также для следующего кадра;
- 4-битовое поле, определяющее тип применяемого уменьшения отношения пиковой и средней мощностей (PAR), если оно применяется;
- 2-битовое поле, обозначающее тип кодирования блока данных сигнализации постобработки L1 (исходно указывается только скорость кода, равная 1/2);

- 4-битовое поле, характеризующее созвездие блока данных сигнализации постобработки L1;
- 2-битовое поле, определяющее тип FEC-кода сигнализации L1 (исходно — LDPC 16 200);
- 18-битовое поле, определяющее размер кодированного и модулированного блока данных сигнализации постобработки L1 в ячейках OFDM;
- 18-битовое поле, задающее размер информационной части блока данных сигнализации постобработки L1, в битах, включая поле расширения, если оно присутствует, но не включая CRC-код;
- 1-битовое поле, обозначающее наличие поля расширения сигнализации постобработки L1;
- 1-битовое поле, определяющее, применяется ли расширенный режим несущих в случаях, когда длина БПФ составляет 8K, 16K и 32K;
- 8-битовое поле, используемое для сигнализации присутствия сигналов идентификации передатчика в текущей географической сотовой;
- 16-битовое поле, однозначно определяющее географическую сотовую в сети DVB-T2;
- 16-битовое поле, однозначно определяющее текущую сеть DVB-T2;
- 16-битовое поле, однозначно определяющее систему T2 в сети DVB-T2;
- 3-битовое поле, определяющее число частот (N_{RF}) в системе T2; частоты перечисляются в конфигурируемых параметрах сигнализации постобработки L1;
- 4-битовое поле, характеризующее расположение рассеянных пилотных несущих, используемых для символов данных OFDM;
- 12-битовое поле, определяющее, число символов данных OFDM в T2-кадре, исключая символы P1 и P2;
- 8-битовое поле, указывающее на число T2-кадров (N_{T2}) в суперкадре;
- 3-битовое поле, имеющее такое же значение, что S1 в сигнализации P1;
- 4-битовое поле, имеющее такое же значение, что S2 в сигнализации P1;
- 3-битовое поле, определяющее защитный интервал в текущем суперкадре (1/32, 1/16, 1/8, 1/4, 1/128, 19/128 или 19/256);
- 10-битовое поле, зарезервированное для последующего использования.

Для обнаружения ошибок сигнализации предобработки L1 применяется 32-битовая кодовая последовательность CRC-32. При этом используется 32-разрядный сдвиговый регистр, инициализируемый единицами. Код CRC-32 основан на следующем полиноме:

$$G_{32}(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + \\ + x^{10} + x^8 + x^7 + x^{12} + x^5 + x^4 + x^2 + x + 1. \quad (21.20)$$

Сигнализация постобработки L1 содержит параметры, обеспечивающие приемник достаточной информацией для декодирования нужных магистралей физического уровня. Эта сигнализация содержит два типа параметров, конфигурируемых и динамических, а также опциональное поле расширения. Конфигурируемые параметры должны всегда оставаться одинаковыми на протяжении одного суперкадра, в то время как динамические параметры предоставляют информацию,

относящуюся к текущему Т2-кадру. Значения динамических параметров могут изменяться на протяжении одного суперкадра, в то время как размер каждого поля должен оставаться неизменным.

Поля сигнализации конфигурируемой сигнализации постобработки L1 содержат:

- 15-битовое поле, определяющее суммарное число субслайсов ($N_{sublices_total}$) для PLP данных во всех РЧ-каналах в одном Т2-кадре;
- 8-битовое поле, определяющее число PLP, передаваемых в текущем суперкадре;
- 4-битовое поле, определяющее число вспомогательных потоков;
- 8-битовое поле, зарезервированное для последующего использования;
- 3-битовое поле, определяющее индекс каждой частоты, перечисленной в цикле;
- 32-битовое поле, определяющее центральные частоты РЧ-каналов в Гц, перечисленные в цикле.

Следующие поля появляются только в том случае, если младший бит поля S2 равен «1»:

- 4-битовое поле, которое должно определять тип кадра последующего расширения стандарта (FEF — Future Extension Frame);
- 22-битовое поле, определяющее длину соответствующей части FEF как число элементарных периодов Т от начала символа Р1 части FEF до начала символа Р1 следующего Т2-кадра;
- 8-битовое поле, определяющее число Т2-кадров между двумя частями FEF.

Следующие поля появляются в цикле PLP:

- 8-битовое поле, однозначно определяющее PLP в системе Т2;
- 3-битовое поле, определяющее тип соответствующей PLP;
- 5-битовое поле, определяющее тип полезных данных, передаваемых в соответствующей PLP;
- флаг, устанавливаемый в «1», если PLP типа 1 в системе ТFS встречается в каждом Т2-кадре одного и того же РЧ-канала;
- 3-битовое поле, указывающее, в каком РЧ-канале PLP данных типа 1 встречается в первом кадре суперкадра в системе ТFS;
- 8-битовое поле, содержащее значение идентификатора первого кадра суперкадра, в котором встречается текущая PLP;
- 8-битовое поле, определяющее, к какой группе PLP в системе Т2 относится текущая PLP;
- 3-битовое поле, определяющее скорость кода, используемую связанный PLP (1/2, 3/5, 2/3, 3/4, 4/5 или 5/6);
- 3-битовое поле, определяющее модуляцию, используемую связанный PLP (QPSK, 16-QAM, 64-QAM или 256-QAM);
- 1-битовый флаг, определяющий, использует ли связанный PLP поворот со-звездия;
- 2-битовое поле, определяющее тип FEC, используемого связанный PLP (LDPC 16 200 или LDPC 64 800);
- 10-битовое поле, определяющее максимальный размер полей PLP;

- 8-битовое поле, определяющее интервал Т2-кадров (I_{LUMP}) в суперкадре для соответствующей PLP;
- 8-битовое поле, определяющее тип и порядок перемежения блоков в соответствующих кадрах;
- 1-битовое поле, определяющее тип временного перемежения;
- 1-битовое поле, определяющее, содержит ли текущая PLP информацию внутриполосной сигнализации;
- 16-битовое поле, зарезервированное для последующего использования;
- 32-битовое поле, зарезервированное для последующего использования.

Следующее 32-битовое поле зарезервировано в цикле вспомогательных потоков. Динамическая сигнализация постобработки L1 содержит:

- 8-битовое поле, определяющее индекс текущего Т2-кадра в суперкадре;
- 22-битовое поле, определяющее число ячеек OFDM от начала одного субслайса одной PLP до начала следующего субслайса той же самой PLP, передаваемой по тому же РЧ-каналу для следующего Т2-кадра (или следующего через один Т2-кадр);
- 22-битовое поле, характеризующее стартовую позицию первой из PLP типа 2, использующих схему адресации ячеек;
- 8-битовое поле, определяющее число стоящих перед ним суперкадров, конфигурация которых (т. е. содержимое полей в сигнализации предобработки или постобработки L1) должна измениться;
- 3-битовое поле, содержащее идентификатор (ID) стартовой частоты кадра TFS по расписанию, для следующего Т2-кадра;
- 8-битовое поле, зарезервированное для последующего использования.

Следующие поля появляются в цикле PLP:

- 8-битовое поле, однозначно определяющее PLP в системе Т2;
- 22-битовое поле, определяющее стартовую позицию соответствующей PLP в текущем Т2-кадре (следующий Т2-кадр в случае TFS);
- 10-битовое поле, определяющее число блоков FEC, содержащихся в текущем кадре перемежения для текущей PLP (в случае TFS это относится к кадру перемежения, который отображается на следующий Т2-кадр);
- 8-битовое поле, зарезервированное для последующего использования.

Следующее 32-битовое поле зарезервировано для последующего использования при вспомогательной сигнализации.

Для повышения надежности динамической части сигнализации постобработки L1 информация может быть повторена в начальных полях двух следующих друг за другом Т2-кадрах.

Поле расширения сигнализации постобработки L1 обеспечивает возможность дальнейшего развития сигнализации L1. На его присутствие указывает поле сигнализации предобработки L1.

32-битовый код (поле CRC) обнаружения ошибок применяется ко всей сигнализации постобработки L1, включая конфигурируемую сигнализацию, динамическую сигнализацию для текущего Т2-кадра, динамическую сигнализацию для

следующего Т2-кадра, если она присутствует, и поле расширения сигнализации постобработки L1, если оно присутствует.

Поле заполнения L1 вставляется после поля CRC сигнализации постобработки L1, обеспечивая одинаковый информационный размер для нескольких блоков кода LDPC сигнализации постобработки L1, когда сигнализация постобработки L1 сегментируется на несколько блоков, и эти блоки кодируются по отдельности.

Сигнализация предобработки L1 защищается с помощью последовательного соединения внешнего кода БЧХ и внутреннего кода LDPC. Биты сигнализации предобработки L1 имеют фиксированную длину и сначала закодированы с помощью кода БЧХ, при этом проверочные биты кода БЧХ сигнализации предобработки присоединяются в конец сигнализации предобработки. Последовательно соединенные сигнализация предобработки L1 и проверочные биты кода БЧХ подвергаются дальнейшей защите с помощью сокращенного и прореженного кода LDPC со скоростью 1/4 ($N_{ldpc} = 16\ 200$). Следует обратить внимание на то, что код LDPC 16 200 со скоростью 1/4 имеет эффективную скорость кода 1/5, определяемую отношением информационной длины к длине блока на выходе кодера. Для определения операции сокращения используется входной параметр $K_{sig} = 200$, равный информационной длине сигнализации предобработки K_{pre} . Параметр N_{punc} , используемый для определения операции прореживания, равен:

$$N_{punc} = (K_{bch} - K_{sig}) \times \left(\frac{1}{R_{eff}} - 1 \right) = 11488, \quad (21.21)$$

где $K_{bch} = 3072$ определяет число информационных бит кода БЧХ, а $R_{eff} = 1/5$ — эффективная скорость кода LDPC для сигнализации предобработки L1.

После сокращения и прореживания кодированные биты сигнализации предобработки L1 отображаются на $(K_{sig} + N_{bch_parity}) \times \frac{1}{R_{eff}} = 1840$ символов, где $N_{bch_parity} = 168$ — число проверочных битов кода БЧХ для кодов LDPC 16 200.

Число битов сигнализации постобработки L1 является переменной величиной, и биты передаются с помощью одного или нескольких блоков кода LDPC 16 200 в зависимости от длины сигнализации постобработки L1. Число $N_{post_FEC_Block}$ блоков кода LDPC для сигнализации постобработки L1 определяется следующей формулой:

$$N_{post_FEC_Block} = \left\lceil \frac{K_{post_ex_pad}}{K_{bch}} \right\rceil, \quad (21.22)$$

где $\lceil x \rceil$ означает целое число большее или равное x , $K_{bch} = 7032$ для кода LDPC 16 200 со скоростью кода 1/2 (эффективная скорость кода 4/9), и $K_{post_ex_pad}$ определяет число информационных битов для сигнализации постобработки L1.

Число информационных битов K_{sig} в каждом из блоков $N_{post_FEC_Block}$ определяется по формуле

$$K_{sig} = \frac{K_{post}}{N_{post_FEC_Block}}, \quad (21.23)$$

где K_{post} — полная длина всей сигнализации постобработки L1, включая поле заполнения.

Каждый блок с информационным размером K_{sig} защищается с помощью последовательного соединения внешних кодов БЧХ и внутренних кодов LDPC.

Каждый блок должен быть сначала закодирован кодом БЧХ, где его проверочные биты кода БЧХ $N_{\text{bch_parity}} = 168$ присоединяются в конец информационных битов каждого блока. Последовательно соединенные информационные биты каждого блока и проверочные биты кода БЧХ подвергаются дальнейшей защите с помощью сокращенного и прореженного кода LDPC 16 200 со скоростью 1/2 (эффективная скорость кода LDPC 16 200 со скоростью кода 1/2, $R_{\text{eff}} \text{ 16K_LDPC_1_2}$, составляет 4/9).

Для заданного K_{sig} и порядка модуляции (BPSK, QPSK, 16-QAM или 64-QAM используются для сигнализации постобработки L1), вычисляется число проверочных битов кода LDPC, которые должны быть исключены при прореживании.

Если используются 16-QAM или 64-QAM, побитовое перемежение должно применяться к каждому блоку кода LDPC. При использовании BPSK или QPSK, побитовое перемежение не применяется.

Число K_{sig} битов должно быть закодировано в кодовое слово LDPC 16 200 после кодирования БЧХ. Если K_{sig} меньше, чем число информационных битов БЧХ-кода (K_{bch}) для заданной скорости кода, код БЧХ должен быть сокращен. Часть информационных битов кода LDPC 16 200 при этом заполняется нулями до заполнения K_{bch} информационных битов. Биты заполнения не должны передаваться. Все K_{bch} информационных битов кода БЧХ, обозначенные как $(m_0, m_1, \dots, m_{K_{\text{bch}}-1})$, делятся на $N_{\text{group}} K_{\text{ldpc}} / 360$ групп в соответствии со следующим правилом:

$$X_j = \left\{ m_k \left| j = \left\lfloor \frac{k}{360} \right\rfloor, \quad 0 \leq k < K_{\text{bch}} \right. \right\} \text{ для } 0 \leq j < N_{\text{group}}, \quad (21.24)$$

где X_j обозначает j -ю группу битов. Параметры кодов сигнализации предобработки и постобработки L1 ($K_{\text{bch}}, K_{\text{ldpc}}$) приведены в табл. 21.31.

Таблица 21.31. Параметры кодов сигнализации предобработки и постобработки L1

	K_{bch}	K_{ldpc}
Сигнализация предобработки L1	3072	3240
Сигнализация постобработки L1	7032	7200

Для $0 \leq j \leq N_{\text{group}} - 2$ каждая группа битов X_j содержит 360 битов, и последняя группа битов $X_{N_{\text{group}}-1}$ содержит $360 - (K_{\text{ldpc}} - K_{\text{bch}}) = 192$ бита, как показано на рис. 21.34а.

Для заданного K_{sig} , число битов заполнения нулями вычисляется как $(K_{\text{bch}} - K_{\text{sig}})$. На рис. 21.34б в качестве примера показано сокращение информационных битов кода БЧХ — заполнение позиций информационных битов кода БЧХ K_{sig} информационными битами с ненулевым заполнением.

Информационные биты K_{bch} (включая $K_{\text{bch}} - K_{\text{sig}}$ биты, заполненные нулями) сначала кодируются кодом БЧХ для генерирования $N_{\text{bch}} = K_{\text{ldpc}}$ выходных битов $(i_0 \dots i_{N_{\text{bch}}-1})$.

$N_{\text{bch}} K_{\text{ldpc}}$ выходные биты $(i_0 \dots i_{N_{\text{bch}}-1})$ ВСН кодера, включая $(K_{\text{bch}} - K_{\text{sig}})$ биты, заполненные нулями, и $(K_{\text{ldpc}} - K_{\text{bch}})$ проверочные биты кода БЧХ формируют K_{ldpc} информационные биты $I = (i_0, i_1, \dots, i_{K_{\text{ldpc}}-1})$ для LDPC-кодера. Кодер LDPC систематически кодирует K_{ldpc} информационные биты в кодовое слово Λ размером N_{ldpc} :

$$\Lambda = (i_0, i_1, \dots, i_{K_{\text{ldpc}}-1}, p_0, p_1, \dots, p_{N_{\text{ldpc}}-K_{\text{ldpc}}-1}).$$

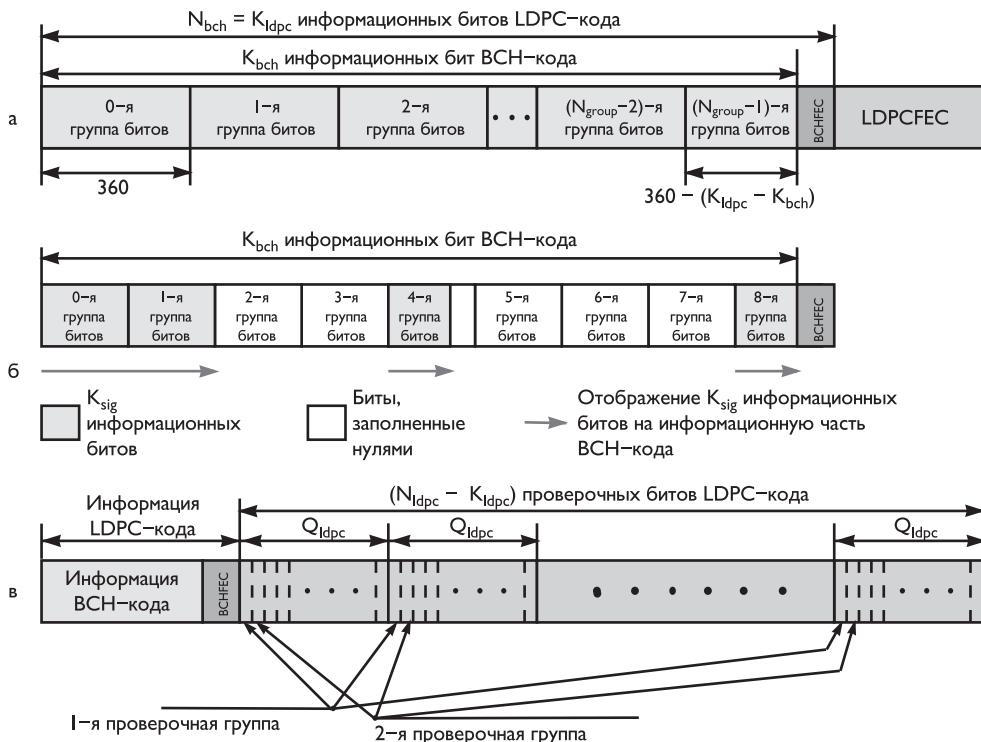


Рис. 21.34. Формат данных после LDPC кодирования сигнализации L1(а); пример сокращения информационной части кода БЧХ (б); группы проверочных битов в блоке FEC (в)

После кодирования LDPC некоторое количество проверочных битов LDPC-кода исключается при прореживании.

Все $N_{ldpc} - K_{ldpc}$ проверочные биты LDPC-кода ($p_0, p_1, \dots, p_{N_{ldpc} - K_{ldpc} - 1}$) делятся на Q_{ldpc} проверочных групп, где каждая проверочная группа образуется из подмножества $N_{ldpc} - K_{ldpc}$ проверочных битов LDPC-кода следующим образом:

$$P_j = \{p_k \mid k \bmod q = j, 0 \leq k < N_{ldpc} - K_{ldpc}\} \text{ для } 0 \leq j < q,$$

где P_j обозначает j -ю проверочную группу. Каждая группа содержит $(N_{ldpc} - K_{ldpc})/Q_{ldpc} = 360$ битов, как показано на рис. 21.34в.

Далее производится операция перестановки данных, различная для сигнализации предобработки и постобработки L1 и зависящая от скорости кода и вида модуляции.

Биты ($K_{bch} - K_{sig}$), заполненные нулями, удаляются и не передаются. При этом в составе слова остается K_{sig} информационных битов, за которыми следуют 168 проверочных битов кода БЧХ и $(N_{ldpc} - K_{ldpc} - N_{punc})$ проверочных битов LDPC-кода.

Если для сигнализации постобработки L1 применяется модуляция 16-QAM или 64-QAM, кодовое слово кода LDPC длиной N_{post} , состоящее из K_{sig} информационных битов, 168 проверочных битов кода БЧХ и $(9000 - N_{punc})$ проверочных битов LDPC-кода.

рочных битов LDPC-кода, подвергаются побитовому перемежению с помощью перемежителя (табл. 21.32).

Таблица 21.32. Структура побитового перемежителя

Модуляция и скорость кода	Строки N_r	Столбцы N_c
16-QAM	$1/2$	$N_{\text{post}}/8$
64-QAM	$1/2$	$N_{\text{post}}/12$

Каждый бит сигнализации предобработки L1 отображается непосредственно на созвездие BPSK, в то время как сигнализация постобработки L1 сначала демультиплексируется на кодовые слова несущих, а затем кодовые слова несущих отображаются на созвездия.

Каждое кодовое слово кода LDPC после побитового перемежения, прореживания и сокращения, представляющее собой последовательность из N_{post} битов, $V = (\nu_0 \dots \nu_{N_{\text{post}}-1})$, где $N_{\text{post}}K_{\text{sig}} + 168 + 9000 - N_{\text{punc}}$, отображается на созвездия. Для этого сначала входные биты демультиплексируются в параллельные кодовые слова ячеек, а затем эти кодовые слова ячеек отображаются на значения созвездий. Число выходных ячеек данных и эффективное число битов на ячейку η_{mod} определены в табл. 21.33.

Входной битовый поток v_{di} демультиплексируется на $N_{\text{substreams}}$ (см. табл. 21.33) подпотоков $b_{e,do}$.

Биты сигнализации предобработки $y_{0,q}$ и кодовые слова несущих сигнализации постобработки L1 $[y_{0,q} \dots y_{\eta_{\text{mod}}-1,q}]$ отображаются на созвездия $f_{\text{pre},q}$ и $f_{\text{post},q}$ соответственно, где q — индекс несущих в каждом кодовом слове LDPC-кода после побитового перемежения. Для сигнализации предобработки $0 \leq q < 1840$, а для сигнализации постобработки L1 $0 \leq q < N_{\text{MOD_per_Block}}$. После этого кодированные и модулированные несущие сигнализации постобработки L1, соответствующие каждому кодовому слову Т2-кадра номера m , объединяются для формирования единичного блока несущих $f_{\text{post},m,i}$, где i — индекс несущих в единичном блоке $0 \leq i < N_{\text{MOD_Total}}$. Кодированные и модулированные несущие сигнализации предобработки L1 для Т2-кадра номера m формируют единичный блок несущих $f_{\text{pre},m,i}$, где i — индекс несущих в единичном блоке $0 \leq i < 1840$. Затем кодированные и модулированные несущие сигнализации предобработки и постобработки L1 отображаются на символы P2.

Таблица 21.33. Параметры для побитового отображения на созвездия

Режим модуляции	η_{mod}	Число выходных ячеек данных в кодовом слове	Число подпотоков, $N_{\text{substreams}}$
BPSK	1	N_{post}	1
QPSK	2	$N_{\text{post}}/2$	2
16-QAM	4	$N_{\text{post}}/4$	8
64-QAM	6	$N_{\text{post}}/6$	12

Функция модуля формирования кадров заключается в компоновке ячеек, генерируемым временными перемежителями для каждой PLP, и ячеек модулированных данных сигнализации L1 в массивы активных ячеек OFDM, соответствующие каждому из символов OFDM, составляющему общую кадровую структуру.

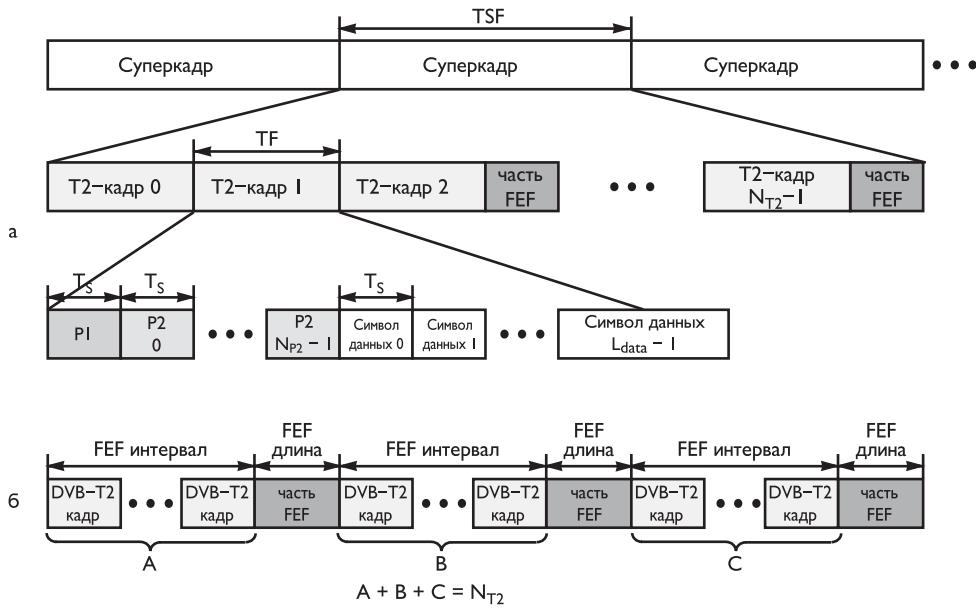


Рис. 21.35. Кадровая структура (а); суперкадр, включающий Т2-кадры и части FEF (б)

Структура Т2-кадра показана на рис. 21.35а и состоит из суперкадров, которые делятся на Т2-кадры, которые, в свою очередь, делятся на символы OFDM. Суперкадр состоит из N_{T2} Т2-кадров. Т2-кадр начинается с одного символа Р1 и N_{P2} символов Р2, за которыми следуют L_{data} символов данных. Кроме того, суперкадр может содержать несколько частей FEF (Future Extension Frame) — кадров для последующего расширения, использование которых является optionalным. Эти кадры могут передавать данные способом, не известным приемнику DVB-T2, имея в виду текущую версию стандарта (рис. 21.35б).

Длительность суперкадра T_{SF} определяется следующим образом:

$$T_{SF} = N_{T2} \cdot T_F + N_{FEF} \cdot T_{FEF}, \quad (21.25)$$

где N_{FEF} — число частей FEF в суперкадре, T_{FEF} — длительность части FEF.

Максимальное значение размера суперкадра T_{SF} равно 64 с, если кадры FEF не используются (что эквивалентно 256 кадрам размером 250 мс), и 128 с в случае использования кадров FEF.

Т2-кадр содержит один символ начального поля Р1 (идентификации базового режима передачи символов DVB-T2), за которым следует один или несколько символов начального поля Р2 (передает информацию о кодировании и модуляции, включая базовую информацию о системе Т2 и данные, необходимые для декодирования сигнализации постобработки L1), за которыми следуют конфигурируемое количество символов данных.

Длительность Т2-кадра определяется как:

$$T_F = L_F \cdot T_s + T_{P1}, \quad (21.26)$$

где T_s — общая длительность символа OFDM, и T_{P1} — длительность символа Р1.

Максимальное значение длительности кадра T_F равно 250 мс. Таким образом, максимальное значение L_F для ширины полосы канала 8 МГц приведено в табл. 21.34.

Таблица 21.34. Максимальная длина кадра L_F в символах OFDM

Длина БПФ	T_u , мс	Защитный интервал						
		1/128	1/32	1/16	19/256	1/8	19/128	1/4
32К	3,584	68	66	64	64	60	60	н/п
16К	1,792	138	135	131	129	123	121	111
8К	0,896	276	270	262	259	247	242	223
4К	0,448	н/п	540	524	н/п	495	н/п	446
2К	0,224	н/п	1081	1049	н/п	991	н/п	892
1К	0,112	н/п	н/п	2098	н/п	1982	н/п	1784

н/п — не предусмотрено

Модуль формирования кадров должен отображать ячейки, поступающие как от временного перемежителя (для PLP), так и от модуля отображения созвездий (для сигнализации предобработки и постобработки L1) на ячейки данных $x_{m,l,p}$ каждого символа OFDM в каждом кадре, где m — число Т2-кадров; l — индекс символа в кадре, начинающийся с 0 для первого символа Р2, $0 \leq l < L_F$; p — индекс ячейки данных в символе до частотного перемежения и вставки пилотной несущей.

Число активных несущих C_{P2} , не используемых в качестве пилотных или резервируемых, в одном символе Р2 определено в табл. 21.35, т. е. общее число активных несущих во всех символах Р2 равно $N_{P2} \cdot C_{P2}$.

Таблица 21.35. Число ячеек данных C_{P2} в одном символе Р2

Длина БПФ	C_{P2}	
	SISO	MISO
1К	558	546
2К	1118	1098
4К	2236	2198
8К	4472	4398
16К	8944	8814
32К	22432	17612

Число активных несущих C_{data} , также не используемых в качестве пилотных, в одном нормальном символе при обычном и расширенном режиме несущих для 8К, 16К и 32К приведены в табл. 21.36 (РР: 4-битовое поле, определяющее расположение рассеянных пилотных несущих используемых для символов данных OFDM).

В некоторых сочетаниях длины БПФ, защитного интервала и расположения пилотных несущих, последний символ Т2-кадра является специальным последним символом кадра FC. Пилотные несущие расположены в нем с большей плотностью, чем в других символах данных, и некоторые из ячеек с целью поддержания одинаковой общей энергии символа не модулируются. Если последний символ кадра присутствует, число содержащихся в нем ячеек данных обозначается как N_{FC} , а меньшее число активных ячеек обозначается как C_{FC} .

Таблица 21.36. Число ячеек данных C_{data} в одном символе данных

Длина БПФ		C_{data} (без резервирования)								Резервируемые ячейки
		PP1	PP2	PP3	PP4	PP5	PP6	PP7	PP8	
1K		764	768	798	804	818				10
2K		1522	1532	1596	1602	1632		1646		18
4K		3084	3092	3228	3234	3298		3328		36
8K	Обычный	6208	6214	6494	6498	6634		6698	6698	72
	Расширенный	6296	6298	6584	6588	6728		6788	6788	72
16K	Обычный	12418	12436	12988	13002	13272	13288	13416	13406	144
	Расширенный	12678	12698	13262	13276	13552	13568	13698	13688	144
32K	Обычный		24886		26022		26592	26836	26812	288
	Расширенный		25412		26572		27152	27404	27376	288

Примечание: Незаполненные ячейки таблицы указывают на то, что соответствующее сочетание длины БПФ и расположения пилотных несущих никогда не используется.

Таким образом, число активных ячеек OFDM в одном Т2-кадре (C_{tot}) зависит от параметров кадровой структуры, включая присутствие или отсутствие последнего символа кадра, и задано следующим выражением:

$$C_{tot} = \begin{cases} N_{P2} \cdot C_{P2} + (L_{data} - 1) \cdot C_{data} + C_{FC}, & \text{если присутствует FC,} \\ N_{P2} \cdot C_{P2} + L_{data} \cdot C_{data}, & \text{если отсутствует FC.} \end{cases} \quad (21.27)$$

Число N_{P2} символов Р2 зависит от используемой длины БПФ и определено в табл. 21.37.

Таблица 21.37. Число символов Р2 (N_{P2}) для различных режимов БПФ

Длина БПФ	N_{P2}
1K	16
2K	8
4K	4
8K	2
16K	1
32K	1

Число ячеек OFDM, предназначенное для передачи PLP в одном Т2-кадре, задается формулой:

$$D_{PLP} = C_{tot} - D_{L1}, \quad (21.28)$$

где D_{L1} — число ячеек OFDM, требуемое для передачи всей сигнализации L1.

Значения D_{L1} и D_{PLP} не изменяются от Т2-кадра к Т2-кадру, но могут изменяться от суперкадра к суперкадру.

PLP данные передаются в субслайсах, число которых может быть от 1 до 6480. Структура Т2-кадра изображена на рис. 21.36а.

Конфигурация структуры Т2-кадров передается с помощью сигнализации предобработки и постобработки L1. Местоположение самих PLP в Т2-кадре может динамически изменяться от Т2-кадра к Т2-кадру, и это сигнализируется как в динамической части сигнализации постобработки L1, в символах Р2, так и во внутриволосной сигнализации (рис. 21.36б).

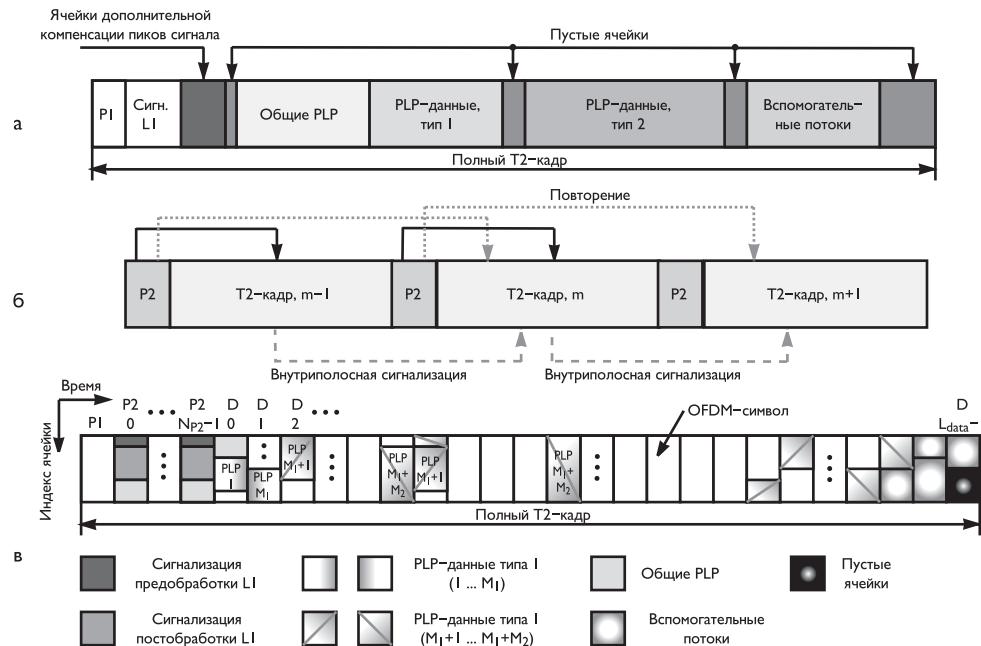


Рис. 21.36. Структура T2-кадра (а); сигнализация L1 в системе с одним РЧ-каналом (б); отображение PLP-данных на символы (в)

Слайсы и субслайсы PLP вспомогательных потоков и пустых ячеек отображаются на символы T2-кадра, как показано на рис. 21.36в. T2-кадр начинается с символа P1, за которым следуют N_{P2} символов P2. Сигнализация предобработки и постобработки L1 сначала отображается на символы P2. После этого общие PLP отображаются сразу после сигнализации L1. PLP-данные передаются после общих PLP, начиная с PLP1 типа 1. PLP типа 2 передаются после PLP типа 1. Вспомогательный поток или потоки, если они присутствуют, передаются после PLP типа 2, и после них могут следовать пустые ячейки. Вместе PLP, вспомогательные потоки и пустые ячейки должны в точности заполнять оставшиеся ячейки в кадре.

Кодированные и модулированные ячейки сигнализации предобработки и постобработки L1 для T2-кадра m ($f_{pre,m,i}$ и $f_{post,m,i}$) отображаются на символы P2 следующим образом: ячейки сигнализации предобработки L1 построчно зигзагообразно отображаются на активные ячейки символов P2 в соответствии с соотношением:

$$x_{m,l,p} = f_{pre,m,p \times N_{P2} + l} \text{ для } 0 \leq l < N_{P2} \text{ и } 0 \leq p < \frac{D_{L1pre}}{N_{P2}}, \quad (21.29)$$

где: $D_{L1pre} = 1840$ — число ячеек сигнализации предобработки L1 на T2-кадр, N_{P2} — число символов P2, приведенное в табл. 21.37, $x_{m,l,p}$ — активные несущие каждого символа OFDM; ячейки сигнализации постобработки L1 построчно зигзагообразно отображаются на активные ячейки символов P2 после ячеек сигнализации предобработки L1 и описываются следующим уравнением:

$$x_{m,l,p+\frac{D_{\text{L1pre}}}{N_{P2}}} = f_post_{m,p \times N_{P2} + l} \text{ для } 0 \leq l < N_{P2} \text{ и } 0 \leq p < \frac{D_{\text{L1post}}}{N_{P2}}, \quad (21.30)$$

где D_{L1post} — число ячеек сигнализации постобработки L1 в T2-кадре, D_{L1post} — $D_{\text{MOD_Total}}$.

После отображения данных L1 на символы P2 оставшиеся активные ячейки данных $x_{m,l,p}$ в символах P2 и символы данных могут быть использованы в PLP.

PLP подразделяются на 3 типа, которые сигнализируются в поле сигнализации постобработки: общие PLP, PLP типа 1 и типа 2. Общие PLP и PLP типа 1 содержат в точности один субслайс на T2-кадр, тогда как PLP типа 2 содержат от 2 до 6480 субслайсов на T2-кадр.

В кадре передаются M_{common} общих PLP, M_1 PLP типа 1 и M_2 PLP типа 2, и планировщик выбирает количество блоков для каждого кадра перемежения каждой PLP, такие, чтобы общее число ячеек всех PLP плюс любые вспомогательные потоки не превышало число ячеек, зарезервированных для данных.

Результирующие значения D_i должны удовлетворять следующему выражению:

$$\sum_{i=1}^{M_{\text{common}}} D_{i,\text{common}} + \sum_{i=1}^{M_1} D_{i,1} + \sum_{i=1}^{M_2} D_{i,2} + \sum_{i=1}^{M_{\text{aux}}} D_{i,\text{aux}} \leq D_{\text{PLP}}, \quad (21.31)$$

где $D_{i,\text{common}}$ — число ячеек OFDM D_i , требуемое для передачи общей PLP с индексом i , $D_{i,j}$ — число ячеек OFDM D_i , требуемое для передачи PLP данных i типа j , M_{aux} — число вспомогательных потоков и $D_{i,\text{aux}}$ — число ячеек, занятых вспомогательным потоком i .

Одномерная схема адресации $(0 \dots D_{\text{PLP}} - 1)$ определена для активных ячеек данных, которые не используются для сигнализации L1. Схема адресации определяет порядок, в котором ячейки из субслайсов PLP назначаются активным ячейкам данных, а также используются для сигнализации местоположений субслайсов всех PLP в динамической части сигнализации постобработки L1.

В качестве примера на рис. 21.37 приведена адресация ячеек OFDM общих PLP и PLP 1-го и 2-го типа.

Ячейки общей PLP для конкретного T2-кадра отображаются последовательно одна за другой на отдельный непрерывный диапазон адресов ячеек кадра, в порядке повышения адресов. Общие PLP, если они присутствуют, отображаются, начиная с адреса 0. Если используется несколько общих PLP, ячейки следующей общей PLP начинаются с адреса, следующего сразу после последней ячейки предыдущей общей PLP, и всегда с данных, записанных в порядке увеличения адресов.

Ячейки PLP типа 1, если они присутствуют, начинаются с адреса, следующего сразу после последней ячейки последней общей PLP, или с адреса 0 в случае отсутствия общих PLP. Ячейки первого субслайса первой PLP типа 2 начинаются с адреса, следующего сразу после последней ячейки последней PLP типа 1.

После PLP типа 2 могут быть добавлены один или несколько вспомогательных потоков. Стартовая позиция и число ячеек для каждого вспомогательного потока могут изменяться от T2-кадра к T2-кадру, а для сигнализации этих параметров в динамической сигнализации L1 резервируются биты.

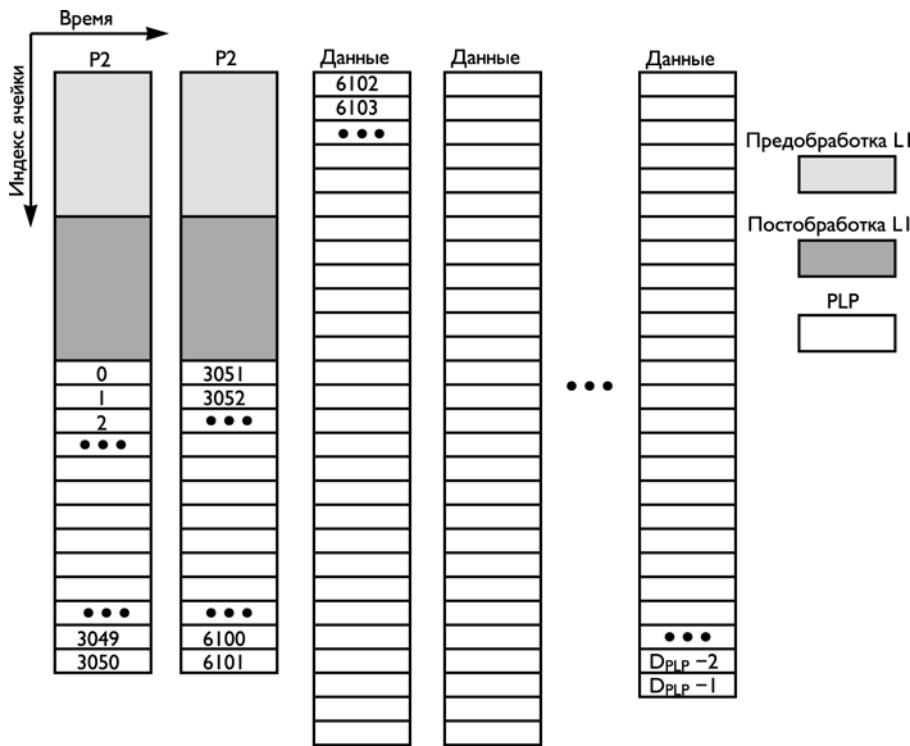


Рис. 21.37. Адресация ячеек OFDM для общих PLP и PLP данных

Если PLP данные и вспомогательные потоки в точности не заполняют ячейки, назначенные для данных, то в оставшиеся ячейки T2-кадра вставляются пустые ячейки.

Вставка кадров для последующего расширения (FEF) позволяет передавать кадры, определенные в последующих расширениях стандарта DVB-T2, в том же самом мультиплексированном потоке, что и обычные T2-кадры. Использование кадров для последующего расширения является optionalным.

Отображение ячеек данных от устройства формирования кадров на несущие осуществляется частотным перемежителем.

При режимах 1K, 2K, 4K, 8K, 16K используются основанные на последовательностях R'_i две функции перестановки: $H_0(p)$ — для четных символов кадра и $H_1(p)$ — для четных символов кадра. В режиме 32K применяется функция перестановки $H(p)$ только для нечетных символов кадра.

Вектор R_i вычисляется из вектора R'_i с помощью битовых перестановок, приведенных в табл. 21.38.

Режим MISO, при котором используется несколько передающих антенн, может применяться ко всем символам сигнала DVB-T2 на уровне ячеек. При этом используется преобразованное кодирование Аламути [8.23] для получения двух наборов ячеек данных, за исключением того, что кодирование никогда не применяется к символу P1 начального поля.

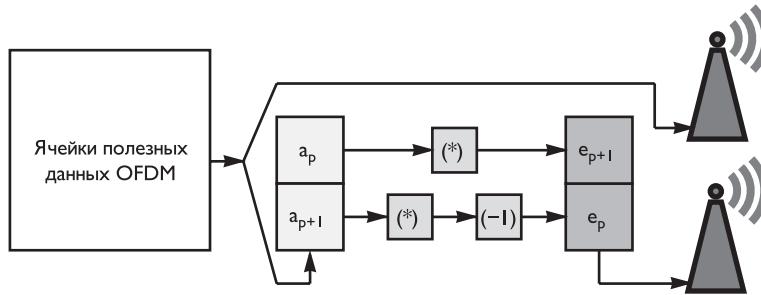


Рис. 21.38. Кодирование ячеек полезных данных OFDM в режиме MISO

Процесс кодирования выполняется над парами ячеек полезных данных OFDM ($a_{m,l,p}, a_{m,l,p+1}$) с выхода частотного перемежителя. Кодированные ячейки полезных данных OFDM $e_{m,l,p}$ для 1-го передатчика ($Tx1$) и $e_{m,l,p}$ для 2-го передатчика ($T \times 2$) генерируются из входных ячеек следующим образом:

$$\begin{aligned} e_{m,l,p}(Tx1) &= a_{m,l,p}, & e_{m,l,p+1}(Tx1) &= a_{m,l,p+1}, \\ e_{m,l,p}(Tx2) &= -a_{m,l,p+1}^*, & e_{m,l,p+1}(Tx2) &= a_{m,l,p+1}^*, \end{aligned}$$

где $p \in \{0, 2, 4, 6, \dots, N_{data} - 2\}$; знак * обозначает действие комплексного сопряжения, N_{data} — число ячеек на выходе частотного перемежителя для текущего символа l (рис. 21.38).

Таблица 21.38. Битовые перестановки

	Позиции битов R'_i									
1K	Позиции битов R_i (H_0)									
	4 3 2 1 0 5 6 7 8									
	Позиции битов R_i (H_1)									
	3 2 5 0 1 4 7 8 6									
2K	Позиции битов R'_i									
	9 8 7 6 5 4 3 2 1 0									
	Позиции битов R_i (H_0)									
	0 7 5 1 8 2 6 9 3 4									
	Позиции битов R_i (H_1)									
	3 2 7 0 1 5 8 4 9 6									
4K	Позиции битов R'_i									
	10 9 8 7 6 5 4 3 2 1 0									
	Позиции битов R_i (H_0)									
	7 10 5 8 1 2 4 9 0 3 6									
	Позиции битов R_i (H_1)									
	6 2 7 10 8 0 3 4 1 9 5									
8K	Позиции битов R'_i									
	11 10 9 8 7 6 5 4 3 2 1 0									
	Позиции битов R_i (H_0)									
	5 11 3 0 10 8 6 9 2 4 1 7									
	Позиции битов R_i (H_1)									
	8 10 7 6 0 5 2 1 3 9 4 11									
16K	Позиции битов R'_i									
	12 11 10 9 8 7 6 5 4 3 2 1 0									
	Позиции битов R_i (H_0)									
	8 4 3 2 0 11 1 5 12 10 6 7 9									
	Позиции битов R_i (H_1)									
	7 9 5 3 11 1 4 0 2 12 10 8 6									
32K	Позиции битов R'_i									
	13 12 11 10 9 8 7 6 5 4 3 2 1 0									
	Позиции битов R_i									
	6 5 0 10 8 1 11 12 2 9 4 3 13 7									

Пилотные несущие применяются для синхронизации кадров, частоты, времени, для оценки канала, идентификации режима передачи и могут также быть использованы для контроля фазового шума. Ячейки, содержащие опорную информацию, передаются на «усиленном» уровне мощности, а информация, передаваемая в этих ячейках, представляет собой рассеянные, непрерывные, граничные

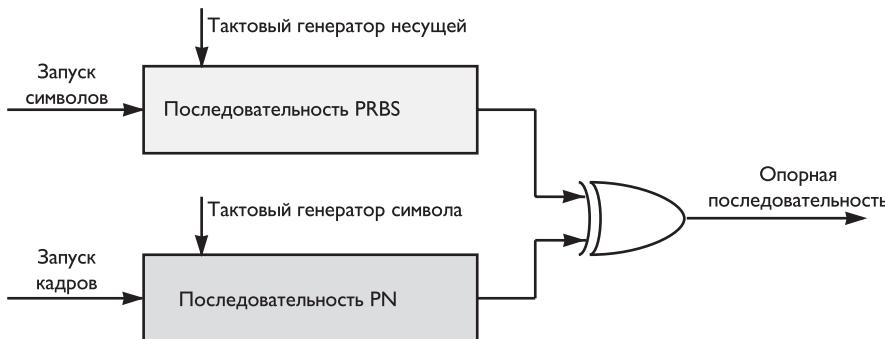


Рис. 21.39. Формирование опорной последовательности из последовательностей PN и PRBS

пилотные ячейки, пилотные ячейки Р2 или пилотные ячейки последнего символа кадра. Местоположение и амплитуды пилотных несущих определяются для передачи в режиме SISO и преобразуются для передачи в режиме MISO.

Пилотные несущие модулируются в соответствии с опорной последовательностью, которая вычисляется с помощью псевдослучайной двоичной последовательности (PRBS) на уровне символа (r_k , где k — индекс несущей) с порождающим полиномом $x^{11} + x^2 + 1$ (сдвиговый регистр инициализируется единицами) и последовательностью пилотных несущих (PN) на уровне кадра (p_{nl} , где l — индекс символа) (рис. 21.39).

Каждое значение PN-последовательности на уровне кадра применяется к одному символу OFDM T2-кадра. Следовательно, длина PN-последовательности на уровне кадра N_{PN} равна числу символов в T2-кадре, за исключением Р1 (табл. 21.39 для каналов с полосой пропускания 8 МГц).

Таблица 21.39. Максимальные длины PN-последовательностей для различных режимов БПФ

Режим БПФ	Максимальная длина последовательности, N_{PN} (элементов)
1К	2098
2К	1081
4К	540
8К	276
16К	138
32К	69

Опорная информация, взятая из опорной последовательности, передается в рассеянных пилотных ячейках в каждом символе, за исключением Р1, Р2 и последнего символа кадра (если он применяется).

Используется несколько видов рассеянных пилотных несущих (PPi), расположение которых определяется параметрами разноса D_X и D_Y (табл. 21.40).

Сочетания расположения рассеянных пилотных несущих, длины БПФ и защитного интервала, допустимые для использования, определены табл. 21.41 для режимов SISO и MISO.

Амплитуды рассеянных пилотных несущих, A_{SP} , зависят от их расположения (табл. 21.42).

Таблица 21.40. Параметры, определяющие расположение рассеянных пилотных несущих

Расположение пилотных несущих	Разнос пилотных несущих (D_X)	Число символов, формирующих одну последовательность рассеянных пилотных несущих (D_Y)
PP1	3	4
PP2	6	2
PP3	6	4
PP4	12	2
PP5	12	4
PP6	24	2
PP7	24	4
PP8	6	16

Таблица 21.41. Расположение рассеянных пилотных несущих, используемое для каждого допустимого сочетания длины БПФ и защитного интервала

Режим	Длина БПФ	Заданный интервал						
		1/128	1/32	1/16	19/256	1/8	19/128	1/4
SISO	32K	PP7	PP4, PP6	PP2, PP8 PP4	PP2, PP8 PP4	PP2, PP8	PP2, PP8	н/п
		PP8, PP4 PP6	PP8, PP4	PP2, PP8	PP2, PP8	н/п	н/п	н/п
SISO	16K	PP7	PP7, PP4, PP6	PP2, PP8, PP4, PP5	PP2, PP8, PP4, PP5	PP2, PP3, PP8	PP2, PP3, PP8	PP1, PP8
		PP8, PP4, PP5	PP8, PP4, PP5	PP3, PP8	PP3, PP8	PP1, PP8	PP1, PP8	н/п
SISO	8K	PP7	PP7, PP4	PP8, PP4, PP5	PP8, PP4, PP5	PP2, PP3, PP8	PP2, PP3, PP8	PP1, PP8
		PP8, PP4, PP5	PP8, PP4, PP5	PP3, PP8	PP3, PP8	PP1, PP8	PP1, PP8	н/п
SISO	4K, 2K	н/п	PP7, PP4	PP4, PP5	н/п	PP2, PP3	н/п	PP1
		н/п	PP4, PP5	PP3	н/п	PP1	н/п	н/п
SISO	1K	н/п	н/п	PP4, PP5	н/п	PP2, PP3	н/п	PP1
		н/п	н/п	PP3	н/п	PP1	н/п	н/п

Уровень рассеянных пилотных несущих вычисляется следующим образом:

$$\operatorname{Re}\{c_{m,l,k}\} = 2 \cdot A_{SP} \cdot (1/2 - r_{l,k}), \quad \operatorname{Im}\{c_{m,l,k}\} = 0, \quad (21.32)$$

где $r_{l,k} = \begin{cases} w_{k+K_{\text{ext}}} \oplus p_{nl} & \text{в обычном режиме несущих,} \\ w_k \oplus p_{nl} & \text{в расширенном режиме несущих,} \end{cases} \quad K_{\text{ext}} — \text{число не-}$

Таблица 21.42. Максимальные амплитуды рассеянных пилотных несущих

Расположение рассеянных пилотных несущих	Амплитуда (A_{SP})	Эквивалентное усиление (дБ)
PP1, PP2	4/3	2,5
PP3, PP4	7/4	4,9
PP5, PP6, PP7, PP8	7/3	7,4

сущих, добавляемое с каждой стороны в расширенном режиме для несущих, m индекс Т2-кадра.

В дополнение к рассеянным пилотным несущим, описанным выше, некоторое количество непрерывных пилотных несущих вставляется в каждый символ кадра, за исключением символов P1, P2 и последнего символа кадра (если он присутствует).

В табл. 21.43 приведены варианты групп непрерывных пилотных несущих СР при различных режимах БПФ.

Таблица 21.43. Группы непрерывных пилотных несущих, используемые для каждой длины БПФ и их максимальные амплитуды

Длина БПФ	Используемые группы непрерывных пилотных несущих (СР)	Максимальные амплитуды непрерывных пилотных несущих (A_{CP})
1K	CP1	4/3
2K	CP1, CP2	4/3
4K	CP1, CP2, CP3	(4 $\sqrt{2}$)/3
8K	CP1, CP2, CP3, CP4	8/3
16K	CP1, CP2, CP3, CP4, CP5	8/3
32K	CP1, CP2, CP3, CP4, CP5, CP6	8/3

Модуляция амплитуд непрерывных пилотных несущих осуществляется в соответствии с формулой (21.32), в которой вместо величины A_{SP} используется значение A_{CP} .

Пилотные ячейки в символах P2 передаются на повышенных уровнях мощности. Максимальная их амплитуда $A_{P2} \frac{\sqrt{31}}{5}$ во всех режимах, за исключением режима 32K SISO, при котором $A_{P2} \frac{\sqrt{37}}{5}$. Модуляция амплитуд этих несущих осуществляется в соответствии с формулой (21.32), в которой вместо величины A_{SP} используется значение A_{P2} .

Передаваемый ОБПФ сигнал организован в виде кадров. Каждый кадр имеет длительность T_F и состоит из L_F символов OFDM. N_{T2} кадров составляют один суперкадр. Каждый символ составлен из множества несущих K_{total} и имеет длительность T_S . Он состоит из двух частей: полезная часть с длительностью T_U и защитный интервал с длительностью T_G .

Символы в кадре OFDM (за исключением P1) пронумерованы от 0 до $L_F - 1$. Все символы содержат данные и опорную информацию.

Поскольку сигнал OFDM содержит много по отдельности промодулированных несущих, каждый символ, в свою очередь, можно рассматривать как разделенный на ячейки, каждая из которых соответствует информации, передаваемой одной модулируемой несущей на интервале одного символа.

Генерируемый сигнал, в случае если не применяется ни кадр последующего расширения стандарта (FEF), ни уменьшение соотношения пиковой и средней мощностей (PAPR — Peak to Average Power Ratio), определяется следующим выражением:

$$s(t) = \operatorname{Re} \left\{ e^{j2\pi f_c t} \sum_{m=0}^{\infty} \left[p_1(t - mT_F) + \frac{5}{\sqrt{27 \times K_{\text{total}}}} \times \right. \right. \\ \left. \times \sum_{l=0}^{L_F-1} \sum_{k=K_{\min}}^{K_{\max}} c_{m,l,k} \times \psi_{m,l,k}(t) \right] \right\}, \quad (21.33)$$

где

$$\psi_{m,l,k}(t) = \begin{cases} e^{j2\pi \frac{k'}{T_U} (t - T_G - T_{P1} - lT_S - mT_F)} & \text{при } \begin{aligned} mT_F + T_{P1} + lT_S &\leq t \leq \\ &\leq mT_F + T_{P1} + (l+1)T_S, \end{aligned} \\ 0 & \text{иначе,} \end{cases}$$

$k \in [K_{\min}; K_{\max}]$ — номер несущей; l — число символов OFDM, начиная с 0, для первого символа P2 кадра; m — число T2-кадров; K_{total} — число переданных несущих; L_F — число символов OFDM на кадр; T_S — общая длительность символа для всех символов за исключением P1, и $T_S = T_U + T_G$; T_U — активная длительность символа; T_G — длительность защитного интервала; f_c — центральная частота РЧ-сигнала; k' — индекс несущей относительно центральной частоты, $k' = k - (K_{\max} + K_{\min})/2$; $c_{m,l,k}$ — комплексное модулирующее значение для несущей k символа OFDM номер l в T2-кадре номер m ; T_{P1} — длительность символа P1, заданная выражением $T_{P1} = 2048 \cdot T$; T_F — длительность кадра, $T_F = L_F \cdot T_S + T_{P1}$, $p_1(t)$ — форма сигнала P1.

В табл. 21.44 приведена величина элементарного периода T как функция ширины полосы используемого канала связи, а в табл. 21.45 приведены параметры OFDM сигнала в зависимости от используемых режимов ОБПФ.

Таблица 21.44. Элементарный период как функция полосы

Ширина полосы	1,7 МГц	5 МГц	6 МГц	7 МГц	8 МГц	10 МГц *
Элементарный период T	71/131 мкс	7/40 мкс	7/48 мкс	1/8 мкс	7/64 мкс	7/80 мкс

*только для профессиональных приложений

Генерируемый сигнал включает в себя вставку защитных интервалов, абсолютная длительность которого, выраженная в числах, кратных элементарному периоду T , приведена в табл. 21.46.

Для уменьшения отношения пиковой и средней мощностей PAPR возможна реализация двух модификаций сигнала OFDM, использующих метод активного расширения созвездия и метод резервирования несущих. Метод активного расширения созвездий не применяется к пилотным или резервируемым несущим, а также в случае если используются повернутые созвездия.

Алгоритм активного расширения созвездий (ACE) реализуется путем формирования сигнала во временной области \mathbf{x}_{ACE} , который замещает исходный сигнал во временной области $\mathbf{x} = [x_0, x_1, \dots, x_{N_{\text{FFT}}-1}]$, генерированный с помощью ОБПФ из множества значений частотной области $\mathbf{X} = [X_0, X_1, \dots, X_{N_{\text{FFT}}-1}]$ (рис. 21.40).

Таблица 21.45. Параметры OFDM

Параметр		Режим 1К	Режим 2К	Режим 4К	Режим 8К	Режим 16К	Режим 32К
Число несущих K_{total}	обычный режим	853	1705	3409	6817	13633	27265
	расширенный режим	н/п	н/п	н/п	6913	13921	27841
Значение числа несущих K_{min}	обычный режим	0	0	0	0	0	0
	расширенный режим	н/п	н/п	н/п	0	0	0
Значение числа несущих K_{max}	обычный режим	852	1704	3408	6816	13632	27264
	расширенный режим	н/п	н/п	н/п	6912	13920	27840
Число несущих K_{ext} , добавляемое с каждой стороны в расширенном режиме		0	0	0	48	144	288
Длительность T_U		$1024 \cdot T$	$2048 \cdot T$	$4096 \cdot T$	$8192 \cdot T$	$16384 \cdot T$	$32768 \cdot T$
Длительность T_U , мкс		112	224	448	896	1792	3584
Интервал между несущими $1/T_U$ (Гц)		8929	4464	2232	1116	558	279
Интервал между несущими K_{min} и K_{max} ($K_{\text{total}} - 1$) / $/T_U$	обычный режим	7,61 МГц	7,61 МГц				
	расширенный режим	н/п	н/п	н/п	7,71 МГц	7,77 МГц	7,77 МГц

Таблица 21.46. Длительность защитного интервала в элементарных периодах T

Длина БПФ	Доля, занимаемая защитным интервалом (T_G/T_U)						
	1/128	1/32	1/16	19/256	1/8	19/128	1/4
32К	$256 \cdot T$	$1024 \cdot T$	$2048 \cdot T$	$2432 \cdot T$	$4096 \cdot T$	$4864 \cdot T$	н/п
16К	$128 \cdot T$	$512 \cdot T$	$1024 \cdot T$	$1216 \cdot T$	$2048 \cdot T$	$2432 \cdot T$	$4096 \cdot T$
8К	$64 \cdot T$	$256 \cdot T$	$512 \cdot T$	$608 \cdot T$	$1024 \cdot T$	$1216 \cdot T$	$2048 \cdot T$
4К	н/п	$128 \cdot T$	$256 \cdot T$	н/п	$512 \cdot T$	н/п	$1024 \cdot T$
2К	н/п	$64 \cdot T$	$128 \cdot T$	н/п	$256 \cdot T$	н/п	$512 \cdot T$
1К	н/п	н/п	$64 \cdot T$	н/п	$128 \cdot T$	н/п	$256 \cdot T$

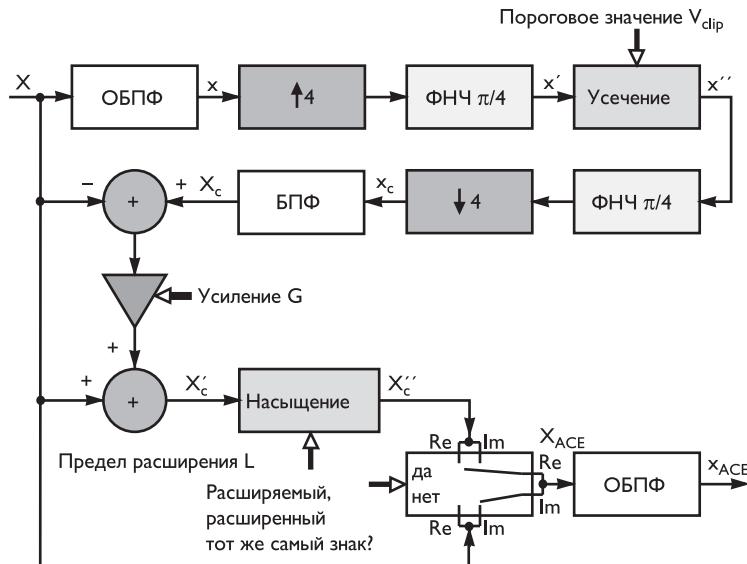


Рис. 21.40. Реализация алгоритма активного расширения созвездий

Отсчеты $\mathbf{x}' = [x'_0, x'_1, \dots, x'_{4 \cdot N_{\text{FFT}} - 1}]$ формируются из \mathbf{x} путем четырехкратной интерполяции. Сочетание ОБПФ, передискретизации и низкочастотной фильтрации реализуется с помощью заполнения нулями до четырехкратной длины и последующего преобразования ОБПФ.

Сигнал $\mathbf{x}'' = [x''_0, x''_1, \dots, x''_{N_{\text{FFT}}-1}]$ получается с помощью применения к \mathbf{x}' оператора ограничения, определенного следующим образом:

$$x_k'' = \begin{cases} x_k', & \|x_k'\| \leq V_{\text{clip}}; \\ V_{\text{clip}} \cdot \frac{x_k'}{\|x_k'\|}, & \text{если } \|x_k'\| \geq V_{\text{clip}}, \end{cases} \quad (21.34)$$

где порог ограничения V_{clip} — параметр алгоритма ACE. Отсчеты $\mathbf{x}_c = [x_{c0}, x_{c1}, \dots, x_{cN_{FFT}-1}]$ формируются из \mathbf{x}'' путем четырехкратного прореживания. Сочетание низкочастотной фильтрации, передискретизации и БПФ реализуется с помощью заполнения нулями до четырехкратной длины и последующего преобразования БПФ. БПФ преобразует отсчеты \mathbf{x}_c в сигнал \mathbf{X}_c .

Новый сигнал \mathbf{X}'_c создается с помощью суммирования \mathbf{X}_c и \mathbf{X} следующим образом:

$$\mathbf{X}'_c = \mathbf{X} + G \cdot (\mathbf{X}_c - \mathbf{X}). \quad (21.35)$$

Коэффициент расширения G также является параметром алгоритма ACE.

\mathbf{X}'_c получается из \mathbf{X}'_c с помощью «оператора насыщения», который по отдельности обрабатывает реальные и мнимые компоненты с тем, чтобы модуль отдельных компонент не превышал заданное значение L :

$$\begin{aligned} \operatorname{Re}\{X''_{c,k}\} &= \begin{cases} \operatorname{Re}\{X'_{c,k}\}, & \text{если } \left|\operatorname{Re}\{X'_{c,k}\}\right| \leq L; \\ L, & \text{если } \operatorname{Re}\{X'_{c,k}\} \geq L; \\ -L, & \text{если } \operatorname{Re}\{X'_{c,k}\} < -L; \end{cases} \\ \operatorname{Im}\{X''_{c,k}\} &= \begin{cases} \operatorname{Im}\{X'_{c,k}\}, & \text{если } \left|\operatorname{Im}\{X'_{c,k}\}\right| \leq L; \\ L, & \text{если } \operatorname{Im}\{X'_{c,k}\} \geq L; \\ -L, & \text{если } \operatorname{Im}\{X'_{c,k}\} < -L, \end{cases} \end{aligned} \quad (21.36)$$

где предел расширения L — параметр алгоритма ACE.

Тогда \mathbf{X}_{ACE} составляется с помощью простого выбора действительных и мнимых компонент из тех, которые относятся к \mathbf{X} , \mathbf{X}'_c .

$$\operatorname{Re}\{X_{ACE,k}\} = \begin{cases} \operatorname{Re}\{X''_{c,k}\}, & \text{если } \operatorname{Re}\{X_k\} \text{ — расширяемое,} \\ & \text{а также } |\operatorname{Re}\{X''_{c,k}\}| > |\operatorname{Re}\{X_k\}| \\ & \text{и } \operatorname{Re}\{X''_{c,k}\} \cdot \operatorname{Re}\{X_k\} > 0; \\ \operatorname{Re}\{X_k\}, & \text{иначе;} \end{cases} \quad (21.37)$$

$$\operatorname{Im}\{X_{ACE,k}\} = \begin{cases} \operatorname{Im}\{X''_{c,k}\}, & \text{если } \operatorname{Im}\{X_k\} \text{ — расширяемое,} \\ & |\operatorname{Im}\{X''_{c,k}\}| > |\operatorname{Im}\{X_k\}|, \\ & \operatorname{Im}\{X''_{c,k}\} \cdot \operatorname{Im}\{X_k\} > 0; \\ \operatorname{Im}\{X_k\}, & \text{иначе.} \end{cases} \quad (21.38)$$

Сигнал \mathbf{x}_{ACE} получается из \mathbf{X}_{ACE} с помощью ОБПФ.

Компонента определяется как расширяемая, если она принадлежит к модулированной ячейке данных и если ее абсолютное значение равно максимальному значению компоненты, связанной с модулирующим созвездием, используемым для этой ячейки. Как пример, компонента, принадлежащая к 256-QAM модулированной ячейке, является расширяемой, если ее значение равно $\pm 15/\sqrt{170}$.

Выбор значения коэффициента G осуществляется в пределах от 0 до 31 с шагом 1. Выбор порога ограничения V_{clip} осуществляется в диапазоне от +0 дБ и +12,7 дБ с шагом 0,1 дБ выше среднеквадратичного отклонения исходного сигнала во временной области. Выбор максимального значения расширения L осуществляется в диапазоне от 0,7 дБ до 1,4 дБ с шагом 0,1 дБ.

Если L устанавливается в максимальное значение, то максимальное увеличение мощности на несущую после расширения получается при QPSK, и оно ограничивается +6 дБ.

При использовании алгоритма уменьшения PAPR с использованием зарезервированных несущих пиковые значения сигнала во временной области последовательно удаляются с помощью набора импульснаподобных ядер, образуемых с помощью зарезервированных несущих. Опорная функция ядра определяется как:

$$p = \frac{\sqrt{N_{FFT}}}{N_{TR}} \text{ОБПФ}(\mathbf{1}_{TR}), \quad (21.39)$$

где N_{FFT} и N_{TR} обозначают длину БПФ и число зарезервированных несущих, соответственно. Вектор $\mathbf{1}_{TR}$ содержит N_{TR} элементов, состоящих из единиц, в позициях, соответствующих индексам зарезервированных несущих, и $(N_{FFT} - N_{TR})$ элементов, состоящих из нулей, в остальных позициях.

Пусть вектор для уменьшения пикового значения сигнала обозначен как \mathbf{c} , вектор сигнала данных во временной области — как \mathbf{x} , тогда процедура алгоритма уменьшения PAPR заключается в следующем:

Инициализация.

Начальные значения для процедуры уменьшения пикового значения сигнала устанавливаются равными нулю:

$$\mathbf{c}^{(0)} = [0 \dots 0]^T,$$

где $\mathbf{c}^{(i)}$ обозначает вектор для уменьшения пикового значения сигнала, вычисленный в i -й итерации.

Итерационный процесс.

- 1) i начинается с 1.
- 2) Определяется максимальная величина $(\mathbf{x} + \mathbf{c}^{(i)})$, y_i , и индекс соответствующего отсчета, m_i , в i -й итерации.

$$\begin{cases} y_i = \max_n |x_n + c_n^{(i)}| \\ m_i = \arg \max_n |x_n + c_n^{(i)}| \end{cases} \quad \text{для } n = 0, 1, \dots, N_{\text{FFT}} - 1,$$

где x_n и $c_n^{(i)}$ обозначают n -й элемент вектора \mathbf{x} и $\mathbf{c}^{(i)}$, соответственно. Если y_i меньше или равен уровню требуемой величины ограничения, то V_{clip} уменьшает i на 1 и переходит к шагу 5.

- 3) Обновление вектора для уменьшения пикового значения сигнала $\mathbf{c}^{(i)}$:

$$c^{(i)} = c^{(i-1)} - \alpha_i \mathbf{p}(m_i),$$

где $\alpha_i = \frac{x_{m_i} + c_{m_i}^{(i-1)}}{y_i}$ ($y_i - V_{\text{clip}}$), где $\mathbf{p}(m_i)$ обозначает вектор, циклически сдвинутый на m_i , k -й элемент которого имеет вид $p_k(m_i) = p_{(k-m_i) \bmod N_{\text{FFT}}}$.

- 4) Если i меньше, чем максимально допустимое значение итераций, i увеличивается на 1 и выполняется возврат к шагу 2. В противном случае выполняется переход к шагу 5.
- 5) Завершение итераций. Переданный сигнал, \mathbf{x}' , получается с помощью прибавления сигнала уменьшения пикового значения к сигналу данных:

$$\mathbf{x}' = \mathbf{x} + \mathbf{c}^{(i)}.$$

Символы OFDM состоят из равнотстоящих друг от друга ортогональных несущих. Амплитуды и фазы несущих, соответствующих ячейкам данных, изменяются от символа к символу в соответствии с вышеописанным процессом отображения.

Спектральная плотность мощности $P_{k'}(f)$ каждой несущей на частоте равна:

$$P_{k'}(f) = \left[\frac{\sin \pi(f - f_{k'})T_S}{\pi(f - f_{k'})T_S} \right]^2, \quad (21.40)$$

где $f_{k'} = f_c + \frac{k'}{T_u}$ при $\left(-\frac{K_{\text{total}}-1}{2}\right) \leq k' \leq \frac{K_{\text{total}}-1}{2}$.

Теоретический спектр сигнала передачи в системе DVB-T2 показан на рис. 21.41а (для каналов с шириной полосы пропускания 8 МГц). Поскольку длительность символа OFDM больше, чем обратная величина интервала между несущими, основной лепесток спектральной плотности мощности каждой несущей уже, чем

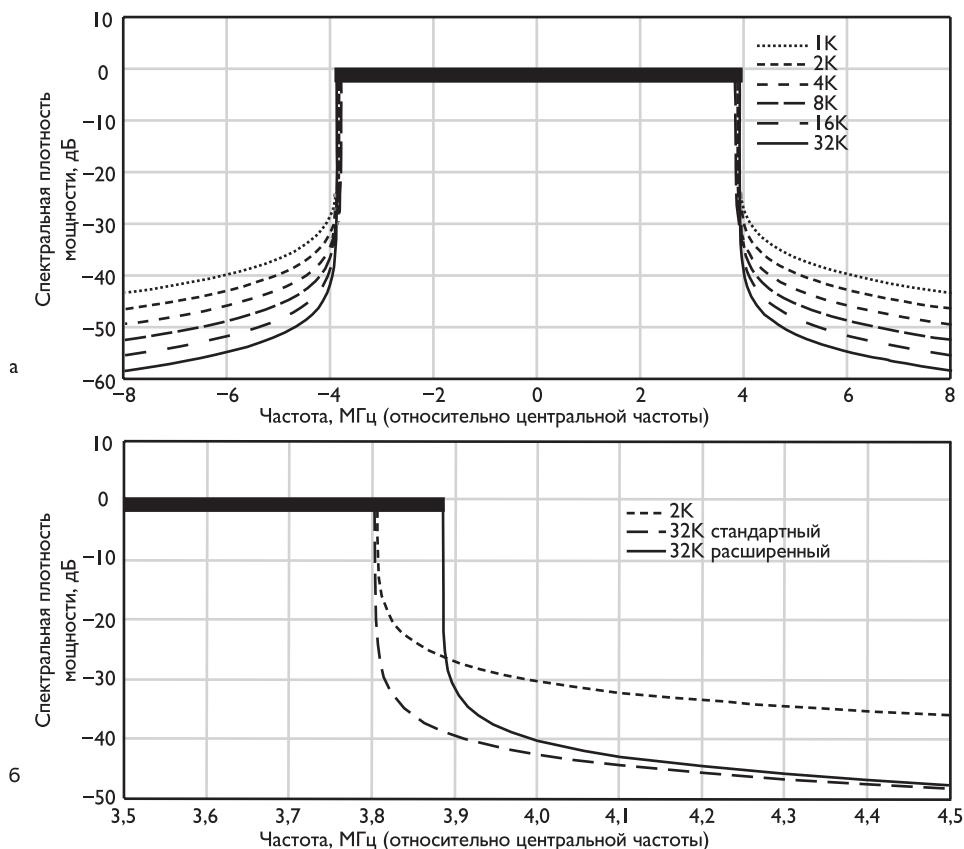


Рис. 21.41. Теоретический спектр сигнала DVB-T2 (а) и подробное изображение теоретического спектра DVB-T2 при 1/8 доли защитного интервала (для каналов с полосой пропускания 8 МГц) (б)

удвоенный интервал между несущими. Следовательно, спектральная плотность не является постоянной в пределах номинальной полосы пропускания.

Применение методов уменьшения PAPR может существенно помочь понизить уровень внеполосных излучений после большого усиления мощности.

В табл. 21.47 в качестве примера приведена реализуемая стандартом DVB-T2 максимальная скорость битового потока при ширине полосы телевизионного канала 8 МГц, 32К номинальных поднесущих, относительном защитном интервале 1/128 и схеме пилотных поднесущих PP7 [8.24–8.26].

21.4. Система цифрового спутникового телевизионного вещания DVB-S

Методы модуляции и канального кодирования DVB-S используются для первичного и вторичного распределения спутникового цифрового многопрограмм-

Таблица 21.47. Максимальная скорость битового потока при ширине полосы телевизионного канала 8 МГц

Тип модуляции	Скорость кодирования	Рекомендуемая максимальная скорость, Мбит/с	Длина Т2-кадра, OFDM-символов	Число кодовых слов в кадре
QPSK	1/2	7,4442731	60	50
	3/5	8,9457325		
	2/3	9,9541201		
	3/4	11,197922		
	4/5	12,948651		
	5/6	12,456553		
16-QAM	1/2	15,037432	60	101
	3/5	18,07038		
	2/3	20,107323		
	3/4	22,619802		
	4/5	24,136276		
	5/6	25,162236		
64-QAM	1/2	22,481705	60	151
	3/5	27,016112		
	2/3	30,061443		
	3/4	33,817724		
	4/5	36,084927		
	5/6	37,618789		
256-QAM	1/2	30,074863	60	202
	3/5	36,140759		
	2/3	40,214645		
	3/4	45,239604		
	4/5	48,272552		
	5/6	50,524472		

ного ТВ/ТВЧ в полосах системы стационарной спутниковой связи (FSS — Fixed Satellite Service) и системы спутникового вещания (BSS — Broadcast Satellite Service). Система предназначена для обеспечения сервиса «непосредственно-на-дом» (Direct To Home — DTH) с использованием потребительского интегрированного приемника-декодера (IRD — Integrated Receiver Decoder), а также для систем коллективного приема (SMATV — Satellite Master Antenna Television) и головных станций кабельного телевидения с возможностью повторной модуляции.

В системе применена модуляция QPSK и защита от ошибок на основе сверточного кода и сокращенного кода Рида–Соломона. Система может быть использована в спутниковых ретрансляторах с различной шириной полосы.

На рис. 21.42 приведена функциональная структурная схема передающей части системы DVB-S. Система непосредственно совместима с телевизионными сигналами, закодированными по стандарту MPEG-2.

Техника помехоустойчивого кодирования, принятая в системе, разработана в целях достижения «квазибезошибочного» (QEФ — Quasi-Error-Free) режима работы, при котором возможно возникновение менее одного случая неисправимой ошибки на час передачи, что соответствует уровню ошибки (BER — Bit Error

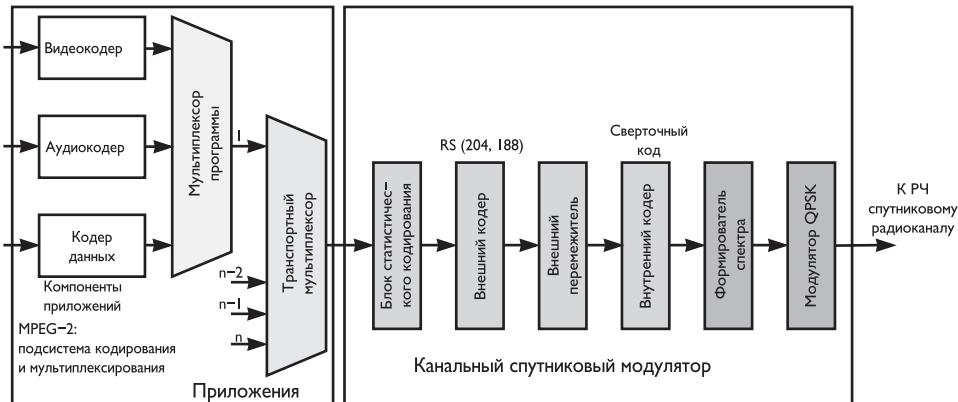


Рис. 21.42. Структурная схема передающей части системы DVB-S

Ratio) 10^{-10} – 10^{-11} на входе демультиплексора MPEG-2.

При адаптации сигнала к спутниковому каналу связи осуществляются следующие операции:

- адаптация транспортного мультиплексирования с использованием статистического кодирования, аналогичного используемому в системе DVB-T (см. рис. 21.13);
- внешнее кодирование с использованием кода Рида–Соломона (рис. 21.14) RS(204,188, $t = 8$);
- сверточное перемежение (рис. 21.15);
- внутреннее кодирование с использованием сверточного кода с выкальванием (рис. 21.16); система предусматривает сверточное кодирование со скоростями кода $1/2$, $2/3$, $3/4$, $5/6$ и $7/8$;
- формирование сигнала в основной полосе частот;
- модуляция QPSK.

В системе применяется обычная модуляция типа QPSK с кодами Грэя и абсолютным отображением (без дифференциального кодирования).

Прежде чем сигналы I и Q (математически представленные последовательностью дельта-функций Дирака, расположенных на расстоянии длительности символа $T_S = 1/R_S$, с соответствующим знаком) будут промодулированы, они должны быть обработаны фильтром с характеристикой, имеющей фактор спада $\alpha = 0,35$:

$$H(f) = \begin{cases} 1, & |f| < f_N(1 - \alpha); \\ \left[\frac{1}{2} + \frac{1}{2} \sin \frac{\pi}{2f_N} \left(\frac{f_N - |f|}{\alpha} \right) \right]^{1/2}, & \text{при } f_N(1 - \alpha) \leq |f| \leq f_N(1 + \alpha), \\ 0, & |f| > f_N(1 + \alpha), \end{cases} \quad (21.41)$$

где $f_N = \frac{1}{2T_S} = \frac{R_S}{2}$ — частота Котельникова–Найквиста; R_S — скорость передачи символов.

Для согласования передаваемого сигнала с полосой и энергетическими характеристиками конкретного транспондера устанавливается требуемое соотно-

шение BW/R_S , где BW — полоса транспондера по уровню -3 дБ. При QPSK-модуляции, скорости сверточного кода R и скорости RS-кода $188/204$ соответствующая скорость передачи информационных символов равна:

$$R_U = R \cdot (2R_S) \cdot (188/204). \quad (21.42)$$

В табл. 21.48 приведены примеры параметров передачи сигналов в полосе транспондера.

Таблица 21.48. Примеры зависимости скоростей битового потока от полосы транспондера

Полоса (-3 дБ), МГц	Полоса (-1 дБ), Мбит/с	R_S при $BW/R_S = 1,28$, Мбит/с	R_U при $R = 1/2$, Мбит/с	R_U при $R = 2/3$, Мбит/с	R_U при $R =$ $3/4$, Мбит/с	R_U при $R =$ $5/6$, Мбит/с	R_U при $R =$ $7/8$, Мбит/с
54	48,6	42,2	38,9	51,8	58,3	64,8	68,0
46	41,4	35,9	33,1	44,2	49,7	55,2	58,0
40	36,0	31,2	28,8	38,4	43,2	48,0	50,4
36	32,2	28,1	25,9	34,6	38,9	43,2	45,4
33	29,7	25,8	23,8	31,7	35,6	39,6	41,6
30	27,0	23,4	21,6	28,8	32,4	36,0	37,8
27	24,3	21,1	19,4	25,9	29,2	32,4	34,0
26	23,4	20,3	18,7	25,0	28,1	31,2	32,8

На рис. 21.43а приведен шаблон спектра сигнала на выходе модулятора, а на рис. 21.43б приводится относительная характеристика групповой задержки для аппаратной реализации фильтра Найквиста.

Определение точек от А до S, изображенных на рис. 21.43, дается в табл. 21.49. Частотная характеристика фильтра строится исходя из того, что на вход подаются идеальные импульсы дельта-функции Дирака, расстояние между которыми равно длине символа $T_S = 1/R_S^{1/2}f_N$, причем в случае П-образных входных сигналов должна быть проведена соответствующая их sinc-коррекция.

21.5. Система высокоскоростного цифрового спутникового ТВ-вещания DVB-S2

Стандарт [8.29–8.33] спутниковой системы второго поколения DVB-S2 был разработан консорциумом DVB для видеовещания, интерактивных услуг, сбора новостей и других широкополосных спутниковых приложений и является дополнением к стандарту [8.27–8.28] вещания DVB-S.

Важнейшим фактором создания этого стандарта стали планы массового запуска ТВЧ, что потребовало разработки форматов канального кодирования, более эффективно использующих имеющиеся частотные ресурсы.

Другой причиной создания стандарта DVB-S2 были дефекты в работе приемных систем Ка-диапазона (15,40... 27,50 ГГц), их сильная зависимость от погодных условий, в частности от дождя. Поэтому для трансляций в этом диапазоне

Таблица 21.49. Определение точек шаблонов, приведенных на рис. 21.43

Точка	Относительная частота f/f_N	Относительная мощность, дБ	Относительная групповая задержка $\tau_{\text{пгр}} \times f_N$
A	0,0	+0,25	+0,07
B	0,0	-0,25	-0,07
C	0,2	+0,25	+0,07
D	0,2	-0,40	-0,07
E	0,4	+0,25	+0,07
F	0,4	-0,40	-0,07
G	0,8	+0,15	+0,07
H	0,8	-1,10	-0,07
I	0,9	-0,50	+0,07
J	1,0	-2,00	+0,07
K	1,0	-4,00	-0,07
L	1,2	-8,00	
M	1,2	-11,00	
N	1,8	-35,00	
P	1,4	-16,00	
Q	1,6	-24,00	
S	2,12	-40,00	

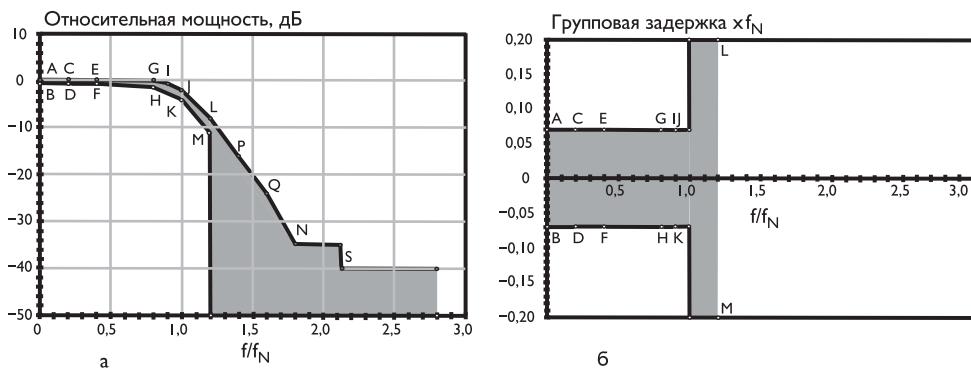


Рис. 21.43. Шаблоны спектра сигнала на выходе модулятора (а) и характеристики групповой задержки фильтра (б)

часто требуется более высокая помехозащищенность, чем в С- и К-диапазонах (3,5...4,2 ГГц и 10,7...12,75 ГГц, соответственно).

И, наконец, еще одной причиной реализации стандарта DVB-S2 является использование интерактивных спутниковых сетей с адресными услугами, требующих значительного транспортного ресурса.

Таким образом, универсальный стандарт DVB-S2 позволяет создавать сети для распространения ТВ-программ стандартной или высокой четкости, сети для предоставления интерактивных услуг, например доступа в Интернет, сети для профессиональных приложений, таких как передача цифрового ТВ от студии к

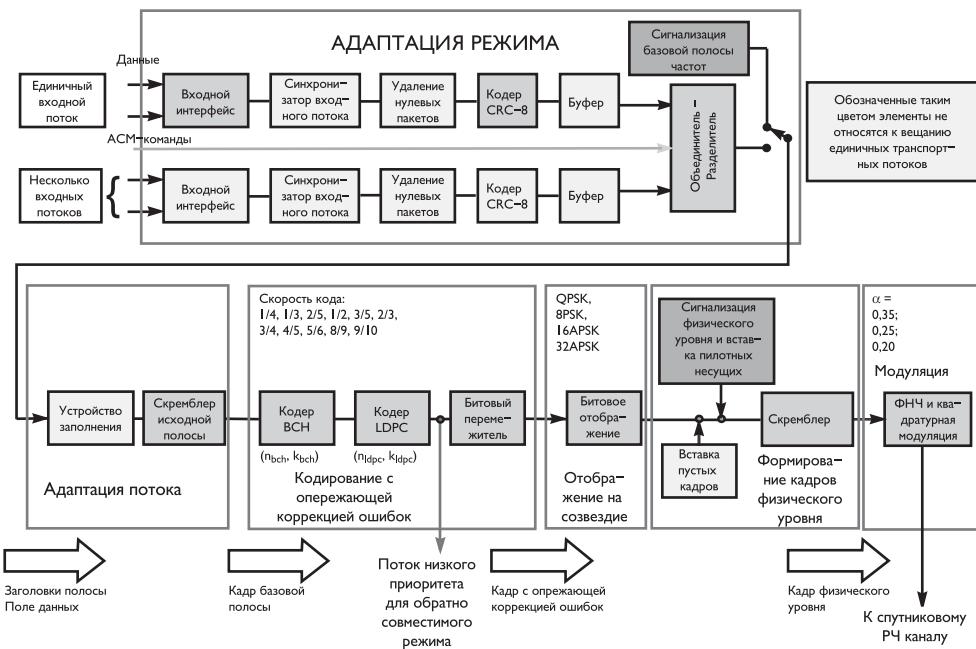


Рис. 21.44. Блок-схема системы по стандарту DVB-S2

студии, сбор новостей и раздача сигнала на эфирные ретрансляторы. Стандарт удобен для формирования сетей передачи данных и создания IP-магистралей.

Большинство эффективных механизмов, заложенных в DVB-S2, несовместимы со старыми стандартами. Потому для выполнения требования совместимости вниз в стандарт введено два режима: один — менее эффективный, совместимый вниз, а другой — реализующий все новые возможности, но не позволяющий использовать приемники стандарта DVB-S.

Функциональная структурная схема системы DVB-S2 приведена на рис. 21.44.

Адаптация режима обеспечивает сопряжение системы с входным потоком, синхронизацию входного потока (опционально), удаление нулевых пакетов (только при адаптивном кодировании и модуляции (ACM) и входном сигнале в формате транспортного потока), кодирование CRC-8 для обнаружения ошибок на уровне пакета в приемнике (только для пакетированных входных потоков), объединение входных потоков (только для режимов с несколькими входными потоками) и разделение на поля данных DATA FIELD. Для постоянного кодирования и модуляции (CCM) и единичного входного транспортного потока адаптация режима должна состоять из «прозрачного» DVB-ASI (или DVB-параллельного) логического битового преобразования и кодирования CRC-8.

Заголовок базовой полосы должен быть присоединен перед полем данных для уведомления о формате входного потока и типе адаптации режима.

Адаптация потока применяется для обеспечения заполнения кадра базовой полосы и для ее скремблирования.

Кодирование с опережающей коррекцией ошибок (FEC) выполняется с помощью последовательного соединения внешних кодов BCH и внутренних ко-

дов LDPC с кодовыми скоростями $1/4, 1/3, 2/5, 1/2, 3/5, 2/3, 3/4, 4/5, 5/6, 8/9, 9/10$. В зависимости от области применения длина FEC-кодированного блока равна $N_{ldpc} = 64\,800$ или $16\,200$ битов. Если применяется переменная или адаптивная система кодирования и модуляции (VCM или ACM), режим модуляции FEC может изменяться от одного кадра к другому, но оставаться постоянным в пределах кадра.

При использовании обратносовместимых режимов (BC — Backwards-Compatible) обеспечивается передача по отдельному спутниковому каналу двух транспортных потоков. Первый из них (высокого приоритета, ВП) совместим с приемниками DVB-S и DVB-S2, второй (низкого приоритета, НП) совместим только с приемниками DVB-S2.

Обратная совместимость может быть в optionalном порядке реализована в соответствии с двумя подходами:

- многоуровневые модуляции, где сигналы DVB-S2 и DVB-S асинхронно объединяются в радиочастотном канале;
- иерархическая модуляция, где два транспортных потока ВП и НП синхронно объединяются на уровне модулированных символов асимметричного созвездия 8-PSK.

Побитовое перемежение применяется к FEC-кодированным битам при **отображении** на созвездия 8-PSK, 16-APSK и 32-APSK (рис. 21.45).

Формирование кадров физического уровня применяется синхронно с кадрами FEC для обеспечения вставки пустых кадров PLFRAME, сигнализации физического уровня (PL), вставки пилотных символов (optionalно) и скремблирования физического уровня для рассеивания энергии. Пустые кадры передаются в том случае, если отсутствуют полезные данные, готовые для отправки по каналу. Система предоставляет регулярную структуру формирования кадров, основанную на слотах (полях SLOT), состоящих из $M = 90$ модулированных символов, и позволяющую надежно синхронизировать приемник на основе структуры FEC-блоков. Слот предназначен для сигнализации физического уровня, включая определение границы начала кадра и определение режима передачи. Для улучшения восстановления несущих в приемниках возможно введение регулярного растра пилотных символов ($P = 36$ пилотных символов на каждые 16 слотов из 90 символов), в то время как возможен также режим беспилотной передачи, дающий дополнительную полезную мощность 2,4%.

Фильтрация базовой полосы и квадратурная модуляция применяются для формирования спектра сигнала в соответствии с соотношением (21.41) с использованием коэффициентов спада $\alpha = 0,35, 0,25$ или $0,20$ и для генерации РЧ-сигнала.

На рис. 21.46а приведена структура формата кадра на выходе адаптера потока, который содержит заголовок, поле данных и, при необходимости, блок, заполняемый нулевыми битами.

10-байтовый заголовок базовой полосы содержит описание формата входного потока(ов), тип адаптации режима и передачи коэффициента спада (2 байта). Первый байт определяет:

- поле TS/GS (2 бита): вход транспортного потока или общего потока (пакетированного или непрерывного);

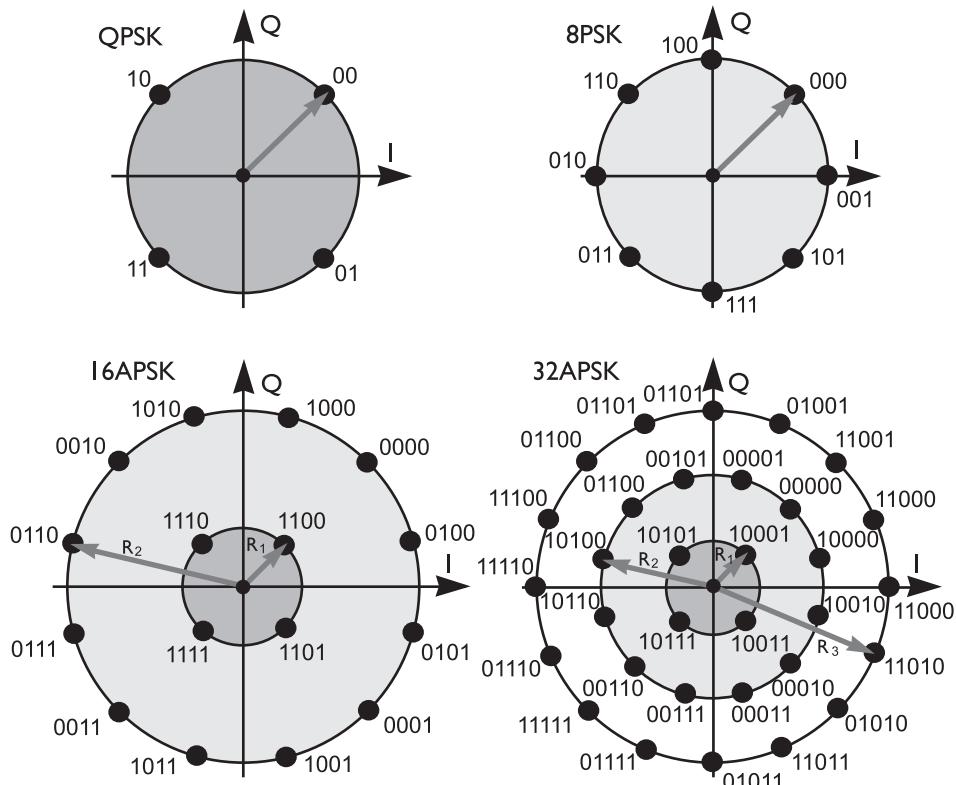


Рис. 21.45. Используемые варианты созвездий в стандарте DVB-S2

- поле SIS/MIS (1 бит): единичный или несколько входных потоков;
- поле CCM/ACM (1 бит): постоянное или адаптивное кодирование и модуляция;
- индикатор синхронизации входного потока ISSYI (1 бит); если ISSYI=1 (active), то поле ISSYI вставляется после пользовательских пакетов;
- удаление нулевых пакетов активно/не активно — NPD (1 бит);
- коэффициент спада (α) — RO (2 бита).

Второй байт используется как идентификатор входного потока (ISI), если кодируется несколько входных потоков. Последующие байты характеризуют:

- UPL (2 байта) — длину пользовательского пакета в битах, в диапазоне [0... 65 535];
- DFL (2 байта) — длину поля данных в битах, в диапазоне [0... 58 112];
- SYNC (1 байт) — копию синхробайта пользовательского пакета, например 47_{HEX} для пакетов транспортного потока MPEG;
- SYNCD (2 байта) — расстояние в битах от начала поля данных до первого пользовательского пакета данного кадра;
- CRC-8 (1 байт) — код обнаружения ошибок (применяется к первым 9 байтам заголовка).

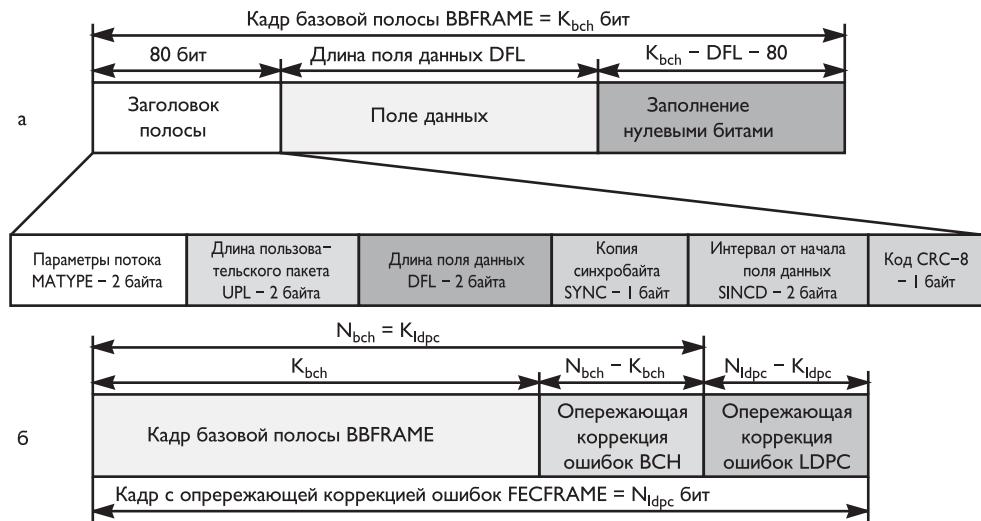


Рис. 21.46. Структуры форматов кадров на выходе адаптера потока (а) и на выходе кодера с опережающей коррекцией ошибок (б)

Структура формата кадра с опережающей коррекцией ошибок BCH и LDPC приведена на рис. 21.46б. Параметры такого кадра для основного размера 64 800 битов и сокращенного размера 16 200 битов приведены в табл. 21.50 и 21.51 соответственно.

Таблица 21.50. Параметры кодирования для кадров основного размера FECFRAME $N_{ldpc} = 64\,800$

Код LDPC	Некодированный блок BCH-кода K_{bch}	Кодированный блок BCH-кода N_{bch} , некодированный блок LDPC-кода K_{ldpc}	Коррекция t ошибок BCH-кода	Кодированный блок LDPC-кода N_{ldpc}
1/4	16 008	16 200	12	64 800
1/3	21 408	21 600	12	64 800
2/5	25 728	25 920	12	64 800
1/2	32 208	32 400	12	64 800
3/5	38 688	38 880	12	64 800
2/3	43 040	43 200	10	64 800
3/4	48 408	48 600	12	64 800
4/5	51 648	51 840	12	64 800
5/6	53 840	54 000	10	64 800
8/9	57 472	57 600	8	64 800
9/10	58 192	58 320	8	64 800

Полиномы кода BCH для основных и сокращенных кадров FECFRAME соответствуют табл. 21.19.

Кодер LDPC преобразует блок информации размером K_{ldpc} , $i = (i_0, \dots, i_{K_{ldpc}-1})$ в кодовое слово размером N_{ldpc} , $c = (i_0, i_1, \dots, i_{K_{ldpc}-1}, p_0, p_1, \dots, p_{N_{ldpc}-K_{ldpc}-1})$.

Таблица 21.51. Параметры кодирования для кадров сокращенного размера FECFRAME
 $N_{ldpc} = 16\,200$

Код LDPC	Некодированный блок БЧХ-кода K_{bch}	Кодированный блок БЧХ-кода N_{bch} , некодированный блок LDPC-кода K_{ldpc}	Коррекция t ошибок БЧХ-кода	Эффективная скорость LDPC-кода $K_{ldpc}/16\,200$	Кодированный блок LDPC-кода N_{ldpc}
1/4	3072	3240	12	1/5	16 200
1/3	5232	5400	12	1/3	16 200
2/5	6312	6480	12	2/5	16 200
1/2	7032	7200	12	4/9	16 200
3/5	9552	9720	12	3/5	16 200
2/3	10632	10800	12	2/3	16 200
3/4	11712	11880	12	11/15	16 200
4/5	12432	12600	12	7/9	16 200
5/6	13152	13320	12	37/45	16 200
8/9	14232	14400	12	8/9	16 200

Передача кодового слова начинается в заданном порядке с i_0 и заканчивается $p_{N_{ldpc}-K_{ldpc}-1}$. Параметры кода LDPC (N_{ldpc}, K_{ldpc}) приведены в табл. 21.50 и 21.51.

После кодирования кадров FECFRAME и побитового перемежения осуществляется их последовательное отображение на выбранные варианты созвездия с формированием кадров XFECFRAME (compleX FECFRAME) и последующей реализацией кадров физического уровня — PLFRAME (рис. 21.47).

При генерации кадра физического уровня выполняются следующие процессы:

- разделение кадра XFECFRAME на целое число S слотов постоянной длины $M = 90$ символов (табл. 21.52);
- генерация и вставка заголовка PLHEADER размером M символов перед кадром XFECFRAME для конфигурации приемника;
- вставка пилотного блока для синхронизации приемника (для режимов, требующих пилотных символов) через каждые 16 слотов; этот блок состоит из $P = 36$ символов;
- рандомизация (I, Q) модулированных символов в устройстве скремблирования физического уровня.

Таблица 21.52. Число слотов в кадре физического уровня

Число битов на символ созвездия η_{mod} (бит/с/Гц)	$N_{ldpc} = 64\,800$ (кадр основного размера)		$N_{ldpc} = 16\,200$ (кадр сокращенного размера)	
	S	η % беспилотный	S	η % беспилотный
2	360	99,72	90	98,90
3	240	99,59	60	98,36
4	180	99,45	45	97,83
5	144	99,31	36	97,30

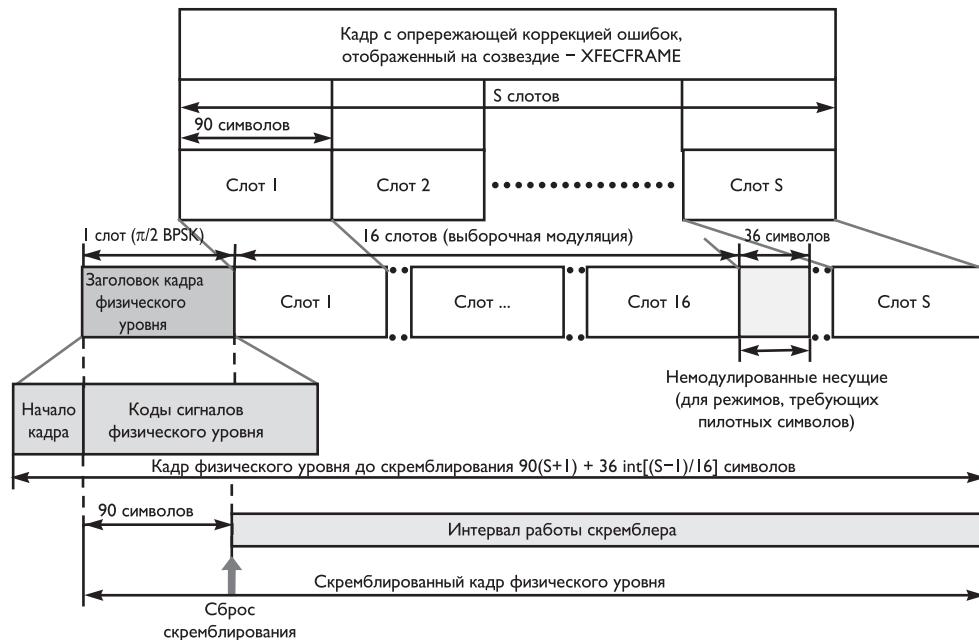


Рис. 21.47. Формирование кадра физического уровня PLFRAME

Эффективность PLFRAMING равна

$$\eta = 90 \cdot S / [90 \cdot (S + 1) + P \cdot \text{int}\{(S - 1)/16\}],$$

где $P = 36$.

Заголовок кадра физического уровня PLHEADER (один слот, 90 символов) состоит из следующих полей:

- начала кадра — SOF (26 символов);
- сигнализации физического уровня — PLS (64 символа) в виде несистематического двоичного кода длиной 64, размерностью 7, с минимальным кодовым расстоянием $d_{\min} = 32$, равнозначный коду Рида–Мюллера первого порядка с перестановкой.

7 битов заголовка включают два поля: MODCOD и TYPE, определенные следующим образом:

- MODCOD (5 битов), определяющие модуляцию XFECFRAME и скорость кодирования FEC;
- TYPE (2 бита), определяющие длину FECFRAME (64 800 битов или 16 200 битов) и присутствие/отсутствие пилотных символов.

Возможны две конфигурации PLFRAME:

- без пилотных символов;
- с пилотными символами.

В последнем случае пилотный блок (PILOT BLOCK) должен состоять из $P = 36$ пилотных символов. Каждый пилотный символ должен представлять собой немодулированный символ, содержащий сигнал $I = (1/\sqrt{2})$, $Q = (1/\sqrt{2})$.

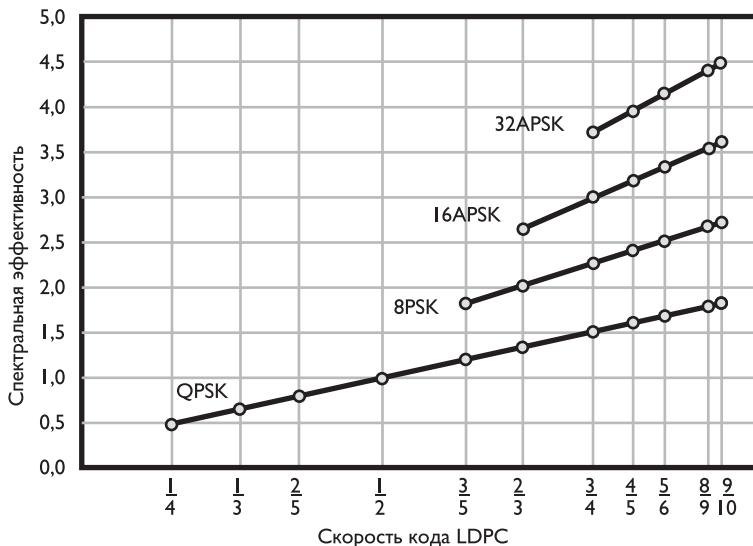


Рис. 21.48. Зависимости скоростей передачи полезных битов R_U от скорости кода LDPC относительно скорости передачи символа R_S

До модуляции каждый физический кадр, исключая заголовок, рандомизируется (см. рис. 21.47) для рассеивания энергии с помощью умножения отсчетов $(I + jQ)$ на комплексную рандомизирующую последовательность $(C_I + jC_Q)$:

$$I_{\text{SCRAMBED}} = [I \cdot C_I - Q \cdot C_Q]; Q_{\text{SCRAMBED}} = (I \cdot C_Q + Q \cdot C_I).$$

На рис. 21.48 приведены графики спектральной эффективности (в пересчете на символьную скорость), рассчитанные для кадров с опережающей коррекцией FECFRAME основного размера при отсутствии пилотных символов. При заданной спектральной эффективности системы η_{tot} отношение энергии на информационный бит к спектральной плотности мощности шума одной боковой полосы равно:

$$E_b/N_0 = E_s/N_0 - 10 \cdot \lg(\eta_{\text{tot}}),$$

где E_s — средняя величина энергии на передаваемый символ.

Система DVB-S2 может использоваться в конфигурациях (FDM) «одна несущая на транспондер» или «несколько несущих на транспондер». В конфигурациях с одной несущей на транспондер скорость передачи символа R_S может быть приведена в соответствие с заданной полосой пропускания транспондера BW (по уровню -3 дБ), для достижения максимальной пропускной способности, согласуемой с приемлемым ухудшением сигнала из-за ограничений, накладываемых на пропускную способность транспондера.

Шаблоны спектра сигнала на выходе модулятора и относительной характеристике групповой задержки аппаратной реализации фильтра Найквиста при $\alpha = 0,35$ соответствуют приведенным на рис. 21.43.

В конфигурации FDM с несколькими несущими, R_S может быть приведена в соответствие с частотным слотом BS , присвоенным данному сервису согласно частотному плану для оптимизации пропускной способности при сохранении взаимной интерференции между соседними несущими на приемлемом уровне.

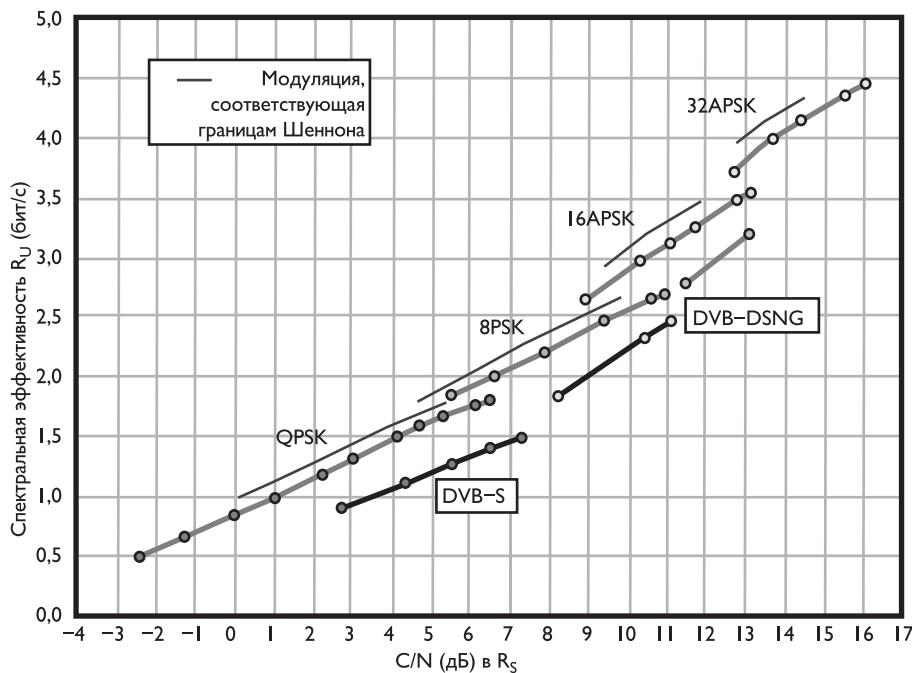


Рис. 21.49. Зависимость C/N от спектральной эффективности, полученная с помощью компьютерного моделирования для канала с гауссовским шумом

Рис. 21.49 иллюстрирует требуемое отношение мощности несущей к шуму C/N в полосе пропускания, соответствующей символьной скорости в зависимости от спектральной эффективности – скорости передачи полезного битового потока R_U в канале с гауссовским шумом при применении идеального демодулятора и отсутствии фазового шума.

Для сравнения показана также производительность систем DVB-S и DVB-DSNG [8.34] при частоте появления ошибок $BER = 10^{-7}$. Использование более крутого спада $\alpha = 0,25$ и $\alpha = 0,20$ дает возможность повышения пропускной способности передачи, но может также создать значительные нелинейные ухудшения со спутников при работе с одной несущей.

Эффективная система адаптивного кодирования и модуляции (ACM) DVB-S2 применяется для приложений класса «точка–точка» (например, IP-вещание в один адрес или DSNG-услуги цифрового спутникового сбора новостей). Суть режима ACM сводится к тому, что в зависимости от приема сигнала (например, наличия дождя), меняется режим работы модулятора DVB-S2, т. е. изменяются скорость кодирования (SR) и формат модуляции, вследствие чего меняется и требуемое отношение C/N у абонента. Порог C/N устанавливается на приемной стороне потребителем данной услуги за счет непрерывного измерения $C/N+I$ (отношение несущая/шум + интерференционная помеха) и посылки измеренного значения на вещательную наземную передающую станцию посредством обратного канала. При этом параметры кодирования и модуляции могут изменяться от кадра к кадру.

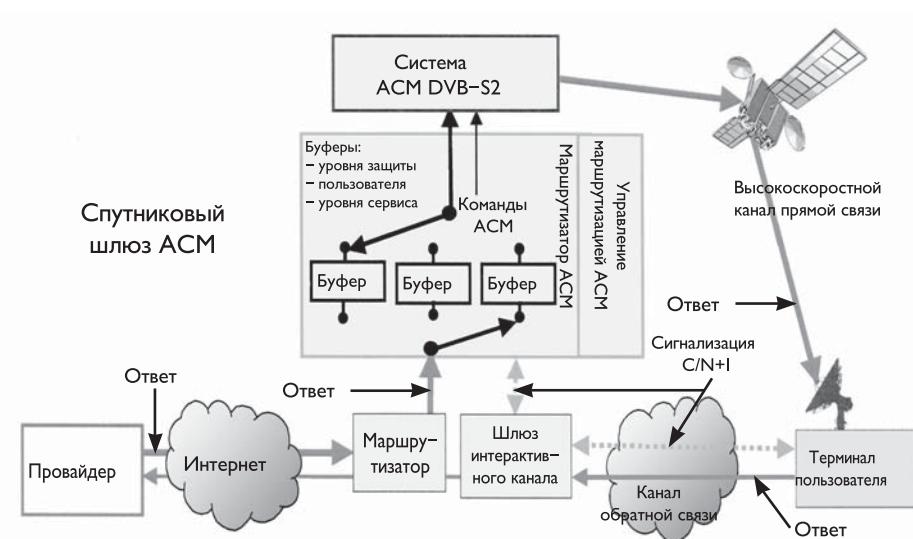


Рис. 21.50. Пример IP-сервисов с использованием канала связи ACM стандарта DVB-S2

На рис. 21.50 приведен возможный обмен информацией (запрос и ответ на запрос) между пользователем, спутниковым шлюзом и провайдером информации во время сеанса работы с Интернетом с помощью спутника. Такие интерактивные сервисы передачи данных могут использовать:

- неравномерную защиту от ошибок (ACM);
 - дифференцированные уровни сервиса (приоритет в очередности доставки).

В соответствии с согласованием между пользовательским терминалом и «управлением маршрутизацией АСМ», «маршрутизатор АСМ» в принципе может разделить IP-пакеты, предназначенные для пользователя, для требуемой защиты от ошибок и для уровня сервиса. Совокупный входной трафик на различных уровнях защиты не должен перегружать имеющуюся пропускную способность канала.

Для того чтобы это правило выполнялось, IP-пакеты низкого приоритета могут быть задержаны (или даже потеряны) в пользу пакетов высокого приоритета, когда общий трафик начинает превышать пропускную способность канала. Если задержки цикла управления (включая управление маршрутизацией и маршрутизатор ACM) слишком велики и не допускают безошибочного приема в условиях быстрого затухания, сервисы реального времени (например, потоковое видео/аудио) могут быть жестко привязаны к ветвям с высокой степенью защиты, тогда как сервисы более низкого приоритета могут работать с большей эффективностью, что обеспечивается ACM. Следует отметить, что стратегия опроса входных буферов может статически или динамически настраиваться в соответствии со статистической оценкой параметров трафика, характеристиками среды распространения и политикой установления оператором сервиса приоритетов трафика.

Маршрутизатор ACM может связываться с модулятором DVB-S2:

- через вход единичного общего потока и вход ACM Command; в данном случае маршрутизатор ACM независим от модулятора DVB-S2 и может реализовывать любую политику маршрутизации; модулятор DVB-S2 незамедлительно передает пользовательские данные в соответствии с командой ACM, следовательно, задержки обратной связи могут быть минимизированы;
- через входы нескольких (транспортных или общих) потоков, по одному на каждый активный уровень защиты (не требуется, чтобы интерфейс команд ACM был активным); в данном случае объединитель/разделитель DVB-S2 частично включает функциональность ACM.

21.6. Система цифрового кабельного телевизионного вещания DVB-C

Система цифрового кабельного телевидения определяется как функциональный блок оборудования, выполняющий адаптацию ТВ-сигналов к характеристикам кабельного канала (см. рис. 21.51). Система DVB-C максимально гармонизирована со спутниковой системой DVB-S и может использовать источник местных ТВ-программ.

В связи с высокой помехозащищенностью кабельных каналов связи в системе DVB-C не используется сверточное кодирование, но применяется многопозиционная QAM-модуляция — от 16-QAM до 256-QAM.

В качестве входных сигналов на головной станции используются транспортные пакеты MPEG-2, поступающие через входной интерфейс на модуль, инвертирующий 1-й синхронизирующий байт кадровой структуры MPEG-2 и рандомизирующий поток поступающих данных. Схема рандомизатора/дерандомизатора аналогична используемой в стандарте DVB-T.

Вслед за процессом рандомизации распределения энергии выполняется систематическое сокращенное кодирование Рида–Соломона каждого рандомизированного транспортного пакета MPEG-2, при $t = 8$, что обеспечивает возможность корректировки 8 ошибочных байтов в каждом транспортном пакете. Данный процесс добавляет 16 байтов четности к транспортному пакету MPEG-2 для получения кодового слова (204, 188).

Затем следует сверточный перемежитель состоящий из $l = 12$ звеньев, циклически присоединенных к входному потоку байтов с помощью коммутатора. Каждое звено представляет собой регистр FIFO размером $M \cdot j$ ячеек ($M = 17 = N/l$, $N = 204$ — длина защищенного от ошибок кадра, $l = 12$ — глубина перемежения, j — индекс звена). Ячейки FIFO содержат 1 байт, а работа коммутаторов входа и выхода синхронизирована. При этом неинвертированные и инвертированные синхронизирующие байты должны быть всегда адресованы в нулевое звено компоновщика, соответствующее нулевой задержке.

После сверточного перемежения производится точное перекодирование байтов в символы. Перекодирование должно быть основано на использовании границ байтов в системе модуляции. Длина символов $m = \log_2 M$, где M — число позиций

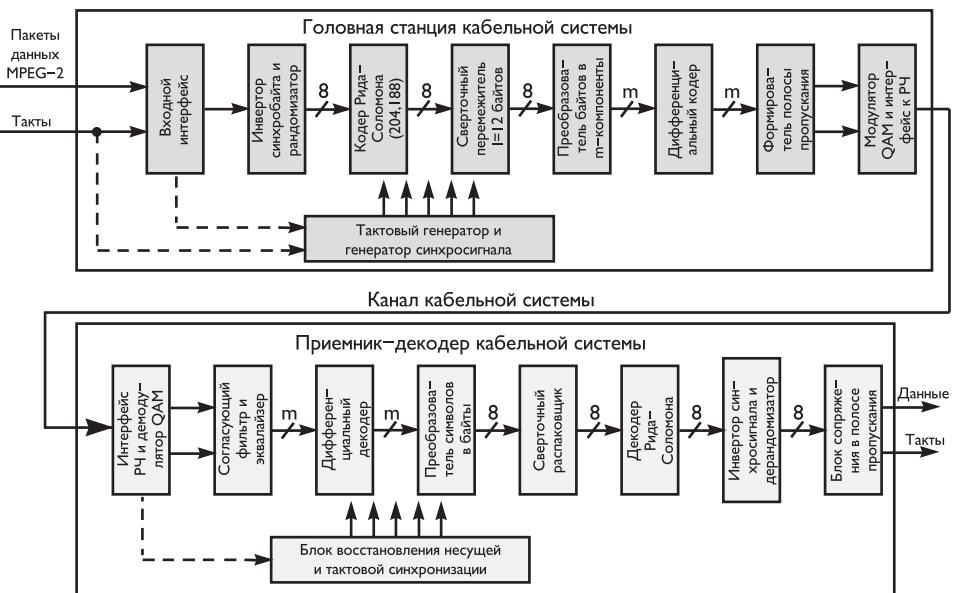


Рис. 21.51. Структурная схема системы цифрового кабельного телевидения DVB-C

QAM-созвездия. Циклическая задача отображения для одного цикла определяется соотношением:

$$8k = n \cdot m, \quad (21.43)$$

где k и n — числа преобразуемых байтов и последовательности двоичных символов, соответственно (см. табл. 21.53).

Таблица 21.53. Коэффициенты преобразования байтов в последовательности символов

Модуляция	m	n	k	$8k$
16-QAM	4	2	1	8
32-QAM	5	8	5	40
64-QAM	6	4	3	24
128-QAM	7	8	7	56
256-QAM	8	1	1	8

Для устранения потерь из-за скачков фазы несущей применяется дифференциальное кодирование двух старших битов (A_k и B_k) последовательности символов. Эти два старших бита каждого символа должны быть дифференциальными закодированы для получения инвариантного относительно фазового сдвига на $\pi/2$ созвездия QAM. Дифференциальное кодирование двух старших битов (MSB) должно осуществляться согласно следующему Булеву выражению:

$$I_k = \overline{(A_k \oplus B_k)} \cdot (A_k \oplus I_{k-1}) \vee (A_k \oplus B_k) \cdot (A_k \oplus Q_{k-1}); \\ Q_k = \overline{(A_k \oplus B_k)} \cdot (A_k \oplus Q_{k-1}) \vee (A_k \oplus B_k) \cdot (A_k \oplus I_{k-1}). \quad (21.44)$$

На рис. 21.52 приведен пример реализации преобразования байтов в символы.

Квадратурные сигналы I и Q с выхода схемы отображения перед модуляцией подвергаются фильтрации с помощью фильтра, характеристика которого соответствует соотношению (21.41) при коэффициенте $\alpha = 0,15$.

В табл. 21.54 приведены примеры расчетных значений символьной и информа-

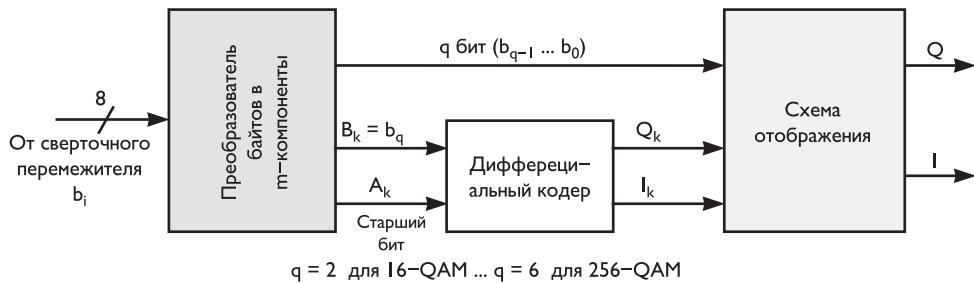


Рис. 21.52. Пример реализации преобразования байта в m -мерный вектор и дифференциального кодирования двух старших битов

ционной скоростей при разных кратностях модуляции в канале с полосой 8 МГц. Максимальная скорость достигает 38,1 Мбит/с, что соответствует пропускной способности ствола спутникового ретранслятора с полосой 33 МГц в типовом режиме $F_{\text{симв}} = 27,5$ Мсимв/с, $R = 3/4$.

Таблица 21.54. Примеры расчетных значений символьной и информационной скоростей при использовании стандарта DVB-C

Полезная информационная скорость (транспортный уровень MPEG-2), Мбит/с	Общая скорость, включая RS (204,188), Мбит/с	Кабельная символьная скорость, Мбод/с	Занимаемая полоса частот, МГц	Вид модуляции
38,1	41,34	6,89	7,92	64-QAM
31,9	34,61	6,92	7,96	32-QAM
25,3	27,34	6,84	7,86	16-QAM
18,9	20,52	3,42	3,93	64-QAM
16,0	17,40	3,48	4,00	32-QAM
12,8	13,92	3,48	4,00	16-QAM
9,6	10,44	1,74	2,00	64-QAM
8,0	8,70	1,74	2,00	32-QAM
6,4	6,96	1,74	2,00	16-QAM

21.7. Система высокоскоростного цифрового кабельного ТВ-вещания DVB-C2

Стандарт кабельного цифрового телевизионного вещания DVB-C2 максимально унифицирован с новыми стандартами, обслуживающими спутниковую (DVB-S2) и эфирную (DVB-T2) транспортные среды [8.38, 8.39].

На рис. 21.53а-г приводится достаточно подробная структурная схема передающей части DVB-C2. Как в DVB-S2 и DVB-T2, в новом кабельном стандарте внутри одного физического канала предусмотрено выделение транспортных PLP

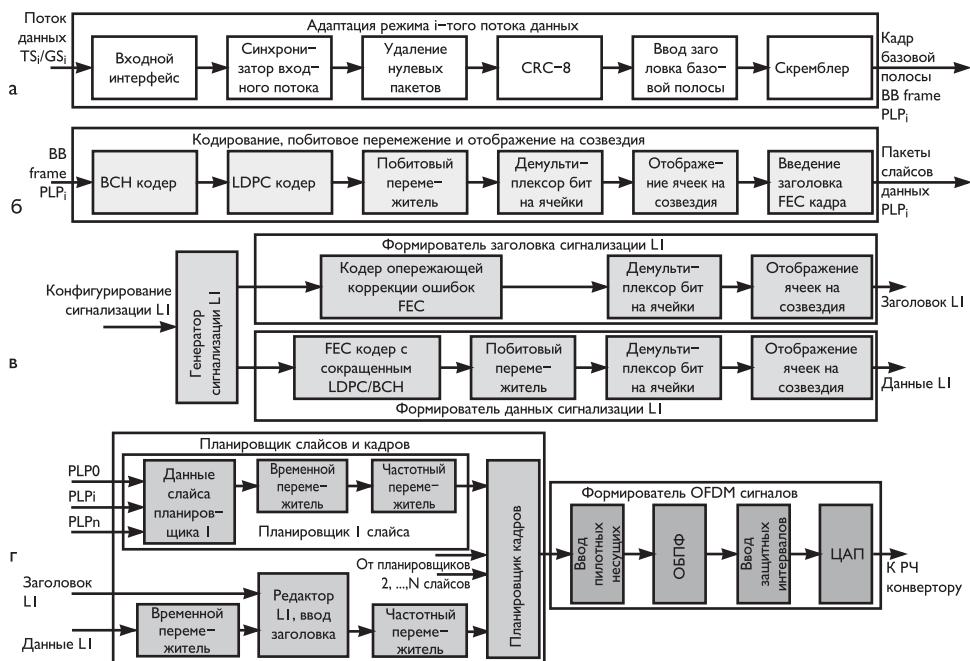


Рис. 21.53. Структурная схема передающей части системы DVB-C2

физических каналов, которые могут обрабатывать и переносить обычный поток MPEG-2 TS или использоваться для передачи IP с применением GSE-протокола.

Вход системы DVB-C2 состоит из одного или из множества логических потоков данных. По одной магистрали физического уровня (PLP) передается один логический поток данных. Модули адаптации режима, по отдельности обрабатывающие содержимое каждой PLP, разбивают входной поток данных на поля данных, которые после адаптации потока должны сформировать кадры базовой полосы (BBFrame). Модуль адаптации режима включает в себя входной интерфейс, за которым следуют три опциональные подсистемы (синхронизатор входного потока, модуль удаления нулевых пакетов и кодер CRC-8), после которых он разбивает входной поток данных на поля данных и выполняет вставку заголовка базовой полосы (BBHeader) в начале каждого поля данных.

Подсистема входного интерфейса обеспечивает преобразование входного сигнала во внутренний логически-битовый формат для каждой единичной магистрали физического уровня (PLP, рис. 21.53а). Длина поля данных DFL на выходе интерфейса должна находиться в пределах:

$$0 < DFL < (K_{bch} - 80),$$

где K_{bch} — количество битов в поле данных, защищенных кодами BCH и LDPC; 10-байтовый (80 битов) заголовок BBHeader присоединяется к началу поля данных и также защищен кодами LDPC и BCH.

Синхронизатор входного потока формирует поле, состоящее из 2 или 3 байтов (поле ISSY — Input Stream Synchronisation), в котором передается значение счетчика, тактируемого с тактовой частотой модулятора ($1/T$, где $T = 7/64$ мкс

или $T = 7/48$ мкс для каналов с полосой пропускания 8 МГц или 6 МГц, соответственно), используемого приемником для восстановления точной синхронизации восстановленного выходного потока. Содержание поля ISSY зависит от формата входного потока и режимов обычной или повышенной эффективности, указанных в заголовке базовой полосы.

Установленные для передачи транспортных потоков требования предусматривают, чтобы скорости битовых потоков на выходе мультиплексора передатчиков и на входе демультиплексора приемников были постоянными на протяжении длительных периодов времени и сквозная задержка также была постоянной. Во входных транспортных потоках может присутствовать большая доля нулевых пакетов для адаптации сервисов с переменной скоростью битового потока в транспортных потоках с постоянной скоростью. В таком случае, во избежание излишних накладных расходов при передаче, нулевые пакеты TS должны быть удалены. Процесс выполняется таким образом, чтобы удаленные нулевые пакеты могли быть повторно вставлены в приемнике в точности на то же самое место, где они находились первоначально.

Кодирование CRC-8, как и в системах DVB-T2 и DVB-S2, применяется для детектирования ошибок на уровне пользовательского пакета, а 10-байтовый заголовок базовой полосы (BBHeader) фиксированного размера вводится перед полем данных для описания формата поля данных.

Перед поступлением на вход системы помехоустойчивого кодирования (см. рис. 21.53б) цифровой поток базовой полосы скремблируется сдвиговым регистром с обратной связью. Порождающий полином последовательности PRBS — $1 + x^{14} + x^{15}$ — с инициируемой в начале каждого кадра BBFrame загрузкой в регистр кода 100101010000000.

Структура формата кадра с опережающей коррекцией ошибок BCH и LDPC соответствует на рис. 21.46б, а параметры такого кадра для основного размера 64 800 битов и сокращенного размера 16 200 битов могут быть извлечены из табл. 21.50 (для LDPC кодов — 2/3, 3/4, 4/5, 5/6 и 9/10) и табл. 21.51 (для LDPC кодов — 1/2, 2/3, 3/4, 4/5, 5/6 и 8/9), соответственно.

Сигнал с выхода кодера LDPC подвергается побитовому перемежению, которое состоит из перемежения проверочных битов, за ним следует перемежение со сдвигом начала столбцов в соответствии с правилом, аналогичным соотношениям (21.16) системы DVB-C2 (информационные биты не перемежаются). Значения параметра Q_{ldpc} определены в табл. 21.55.

Таблица 21.55. Значения Q_{ldpc} для основных и сокращенных кадров

Скорость кода	$N_{ldpc} = 64\,800$	$N_{ldpc} = 16\,200$
1/2	—	25
2/3	60	15
3/4	45	12
4/5	36	10
5/6	30	8
8/9	—	5
9/10	18	—

Конфигурация перемежения со сдвигом начала столбцов для каждого формата модуляции определена в табл. 21.56.

Таблица 21.56. Структура побитового перемежителя

Модуляция	Строки N_r		Столбцы N_c
	$N_{ldpc} = 64\ 800$	$N_{ldpc} = 16\ 200$	
16-QAM	8100	2025	8
64-QAM	5400	1350	12
256-QAM	4050	—	16
	—	2025	8
1024-QAM	3240	810	20
4096-QAM	5400	—	12
	—	675	24

При перемежении со сдвигом начала столбцов биты данных u_i с перемежителем проверочных битов последовательно записываются в перемежитель со сдвигом начала столбцов по столбцам, и последовательно считываются по строкам (старший бит заголовка считывается первым). Запись стартовой позиции каждого столбца сдвигается на величину t_c в соответствии с табл. 21.57.

Таблица 21.57. Параметр сдвига начала столбцов t_c

Модуляция	16-QAM		64-QAM		256-QAM		1024-QAM		4096-QAM	
	N_{ldpc}	64 800	16 200	64 800	16 200	64 800	16 200	64 800	16 200	64 800
Столбцы N_c	8		12		16	8	20		12	24
Сдвиг t_c	0	0	0	0	0	0	0	0	0	0
	1	0	0	0	0	2	0	1	0	0
	2	2	0	2	0	2	0	3	0	2
	3	4	1	2	2	2	1	4	2	2
	4	4	7	3	2	2	7	5	2	3
	5	5	20	4	2	3	20	6	2	4
	6	7	20	4	3	7	20	6	2	4
	7	7	21	5	3	15	21	9	2	5
	8	—	—	5	3	16	—	13	5	1
	9	—	—	7	6	20	—	14	5	7
	10	—	—	8	7	22	—	14	5	8
	11	—	—	9	7	22	—	16	5	9
	12	—	—	—	—	27	—	21	5	—
	13	—	—	—	—	27	—	21	7	—
	14	—	—	—	—	28	—	23	7	—
	15	—	—	—	—	32	—	25	7	—
	16	—	—	—	—	—	—	25	7	—
	17							26	8	—
	18							28	8	—
	19							30	10	—
	20									10
	21									10
	22									10
	23									11

Каждый кадр FECFRAME преобразовывается в кодированный и модулированный FEC блок с опережающей коррекцией ошибок. Для этого входные биты

сначала демультиплексируются на параллельные модулирующие значения ячеек, и затем эти модулирующие значения отображаются на значения созвездия. Количество ячеек выходных данных и эффективное количество битов на ячейку η_{mod} заданы в табл. 21.58.

Таблица 21.58. Параметры побитового отображения на созвездия

Длина блока LDPC-кода (N_{ldpc})	Режим модуляции	η_{mod}	Число выходных ячеек данных
64 800	4096-QAM	12	5400
	1024-QAM	10	6480
	256-QAM	8	8100
	64-QAM	6	10 800
	16-QAM	4	16 200
16 200	4096-QAM	12	1350
	1024-QAM	10	1620
	256-QAM	8	2025
	64-QAM	6	2700
	16-QAM	4	4050
	QPSK	2	8100

Битовый поток v_{di} , от побитового перемежителя демультиплексируется на $N_{substreams}$ подпотоков (табл. 21.59).

Таблица 21.59. Количество подпотоков в демультиплексоре

Модуляция	N_{ldpc}	Количество подпотоков $N_{substreams}$
QPSK	Любое	2
16-QAM	Любое	8
64-QAM	Любое	12
256-QAM	64 800	16
	16 200	8
1024-QAM	Любое	20
4096-QAM	64 800	12
	16 200	24

Демультиплексирование определяется как отображение подвергнутых побитовому перемежению входных битов, v_{di} , на выходные биты $b_{e,do}$ на выходе демультиплексора ($di \bmod N_{substreams}$ — число входных битов; $do = di \bmod N_{substreams}$ — число битов в заданном потоке на выходе демультиплексора; e — количество демультиплексированных битовых потоков, $(0 \leq e < N_{substreams})$, зависящее от di , табл. 21.60).

Каждое модулирующее значение ячеек, поступающее от демультиплексора, модулируется с использованием созвездий одного из типов: QPSK, 16-QAM, 64-QAM, 256-QAM, 1024-QAM или 4096-QAM. Созвездия и детализация применяемого к ним отображения реализуются в соответствии с кодом Грея.

Точки созвездий z_q для каждого входного модулирующего значения ячеек $[y_{0,q} \dots y_{\eta_{mod}-1,q}]$ нормализуются в соответствии с табл. 21.61.

Таблица 21.60. Параметры демультиплексирования битов на подпотоки для всех кодовых скоростей (за исключением 2/3)

QPSK	
Вход	0 1
Выход	0 1
16-QAM	
Вход	0 1 2 3 4 5 6 7
Выход	7 1 4 2 5 3 6 0
64-QAM	
Вход	0 1 2 3 4 5 6 7 8 9 10 11
Выход	11 7 3 10 6 2 9 5 1 8 4 0
256-QAM ($N_{ldpc} = 64\ 800$)	
Вход	0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
Выход	7 2 9 0 4 6 13 3 14 10 15 5 8 12 11 1
256-QAM ($N_{ldpc} = 16\ 200$)	
Вход	0 1 2 3 4 5 6 7
Выход	7 3 1 5 2 6 4 0
1024-QAM ($N_{ldpc} = 64\ 800$)	
Вход	0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
Выход	8 16 7 19 4 15 3 12 0 11 10 9 13 2 14 5 17 6 18 1
1024-QAM ($N_{ldpc} = 16\ 200$)	
Вход	0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
Выход	8 3 7 10 19 4 9 5 17 6 14 11 2 18 16 15 0 1 13 12
4096-QAM ($N_{ldpc} = 64\ 800$)	
Вход	0 1 2 3 4 5 6 7 8 9 10 11
Выход	8 0 6 1 4 5 2 3 7 10 11 9
4096-QAM ($N_{ldpc} = 64\ 800$)	
Вход	0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
Выход	8 0 6 1 4 5 2 3 7 10 11 9 6 17 13 20 1 3 9 2 7 8 12 0

Таблица 21.61. Нормирующие множители для ячеек данных

Модуляция	Нормирующие множители
QPSK	$f_q = z_q / \sqrt{2}$
16-QAM	$f_q = z_q / \sqrt{10}$
64-QAM	$f_q = z_q / \sqrt{42}$
256-QAM	$f_q = z_q / \sqrt{170}$
1024-QAM	$f_q = z_q / \sqrt{682}$
4096-QAM	$f_q = z_q / \sqrt{2730}$

На рис. 21.53в приведена структура датчика сигналов синхронизации кадров DVB-C2, содержащего два канала — формирователей заголовка и данных сигнализации L1.

Кодирование заголовка сигнализации L1 осуществляется первоначально с помощью кодера Рида–Мюллера (32, 16). При этом каждый бит 32-битового кодового слова Рида–Мюллера разбивается на две ветви. Затем данные отображаются на созвездие QPSK для устойчивого к ошибкам заголовка кадров FECFrame, или на созвездие 16-QAM для заголовка кадров FECFrame с повышенной эффективностью.

Данные сигнализации L1 подвергаются сокращенному LDPC/BCH-кодированию с последующим побитовым перемежением, демультиплексированием и 16-QAM модуляцией.

На рис. 21.53 г. изображена структура планировщика слайсов, формирователя кадров и OFDM-сигналов системы DVB-C2.

Один или множество логических каналов PLP организуются в группу PLP, и одна или множество таких групп PLP образуют слайсы данных. Система C2 может состоять из одного или множества слайсов данных. Предполагается, что приемник всегда должен иметь возможность принимать одну PLP-данных и связанную с ней общую PLP при ее наличии.

Для канала с шириной полосы 8 МГц максимальное число OFDM-несущих при передаче каждого слайса должно быть не более $nK_{DC\max} - K_{DC\min} \leq 3408$ при $f_{\max} - f_{\min} = 7,61$ МГц и длительности символа $T_U = 448$ мкс.

Данные слайса от каждого планировщика подвергаются временному и частотному перемежению. Временной перемежитель обычно содержит два банка памяти, в первый из которых осуществляется запись, а из второго производится считывание данных слайса, затем производится переключение режимов работы этих банков. Для реализации частотного перемежения производится изменение порядка считывания символов звездной диаграммы. Данные от планировщиков слайсов поступают на планировщик кадров, на который подаются также специальным образом обработанные данные синхронизации L1.

Структура кадра системы DVB-C2 поясняется рис. 21.54а.

Пreamble кадра включает L_P символов ($L_P \geq 1$), за ней следует L_{Data} символов данных. Пreamble несет информацию о символах блока синхронизации L1 (3408 поднесущих частот в полосе 7,71 МГц). Данные слайсов могут передаваться в произвольной полосе частот, не превышающей полосу частот передачи символов блока L1. Неиспользуемые частоты могут занимать часть всего кадра DVB-C2.

Данные сигнализации L1 циклически повторяются, что обеспечивает возможность восстановить полный L1 блок из частей двух блоков, как показано на рис. 21.54б.

На входе формирователя OFDM-сигналов установлена схема ввода пилотных несущих, в состав которых входят пилотные несущие preamble, постоянные и рассеянные пилотные несущие в составе передаваемых символов данных, а также граничные пилотные несущие, несущие информацию о границах передачи символов данных.

Номера пилотных несущих preamble связаны соотношением:

$$k \bmod D_P = 0, \quad (21.45)$$

где $D_P = 6$, т. е. эти пилотные несущие соответствуют значениям $k = 0, 6, 12, \dots, 3402$. Параметры пилотных несущих определены следующим образом:

$$\operatorname{Re} \{c_{m,l_P,k}^P\} = A_{PP} \cdot 2(1/2 - r_k), \quad \operatorname{Im} \{c_{m,l_P,k}^P\} = 0, \quad (21.46)$$

где $A_{PP} = 1$, m — номер кадра, l_P — номер символа preamble, k — индекс несущей, а r_k определено только для значений k , кратных 6, и вычисляется по формуле:

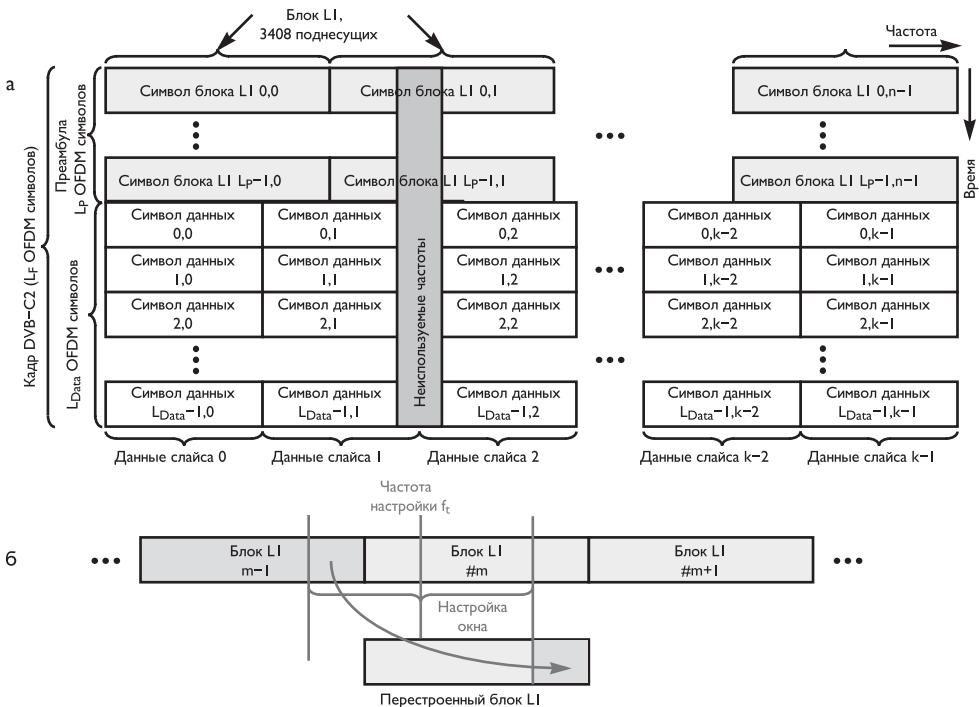


Рис. 21.54. Структура кадра системы DVB-C2

$$r_k = \begin{cases} w_k^P, & k \bmod K_{L1} = 0; \\ r_{k-6} \oplus w_k^P, & \text{иначе;} \end{cases} \quad w_k^P = w_k \oplus w'_i, i = (k \bmod K_{L1})/D_P; \quad (21.47)$$

w'_i — PRBS регистра сдвига, определяемого соотношением $x^{10} + x^3 + 1$, инициируемого последовательностью единиц, т. е. $w'_0, w'_1, w'_2, \dots = 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, \dots$; w_k — PRBS регистра сдвига, определяемого соотношением $x^{11} + x^2 + 1$, инициируемого последовательностью единиц, т. е. $w_0, w_1, w_2, \dots = 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, \dots$.

Номера локальных рассеянных пилотных несущих определяются следующим соотношением:

$$k \bmod (D_X \cdot D_Y) = D_X \cdot (l \bmod D_Y), \quad (21.48)$$

где k — индекс несущей, l — индекс символа данных, D_X, D_Y определены в табл. 21.62.

Таблица 21.62. Коэффициенты, определяющие положение рассеянных пилотных несущих

Коэффициент защитного интервала	Выделение пилотных несущих (D_X)	Номер символов формирования рассеянных несущих (D_Y)
1/64	12	4
1/128	24	4

Кроме рассеянных пилотных несущих в каждый символ кадра, за исключением преамбулы, вводятся постоянные несущие. Индексы этих несущих локально в пределах блока из $K_{L1} = 3408$ приведены в табл. 21.63.

Таблица 21.63. Индексы постоянных пилотных несущих

96	216	306	390	450	486	780	804
924	1026	1224	1422	1554	1620	1680	1902
1956	2016	2142	2220	2310	2424	2466	2736
3048	3126	3156	3228	3294	3366		

В дополнение к рассеянным и постоянным пилотным несущим вводятся также граничные пилотные несущие, определяющие «края» в каждом символе. Такие пилотные несущие вводятся также на границах областей неиспользуемых частот.

Параметры рассеянных, постоянных и граничных пилотных несущих определяются соотношением, аналогичным (21.44), в котором вместо величины A_{PP} должна быть введена амплитуда $A_{SP} = 7/3$.

Некоторые OFDM-ячейки могут быть зарезервированы для подавления пикировых значений формируемого радиосигнала (PAPR).

Резервируемые ячейки OFDM-сигнала S_0 определяются соотношением:

$$[k \bmod (8 \cdot K_{L1})] - D_x(l \bmod D_Y) \in S_0, \quad 0 \leq l \leq L_{Data}, \quad (21.49)$$

где k — абсолютный индекс несущей, L_{Data} — количество символов данных в кадре DVB-C2.

Формируемый на выходе ОБПФ-сигнал может быть записан в следующем виде:

$$s(t) = \operatorname{Re} \left\{ \sum_{m=0}^{\infty} \left[\frac{1}{\sqrt{K_{\text{total}}}} \sum_{l=0}^{L_F-1} \sum_{k=K_{\min}}^{K_{\max}} c_{m,l,k} \cdot \psi_{m,l,k}(t) \right] \right\}, \quad (21.50)$$

где

$$\psi_{m,l,k}(t) = \begin{cases} \exp \left[2\pi j \frac{k}{T_U} (t - T_G - lT_S - mT_F) \right] & \text{при } mT_F + lT_S \leq t \leq \\ & \leq mT_F + (l+1)T_S; \\ 0 & \text{иначе,} \end{cases}$$

k — номер поднесущей; l — номер символа OFDM от «0» до первого символа преамбулы кадра; m — номер кадра DVB-C2; K_{total} — количество используемых несущих, $K_{\text{total}} = K_{\max} - K_{\min}$; L_F — количество OFDM символов в кадре (исключая преамбулу); T_U — длительность активной части символа; T_G — длительность защитного интервала; T_S — полная длительность символа; $c_{m,l,k}$ — комплексное значение k -й несущей в l -м OFDM-символе m -го кадра DVB-C2; T_F — длительность кадра; K_{\min} и K_{\max} — индексы первой нижней и последней верхней активных несущих, соответственно. Основные параметры OFDM-сигналов приведены в табл. 21.64.

Поскольку в системе DVB-C2 применены более эффективные методы помехоустойчивого кодирования по сравнению с системой DVB-C, возможно использование звездных диаграмм OFDM-сигналов более высокой размерности. Допустимые комбинации модуляций и параметров LDPC-кодирования приведены в табл. 21.65, в которой также указаны величины отношения сигнал/шум, требуемые для приема, квазисвободного от ошибок.

Таблица 21.64. Параметры OFDM-сигналов для каналов с полосой пропускания 6 МГц и 8 МГц

Параметр	6 МГц 1/64	6 МГц 1/128	8 МГц 1/64	8 МГц 1/128
Количество OFDM-несущих в блоке сигнализации L1 — K_{L1}	3408	3408	3408	3408
Ширина полосы блока сигнализации L1	5,61 МГц	5,61 МГц	7,61 МГц	7,61 МГц
Длительность T_U в элементарных периодах T	4096T	4096T	4096T	4096T
Длительность T_U в мкс	597,3	597,3	448	448
Частотный интервал между несущими $1/T_U$ в Гц	1674	1674	2232	2232
Длительность T_G в элементарных периодах T	64T	32T	64T	32T
Длительность T_G в мкс	9,33	4,66	7	3,5

Таблица 21.65. Отношение сигнал/шум при различных параметрах системы DVB-C2

Параметры LDPC кода	16-QAM	64-QAM	256-QAM	1024-QAM	4096-QAM
2/3	—	13,5 дБ	—	—	—
3/4	—	—	20,0 дБ	24,8 дБ	—
4/5	10,7 дБ	16,1 дБ	—	—	—
5/6	—	—	22,0 дБ	27,2 дБ	32,4 дБ
9/10	12,8 дБ	18,5 дБ	24,0 дБ	29,5 дБ	35,0 дБ

Для сравнения эффективностей использования систем кабельного цифрового телевизионного вещания DVB-C2 и DVB-C в табл. 21.66 приведены допустимые скорости передачи информации при эквивалентной ширине канала 8 МГц.

Таблица 21.66. Максимальные скорости передачи информации в системах DVB-C и DVB-C2 при эквивалентной ширине канала 8 МГц

Система	16-QAM	64-QAM	256-QAM	1024-QAM	4096-QAM
DVB-C	25 Мбит/с	38,4 Мбит/с	51,2 Мбит/с	—	—
DVB-C2	2/3	—	31,4 Мбит/с	—	—
	3/4	—	—	47,1 Мбит/с	58,9 Мбит/с
	4/5	25,1 Мбит/с	37,7 Мбит/с	—	—
	5/6	—	—	52,4 Мбит/с	65,4 Мбит/с
	9/10	28,3 Мбит/с	41,4 Мбит/с	56,6 Мбит/с	70,7 Мбит/с
					84,8 Мбит/с

В отличие от стандартов эфирного вещания, стандарт DVB-C2 может не подчиняться жесткой частотной сетке, поскольку кабельная сеть является закрытой экранированной средой и нет необходимости координировать использование ее спектра с эфирными присвоениями. Напротив, можно гибко адаптировать полосу канала под свои конкретные потребности, что позволяет расширить полосу

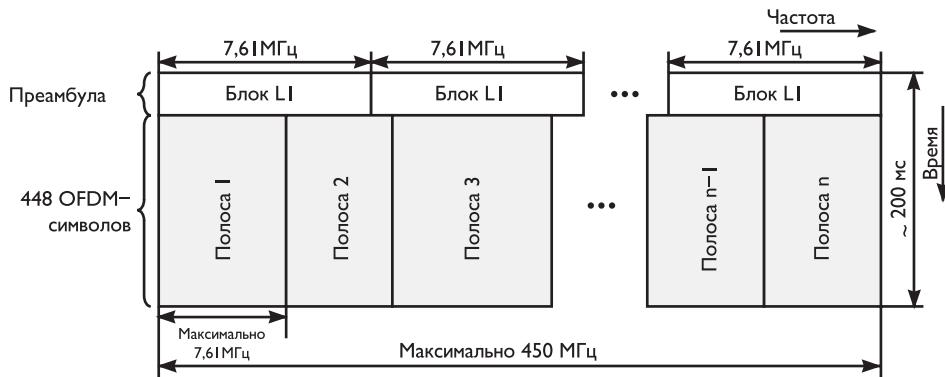


Рис. 21.55. Структура кадра DVB-C2 в частотно-временной области

передаваемого сигнала для размещения в нем большего количества услуг. Чтобы не усложнять и не удорожать абонентское оборудование, реализуется сегментированный прием таких каналов, аналогичный используемому в японской системе эфирного телевидения ISDB-T [8.40]. Приемник со стандартной полосой пропускания извлекает из широкого пакета только необходимую часть спектра, не превышающую, например, 8 МГц.

Структура кадра DVB-C2 показана на рис. 21.55. Каждый кадр C2 начинается с преамбулы, состоящей из одного или более OFDM-символов и выполняющей две основные функции. С одной стороны, она обеспечивает надежную временную и частотную синхронизацию OFDM-сигнала и самой структуры. С другой стороны, преамбула содержит сигнализацию уровня L1, необходимую для декодирования потоков данных и содержащейся в них полезной информации. Преамбула состоит из циклически передаваемых блоков сигнализации L1, повторяющихся в каждой полосе 7,61 МГц широкого канала. Фиксированное расположение блоков L1 и их повторение с шагом 7,61 МГц обеспечивают их прием при настройке тюнера на любые 8 МГц из занимаемого кадром диапазона.

21.8. Система наземного цифрового телевизионного вещания ISDB-T

В системе ISDB-T (Integrated Services Digital Broadcasting-Terrestrial), разработанной в Японии Ассоциацией радиовещательной индустрии и бизнеса — ARIB, используется принцип многочастотной модуляции с передачей цифрового потока на OFDM несущих в нескольких полосах частот (так называемая частотно-сегментированная передача — BST-OFDM) [8.12, 8.40–8.42].

Система ISDB-T может быть использована в каналах ТВ-вещания UHF-диапазона с полосами $B = 6, 7$ или 8 МГц. Оригинальным свойством системы является разделение каждого канала вещания на сегменты, в каждом из которых передаются BST-OFDM-кадры, содержащие заданное число транспортных пакетов стандарта MPEG-2, зависящее от вида модуляции несущих, скорости

сверточного кода и используемых трех режимов работы.

1-й, 2-й и 3-й режимы работы различаются частотным интервалом между соседними несущими и, следовательно, используемым количеством несущих в каждом сегменте. При этом в 1-м режиме используется наибольший интервал между несущими и система обеспечивает передачу наименьшего объема информации, но возможность ее мобильного приема средствами, установленными на движущихся объектах.

Сигналы системы состоят из 13 базовых сегментов OFDM, каждый из которых занимает полосу частот, равную $B/14$ (см. табл. 21.67).

На рис. 21.56 приведены расположения сегментов системы ISDB-T для каналов 6 МГц и возможностей их использования при трансляции либо одной программы ТВ высокой четкости, либо одной ТВ-программы повышенной четкости и одной ТВ-программы стандартной четкости, либо трех ТВ-программ стандартной четкости.

Таблица 21.67. Характеристики спектральных компонент системы ISDB-T для каналов 6, 7, и 8 МГц

Полоса канала	6 МГц	7 МГц	8 МГц
Полоса сегмента	428,571 кГц	500,00 кГц	571,428 кГц
Рабочая полоса частот	5,571 МГц	6,500 МГц	7,428 МГц
Число несущих в сегменте	Режим 1	108	
	Режим 2	216	
	Режим 3	432	
Интервал между несущими	Режим 1	3,968 кГц	1,984 кГц
	Режим 2	4,629 кГц	2,314 кГц
	Режим 3	5,291 кГц	2,645 кГц

В табл. 21.68 приведена информация о числе транспортных пакетов, передаваемых в одном кадре BST-OFDM-сегмента. Структурная схема формирования сигналов системы ISDB-T приведена на рис. 21.57.

Таблица 21.68. Число транспортных пакетов в каждом BST-OFDM кадре

Скорость сверточного кода	Вид модуляции несущих		
	QPSK/DQPSK (режимы 1/2/3)	16-QAM (режимы 1/2/3)	64-QAM (режимы 1/2/3)
1/2	12/24/48	24/48/96	36/72/144
2/3	16/32/64	32/64/128	48/96/192
3/4	18/36/72	36/72/144	54/108/216
5/6	20/40/80	40/80/160	60/120/240
7/8	21/42/84	42/84/168	63/126/252

Особенностью этой системы является три тракта кодирования и модуляции, обеспечивающие обработку данных для каждого из трех иерархических режимов передачи. Транспортные пакеты через мультиплексор и внешний кодер, реализующий укороченный код Рида–Соломона RS (204, 188, $t = 8$), коммутируются на вход того или иного тракта в зависимости от установленного приоритета.

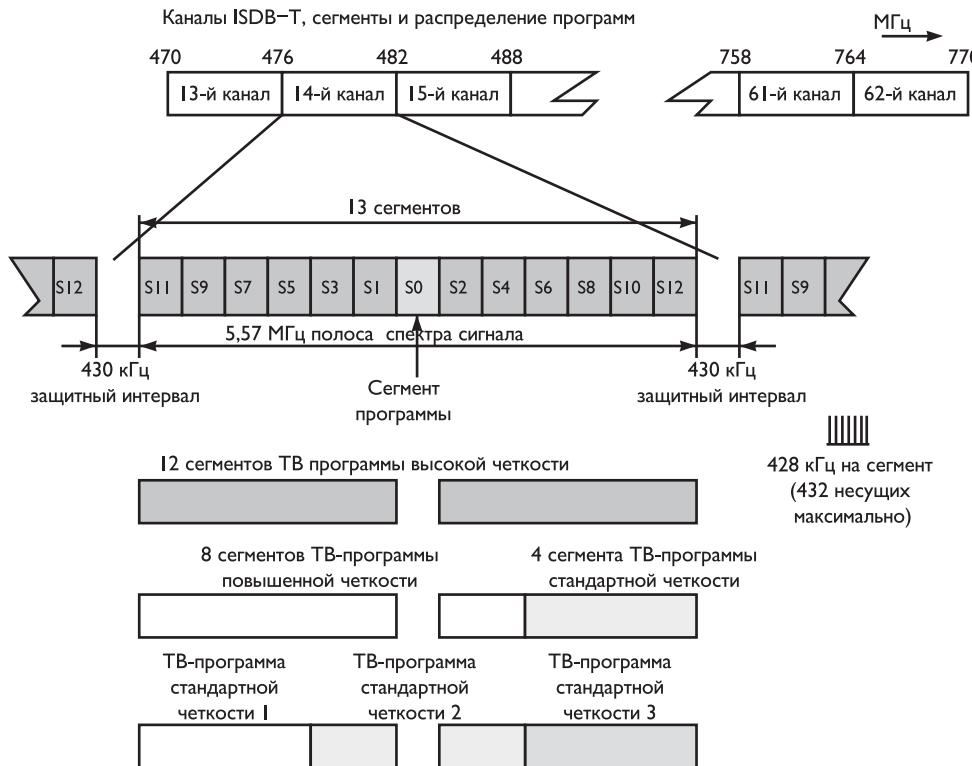


Рис. 21.56. Принцип распределения сегментов системы ISDB-T для каналов 6 МГц

Статистические кодеры, установленные в каждом из этих трактов, обеспечивают рандомизацию данных, реализуя рассеивание энергии, близкое к случайной статистике переходов между битами потоков, аналогичны используемым в стандарте DVB-T и определяемым соотношением (21.3).

Блоки коррекции задержки используются для выравнивания задержек, обусловленных различными процессами обработки сигналов в трактах с разными приоритетами.

Структура байтового перемежителя аналогична приведенному на рис. 21.15 и используемому в системе DVB-T. Порождающий полином материнского сверточного кода аналогичен рис. 21.16, а используемые в системе ISDB-T сверточные коды переменной длины соответствуют табл. 21.5.

Структуры перемежителей битов и мапперов для разных видов модуляции — DQPSK, QPSK, 16-QAM и 64-QAM, — приведены на рис. 21.58.

Специфической особенностью расчета I - и Q -компонент при реализации дифференциальной QPSK-модуляции является использование вычислителя фазы (рис. 21.58а), алгоритм реализации которого определяется соотношением:

$$\Theta_j = \begin{cases} \pi/4, & \text{при } b'_{2j} b'_{2j+1} \Rightarrow 00, \quad j = 0, 1, 2, \dots, \\ -\pi/4, & \text{при } b'_{2j} b'_{2j+1} \Rightarrow 01, \quad j = 0, 1, 2, \dots, \\ 3\pi/4, & \text{при } b'_{2j} b'_{2j+1} \Rightarrow 10, \quad j = 0, 1, 2, \dots, \\ -3\pi/4, & \text{при } b'_{2j} b'_{2j+1} \Rightarrow 11, \quad j = 0, 1, 2, \dots, \end{cases} \quad (21.51)$$

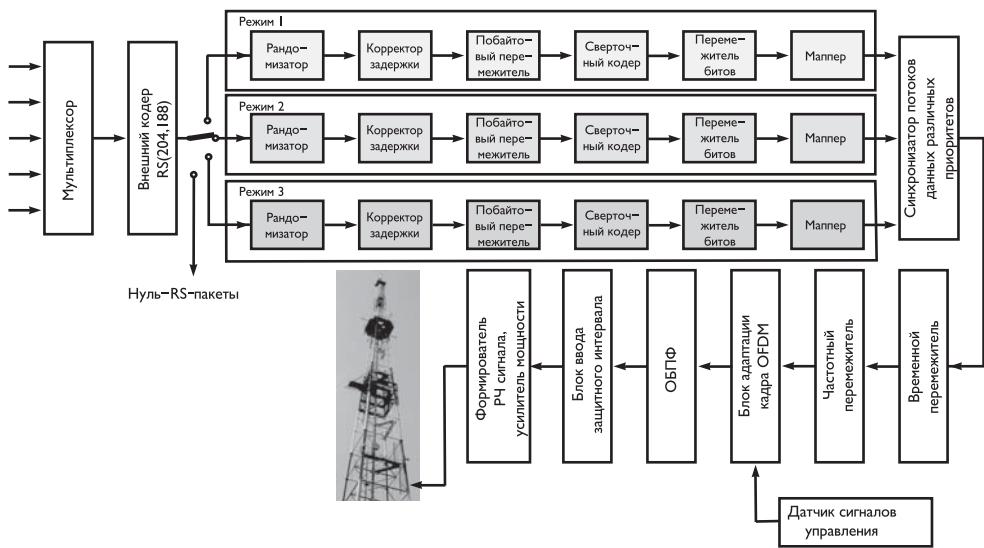


Рис. 21.57. Структурная схема кодирования и модуляции сигналов системы ISDB-T

где b'_{2j} и b'_{2j+1} — четные и нечетные биты, поступающие на два входа вычислителя. I - и Q -компоненты рассчитываются по формуле:

$$\begin{pmatrix} I_j \\ Q_j \end{pmatrix} = \begin{pmatrix} \cos \Theta_j & -\sin \Theta_j \\ \sin \Theta_j & \cos \Theta_j \end{pmatrix} \begin{pmatrix} I_{j-1} \\ Q_{j-1} \end{pmatrix} \quad (21.52)$$

Звездная диаграмма DPSK-модуляции приведена на рис. 21.586.

Структурные схемы вычисления I - и Q -компонентов при QPSK-, 16-QAM- и 64-QAM-модуляции приведены на рис. 21.58в, г, д, соответственно.

Синхронизатор потоков данных различных приоритетов обеспечивает распределение 13 сегментов следующим образом:

- один из сегментов распределяется для передачи информации, рассчитанной на прием с использованием упрощенного узкополосного приемника;
- определенное количество сегментов предназначено для применения дифференциальной модуляции;
- оставшиеся сегменты предназначены для применения когерентной модуляции.

Далее в пределах каждого сегмента производится сначала временное, а затем частотное перемежение данных. Блок адаптации кадра OFDM при воздействии сигналов датчика управления обеспечивает формирование кадра каждого сегмента, состоящего из 204 символов, включающих информативные данные и рассеянные и постоянные пилотные несущие.

Структура кадров OFDM в каждом сегменте содержит 108 несущих в 1-м режиме, 216 несущих во 2-м режиме и 432 несущих в 3-м режиме.

Преобразование сформированной цифровой последовательности данных в эффективную часть символов длительностью T_U осуществляется ОБПФ-преобразователем. Затем к этому сигналу добавляется защитный интервал, относительная

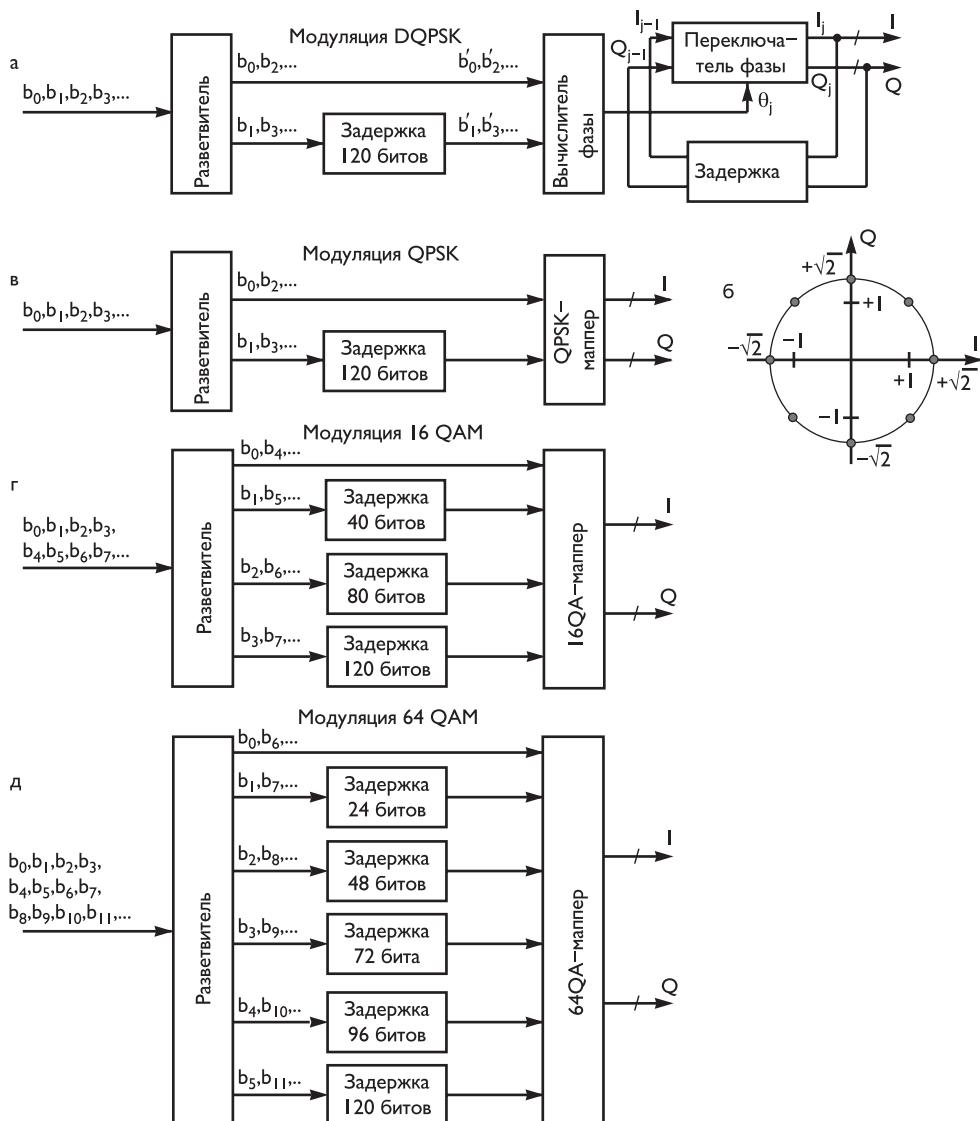


Рис. 21.58. Структурные схемы модуляторов DQPSK, QPSK, 16-QAM и 64-QAM

величина которого T_G/T_U может быть равной $1/4, 1/8, 1/16$ или $1/32$. В табл. 21.69 приведены основные параметры передачи в системе ISDB-T.

21.9. Система цифрового спутникового многопрограммного ТВ-вещания ISDB-S

Система ISDB-S разрабатывалась в Японии Ассоциацией радиовещательной индустрии и бизнеса ARIB как средство доставки различных мультимедийных про-

Таблица 21.69. Основные параметры системы ISDB-T

Параметры		Каналы/Режимы								
		6 МГц			7 МГц			8 МГц		
		1	2	3	1	2	3	1	2	3
Число сегментов		13								
Полоса сегмента		428,571 кГц			500,00 кГц			571,428 кГц		
Рабочая полоса частот		5,571 МГц			6,500 МГц			7,428 МГц		
Число несущих	Общее	1405	2809	5617	1405	2809	5617	1405	2809	5617
	Данных	1248	2496	4992	1248	2496	4992	1248	2496	4992
Эффективная длительность символа T_U , мкс		252	504	1008	216	432	864	189	378	756
Защитный интервал T_G/T_U		1/4, 1/8, 1/16, 1/32								
Модуляция несущих		DQPSK, QPSK, 16-QAM, 64-QAM								
Внутренний код		Сверточный код $R = 1/2, 2/3, 3/4, 5/6, 7/8$								
Внешний код		RS (204, 188, $t = 8$)								
Скорость передачи при 13 сегментах	Минимальная при $T_G/T_U = 1/4$, DQPSK/QPSK, $R = 1/2$	3,651 Мбит/с			4,259 Мбит/с			4,868 Мбит/с		
	Максимальная при $T_G/T_U = 1/32$, 64-QAM, $R = 7/8$	23,234 Мбит/с			27,107 Мбит/с			30,979 Мбит/с		

грамм, содержащих динамические изображения различной четкости, аудиоинформацию, текст, графику, другие виды данных. При этом используется широкий набор методов модуляции и помехоустойчивого кодирования [8.44–8.46].

Структурная схема передающей части системы ISDB-T приведена на рис. 21.59. В составе этой схемы используются три группы формирования сигналов:

- канал формирования основного сигнала, обеспечивающий обработку нескольких (до восьми) транспортных потоков MPEG-2 и передачу, в частности, ТВ-программ различного качества вплоть до телевидения высокой четкости;
- канал формирования сигналов управления конфигурацией передачи и мультиплексирования (TMCC), которые несут информацию о примененном методе модуляции и помехоустойчивого кодирования, идентификации транспортных пакетов и т. д.;
- формирователь сигнала вспышки, обеспечивающего стабильное восстановление несущей в приемнике.

Транспортные потоки MPEG-2 после кодирования Рида–Соломона RS (204, 108, $t = 8$) поступают на формирователь транспортного кадра данных, в котором транспортные пакеты длиной 204 байта структурируются в так называемые слоты (рис. 21.60а). Количество слотов в каждом кадре равно 48, каждые 8 кадров структурируются в суперкадры, как показано на рис. 21.60б.

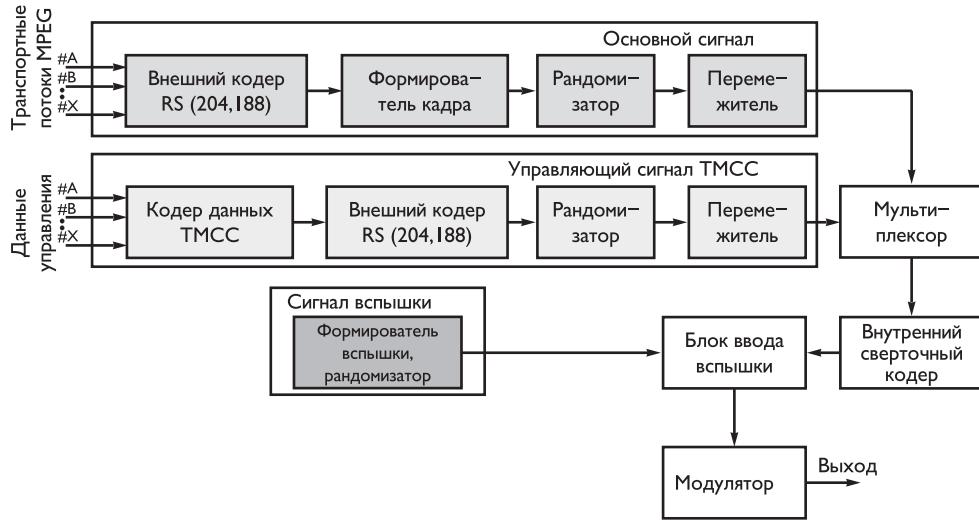


Рис. 21.59. Структурная схема передающей части системы ISDB-S

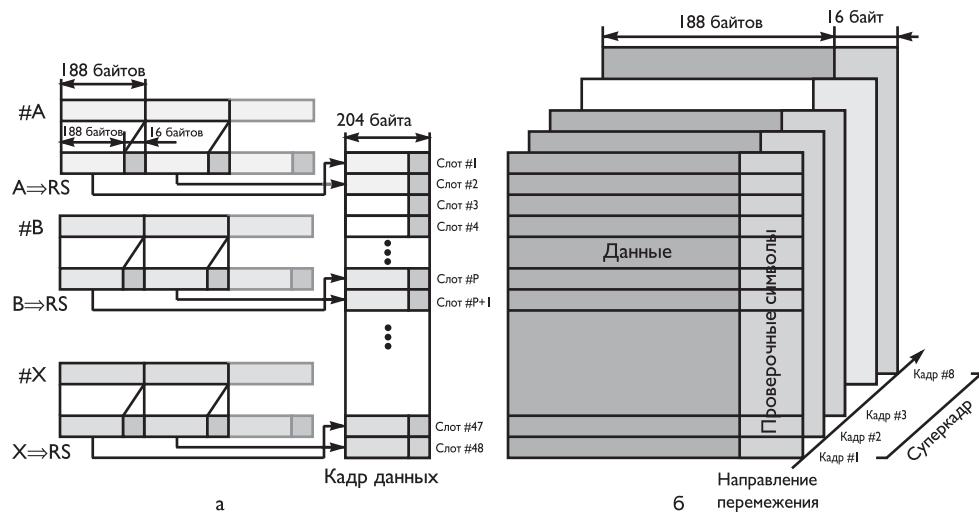


Рис. 21.60. Структуры кадра (а) и суперкадра (б) данных в системе ISDB-S

Вслед за формирователем кадров установлен рандомайзатор, обеспечивающий рассредоточение энергии цифрового сигнала. Порождающий генераторный полином для используемой псевдослучайной двоичной последовательности аналогичен используемому в системе DVB-T и соответствует соотношению (21.3). На рис. 21.60б показано направление перемежения 203 байтов каждого слота последовательности кадров, за исключением первого синхронизирующего байта потока MPEG-2, код которого равен 47_{HEX}.

При перемежении байтычитываются в межкадровом направлении в порядке — $A_{1,1}, A_{2,1}, A_{3,1}, \dots, A_{m,n}$, где m — номер кадра, а n — номер слота в кадре. Глубина перемежения равна количеству кадров в суперкадре — 8. После перемежения байтычитываются в том же порядке, но последовательно по горизонтали в слотах последовательно передаваемых кадров. После перемежения первый синхронизирующий байт транспортных пакетов MPEG-2 исключается, а все временные метки, обеспечивающие на приемной стороне синхронизацию слотов, кадров и суперкадров, передаются в составе управляющего сигнала TMCC, минимально необходимый интервал обновления которого составляет длительность одного каждого суперкадра.

Исходно сигнал TMCC имеет фиксированную длину в 384 бита (48 байтов), в его составе передаются следующие данные:

- обновление передаваемой в TMCC информации — 5 битов;
- тип используемого вида модуляции в каждом слоте — 4 бита;
- идентификации передаваемого в каждом слоте MPEG-2 потока — 16 битов;
- флаг экстренного оповещения — 1 бит;
- сведения о канале земля-борт — 4 бита и др.

Внешнее кодирование сигналов TMCC осуществляется сокращенным кодом Рида-Соломона RS (64, 48, $t = 8$), длина кодовой последовательности становится равной 64 байтам. Эта последовательность разбивается по числу кадров в суперкадре на 8 блоков (рис. 21.61а) и к каждому блоку добавляется по два синхрослова длиной 2 байта:

- синхрослово W_1 (1B95_{HEX}) вводится перед началом каждого блока;
- синхрослово W_2 (A340_{HEX}) вводится после блока, передаваемого в первом кадре;
- синхрослово W_3 (5CBF_{HEX}), инверсное W_2 , вводится в конце всех остальных блоков.

После формирования блоков сигналы TMCC подвергаются рандомизации (исключая синхрослова) и перемежению, если такой вид обработки используется, а затем мультиплексируются с основным сигналом. Полученный таким образом сигнал подвергается сверточному кодированию, причем при обработке сигнала TMCC используется скорость сверточного кода $R = 1/2$, а при обработке основного сигнала возможны варианты $R = 1/2, 2/3, 3/4, 5/6$ и $7/8$, используемые для разных сигнально-кодовых конструкций.

После каждого 203 байтов основного сигнала вводится сформированный и рандомизированный сигнал вспышки, который обеспечивает восстановление несущей в приемнике. Длительность сигнала вспышки равна 4 байтам (см. рис. 21.61а). Принцип модуляции вспышки такой же, как и сигнала TMCC. В схеме восстановления несущей в приемнике для устранения дефектов фазовой подстройки частоты кроме вспышки также используются сигналы TMCC.

В системе используется 7 видов схемы модуляции несущей:

- BPSK при скорости сверточного кода $R = 1/2$;
- QPSK при 5 разных скоростях сверточного кода $R = 1/2, 2/3, 3/4, 5/6$ и $7/8$;
- TS8PSK при скорости сверточного кода $R = 2/3$.

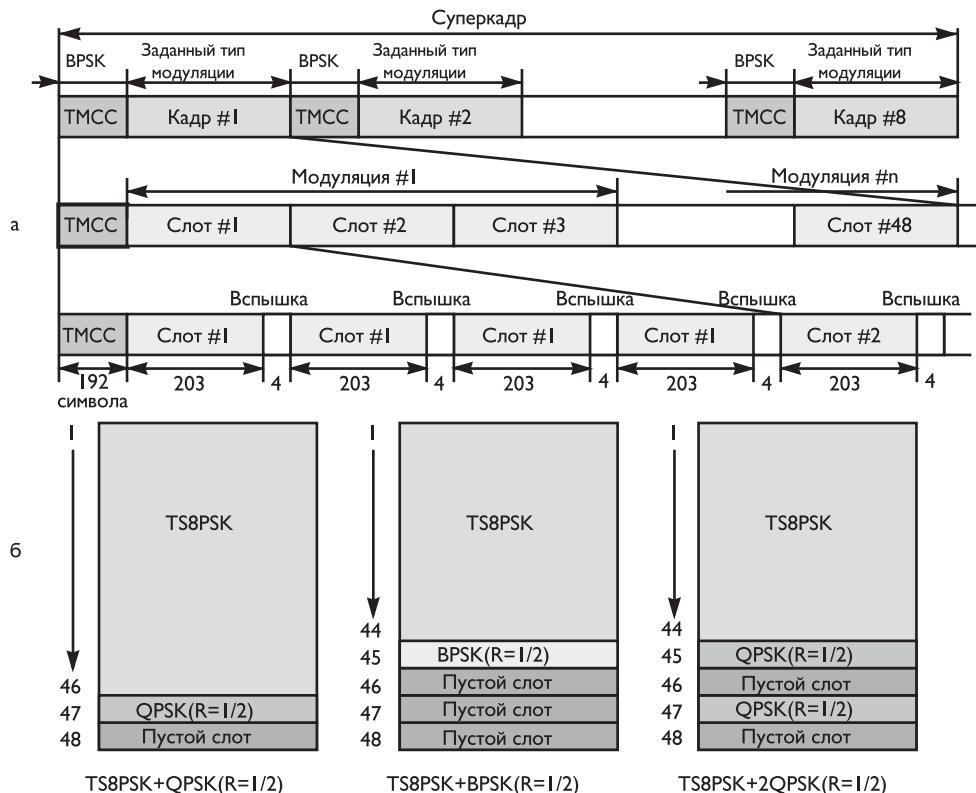


Рис. 21.61. Форма передаваемого сигнала (а) и примеры типов слотов в системе ISDB-S (6)

Прежде чем сигналы I и Q (математически представленные последовательностью дельта функций Дирака, расположенных на расстоянии длительности символа, с соответствующим знаком) будут промодулированы, они должны быть обработаны фильтром, аналогичным применяемому в системе DVB-S с характеристикой (21.41), имеющей фактор спада $\alpha = 0,35$. Примеры присваивания слотов в каждом кадре системы ISDB приведены на рис. 21.61б.

Так как спектральная эффективность зависит от используемой комбинации вида модуляции и скорости сверточного кодирования, то объем передаваемой информации определяется этой комбинацией. Отношение между числом передаваемых пакетов и числом модулирующих символов определяется выражением:

$$I_k = \frac{8 \cdot B \cdot P_k}{E_k}, \quad (21.53)$$

где k — номер схемы модуляции; I_k , P_k — целые числа переданных символов и пакетов, соответственно; $B = 204$ — число байтов в пакете; E_k — спектральная эффективность. При этом число символов в кадре данных I_D равно сумме частных значений величин

$$I_D = \sum_k I_k.$$

Общее число символов в кадре данных системы ISDB-S равно 38 936, а число символов, включая сигналы ТМСС и вспышки, определяется числом 39 936.

В табл. 21.70 иллюстрируются зависимости эффективности и помехоустойчивости системы ISDB-S от используемых видов модуляции.

Таблица 21.70. Спектральная эффективность и помехоустойчивость системы ISDB-S

Вид модуляции	Скорость сверточного кода	Спектральная эффективность, бит/(с·Гц)	Отношение С/Ш при квазибезошибочном приеме, дБ
BPSK	1/2	0,35	1,0
QPSK	1/2	0,7	4,1
	2/3	0,94	6,0
	3/4	1,06	7,2
	5/6	1,18	8,3
	7/8	1,24	9,2
TC8PSK	2/3	1,4	10,7

В табл. 21.71 приведены зависимости скорости передачи полезной информации от полосы пропускания спутникового канала связи и используемых методов модуляции.

Таблица 21.71. Скорости передачи полезной информации

Полоса канала, МГц	Скорость передачи символов, Мсимв/с	Скорость передачи полезной нагрузки MPEG потока, Мбит/с						
		BPSK, $R = 1/2$	QPSK					TC8PSK, $R = 2/3$
			$R = 1/2$	$R = 2/3$	$R = 3/4$	$R = 5/6$	$R = 7/8$	
36	30,1	13,6	27,2	36,3	40,8	45,4	47,6	54,4
34,5	28,9	13,0	26,1	34,8	39,1	43,5	45,6	52,2
33	27,6	12,5	25,0	33,3	37,4	41,6	43,7	49,9
30	25,1	11,6	23,1	30,1	34,7	38,5	40,5	46,3
27	22,6	10,2	20,4	27,2	30,6	34,0	35,7	40,8
24	20,1	9,1	18,1	24,2	27,2	30,2	31,8	36,3

21.10. Система цифрового кабельного многопрограммного ТВ-вещания ISDB-C

Система ISDB-C обеспечивает реализацию функций специального оборудования, выполняющего адаптацию параметров ТВ-сигнала к характеристикам кабельного канала связи. ТВ-сигналы могут поступать на вход оборудования по каналам радиовещания, сети распределения программ, от местных студий и пр. Процесс обработки информации поясняется структурной схемой, приведенной на



Рис. 21.62. Конфигурация системы кабельного ТВ вещания ISDB-C

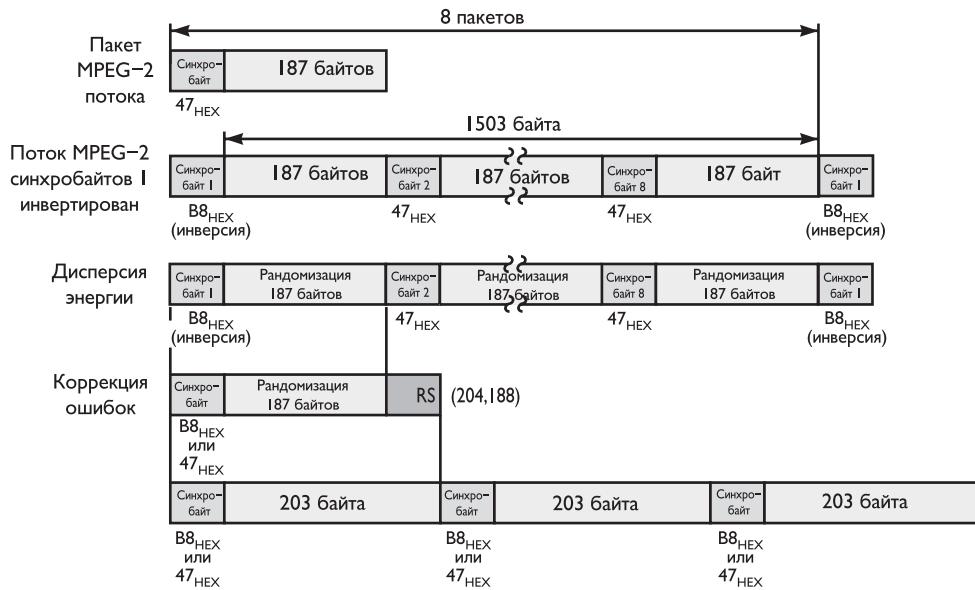


Рис. 21.63. Конфигурация передаваемых сигналов

рис. 21.62 [8.47, 8.48]. На вход системы поступает транспортный поток MPEG-2, каждый пакет которого включает байт синхронизации (47_{HEX}) и 187 байт информационных данных (рис. 21.63).

Во входном блоке головной станции кабельной системы один из восьми байтов синхронизации инвертируется ($B8_{HEX}$), а передаваемые данные подвергаются рандомизации с применением схемы рандомизатора/дерандомизатора, аналогичной используемой в стандарте DVB-C.

Вслед за процессом рандомизации распределения энергии выполняется систематическое сокращенное кодирование Рида–Соломона ($204, 188, t = 8$). Без учета

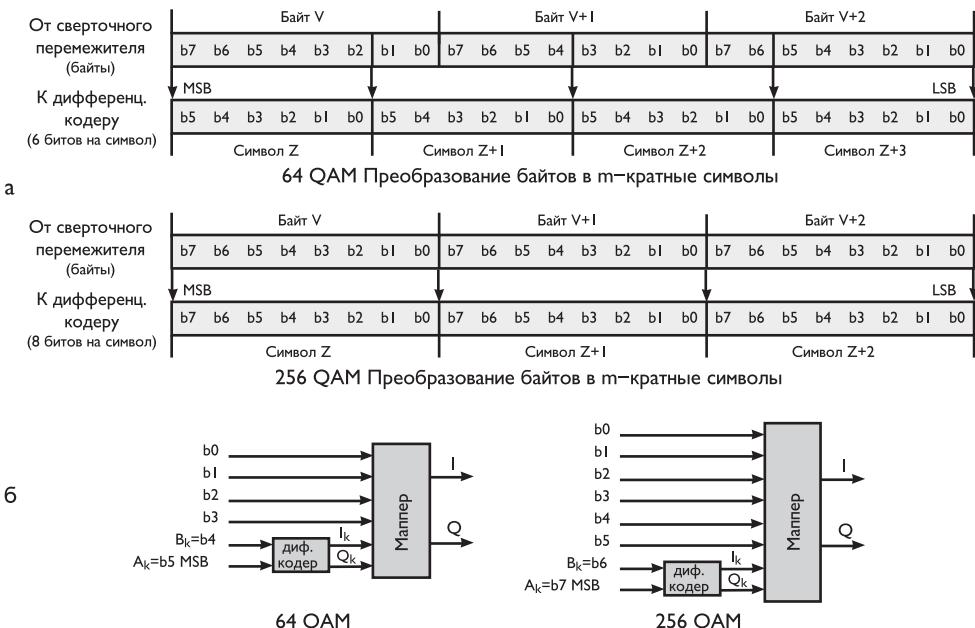


Рис. 21.64. Преобразование байтов в символы (а) и структуры преобразования символов с использованием дифференциального кодирования двух старших битов информации для созвездий 16-QAM и 256-QAM (б)

синхробайтов цифровой поток содержит 203 байта информации, 16 байтов из которых обеспечивают коррекцию ошибок.

Затем следует сверточный перемежитель, состоящий из $l = 12$ звеньев, циклически присоединенных к входному потоку байтов с помощью коммутатора. Каждое звено представляет собой регистр FIFO размером $M \times j$ ячеек ($M = 17 = N/l$, $N = 204$ — длина защищенного от ошибок кадра, $l = 12$ — глубина перемежения, j — индекс звена).

После сверточного перемежения производится точное перекодирование байтов в символы. Перекодирование должно быть основано на использовании границ байтов в системе модуляции, как показано на рис. 21.64а для двух используемых в системе вариантов созвездий — 16-QAM и 256-QAM. Длина символов $m = \log_2 M$, где M — число позиций QAM-созвездия.

Для устранения потерь из-за скачков фазы несущей так же, как и в системе DVB-C, применяется дифференциальное кодирование двух старших битов (A_k и B_k) последовательности символов, определяемых соотношением (21.43). Примеры структуры преобразования символов с использованием дифференциального кодирования двух старших битов информации для созвездий 16-QAM и 256-QAM приведены на рис. 21.64б.

До подачи на QAM-модулятор сформированные таким образом символы подвергаются стандартной фильтрации с использованием ФНЧ, склон характеристики которого соответствует соотношению (21.41) при коэффициенте $\alpha = 0,13$. Допустимые изменения АЧХ такого фильтра приведены на рис. 21.65.

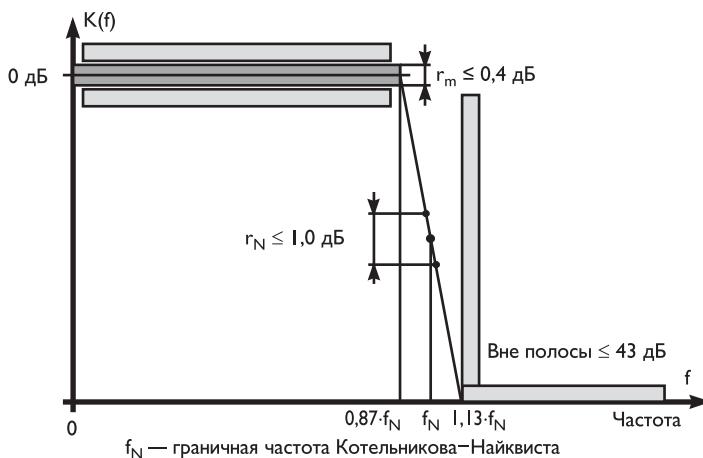


Рис. 21.65. Форма амплитудно-частотной характеристики фильтра

Сводка основных алгоритмов обработки сигналов в системе ISDB-C приведена в табл. 21.72. В приемнике-декодере производятся операции, обратные реализуемым в головной станции кабельной системы.

Таблица 21.72. Параметры сигналов и методы их обработки в системе ISDB-C

Входной сигнал		Транспортный поток MPEG-2
Кадровая структура		Организация на базе транспортных пакетов MPEG-2
Канальное кодирование	Рандомизация	Генераторный полином $1 + x^{14} + x^{15}$ при загрузке «100101010000000» в регистры кодера
	Защита от помех	Кодер Рида–Соломона RS (204, 188, $t = 8$)
	Сверточное перемежение	Сверточный перемежитель из $l = 12$ звеньев
Модуляция	Преобразование байтов в символы	3 байта в 4 символа для 16-QAM 1 байт в 1 символ для 256-QAM
	Дифференциальное кодирование	Используются два старших бита каждого символа
	Вид звездной диаграммы	16-QAM и 256-QAM
	Ширина полосы канала	6 МГц
	Коэффициент склона ФНЧ	$\alpha = 0,13$

21.11. Система цифрового наземного телевизионного вещания DTMB

Стандарт цифрового вещания DTMB (GB 20600–2006 [8.49]) разработан в китайских университетах Jiaotong в Шанхае и Tsinghua в Пекине. Иногда этот стандарт

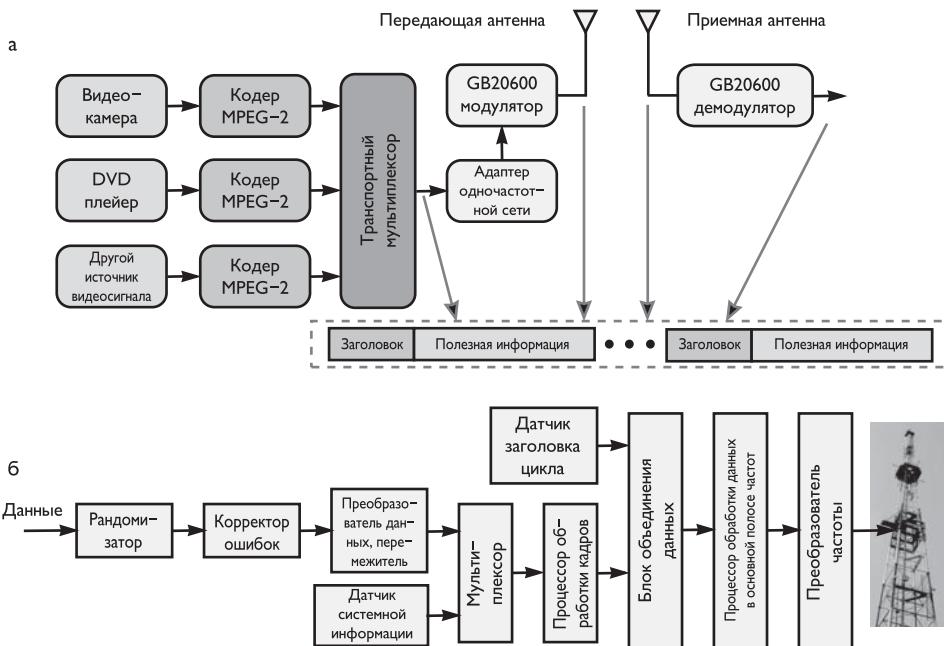


Рис. 21.66. Структурная схема передачи и приема видеонформации в системе DTMB (а) и блок-схема ее передающей части (б)

сокращенно обозначают как DMB-T/H, поскольку он обеспечивает возможность эфирной передачи мультимедийных данных и воспроизведения ТВ-сигналов на малоразмерных ручных приемниках.

Структура системы DTMB и блок-схема ее передающей части приведены на рис. 21.66 [8.50–8.55].

Входной транспортный поток данных, сформированный на выходе транспортного мультиплексора (рис. 21.66а) поступает на вход рандомизатора (рис. 21.66б), выполненного в виде 15 разрядного регистра сдвига. Порождающий генераторный полином для псевдослучайной двоичной последовательности имеет вид, определенный соотношением (21.3). Загрузка последовательности «1001010100000000» в регистры статистического кодера может инициироваться при запуске от 2 до 12 транспортных пакетов.

Предварительная коррекция ошибок осуществляется с использованием внешнего кодера BCN, формирующего укороченный код из кода BCN (1023, 1013), и внутреннего кодера LDPC с тремя возможными скоростями: 2/5 — LDPC (7488, 3048); 3/5 — LDPC (7488, 4572); 4/5 — LDPC (7488, 6096).

Преобразователь данных и перемежитель обеспечивают формирование символной последовательности и временное перемежение блоков данных.

Стандарт предусматривает пять вариантов звездных диаграмм — 64-QAM, 32-QAM, 16-QAM, 4-QAM и 4-QAM-NR. Важно отметить, что в процессе модуляции 4-QAM-NR при формировании сигнала добавляется квазиортогональный код Нордстрема–Робинсона [8.56] перед использованием 4-QAM-отображения.

Перемежитель блоков данных содержит коммутатор на $B = 52$ положения, в каждом из которых осуществляется задержка блока длительностью M битов на

$M \cdot j$ интервалов, где $j = 0, 1, \dots, (B-1)$. Стандарт предусматривает возможность использования $M = 240$ или 720 битов с предпочтением $M = 720$.

В зависимости от принципов обработки сигналов с использованием обратного дискретного преобразования Фурье в системе возможна реализация двух режимов передачи — с множеством несущих ($C = 3780$), на каждой из которых передается один символ с использованием OFDM-модуляции, и с одной несущей ($C = 1$), обеспечивающей передачу 3780 точек ОБПФ при optionalном вводе пилотной несущей (на уровне — 16 dB и смещением по частоте на 1/2 передаваемого символа).

В режиме с множеством несущих в качестве базисного элемента используется сигнальный кадр фиксированной длительности в 500 мкс, он состоит из 3780 символов, 3744 из которых используются для передачи полезной информации, а 36 символами передается системная информация, формируемая специальным датчиком (см. рис. 21.66).

Для устранения влияния интерференционных помех сигнальный кадр дополняется защитным интервалом, длительность которого составляет $1/9$ (55,6 мкс), $1/6$ (78,7 мкс) или $1/4$ (125 мкс) часть активной длительности символа.

В системе используется технология модуляции цифровых сигналов ортогональным частотным и синхронным разделением по времени TDS-OFDM. Для разделения двух соседних кадров структуры заголовок цикла данных содержит псевдослучайную последовательность PN, используемую вместо защитного интервала. При этом применяется три варианта последовательности — PN420, PN595 или PN945 (цифры указывают количество используемых символов, а интервалы их передачи соответственно равны 55,6, 78,7 и 125 мкс). Применение PN-последовательностей в TDS-OFDM заменяет использование непрерывных и рассеянных пилотных несущих. Они используются на приемной стороне в качестве проверочных, повышающих эффективность оценки состояния канала, коррекции его параметров и ускоряющих синхронизацию работы приемника. PN-последовательность может быть либо фиксированной, либо изменяющейся, при которой передается уникальный адрес сигнального кадра в пределах структуры суперкадра.

На рис. 21.67 приведена иерархическая структура кадров в системе DTMB, из которой видно, что в составе суперкадра длительностью 125 мс обеспечивается передача 225, 216 или 200 кадров в зависимости от используемой длительности PN-последовательности.

В процессе формирования OFDM-сигнала составляющие I и Q фильтруются ФНЧ, характеристика которого соответствует соотношению (21.43) при факторе спада $\alpha = 0,05$. Основные параметры обработки сигналов в системе приведены в табл. 21.73.

В табл. 21.74 приведены значения пропускной способности системы при различных методах модуляции, скорости канального кодирования и величин PN-последовательностей, в табл. 21.75 — пороговые отношения мощности несущей к шуму для различных каналов связи.

Система DTMB также обеспечивает реализацию мобильного сервиса, при котором используется протокол многоканального вещания (МРЕ). При этом обеспечивается передача от 20 до 30 телевизионных программ. В отличие от стандартной передачи нескольких программ, как показано на рис. 21.68а, в базовой

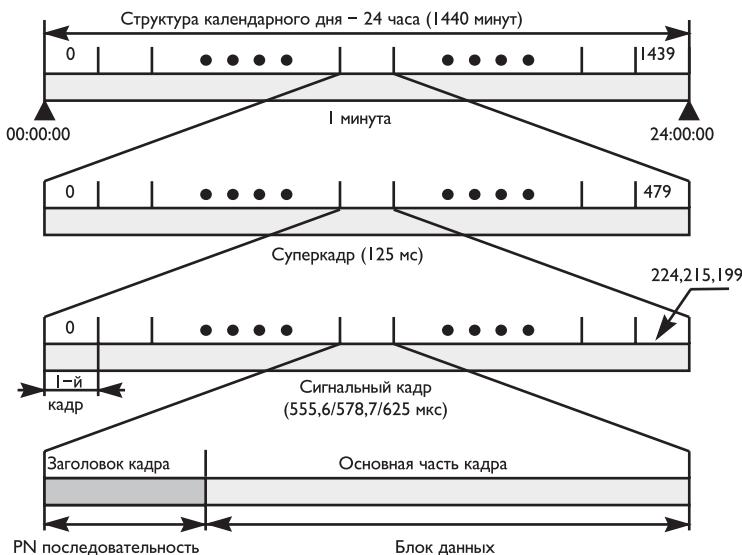


Рис. 21.67. Иерархическая структура кадров системы DTMB

конструкции МРЕ предусмотрена трансляция каждой программы с разбиением на временные слайсы (рис. 21.68б).

Таблица 21.73. Основные параметры системы DTMB

Наименование параметров	Значение параметров
Номинальная полоса канала связи	8 МГц
Используемая полоса частот	7,56 МГц (TDS-OFDM)
Число несущих	$C = 1$ или $C = 3780$
Вид используемого созвездия	64-QAM, 32-QAM, 16-QAM, 4-QAM, 4-QAM-NR
Активная часть длительности символа	500 мкс
Разнос несущих при $C = 3780$	2000 Гц (режим 4К)
Длительность защитного интервала	1/9 (55,6 мкс), 1/6 (78,7 мкс), 1/4 (125 мкс)
Общая длительность символа	555,6; 578,7; 625 мкс
Внутреннее канальное кодирование	LDPC
Скорости внутреннего кодирования	2/5 (7488, 3048), 3/5 (7488, 4572), 4/5 (7488, 6096)
Внешнее канальное кодирование	BCH (762, 752)
Параметры внешнего перемежения	$B = 52$; $M = 240$ или 720
Рандомизация данных	Порождающий генераторный полином: $1 + x^{14} + x^{15}$
Синхронизация по времени и частоте	PN420, PN595, PN945
Сигнализация о параметрах передачи (TPS)	32 несущих TPS, модуляция 4-QAM
Полезная скорость передачи	от 4,81 до 32,49 Мбит/с

Таблица 21.73 (окончание)

Наименование параметров	Значение параметров
Отношение мощности несущей к шуму в канале с аддитивным белым гауссовским шумом	от 2,5 до 22,0 дБ
Коэффициент скругления характеристики фильтра	$\alpha = 0,05$

Таблица 21.74. Пропускная способность системы DTMB, Мбит/с

Скорость кода	2/5			3/5			4/5		
	PN420	PN595	PN945	PN420	PN595	PN945	PN420	PN595	PN945
4-QAM-NR							5,414	5,198	4,813
4-QAM	5,414	5,198	4,813	8,122	7,797	7,219	10,829	10,396	9,626
16-QAM	10,829	10,396	9,626	16,243	15,593	14,438	21,658	20,791	19,251
32-QAM							27,072	25,989	24,064
64-QAM	16,243	15,593	14,438	24,365	23,390	21,568	32,486	31,187	28,877

Таблица 21.75. Пороговое отношение сигнал/шум, дБ

Скорость кода	2/5			3/5			4/5			
	Тип модуляции	Канал Гаусса	Канал Райса	Канал Релея	Канал Гаусса	Канал Райса	Канал Релея	Канал Гаусса	Канал Райса	Канал Релея
4-QAM-NR								2,5	3,5	4,5
4-QAM	2,5	3,5	4,5	4,5	5,0	7,0	7,0	8,0	12,0	
16-QAM	8,0	9,0	10,0	11,0	12,0	14,0	14,0	15,0	18,0	
32-QAM								16,0	17,0	21,0
64-QAM	14,0	15,0	16,0	17,0	18,0	20,0	22,0	23,0	28,0	

В табл. 21.76 даны величины порогового отношения мощности несущей к шуму при мобильном приеме (f_d — доплеровский сдвиг частоты сигнала).

21.12. Эффективность использования систем цифрового ТВ-вещания в России

Применение и развитие современных систем цифрового телевизионного вещания связано с усовершенствованием и созданием новых видеоинформационных систем, реализацией эффективных методов сжатия видеоинформации и ее передачи в реальных сетях связи и вещания.

Наиболее эффективным стандартом сжатия видеоинформации в настоящее время является H.264/AVC (MPEG-4 Part 10). Несмотря на то что он существенно превосходит по эффективности кодирования иные стандарты, мировое сообщество продолжает поиск более совершенных методов обработки. Это обусловлено процессом развития современных видеоинформационных систем и возросшей потребностью в новых, более эффективных методах сжатия для систем с одной (ТВ, ТВЧ, ТУВЧ) или несколькими (3D-TV, Multiview, FTV) точками наблюдения, а также возросшими требованиями к помехоустойчивости.

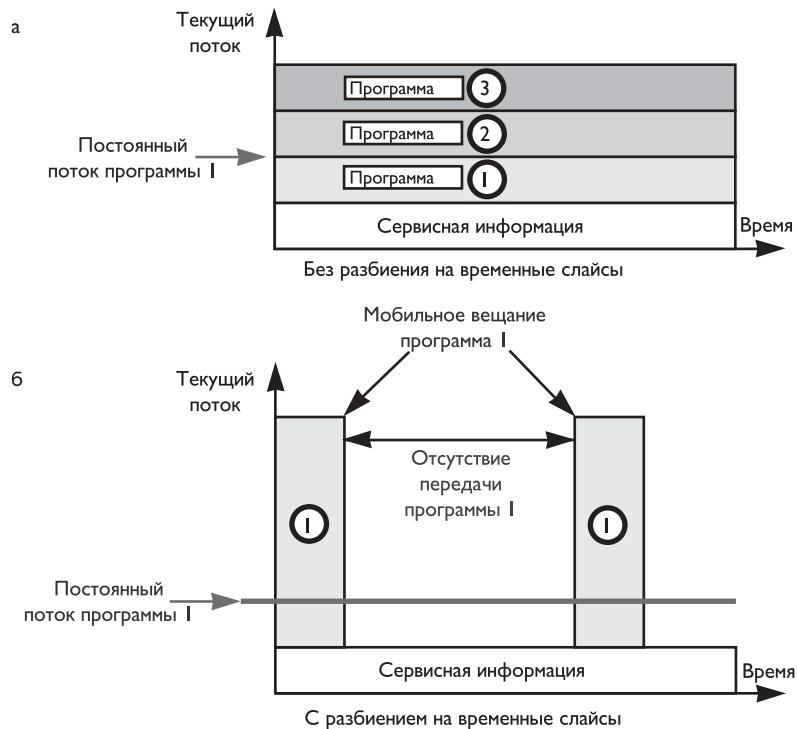


Рис. 21.68. Варианты трансляции телевизионных программ в стандарте DTMB

Таблица 21.76. Пороговое отношение сигнал/шум при мобильном приеме на различных несущих частотах

Вид модуляции	Кодовая скорость	$f_d = 70$ Гц; $(C/N)_{min}$, дБ	f_d , Гц $(C/N)_{min} + 3$ дБ	Скорость, км/час, при $(C/N)_{min} + 3$ дБ			
				65 МГц	200 МГц	500 МГц	700 МГц
4-QAM-NR	4/5	6	162	2692	875	350	250
4-QAM	2/5	6	162	2692	875	350	250
	3/5	10	148	2459	799	320	228
	4/5	14	123	2044	664	266	190
	2/5	12	134	2226	724	290	207
16-QAM	3/5	17	116	1927	626	251	179

Эфирные, спутниковые и кабельные платформы DVB-T2, DVB-S2 и DVB-C2 были разработаны и внедряются сейчас из-за существенно больших скоростей передачи данных по сравнению с их предшественниками DVB-T, DVB-S и DVB-C.

Способы кодирования видео- и аудиоинформации играют важнейшую роль в повышении эффективности систем цифрового вещания, в их конкурентоспособности по сравнению с другими платформами вещания. Применение последних стандартов сжатия видеосигналов позволяет эффективно использовать радиочастотный спектр и сократить сроки перехода на новые стандарт цифрового вещания.

По сравнению с первым поколением стандартов DVB-S/DVB-T/DVB-C в системах DVB-S2/DVB-T2/DVB-C2 были изменены параметры обработки сигналов: использовано новое поколение прямой коррекции ошибок, что позволило вплотную приблизиться к пределу Шеннона; применяются более высокие созвездия, обеспечивающие существенное повышение эффективности; увеличено возможное количество несущих OFDM; введены новые защитные интервалы; осуществлена минимизация количества рассеянных пилотов в зависимости от защитного интервала, в результате чего сокращены накладные расходы; расширена полоса пропускания; повышена эффективность обработки информации за счет расширенного временного и частотного перемежения битов и ячеек. Научно-технический прогресс в сфере цифровых телевизионных вещательных технологий в Европе поясняется на рис. 21.69 [8.57].

Рис. 21.69. Научно-технический прогресс в сфере вещательных технологий в Европе



В табл. 21.77 приведено сравнение достигаемых скоростей цифрового потока различных систем цифрового вещания первого и второго поколений для стандартных сетей связи.

Таблица 21.77. Скорости цифрового потока, Мбит/с

Полоса канала	26 МГц		54 МГц		8 МГц			
	DVB-S	DVB-S2	DVB-S	DVB-S2	DVB-T	DVB-T2	DVB-C	DVB-C2
Стандарт								
Минимальная	18,7	12,9	38,9	26,2	4,9	7,4	6,4	25,0
Максимальная	32,8	116,7	68,8	242,4	31,6	50,5	38,1	78,6

Спутниковый стандарт DVB-S2 был самым первым из второго поколения стандартов DVB, и поэтому процесс его внедрения на сегодняшний день успел продвинуться гораздо дальше других. Появление DVB-S2 совпало по времени с появлением реального рыночного (а не экспериментального) HDTV, а также с масштабным внедрением стандарта MPEG-4 AVC/H.264, вследствие чего DVB-S2 фактически стал обязательным форматом всех спутниковых HDTV-трансляций. В целом, на сегодняшний день переломный момент перехода на DVB-S2 уже остался позади. Оборудование, неспособное принимать спутниковый сигнал в этом стандарте, воспринимается как исчезающий вид, и практически у каждого оператора спутникового ТВ в мире уже разработан план по постепенному переходу на DVB-S2 с применением MPEG-4.

В силу того, что эфирный DVB-T2 появился намного позже спутникового стандарта, его поддержка на сегодняшний день присутствует далеко не в каждом устройстве, способном декодировать MPEG-4. Несомненно, такая ситуация является временной: именно DVB-T2 рассматривается в европейских странах как стандарт для эфирного HDTV. И хотя в данный момент на некоторых территориях ведутся тестовые HD-трансляции в обычном DVB-T, регулярное эфирное HD-вещание в Европе будет вестись исключительно в DVB-T2.

Использование DVB-T2 для телевидения высокой четкости является ключевым элементом долговременной стратегии по переводу на него всего цифрового вещания. Рано или поздно каждый телезритель независимо от его уровня дохода получит возможность смотреть HDTV, а если любой HD-приемник будет поддерживать DVB-T2, то это даст возможность перевода на него и каналов стандартной четкости, что, в свою очередь, позволит освободить место под дополнительные каналы HDTV.

Перспективы использования стандарта кабельного вещания DVB-C2 представляются пока не такими радужными, как у его спутниковых и эфирных сородичей. Основная проблема в том, что DVB-C2 появился на свет гораздо позже момента перехода большинства кабельных сетей на DVB-C и HDTV — момента, когда можно было безболезненно сделать его обязательным элементом каждого современного кабельного ресивера. А так как в кабельных сетях острые проблемы с частотным ресурсом обычно отсутствуют, необходимость идти на какие-либо жертвы ради экономии частот неочевидна. И хотя мертворожденным DVB-C2 называть нельзя (в определенных случаях применение ему все же найдется), у этого стандарта, скорее всего, не будет безоговорочной позиции, которой уже сейчас пользуется DVB-S2, и которую предрекают стандарту DVB-T2.

Этот новый стандарт DVB-T2 обладает существенно более высокой спектральной эффективностью по сравнению со стандартом DVB-T. По сути, это означает, что в том же участке спектра может быть передано большее количество программ или такое же количество передач, но с лучшим аудио/видео качеством или с большей зоной покрытия территории. Применение кодирования источника с лучшим сжатием (например, MPEG-4) дополнительно увеличивает количество передаваемых ТВ-программ. Возможно существенное увеличение числа программ, которые могут быть размещены в одном мультиплексе (при сохранении того же аудио/видео качества). Кроме того, становится возможной передача большего числа программ высокого разрешения. Зона покрытия при применении нового стандарта может быть значительно увеличена, сохранив при этом неизменными характеристики передатчика, а также режим приема, качество видео и число программ.

Среди основных характеристик DVB-T2 следует отметить:

- применение имеющихся приемных и передающих антенн и возможность повторного использования существующей инфраструктуры;
- предоставление услуг для стационарных и портативных приемников;
- увеличение возможностей передачи информации примерно на 50% по сравнению с DVB-T;
- улучшенная производительность одночастотных сетей;
- механизм предоставления спецпотребителям надежных служб, т. е. имеется

возможность предоставления различных уровней надежности; например, в рамках одного канала возможно предоставление части услуг как для стационарного приема, так и других услуг для приема на портативные приемные устройства;

- гибкость пропускной способности и выбора полосы частот;
- механизм уменьшения пик-фактора сигнала (уменьшения затрат на его передачу).

В табл. 21.78 представлено сравнение максимальной скорости передачи и допустимого отношения сигнал/шум (С/Ш) для стандартов DVB-T и DVB-T2. Сравнивая значения максимальной скорости передачи для, скажем, С/Ш = 6 дБ и 20 дБ, видим, что соответствующие максимальные скорости передачи для стандарта DVB-T2 (15 и 45,2 Мб/с — выделены синим) более чем в полтора раза выше соответствующих скоростей для стандарта DVB-T (9 и 31,7 Мб/с — выделены красным). Тем самым очевиден выигрыш в полтора раза, обеспечиваемый стандартом второго поколения.

Таблица 21.78. Сравнение максимальной скорости передачи и допустимого отношения сигнал/шум

Тип модуляции	Скорость кодирования	Максимальная скорость передачи, Мб/с		Допустимое отношение сигнал/шум, дБ	
		DVB-T	DVB-T2	DVB-T	DVB-T2
QPSK	1/2	6,03	7,45	3,1	1,0
	3/5	—	8,94	—	2,3
	2/3	8,04	9,95	4,9	3,1
	3/4	9,05	11,20	5,9	4,1
	4/5	—	11,95	—	4,7
	5/6	10,05	12,45	6,9	5,2
	7/8	10,56	—	7,7	—
16-QAM	1/2	12,06	15,04	8,8	6,0
	3/5	—	18,07	—	7,6
	2/3	16,09	20,11	11,1	8,9
	3/4	18,10	22,62	12,5	10,0
	4/5	—	24,14	—	10,8
	5/6	20,11	25,16	13,5	11,4
	7/8	21,11	—	13,9	—
64-QAM	1/2	18,10	22,48	14,4	9,9
	3/5	—	27,02	—	12,0
	2/3	24,13	30,06	16,5	13,5
	3/4	27,14	33,81	18,0	15,1
	4/5	—	36,08	—	16,1
	5/6	30,16	37,62	19,3	16,8
	7/8	31,67	—	20,1	—
256-QAM	1/2	—	30,07	—	13,2
	3/5	—	36,14	—	16,1
	2/3	—	40,21	—	17,8
	3/4	—	45,24	—	20,0
	4/5	—	48,27	—	21,3
	5/6	—	50,52	—	22,0
	7/8	—	—	—	—

Увеличенный диапазон параметров COFDM позволяет значительно сократить накладные расходы DVB-T2 по сравнению с DVB-T, что в сочетании с улучшенной коррекцией ошибок кодирования позволяет повысить производительность почти на 50% при эксплуатации как многочастотных сетей, так и одночастотных сетей. DVB-T2 также имеет три новые полосы частот: 1,7 МГц (для мультимедийных приложений), 5 МГц и 10 МГц (для профессиональных приложений). На физическом уровне квантование во временной области позволяет эффективно реализовать режим энергосбережения и обеспечивает различный уровень надежности разным каналам физического уровня.

Стандарт DVB-T2 также предоставляет ряд новых возможностей для повышения гибкости и устойчивости в критических условиях приема, таких как:

- поворот созвездий, что содействует получению более высоких скоростей сигналов в сложных каналах передачи;
- специальные методы уменьшения пик-фактора (PAPR) передаваемого сигнала, что приводит к повышению эффективности передатчиков в части увеличения излучаемой мощности до 4 раз при неизменном энергопотреблении, либо существенном уменьшении энергопотребления при сохранении излучаемой мощности;
- режим передачи со многими входами и одним выходом (MISO) с использованием модифицированной формы кодирования, которая позволяет разделить сигналы в пространстве и времени.

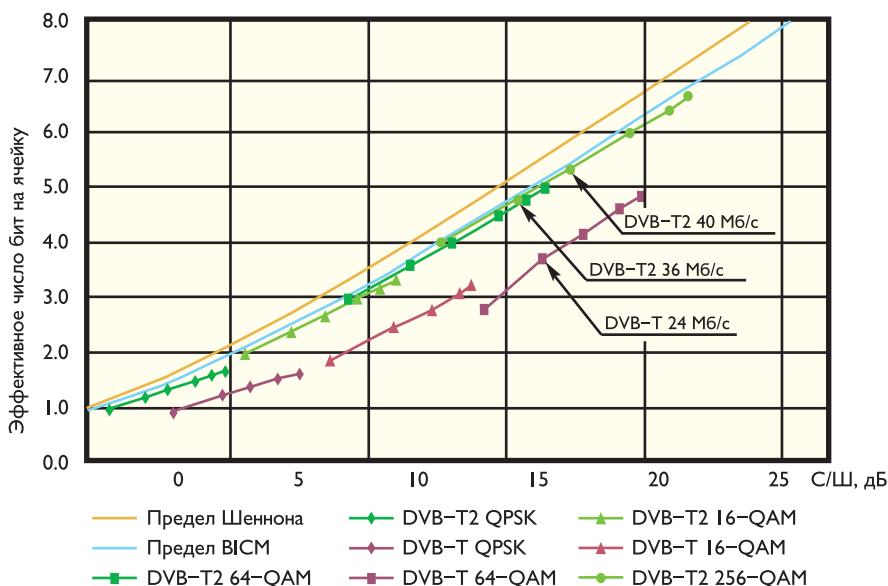


Рис. 21.70. Сравнительные характеристики DVB-T и DVB-T2

На рис. 21.70 приведены графики эффективности систем DVB-T и DVB-T2, которые показывают, сколь близко канальное кодирование для стандарта второго поколения DVB-T2 к пределу Шенна и насколько менее эффективна система, применяющая стандарт первого поколения DVB-T.

Важнейшее значение имеет применяемый способ кодирования телевизионного контента в системах цифрового вещания. На рис. 21.71 приведено несколько примеров наполнения каналов вещания.

Первый пример — это DVB-T мультиплекс (с типичным цифровым потоком 25 Мб/с), обеспечивающий прием 7 MPEG-2 SD-программ, каждая из которых включает компонентное видео 3,2 Мб/с и стерео аудио AAC с отличным качеством изображения и звука.

Второй пример — кодер MPEG-2 заменяется на MPEG-4 и с тем же качеством видео число каналов увеличивается в два раза, при этом будет 14 компонент с видеосигналом 1,5 Мбит/с.

Следующий пример — это применение DVB-T2 с типичным доступным потоком 44 Мб/с, что обеспечивает передачу 25 SD-программ.

Последний пример включает 8 MPEG-4 HD-программ высокого разрешения с общим потоком в 44 Мб/с в рамках мультиплекса DVB-T2. Телеканалы, радио, данные, программы стандартного (SD) и высокого разрешения (HD), потоки MPEG-2, MPEG-4 — все это может быть объединено в общий статистический мультиплекс.

С точки зрения частотного планирования, для перехода на DVB-T2 никакой дополнительной перепланировки частот не потребуется как на национальном, так и на международном уровне. Для стран, которые еще не начали переход на DVB-T, вполне логичен непосредственный ввод стандарта DVB-T2. Так как вещание в стандарте DVB-T2 в странах Европы уже началось, следует считать, что оборудование для DVB-T2 в скором времени будет доступно на массовом рынке. Ожидается, что потребуется период одновременного вещания в разных стандартах, аналогичный используемому при переходе от аналогового телевидения к DVB-T. Длительность такого периода во многом зависит от значимости наземных платформ вещания по сравнению с другими платформами. Для периода одновременного вещания будет необходим дополнительный спектр для параллельной передачи телевизионных услуг.

Так как DVB-T2 не является обратно совместимым с DVB-T, резкий переход от DVB-T к DVB-T2 не представляется возможным. Поэтому требуются более сложные стратегии миграции. Они должны быть основаны на дополнительных предложениях потребителю, например дополнительных программ или новых видов сервисов, таких как ТВЧ (HDTV), 3D-TV. Для стран, которые уже перешли на DVB-T, вопрос о реинвестиции потребителей может стать проблемой. Введение DVB-T в течение последних десяти лет сопровождалось необходимостью для потребителей вкладывать средства в новое приемное оборудование. Теперь, с переходом на DVB-T2, потребителям потребуются новые инвестиции в приемное оборудование. Это ставит потребителей в сложное положение, поскольку они привыкли к более долгим циклам обновления приемного оборудования для ТВ. Стратегия перехода на DVB-T2 должна быть выбрана тщательно, с тем чтобы не потерять клиентов, которые могут уйти на другие платформы, как это произошло в некоторых странах во время перехода от аналогового ТВ к DVB-T.

Особая ситуация возникает в странах, которые начали, но не завершили процесс цифрового перехода от аналогового телевидения к DVB-T, а также начали внедрение DVB-T2. Эта ситуация встречается не так уж редко. Страны, в которых наземные платформы используются большим процентом населения в качес-

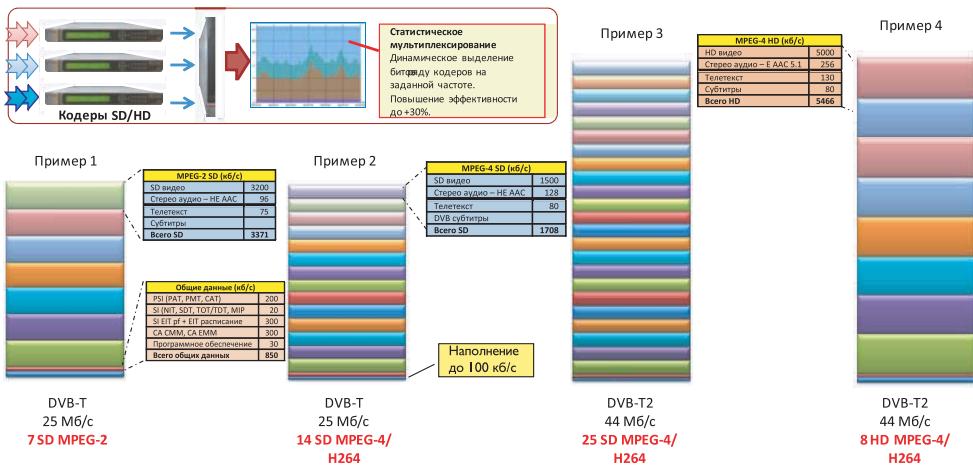


Рис. 21.71. Примеры мультиплексов с использованием стандартов DVB-T и DVB-T2

тве основного средства получения услуг ТВ, обязательно будут иметь более длительный переходный период, и они в настоящее время сталкиваются с проблемой дополнительной миграции.

В табл. 21.79 приведен сравнительный перечень основных характеристик цифровых телевизионных вещательных систем, реализующих стандарты DVB-T и DVB-T2.

Таблица 21.79. Основные характеристики DVB-T и DVB-T2

Параметры	DVB-T	DVB-T2
Сигналы	Поток MPEG-2	Поток MPEG-2 и варианты потоков GS (уменьшение накладных расходов)
Коррекция ошибок	Сверточные коды + RS	LDPC + BCH
Скорость кода	1/2, 2/3, 3/4, 5/6, 7/8	1/2, 3/5, 2/3, 3/4, 4/5
Перемежители	Битов	Бит/ячеек/временное/частотное
Созвездие	QPSK, 16-QAM, 64-QAM	QPSK, 16-QAM, 64-QAM, 256-QAM
Защитный интервал	1/4, 1/8, 1/16, 1/32	1/4, 19/256, 1/8, 19/128, 1/16, 1/32, 1/128
Размер БПФ	2K, 8K	1K, 2K, 4K, 8K, 16K, 16K расшир., 32K, 32K расшир.
C/I (канал Гаусса)	от 3 дБ (QPSK) до 20 дБ (64-QAM)	от 1 дБ (QPSK) до 22 дБ (256-QAM)
Рассеянные пилоты	8%	1%, 2%, 4%, 8%
Постоянные пилоты	2,6%	0,35%
Каналы физического уровня (PLP)	нет	служба или группа служб, делящих заданную кодовую скорость и модуляцию
Вращение созвездий	нет	существенно возросшая устойчивость (5 дБ для скорости 5/6)
Будущие фреймы расширения (FEF)	нет	для будущих MIMO версий DVB-T2 или DVB-NGH
Ширина полосы, МГц	6, 7, 8	1,7, 5, 6, 7, 8, 10
Максимальная емкость канала, Мбит/с	31,66	50,34

Литература к части VIII

- 8.1. *Вишневский В.М., Портной С.Л., Шахнович И.В.* Энциклопедия WiMAX. Путь к 4G. М.: Техносфера, 2010.
- 8.2. ITU-T. Rec. X200. Information technology — Open Systems Interconnection — Basic Reference Model: The basic model. Identical standard:ISO/IEC 7498-1:1994 (Common) Approval date: 1994-07-01
- 8.3. http://en.wikipedia.org/wiki/OSI_model
- 8.4. *Кривошеев М.И.* Международная стандартизация цифрового телевизионного вещания. М.: НИИР, 2006.
- 8.5. ATSC Document A/53. ATSC Digital Television Standard.
- 8.6. ATSC Document A/54. Guide to the Use of the ATSC Digital Television Standard.
- 8.7. ATSC Document A/52 Digital Audio Compression (AC-3) Standard.
- 8.8. ATSC Document A/63, Standart for Coding 25/50 Hz Video.
- 8.9. ATSC Document A/53. ATSC Digital Television Standard, Parts 1-6, 2007. http://www.atsc.org/cms/standards/a53/a_53-Part-1-6-2007.pdf
- 8.10. ITU-R Recommendation BT.1306-1/ Error-Correction, Data Framing, Modulation and Emission Methods for Digital Terrestrial Television Broadcasting – ITU-R Recommendations/ BT Series. Broadcasting Service (Television).
- 8.11. Zenith Electronics Corporation (Glenview, IL). Patent RE36992 VSB HDTV transmission system with reduced NTSC co-channel interference. 1994.
- 8.12. *Зубарев Ю.Б., Кривошеев М.И., Красносельский И.Н.* Цифровое телевизионное вещание. Основы. Методы. Системы. М.: НИИР, 2001.
- 8.13. *Варгузин В.* Принципы цифрового телевидения системы ATSC// Теле-Спутник. 1999. № 9 (47).
- 8.14. *Гласман К.* Стандарт цифрового телевидения ATSC// «625». 1999. № 7.
- 8.15. European Standard (Telecommunications series). ETSI EN 300 744 V1.6.1 (2009-01). Digital Video Broadcasting (DVB); Framing structure, channel coding and modulation for digital terrestrial television.
- 8.16. Стандарт DVB-T. Технологии мультисервисных сетей. Прием, цифровые передатчики, сигнал вещания, параметры приемной и передающей антennы. <http://www.conturm.com/tech.php?id=dvbtg>
- 8.17. European Standard (Telecommunications series). ETSI EN 302 755 V1.1.1 (2009-09): Digital Video Broadcasting (DVB); Frame structure channel coding and modulation for a second generation digital terrestrial television broadcasting system (DVB-T2).
- 8.18. *Уэллс Н., Нокс. К.* DVB-T2: Новый стандарт вещания для телевидения высокой четкости// Теле-Спутник . 2008. № 11 (157).
- 8.19. ISO/IEC 13818-1: «Information technology — Generic coding of moving pictures and associated audio information: Systems».
- 8.20. ETSI TS 102 606: «Digital Video Broadcasting (DVB); Generic Stream Encapsulation (GSE) Protocol».
- 8.21. ETSI EN 302 307: «Digital Video Broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications».



- 8.22. ETSI EN 300 468: «Digital Video Broadcasting (DVB); Specification for Service Information (SI) in DVB systems».
- 8.23. *Alamouti S.M.* «A simple transmit diversity technique for wireless communications»// IEEE Selected Areas Commun. 1998. V. 16. P. 1451–1458.
<http://www.mast.queensu.ca/~fady/Math800/papers/Alamouti>
- 8.24. DVB Document A122 «Frame structure channel coding and modulation for a second generation digital terrestrial television broadcasting system (DVB-T2)».
- 8.25. *Шахнович И.* DVB-T2 — новый стандарт цифрового телевизионного вещания// ЭЛЕКТРОНИКА: Наука, Технология, Бизнес. 2009. № 6.
- 8.26. DVB Document A133. Implementation guidelines for a second generation digital terrestrial television broadcasting system (DVB-T2). February, 2009.
- 8.27. ETSI. EN 300 421 «Framing structure, channel coding and modulation for 11/12 GHz satellite services». 1997-08.
- 8.28. ETSI.TR 101 198 «Implementation of Binary Phase Shift Keying (BPSK) modulation in DVB satellite transmission systems». 1997-09.
- 8.29. ETSI. EN 302 307 «Second generation framing structure, channel coding and modulation systems for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications». 2004-06.
- 8.30. ETSI. TR 102 376 «User guidelines for the second generation system for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications». 2005-02.
- 8.31. ETSI. TS 102 441 «DVB-S2 Adaptive Coding and Modulation for Broadband Hybrid Satellite Dialup Applications». 2005-10.
- 8.32. Стандарт DVB-S2. Система цифрового ТВ вещания.
<http://www.konturm.ru/tech.php?id=dvbs2>
- 8.33. *Андуфриев А.* Стандарт DVB-S2 как средство развития новых сервисов на спутниковых сетях связи// Broadcasting. Телевидение и радиовещание. 2007. № 3.
- 8.34. ETSI EN 301 210: «Digital Video Broadcasting (DVB); Framing structure, channel coding and modulation for Digital Satellite News Gathering (DSNG) and other contribution applications by satellite». 1999-03.
- 8.35. ETSI EN 300 429 «Framing structure, channel coding and modulation for cable systems». 1998-04.
- 8.36. Стандарт DVB-C. Система цифрового ТВ вещания.
<http://www.konturm.ru/tech.php?id=dvbc>
- 8.37. DVB-C — стандарт цифрового телевизионного вещания по кабелю.
<http://www.roks.com.ua/ru/info/standarts/dvb-c>
- 8.38. ETSI EN 302 769 « Digital Video Broadcasting (DVB); Frame structure channel coding and modulation for a second generation digital transmission system for cable systems (DVB-C2)», 2010-04.
- 8.39. *Robert J., Schaaf C., L. Stadelmeier L.* DVB-C2 — стандарт передачи по сетям КТВ следующего поколения// Теле-Спутник — 2009. № 12.
- 8.40. ITU-R. Doc.11A/59-E, 17.05.1999. Delayed contribution (Japan). — Proposed Draft new Recommendation/ Channel Coding, Frame Structure and Modulation Scheme for Terrestrial Integrated Services Digital Broadcasting (ISDB-T).

- 8.41. ITU-R. Doc.11A/61-E, 28.04.1999. (Special Rapporteur). — Updating of the Handbook on Terrestrial Television Broadcasting ISDB-T.
- 8.42. ISDB-T. <http://es.wikipedia.org/wiki/ISDB-T#Introducci.C3.B3n>
- 8.43. Крикошев М.И. Новые системы цифрового вещания// «625». 1999. № 10.
- 8.44. ITU-R Recommendation BO.1408-1 (04/02) Transmission system for advanced multimedia services provided by integrated services digital broadcasting in a broadcasting-satellite channel. 2002.
- 8.45. ISDB-S. <http://uk.ask.com/wiki/ISDB#ISDB-S>
- 8.46. ISDB-S — Satellite Transmission System for Advanced Multimedia Services Provided by Integrated Services Digital Broadcasting.(Courtesy Asia-Pacific Broadcasting Union.) NHK (Japan Broadcasting Corporation).
<http://www.nhk.or.jp/strl/publica/bt/en/pa0005.html>
- 8.47. ITU-T. Recommendation J.83. Digital multi-programme systems for television, sound and data services for cable distribution. 12/2007.
- 8.48. ITU-T. Recommendation J.183. Time-division multiplexing of multiple MPEG-2 transport streams over cable television systems. 03/2001.
- 8.49. General Administration of Quality Supervision, Inspection and Quarantine of the People's Republic of China,Standardization Administration of the People's Republic of China. Framing Structure, Channel coding and Modulation for Digital Terrestrial Broadcasting System (DTMB) // Chinese National Standard GB 20600-2006, Publish Data 2006-08-18, Execution Data 2007-08-01.
- 8.50. ITU-R. China (People's Republic of). Chinese Digital Terrestrial Television Broadcasting System// Doc. 6A/287, 3 December 2009.
- 8.51. *Luong Vinh Quoc Danh*, Ph.D. Lecturer at Cantho University, Vietnam.
<http://lvqdanh.blogspot.com/>
- 8.52. *Karamchedu R.* China's DTV standard revolutionises broadcast.
http://www.embeddeddesignindia.co.in/STATIC/PDF/200902/EDIOL_2009FEB16_DSP_TA_01.pdf?SOURCES=DOWNLOAD
- 8.53. *Jian Song.* DTMB. Technical Review of the Chinese Digital Terrestrial Television Broadcasting Standard (DTMB). DTV Technology R&D Center Tsinghua University. Dec. 18, 2006.
http://www.tnlist.org.cn/jspsthnokiasummit06/keynote_speech/Keynote%20Speech%206-song%20jian-TechnicalReviewoftheDTMB.pdf
- 8.54. *Dr. Yun Tang.* Content Protection in China. A Technical Perspective. 2007, 2008, 2009.
- 8.55. Крикошев М.И., Федунин В.Г.Система цифрового наземного ТВ-вещания DTMB// Телеспутник. 2010. № 4.
- 8.56. *Chun-Chien Tseng, Chi-Nong Hsieh.* Decoding System for LDPC Code Concatenated with 4QAM-NR Code. US Patent AP. Pub. No: US 2010/0074319 A1. Mar. 25, 2010.
- 8.57. Чобану М.К., Дворкович А.В., Дворкович В.П. Эффективность современных цифровых систем вещания// Радиочастотный спектр. 2011. №4.

ЧАСТЬ IX

СТАНДАРТЫ СИСТЕМ МОБИЛЬНОГО РАДИОВЕЩАНИЯ И ВИДЕОКОНФЕРЕНЦСВЯЗИ

Введение

Существует несколько форматов вещания, ориентированных на прием мобильными терминалами. К ним следует отнести принятый в декабре 2004 года Европейской ассоциацией по телекоммуникационным стандартам (ETSI) формат DVB-H, закрытую систему MediaFLO разработки компании Qualcomm, группу форматов, базирующихся на системе радиовещания DAB, к которой относятся Movio System (бывшая Live Time), разработанная British Telecom, корейские форматы T-DMB и S-DMB. К этим форматам, рассмотренным в части VIII, можно также отнести японский стандарт эфирного вещания ISDB-T и китайский формат DTMB (DMB-T/H), по своей гибкости пригодные для любых видов эфирного вещания на любые терминалы (см. табл. IX.1).

Система DVB-H разработана на базе DVB-T, что обеспечивает их частичную совместимость. Она заключается в том, что трансляции DVB-H, за исключением одного режима модуляции, могут приниматься приемниками DVB-T, и в одном мультиплексированном потоке возможно совмещение трансляций DVB-H и DVB-T.

В настоящее время проводится разработка стандарта DVB-NGH (DVB-H2). Ожидается, что спецификации стандартов DVB-NGH (Next Generation Handheld) и DVB-T2 будут взаимосвязаны. Широкое внедрение устройств стандарта DVB-NGH предполагается с 2013 года.

Для мобильного телевидения в различных странах отведены определенные диапазоны частот: как традиционные для эфирного вещания диапазоны (VHF, UHF), так и более высокочастотные диапазоны — L-band (1452–1492 МГц) и IMT2000/UMTS (1900–2200 МГц). Важно отметить, что в качестве системы модуляции в стандартах DVB-H, MediaFLO и DMB-T используется система COFDM. Во всех этих стандартах применяется один из трех видов модуляции: квадратурная амплитудная манипуляция 64-QAM, 16-QAM или QPSK. Принцип помехоустойчивого кодирования во всех перечисленных выше стандартах также примерно одинаков: двухступенчатое кодирование с внешним кодом Рида–Соломона и внутренним — сверточным кодом (в DVB-H и DMB-T) и турбо-кодом (в MediaFLO).

Каждый из стандартов мобильного ТВ имеет свою зону влияния: DVB-H распространен в Европе, MediaFLO популярен в Северной Америке (комерческое вещание — в США, испытания — в Канаде и Японии), а T-DMB — в Азии и ряде европейских стран. Сейчас большинство специалистов отдают предпочтение стандарту DVB-H.

В ноябре 2007 года Европейский союз признал DVB-H официальным стандартом мобильного ТВ в Европе. Возможно, что большинство азиатских, африканских и ближневосточных стран также поддержат этот стандарт.

С DVB-H активно конкурирует стандарт MediaFLO. Этому способствовала договоренность компании Qualcomm с крупнейшими операторами сотовой связи в США — компаниями AT&T Mobile и Verizon Wireless — о развитии мобильного сервиса на базе технологии MediaFLO. Данный стандарт продвигался не только в США, но и в ряде других стран.

Основными достижениями мобильного режима в системах цифрового телевидения следующие факторы:

- устойчивость к многолучевому распространению сигнала и собственным помехам;
- масштабируемая пропускная способность канала;
- технология временного метода разделения сеансов приема–передачи данных (Time Division Duplex — TDD), который позволяет эффективно обрабатывать асимметричный трафик и упрощает управление сложными системами антенн за счет эстафетной передачи сессии между каналами;
- технология сочетания прямого исправления ошибок с автоматическим определением неисправимых ошибок (Hybrid-Automatic Repeat Request — H-ARQ), которая позволяет сохранять устойчивое соединение при резкой смене направления движения клиентского оборудования;
- распределение выделяемых частот и использование субканалов при высокой загрузке позволяет оптимизировать передачу данных с учетом мощности сигнала клиентского оборудования;
- управление энергосбережением позволяет оптимизировать затраты энергии на поддержание связи портативных устройств в режиме ожидания или простоя;
- технология оптимального переключения каналов связи (Network-Optimized Hard Handoff — ННО), реализованная, например, в системе WiMAX, позволяющая до 50 мс и менее сократить время на переключение клиента между каналами;

- технология многоадресной рассылки и передачи (Multicast and Broadcast Service — MBS), объединяющая функции DVB-H, MediaFLO и 3GPP E-UTRA для достижения высокой скорости передачи данных с использованием одночастотной сети, гибкого распределения радиочастот, низкого потребления энергии портативными устройствами, быстрого переключения между каналами;
- технология формирования лучей излучения с учетом направления их на потребителя (Smart Antenna), поддерживающая субканалы и эстафетную передачу сессии между каналами, что позволяет использовать сложные системы антенн, включая формирование диаграммы направленности, пространственно-временное маркирование, пространственное мультиплексирование;
- технология различного набора каналов (Fractional Frequency Reuse), которая позволяет контролировать наложение/пересечение каналов для повторного задействования частот с минимальными потерями;
- малый размер фрейма (порядка 5 мс) создает оптимальный компромисс между надежностью передачи данных за счет использования малых пакетов и накладными расходами за счет увеличения числа пакетов (и как следствие — заголовков).

Таблица IX.1. Стандарты цифрового мобильного радиовещания

Обозначение	Стандарт, Рекомендация	Диапазон частот; ширина спектра, цифровой поток	Год утверждения стандарта
DVB-H/DVB-SH	ETSI EN 302 304	VHF, UHF; 6, 7, 8 МГц; L-band и IMT2000/UMTS 4,98...31,67 Мбитов/с	2007
MediaFLO	QUALCOMM FLO TV TM	UHF; 5, 6 МГц; частотный диапазон в области 700 МГц	2009
DMB/T-DAB	ETSI TS 102 427 ETSI TS 102 428	III band VHF, L-band; 1,5 МГц	2006
ISDB-T	ITU-R BT.1306	VHF, UHF; 6, 7, 8 МГц; 3,65...30,98 Мбитов/с	1999
DTMB	GB 20600–2006	VHF, UHF; 6, 8 МГц; 5,41...28,8 Мбитов/с	2006
РАВИС	ГОСТ Р 54309–2011	I/II band VHF; 250, 200, 100 кГц 80...900 кбитов/с	2011
DRM+	ETSI ES 201 980 V.3.1.1	I-III band VHF; 100 кГц 40...186 кбитов/с	2009

Основным же недостатком цифровых телевизионных мобильных систем является использование широкополосных каналов связи и диапазонов частот более 200 МГц. По этой причине в последнее время разрабатываются весьма эффективные узкополосные системы мультимедийного вещания в VHF-диапазоне частот.

Мультимедийная технология DMB была рождена радиовещательным цифровым стандартом DAB (Digital Audio Broadcasting), утвержденным в 1994 году и изначально ориентированным на звуковое радиовещание на мобильные, в частности, автомобильные приемники.

В стандарт заложена возможность приема сигнала на высокой скорости при наличии значительной величины эффекта Дошлера. Существует наземная (T-DAB) и спутниковая (S-DAB) версии стандарта DAB, однако последняя не получила широкого распространения. Наземная (эфирная) версия стандарта предназначена для доставки сигнала в III полосе VHF диапазона (174–230 МГц) или L-диапазоне (1400–1900 МГц).

В октябре 2004 года южнокорейскими компаниями Samsung и LG Electronics были продемонстрированы терминалы DAB, позволяющие принимать различные виды данных, включая видеинформацию. Эта технология и получила название DMB (T-DMB). Сущность технологии лежит на поверхности. Для сжатия видеосигнала может быть использована любая технически подходящая технология: H.264, Windows Media и т. п. Цифровые аудио- и видеосигналы мультиплексируются в один мультимедийный высокоскоростной транспортный поток. При необходимости данные могут быть упакованы в IP-пакеты. Эти идеи и составляют содержание технологии DMB.

Разработанная в России аудиовизуальная информационная система реального времени РАВИС (RAVIS) обеспечивает решение проблемы цифровизации сети мобильного радиовещания взамен аналоговых систем УКВ-ЧМ и FM вещания в выделенных для этих целей в России диапазонах частот и внедрение собственных мультимедийных систем и соответствующей аппаратуры.

При создании отечественного стандарта системы РАВИС были положены следующие важнейшие технические требования:

- создание цифровой системы аудиовизуального вещания в ОВЧ-диапазоне (66–74 МГц и 87,5–108 МГц), в котором на территории России ведется звуковое радиовещание;
- использование при разработке общедоступных стандартов под эгидой таких организаций, как ETSI или ITU;
- более высокая спектральная эффективность по сравнению с аналоговым FM и УКВ-ЧМ-вещанием;
- гибкость группирования несущих для поддержки различных распределений частот;
- повышенная надежность сервиса по сравнению с аналоговым вещанием, особенно в условиях мобильного вещания;
- системный подход должен способствовать разработке недорогих приемников;
- поддержка одночастотных сетей;
- использование достижений видео- и аудиоинформационных систем нового уровня по качеству воспроизведения изображения и звука в рамках приема информации в движущемся транспорте;
- удовлетворение постоянно возрастающего спроса на частотные присвоения для систем связи без пересмотра частотных планов;
- организация вещания нескольких программ стереозвука высокого качества, включая поддержку объемного звука, или трансляция изображения с допустимой четкостью со стереозвуковым сопровождением в крупном городе с использованием только одного передатчика;
- возможности транспорта данных, допускающие введение новых мультимедийных функций;

- эффективность и возможность разнесенного приема;
- возможность реализации пространственного масштабирования при трансляции динамических изображений.

Все эти требования выполнены при реализации системы РАВИС. Благодаря использованию современных алгоритмов компрессии изображения и звука, эффективных методов канального кодирования и модуляции система РАВИС позволяет в стандартном канале (200–250 кГц) передавать либо много программ стереозвука, включая поддержку объемного звукового вещания, либо качественное цифровое телевизионное изображение с разрешением от 160 × 120 до 640 × 480 пикселов на кадр и соответствующий звуковой стереосигнал.

При этом при меньших уровнях излучаемого сигнала, чем при FM-вещании, радиус уверенного приема сигнала системы РАВИС может достигать стандартных для FM-диапазона значений 80–90 км.

По сравнению с DVB-H система мобильного телевещания РАВИС имеет ряд ключевых преимуществ:

- система РАВИС — полностью отечественная разработка;
- существенное повышение эффективности использования частотного ресурса в ОВЧ диапазоне частот;
- возможность в стандартном канале FM-вещания передачи нескольких стереофонических и многоканальных программ;
- трансляция видеопрограмм для мобильного потребителя при возможности реализации телевизионного вещания в малонаселенных пунктах при резком удешевлении системы кодирования и трансляции;
- простота процедур получения радиочастотных разрешений благодаря использованию стандартной полосы каждого канала в ОВЧ-диапазоне;
- возможность использования существующей на всей территории России радиопередающей инфраструктуры в ОВЧ-диапазоне;
- обеспечение устойчивого мобильного приема сигнала в условиях многолучевого распространения в среде с городской застройкой, в горной и лесистой местности, в водных акваториях;
- радиус покрытия в десятки раз больше, а затраты на создание инфраструктуры мобильного телевещания в десятки раз ниже (к примеру, покрытие мобильным телевещанием всей территории Москвы может быть обеспечено одним, либо максимум тремя передатчиками, работающими в одночастотном режиме);
- возможность локализовать вещание, т. е. на одной и той же частоте в разных городах передавать различные программы;
- возможность создания одночастотных сетей мобильного вещания вдоль железнодорожных и шоссейных дорог (например, на магистрали Москва–Санкт-Петербург);
- существенное снижение энергопотребления передающих станций в пересчете на одну программу вещания;
- реализация эфирных систем оповещения населения и организаций в условиях чрезвычайных ситуаций;
- организация систем для использования различными спецслужбами и др.

Конкурентной технологией системы РАВИС можно считать разработку DRM консорциума, в составе которого более 100 организаций из 34 стран, «альтернативной» системы DRM+, более узкополосной (100 кГц на канал), обеспечивающей передачу в одном канале не более 4 стереофонических звуковых программ, но не позволяющей транслировать видеопрограммы. Заметим, что эта разработка была начата после публикаций патента России РФ № 2219676, статьи в журнале «Broadcasting. Телевидение и радиовещание» и первого вклада России в ITU о модели системы РАВИС.

При всех положительных свойствах системы DRM+ можно считать не очень рациональной нестандартную разбивку используемого диапазона частот, поскольку возникнут сложности при ее внедрении, увеличатся частотные потери за счет увеличения числа межсигнальных интервалов.

В конце XX века одной из бурно развивающихся технологий в области телекоммуникаций стала видеоконференцсвязь (ВКС). Появление видеоконференцсвязи явилось технологическим развитием цифровой телефонии вообще и видеотелефонии в частности. Однако, если видеотелефон, появившийся еще в 1964 г., так и не стал массовым явлением, то новые возможности, предоставленные ВКС, привели к бурному росту количества пользователей этой технологии. Естественно, что развитие ВКС не могло идти без использования предшествующих стандартов ITU и создания новых рекомендаций, позволяющих унифицировать подходы к передаче мультимедийного контента и внедрения дополнительных сервисов.

Разработка рекомендаций ITU-T, используемых при построении систем видеоконференцсвязи, в основном осуществлялась в трех разделах:

- аудиовизуальные и мультимедийные системы (серия H);
- передающие системы и средства, цифровые системы и цепи (серия G);
- терминалы для телематических сервисов (серия T).

В табл. IX.2 приведены некоторые основные рекомендации ITU-T, используемые при построении систем видеоконференцсвязи.

Таблица IX.2. Основные рекомендации ITU-T, используемые при построении

№	Рекомендация ITU-T	Наименование рекомендации	Год принятия 1-й редакции
Серия H: Аудиовизуальные и мультимедийные системы (H Series: Audiovisual and Multimedia Systems)			
1	H.221	Frame structure for a 64 to 1920 kbit/s channel in audiovisual teleservices	1990
2	H.222.0	Information technology — Generic coding of moving pictures and associated audio information: Systems	2006
3	H.222.1	Multimedia multiplex and synchronization for audiovisual communication in ATM environments	1996
4	H.224	A real time control protocol for simplex applications using the H.221 LSD/HSD/MLP channels	2005

Таблица IX.2 (продолжение)

5	H.225.0	Call signalling protocols and media stream packetization for packet-based multimedia communication systems	1998
6	H.231	Multipoint control units for audiovisual systems using digital channels up to 1920 kbit/s	1993
7	H.233	Confidentiality system for audiovisual services	1993
8	H.234	Encryption key management and authentication system for audiovisual services	1994
9	H.239	Role management and additional media channels for H.300-series terminals	2003
10	H.241	Extended video procedures and control signals for H.300-series terminals	2003
11	H.242	System for establishing communication between audiovisual terminals using digital channels up to 2 Mbit/s	1990
12	H.243	Procedures for establishing communication between three or more audiovisual terminals using digital channels up to 1920 kbit/s	1993
13	H.245	Control protocol for multimedia communication	1996
14	H.246	Interworking of H-series multimedia terminals with H-series multimedia terminals and voice/voiceband terminals on GSTN, ISDN and PLMN	2006-
15	H.248.1	Gateway control protocol	2002
16	H.261	Video codec for audiovisual services at p x 64 kbit/s	1990
17	H.263	Video coding for low bit rate communication	1997
18	H.264	Advanced video coding for generic audiovisual services (Identical standard - ISO/IEC 14496-10)	2003
19	H.281	A far end camera control protocol for videoconferences using H.224	1994
20	H.310	Broadband audiovisual communication systems and terminals	1996
21	H.320	Narrow-band visual telephone systems and terminal equipment	1990
22	H.321	Adaptation of H.320 visual telephone terminals to B-ISDN environments	1996
23	H.322	Visual telephone systems and terminal equipment for local area networks which provide a guaranteed quality of service	1996
24	H.323	Packet-based multimedia communications systems	1998
25	H.324	Terminal for low bit-rate multimedia communication	1996
26	H.331	Broadcasting type audiovisual multipoint systems and terminal equipment	1993
27	H.332	H.323 extended for loosely coupled conferences	1998

Таблица IX.2 (продолжение)

28	H.450.1	Generic functional protocol for the support of supplementary services in H.323	1998
29	H.450.2	Call transfer supplementary service for H.323	1998
30	H.450.3	Call diversion supplementary service for ITU-T H.323	1998
31	H.450.4	Call hold supplementary service for H.323	1999
32	H.450.5	Call park and call pickup supplementary services for H.323	1999
33	H.450.6	Call waiting supplementary service for H.323	1999
34	H.450.7	Message waiting indication supplementary service for H.323	1999
35	H.450.8	Name identification supplementary service for H.323	2000
36	H.450.9	Call completion supplementary services for H.323	2000
37	H.450.10	Call offering supplementary services for H.323	2001
38	H.450.11	Call intrusion supplementary service for H.323	2001
39	H.450.12	Common Information Additional Network Feature for H.323	2001
40	H.460.1	Guidelines for the use of the generic extensible framework	2002
41	H.460.4	Call priority designation and country/international network of call origination identification for H.323 priority calls	2002
42	H.460.8	Querying for alternate routes within H.323 systems	2002
43	H.460.9	Support for online QoS-monitoring reporting within H.323 systems	2002
44	H.460.10	Call party category within H.323 systems	2004
45	H.460.11	Delayed call establishment within H.323 systems	2004
46	H.460.12	Glare control indicator within H.323 systems	2004
47	H.460.13	Called user release control within H.323 systems	2004
48	H.460.14	Support for Multi-Level Precedence and Preemption (MLPP) within H.323 systems	2004
49	H.460.15	Call signalling transport channel suspension and redirection within H.323 systems	2004
50	H.460.16	Multiple-message release sequence capability within H.323 systems	2005
51	H.460.17	Using H.225.0 call signalling connection as transport for H.323 RAS messages	2005
52	H.460.18	Traversal of H.323 signalling across network address translators and firewalls	2005
53	H.460.19	Traversal of H.323 media across network address translators and firewalls	2005
54	H.460.20	Location number within H.323 systems	2005

Таблица IX.2 (окончание)

Серия G: Передающие системы и средства, цифровые системы и цепи (G Series: Transmission Systems and Media, Digital Systems and Networks)			
55	G.711	Pulse code modulation (PCM) of voice frequencies	1988
56	G.723.1	Dual rate speech coder for multimedia communications transmitting at 5.3 and 6.3 kbit/s	1996
57	G.728	Coding of speech at 16 kbit/s using low-delay code excited linear prediction	1992
58	G.729	Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear prediction (CS-ACELP)	1996
Т-серия: терминалы для телематических сервисов (T Series: Terminals for Telematic Services)			
59	T.120	Data protocols for multimedia conferencing	1996
60	T.121	Generic application template	1996
61	T.122	Multipoint communication service — Service definition	1993
62	T.123	Network-specific data protocol stacks for multimedia conferencing	1999
63	T.124	Generic Conference Control	1995
64	T.125	Multipoint communication service protocol specification	1994
65	T.126	Multipoint still image and annotation protocol	1995
66	T.127	Multipoint binary file transfer protocol	1995
67	T.128	Multipoint application sharing	1998

В настоящее время системы видеоконференцсвязи реализуются как в аппаратном, так и в программном исполнении.

ГЛАВА 22

СТАНДАРТЫ ЦИФРОВОГО МОБИЛЬНОГО РАДИОВЕЩАНИЯ

В настоящей главе рассматриваются принципы построения систем мобильного радиовещания DVB-H, MediaFLO, T-DMB/T-DAB, РАВИС и DRM+.

22.1. Система мобильного телевизионного вещания DVB-H

Система DVB-H (Digital Video Broadcasting – Handheld) — технология мобильного вещания, позволяющая передавать цифровой видеосигнал на мобильные устройства, такие как КПК, мобильный телефон или портативный телевизор [9.1–9.5]. Формально, этот стандарт был принят ETSI в ноябре 2004 года.

Стандарт DVB-H является логическим продолжением стандарта DVB-T с поддержкой дополнительных возможностей, отвечающих требованиям для переносных мобильных устройств с автономным питанием.

Главное нововведение — циклическая передача пакетов информации, принадлежащих всем транслируемым программам по очереди (time slicing). Приемник включается только в определенные моменты, когда необходимо загрузить очередную порцию видеопотока. Это позволило увеличить функциональную продолжительность работы портативных приемников до 10 часов от одного заряда батареи. Другая особенность стандарта — высокая помехоустойчивость за счет дополнительного введения механизма коррекции ошибок. Еще одной особенностью стандарта DVB-H является использование IP-протокола, что значительно упрощает и удешевляет построение вспомогательной инфраструктуры, позволяет применять готовые и недорогие программные решения. Стандарты и технические отчеты по DVB-IPDC приведены в [9.6–9.14].

Упрощенная структурная схема системы DVB-H изображена на рис. 22.1. Зеленым цветом помечены элементы, добавленные к системе DVB-T.

На физическом уровне система DVB-H максимально приближена к DVB-T. К основным дополнительным возможностям DVB-H необходимо отнести ниже-следующие.

Во-первых, к режимам модуляции 2K и 8K добавлен еще один — 4K, что обеспечило реализацию дополнительной возможности увеличения максимальной скорости передвижения приемника в области охвата одной соты. Очевидно меньшее количество ортогональных несущих при COFDM-модуляции обеспечивается при большем частотном интервале между соседними несущими и, соответственно, повышается возможность увеличения скорости движения терминала, при котором прием информации может срываться из-за допплеровского смещения частот.

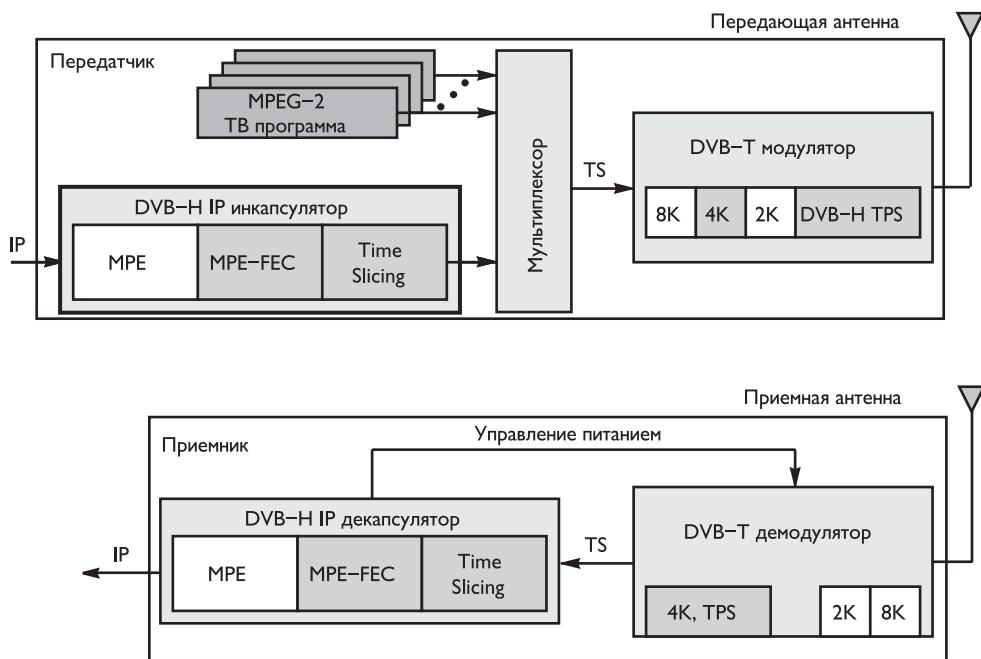


Рис. 22.1. Структурная схема системы DVB-H

С другой стороны, чем меньше несущих, тем короче период, выделенный для передачи каждого COFDM-символа и, соответственно, короче защитный интервал. Сокращение же защитного интервала снижает возможности отстройки от многолучевого приема, т. е. уменьшает допустимый радиус соты. Для сетей DVB-T, рассчитанных в основном на стационарный прием, значительно более важным фактором является зона охвата. Что же касается сетей DVB-H, то там большую значимость приобретает возможность приема сигнала на скорости, а зона охвата в сильной мере ограничивается уровнем сигнала на входе тюнера.

Вторым дополнением на физическом уровне стала возможность более глубокого перемежения данных в режимах 4K и 2K. Канальное кодирование DVB-T предусматривает перемежение данных внутри одного COFDM-символа. Оно в основном предназначено для компенсации селективных замирований несущих при многолучевом приеме. В то же время мобильные терминалы с большей вероятностью могут оказаться в зоне действия широкополосных импульсных помех. И допплеровское смещение частотного спектра при приеме информации при быстром движении терминала также приводит к искажениям сигнала. Поэтому в стандартах мобильного вещания на базе COFDM (DMB, ISDB-T) для борьбы с последствиями длительных помех в цикл канального кодирования введено перемежение длинных серий данных, охватывающее десятки, а то и сотни OFDM-символов.

Очевидно, что борьба с последствиями искажений сигналов при приеме тем эффективнее, чем длиннее последовательность участвующих в перемежении данных. Однако восстановление длинных последовательностей в системе DVB-H

потребовало бы непрерывного приема, что невозможно при использовании энергосберегающего импульсного режима приема и противоречит требованию совместимости с системой DVB-T. Поэтому было выбрано компромиссное решение. Для режима модуляции 8K, наиболее актуального для DVB-T, в DVB-H сохранено перемежение битов в рамках одного символа, а в режимах 4K и 2K введена возможность временного перемежения: для режима 4K перемежение выполняется с глубиной в 2 COFDM-символа, а для режима 2K — с глубиной в 4 COFDM символа. В таких опциях совместная передача трансляций DVB-H и DVB-T невозможна.

Одно из основных отличий DVB-H от DVB-T, обеспечивающих дополнительную помехозащиту, заключается в том, что вся информация передается в форме IP-дейтаграмм, инкапсулируемых в транспортные пакеты MPEG-2 с использованием метода многопротокольной инкапсуляции (MPE — Multi Protocol Encapsulation), единственно пригодного для передачи потоковых услуг (рис. 22.2). IP-пакеты инкапсулируются в MPE-секции, а те, в свою очередь, — в транспортные пакеты MPEG-2, переносящие элементарные потоки. Каждый IP-пакет занимает одну MPE-секцию. В одном пакете может передаваться множество MPE-секций, и, наоборот, одна секция может занимать несколько транспортных пакетов.

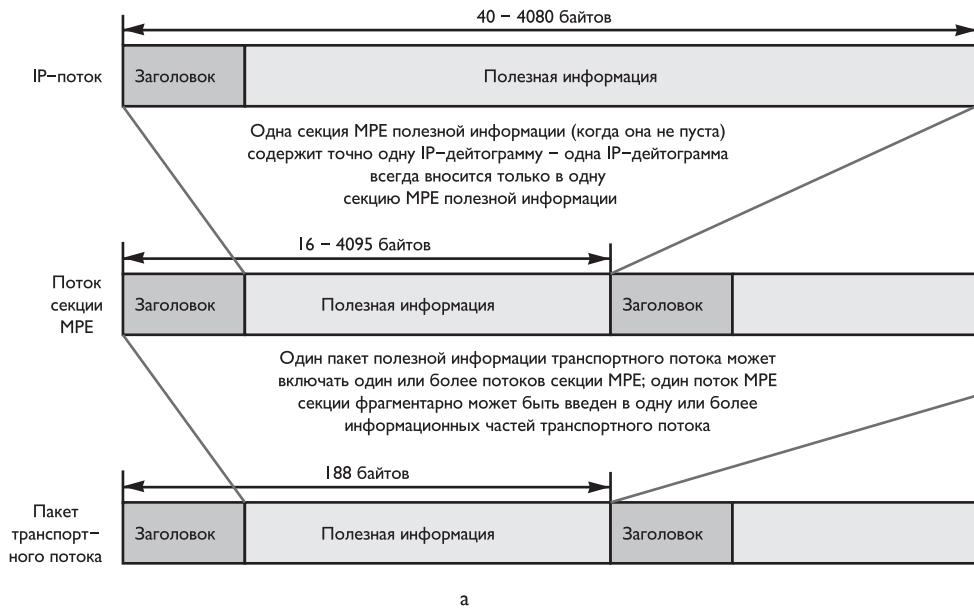
Данные, относящиеся к одной услуге, инкапсулируются в транспортные пакеты MPEG-2 с постоянным идентификационным номером PID. Использование такого стека в системе DVB-H обусловлено тремя причинами:

- ТВ-потоки могут компрессироваться не в формате MPEG-2, а в более эффективных форматах (например, в H.264/AVC), для которых процесс инкапсуляции компрессированных аудио и видео может быть реализован через IP/MPE;
- DVB-H потенциально рассматривается как составная часть гибридной системы доставки мультимедийных услуг (IPDC), и в связи с этим понятие элементарного потока в DVB-H определяется иначе, чем в стандарте MPEG-2 (в DVB-H — это просто поток, передаваемый в пакетах с одним PID-ом);
- снята жесткая корреляция элементарного потока с данными определенного типа — в одном элементарном потоке могут передаваться все данные, относящиеся к определенной ТВ-программе или даже к нескольким программам; в этом случае потоки разных ТВ-программ будут передаваться в дейтаграммах с разным мультикастовым IP адресом и заключаться в MPE-секции с разными MAC-адресами.

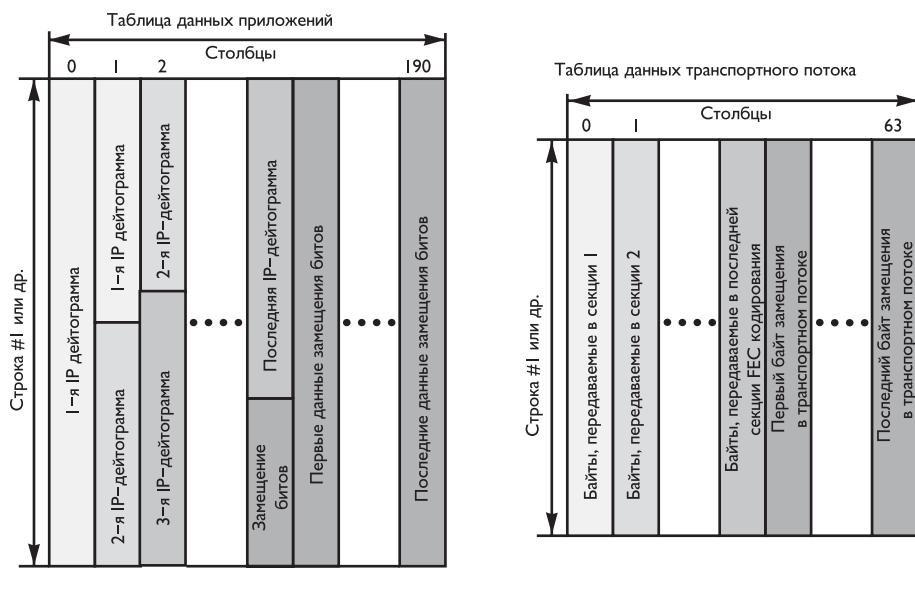
Остальные механизмы внешнего и внутреннего канального кодирования, используемые в DVB-T, без изменения перенесены в DVB-H.

Еще одно дополнение касается транспортной сигнализации (TPS — Transmission Parameter Signalling), в которую добавлены два бита, индицирующие наличие в потоке услуг, передаваемых в формате DVB-H, а также наличие дополнительной кодозащиты, реализуемой на базе IP-дейтаграмм.

Предусмотрена также возможность использования полосы частот 5 МГц при условии, что она выделяется вне вещательного диапазона. Ее применение предусматривается при развертывании сетей DVB-H в США в L-диапазоне (1670–1675 МГц).



а



б

в

Рис. 22.2. Схема инкапсуляции IP-пакетов в транспортные потоки MPEG-2 (а), принцип кодирования IP-дейтаграмм (б, в)

Возможности MPE-секций используются для организации временного разделения услуг внутри транспортного потока (англ. название Time Sliced Multi Protocol Encapsulation). Транспортные пакеты MPEG-2 с одним идентификационным номером (PID) — по существу, определенная услуга или пакет услуг,

передаются не с постоянной скоростью, а импульсами (bursts), позволяющими быстро загрузить в приемное устройство порцию информации, а затем отключить терминал от приема на период воспроизведения загруженной информации.

Экран карманного терминала обычно воспроизводит картинку с разрешением около четверти телевизионного, и для передачи ему ТВ-программы в зависимости от разрешения, системы компрессии и динамики изображения потребуется поток информации в 100–300 кбитов/с. Таким образом, при скорости мультиплексированного потока в 10 Мбитов/с возможно в нем разместить 50 каналов со средней скоростью передачи в 200 кбитов/с. Если на 100 мс предоставить весь канал для передачи одной программы, то за это время в приемник будет загружен 1 Мб информации, воспроизводимой за 5 секунд. При этом тюнер должен быть активен чуть больше 100 мс, а остальное время он может находиться в режиме ожидания. Такая схема передачи позволяет экономить ресурс аккумуляторных батарей на 90–95%.

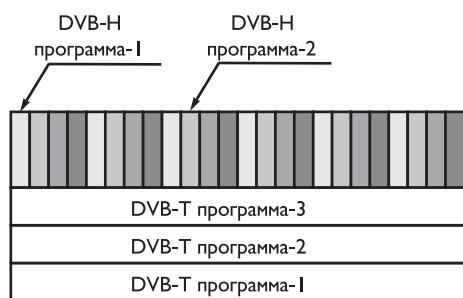


Рис. 22.3. Структура кадра в системе DVB-H

иваться на новую частоту приема. Для распознавания нового сигнала используется упомянутая TPS-сигнализация. Разумеется, для исключения сбоев приема во время перехода трансляции в разных сотах должны быть синхронизированы.

Временное разделение потока (TDM) реализуется на базе MPE-секций. В заголовке каждой секции, передаваемой в слоте, указывается продолжительность текущего слота и начало передачи следующего. Многократное дублирование практически сводит на нет вероятность потери информации о режиме передачи слотов.

Импульсная передача данных в системе DVB-H не препятствует приему передаваемых сигналов приемниками системы DVB-T.

В DVB-H помехозащитное кодирование длинных последовательностей накладывается на всю последовательность IP-дейтаграмм, передаваемых в одном слоте, т. е. на максимальный объем данных. Это кодирование введено в качестве опции и выполняется кодом Рида–Соломона.

IP-дейтаграммы помещаются в таблицу данных приложений (Application Data Table — рис. 22.26), в которой последовательно заполняются столбцы, начиная с левого верхнего угла. Высота столбцов может составлять от 1 до 1024 байтов в зависимости длины дейтаграмм, а их количество всегда одинаково — 191. Если в таблице остается пустое место, то оно заполняется холстыми байтами.

Затем каждый ряд таблицы кодируется кодом Рида–Соломона (255, 191), в результате чего формируется 64 байта, заносимых в соответствующий ряд FEC

Следует отметить, что импульсы могут занимать не всю транспортную полосу, что позволяет обеспечить совместную передачу данных в системах DVB-H и DVB-T (рис. 22.3). В этом случае скорость передачи в каждом слоте будет меньше. Максимальный объем информации, допускаемый для передачи в одном временном слоте, ограничен 2 Мб.

Во время паузы приемник может сканировать спектр в поисках более мощного сигнала, передающего аналогичную услугу, и при его обнаружении перестраиваться на новую частоту приема. Для распознавания нового сигнала используется упомянутая TPS-сигнализация. Разумеется, для исключения сбоев приема во время перехода трансляции в разных сотах должны быть синхронизированы.



таблицы (рис. 22.2в). Затем обе таблицы инкапсулируются в пакеты MPEG-2 TS и транслируются в одном слоте. Вначале — информационная часть, а затем — контрольная. Причем байты сформированной таблицы при инкапсуляции считаются не рядами, как формируются, а колонками. Так создается виртуальное перемежение контрольных байтов, требующее минимальных ресурсов приемника для восстановления их последовательности. Этот метод помехозащитного кодирования получил название MPE-FEC.

MPE-FEC-декодирование проводится на базе модели канала со стиранием, включающем два этапа:

- на первом этапе с помощью циклического (CRC-32) кода выявляются искаженные пакеты и локализуются пораженные части потока;
- на втором этапе выполняется восстановление пораженных пакетов, что при предварительной локализации искажений происходит более эффективно.

При использовании MPE-FEC незащищенными остаются служебные таблицы. Но с учетом того что их содержание в большинстве случаев довольно статично, после нескольких циклов передачи приемнику удается получить нужную информацию даже в сложных условиях.

Особенности канального уровня формата DVB-H не препятствуют приему трансляций приемниками стандарта DVB-T. Они просто будут принимать их неоптимальным образом, различными будут и условия приема трансляций DVB-T и DVB-H. Так для достижения передатчиками DVB-H и DVB-T одинаковой зоны охвата мощность первого должна быть на 20 дБ больше. В то же время требуемый для устойчивого приема уровень несущая/шум в DVB-H в среднем на 30% ниже, а максимально возможная скорость движения приемника — на 40% выше.

Транспортная система DVB-H рассматривается в DVB как потенциальная часть более общей системы передачи мультисервисных услуг, в которой используется транспортный IP-протокол, а в качестве приемников — карманные терминалы с небольшой мощностью аккумуляторов и ограниченными компьютерными возможностями. Это платформа для создания гибридных сетей, в которых односторонние вещательные каналы передачи предоставляются системой DVB-H, а двунаправленные каналы — системами сотовой телефонии (IPDC — IP-Data Cast).

Поэтому в рамках платформы IPDC специфицируется необходимая сигнализация в потоках DVB-H, передаваемая в таблицах SI/PSI, порядок формирования электронной программы передач ESG (Electronic Service Guide) для мультисервисных услуг, стеки протоколов передачи, используемые на верхних уровнях модели ISO/OSI, форматы кодирования видео и аудио, иерархическая система защиты услуг от несанкционированного доступа, включающая контроль за воспроизведением, и даже схемы предоставления платных услуг. Эти механизмы в рамках платформы должны быть едиными и поддерживаться комплексами для создания и администрирования приложений, аппаратурой формирования транспортных потоков и приемными терминалами.

В табл. 22.1 приведены основные параметры стандарта DVB-H в трех режимах COFDM-модуляции.

В табл. 22.2 приведены параметры защитных интервалов и полной символьной длительности в системе DVB-H.

Таблица 22.1. Параметры составляющих стандарта DVB-H

Параметр	Режим 2К	Режим 4К	Режим 8К
Число активных несущих K	1705	3409	6817
Число информационных несущих	1512	3024	6048
Длительность элементарного периода T , мс	7/64		
Полезная символьная часть T_U , мс	224	448	896
Разнос между несущими $1/T_U$, Гц	4464	2232	1116
Разнос между крайними несущими, $(K - 1)/T_U$, МГц	7,61		

Таблица 22.2. Параметры защитных интервалов и длительности символов в системе DVB-H

Параметр	Режим											
	2К	4К	8К	2К	4К	8К	2К	4К	8К	2К	4К	8К
Длительность полезной части символа T_U , мкс	224	448	896	224	448	896	224	448	896	224	448	896
Заданный интервал T_G/T_U	1/4			1/8			1/16			1/32		
Длительность защитного интервала T_G , мс	56	112	224	28	56	112	14	28	56	7	14	28
Длительность символа $T_S = T_G + T_U$, мс	280	560	1120	252	504	1008	238	476	952	231	462	924

В табл. 22.3 представлены расчетные значения цифровых потоков (в Мбитах/с) для разных форматов модуляции.

Таблица 22.3. Расчетные значения цифровых потоков (Мбиты/с), реализуемых системой DVB-H

Модуляция	Скорость кодирования	Заданный интервал			
		1/4	1/8	1/16	1/32
QPSK	1/2	3,74	4,15	4,39	4,52
	2/3	4,98	5,53	5,86	6,03
	3/4	5,6	6,22	6,59	6,79
	5/6	6,22	6,92	7,32	7,54
	7/8	6,53	7,26	7,69	7,92
16-QAM	1/2	7,46	8,3	8,78	9,05
	2/3	9,95	11,06	11,71	12,07
	3/4	11,2	12,44	13,17	13,58
	5/6	12,44	13,82	14,64	15,08
	7/8	13,07	14,51	15,37	15,83
64-QAM	1/2	11,2	12,44	13,17	13,58
	2/3	14,93	16,59	17,57	18,1
	3/4	16,79	18,66	19,76	20,36
	5/6	18,66	20,74	21,95	22,62
	7/8	19,6	21,77	23,06	23,75

Наиболее же важным моментом при построении любой сети вещания является расчет зоны покрытия, который сводится к двум операциям:



- определение минимальной напряженности электрического поля E_{\min} , при которой еще возможен прием сигнала с заданной вероятностью ошибки (BER);
- оценка напряженности поля в любой точке приема при известной эффективно излучаемой мощности передатчика $P_{\text{ЭИМ}}$, высоте подъема передающей антенны, имеющейся ландшафте, времени года, погодных условиях и т. п.

Минимальная напряженность поля для дециметрового диапазона зависит от коэффициента шума мобильного терминала F , коэффициента удаления приемной антенны G_a , несущей частоты f и требуемого отношения сигнал/шум C/N :

$$E_{\min} [\text{дБмкВ/м}] = -24 [\text{дБ}] + 20 \lg(f [\text{МГц}]) + C/N [\text{дБ}] - G_a [\text{дБ}]. \quad (22.1)$$

Коэффициент усиления приемной антенны мобильного терминала имеет типовые значения $-5 \dots -10$ дБ. Для расчета зоны покрытия различают также портативный и мобильный прием, внутри- или внешнедомовой прием, допустимую скорость перемещения мобильного терминала, вероятность приема и т. п.

В силу этого различают четыре класса приема:

- класс А — внешнедомовой портативный прием;
- класс В — портативный прием внутри помещения;
- класс С — внешнедомовой мобильный прием;
- класс D — внутридомовой мобильный прием.

В табл. 22.4–22.7 приведены только расчетные значения минимально допустимой медианной напряженности поля E_{med} для коэффициента шума мобильного терминала $F = 6$ дБ.

Таблица 22.4. Класс А. Городские условия приема, 3-я категория приемного терминала

Название параметра	500 МГц			800 МГц		
Минимальное C/N , дБ	2 14 26			2 14 26		
Усиление антенны, G_a , дБ	-12			-7		
Минимальная эквивалентная напряженность поля, E_{\min} , дБмкВ/м	44	56	68	43	55	67
Минимальная медианская эквивалентная напряженность поля, E_{med} , при высоте антенны $h = 10$ м, вероятности приема 50% по времени, дБмкВ/м:	69	81	93	70	82	94
– для вероятности приема 70% по положению	75	87	99	76	88	100
– для вероятности приема 95% по положению						

22.2. Система мобильного телевизионного вещания MediaFLO

Наиболее близкой к DVB-H по своей идеологии является американская система MediaFLO — это корпоративный формат фирмы Qualcomm, разработанный для поставки услуг телерадиовещания на переносные приемники и использующий модуляцию COFDM в области частоты 700 МГц в Соединенных Штатах [9.15–9.18].

Таблица 22.5. Класс В. Городские условия приема, 3-я категория приемного терминала

Название параметра	500 МГц			800 МГц		
	2	14	26	2	14	26
Минимальное C/N , дБ	–12			–7		
Усиление антенны, G_a , дБ						
Минимальная эквивалентная напряженность поля, E_{min} , дБмкВ/м	44	56	68	43	55	67
Минимальная медианная эквивалентная напряженность поля, E_{med} , при высоте антенны $h = 10$ м, вероятности приема 50% по времени, дБмкВ/м:	81	93	105	82	94	106
– для вероятности приема 70% по положению	91	103	115	92	104	116
– для вероятности приема 95% по положению						

Таблица 22.6. Класс С. Сельские условия приема, 1-я категория приемного терминала

Название параметра	500 МГц			800 МГц		
	2	14	26	2	14	26
Минимальное C/N , дБ	–2			–1		
Усиление антенны, G_a , дБ						
Минимальная эквивалентная напряженность поля, E_{min} , дБмкВ/м	34	46	58	37	49	61
Минимальная медианная эквивалентная напряженность поля, E_{med} , при высоте антенны $h = 10$ м, вероятности приема 50% по времени, дБмкВ/м:	52	64	76	69	81	93
– для вероятности приема 70% по положению	58	70	82	75	87	99
– для вероятности приема 95% по положению						

Таблица 22.7. Класс D. Сельские условия приема, 3-я категория приемного терминала

Название параметра	500 МГц			800 МГц		
	2	14	26	2	14	26
Минимальное C/N , дБ	–12			–7		
Усиление антенны, G_a , дБ						
Минимальная эквивалентная напряженность поля, E_{min} , дБмкВ/м	44	56	68	43	55	67
Минимальная медианная эквивалентная напряженность поля, E_{med} , при высоте антенны $h = 10$ м, вероятности приема 50% по времени, дБмкВ/м:	69	81	93	70	82	94
– для вероятности приема 70% по положению	75	87	99	76	88	100
– для вероятности приема 95% по положению						

Система MediaFLO, будучи схожей DVB-H, оптимизирована за счет снятия требований совместимости с DVB-T. При обработке информации используется режим 4K, турбокодирование и перемежение данных. В MediaFLO, как и в DVB-H, реализован режим энергосбережения за счет импульсной передачи информации. Кроме того, в MediaFLO применяется система мультиплексирования CDMA, позволяющая в одном канале передавать множество услуг, каждая из которых занимает часть COFDM мультиплексированных несущих. В связи с этим имеется возможность кодового сегментирования канала.

Полезным свойством MediaFLO, как и DVB-T, является способность объединения в одном мультиплексе сервисов, предполагающих разные зоны покрытия, т. е. сервисов с заложенными разными уровнями помехозащиты и схемами модуляции. Это в сочетании с одночастотным режимом работы передатчиков позволяет

Таблица 22.8. Параметры системы MediaFlo для канала с полосой 6 МГц

Параметр	Величина
Общее количество поднесущих	4096
Количество постоянных поднесущих	96
Количество пилотных поднесущих	500
Число многоадресно передаваемых логических каналов	1 — 7
Используемая модуляция	QPSK или 16-QAM
Интервал между крайними поднесущими	5,55 МГц
Интервал между соседними поднесущими	1,355 кГц (5,55 МГц / 4096)
Длительность элементарного символа T	0,18 мкс (1/5,55 МГц)
Длительность активной части OFDM символа T_U	738,02 мкс
Длительность интервала окна T_{WGI}	3,03 мкс
Длительность защитного интервала T_{FGI}	92,25 мкс
Суммарная длительность OFDM символа $T_S = T_U + T_{WGI} + T_{FGI}$	833,33 мкс

ет объединять в одном канале индивидуальные или местные услуги, актуальные в небольшой зоне покрытия, и общенациональные программы для вещания на большие территории.

Формат разрабатывался специально для мобильных телефонов, поэтому разрешение ограничено параметрами QVGA (320×240 пикселов). Стандарт обеспечивает передачу 20 потоковых ТВ-программ, передаваемых со скоростью до 30 кадров в секунду, 10 стереофонических аудиопрограмм (формат НЕ-ААС), а также загрузку клипов общей продолжительностью до 800 минут в день в одном мультиплексе (в полосе 5 МГц). При работе в полосе 6 МГц число каналов может быть увеличено еще на 20% и составляет до 11 Мбитов/с.

MediaFLO — комплексная система, включающая разработку платформы для предоставления услуг, схему закрытия контента и контроля за его воспроизведением. Более того, MediaFlow задумана как составляющая более широкой системы передачи медиа — Media Distribution System (MDS), в состав которой войдут и сотовые сети третьего поколения EV-DO CDMA 2000. Очевидна аналогия MDS с европейской системой IPDC (IP DataCast), в которой для вещательных услуг предполагается использовать сети каналы DVB-H, а для интерактивного взаимодействия с абонентом и предоставления индивидуальных услуг — каналы UMTS.

Помимо поддержки трансляции потокового видео в MediaFLO добавлена функция фоновой передачи видеофрагментов, пересылаемых в периоды времени, когда трафик загружен минимально. Некоторые параметры системы MediaFLO приведены в табл. 22.8.

22.3. Система мобильного мультимедийного вещания T-DMB

Мультимедийная технология DMB (Digital Multimedia Broadcasting) основана на использовании радиовещательного цифрового стандарта DAB (Digital Audio Broadcasting). Эти технологии DAB и DMB, а также DAB-IP-представляются единым форумом WorldDMB, в который входят 130 членов из 40 стран мира [9.19–9.24].

Цифровой стандарт DAB был создан в 90-х годах прошлого века на основе технологии Eureka 147 и изначально был ориентирован на звуковое радиовещание на мобильные, в том числе автомобильные приемники.

Имеется две версии стандарта: наземная T-DAB (Terrestrial) и спутниковая S-DAB (Satellite). Наземная (эфирная) версия стандарта предназначена для доставки сигнала в III полосе ОВЧ диапазона (174–230 МГц) или L-полосе УВЧ-диапазона (1400–1900 МГц).

Общие принципы формирования каналов T-DAB заключаются в следующем.

Ширина полосы спектра установлена равной 1,536 МГц, используются мультиплексирование COFDM/QPSK, сверточное кодирование, а также частотное и временное перемежение данных с глубиной до 384 мс.

Структура T-DAB потока включает три составляющие:

- канал синхронизации SC (Synchronisation Channel), передающий метки времени, необходимые для синхронизации приемника;
- быстрый информационный канал FIC (Fast Information Channel) передачи служебной информации о структуре мультиплексированного потока и параметрах кодирования составляющих потока; помимо этого в FIC может передаваться SI (Service Information), служебная информация для систем условного доступа (CA), и FIDC (Fast Information Data Channel); для обеспечения быстрой и в то же время оперативной передачи информации MCI FIC передается без временного перемежения, но с высоким уровнем помехоустойчивой защиты;
- основной сервисный канал MSC (Main Service Channel), в котором непосредственно передается информация; он может делиться на подканалы, каждый из которых отдается под определенный тип потока.

Для MSC предусмотрено два режима передачи — пакетный или потоковый. Если подканалу задан потоковый режим, то в нем обеспечивается непрерывная, фиксированная по скорости передача одного потока. В пакетном режиме один подканал может использоваться для передачи нескольких услуг.

В качестве помехоустойчивого кодирования в MSC используется сверточный кодер, а также временное и частотное перемежение данных. Системы, использующие транспортный формат T-DAB, применяют разные схемы передачи информации в канале MSC. Кроме того, они различаются «надстройкой» на уровне приложений, реализуемой в конкретном проекте.

В Англии развертывание цифрового радиовещания в стандарте T-DAB началось в 1994 году, а в 1999 году такой стандарт радиовещания был реализован компанией BBC.

На начало 2007 года в Англии построена одночастотная сеть (Single Frequency Network, SFN), включающая 100 передатчиков и охватывающая 90% территории страны.

В октябре 2004 года южнокорейскими компаниями Samsung и LG Electronics были продемонстрированы терминалы T-DAB, позволяющие принимать различных виды данных. Эта технология и получила название T-DMB. Для сжатия видеосигнала может быть использована любая технически подходящая технология: H.264, Windows Media и т. п. Цифровые аудио- и видеосигналы могут быть мультиплексированы в один мультимедийный высокоскоростной транспортный поток. Для этого можно использовать оборудование DVB, формирующее транспортный поток MPEG-2. При необходимости данные могут быть упакованы в IP-пакеты. Эти идеи и составляют содержание технологии T-DMB.

Южнокорейская корпорация TU Media 1 мая 2005 года начала трансляцию 7 видеоканалов и 20 аудиоканалов. На мобильные абонентские терминалы стандарта CDMA доставка сигнала с мультимедийными данными осуществлялась главным образом со спутника в S-диапазоне, частично — путем ретрансляции спутникового сигнала Ки-диапазона в S-диапазон. По этой причине терминалы получили название S-DMB (Satellite). Сегодня эти устройства находят использование главным образом на автомобильном рынке.

1 декабря 2005 в Сеуле начато предоставление мультимедийного вещания с использованием наземной технологии T-DMB на коммерческой основе. Услуги начали предоставлять сразу 6 операторов мобильной связи, работающие в стандарте CDMA. Каждый из операторов предоставил своим абонентам 4–5 сервисов в канале T-DAB. В общей сложности всеми операторами было предоставлено 7 телеканалов, 13 аудиоканалов и 9 каналов передачи данных для загрузки web-страниц.

На начало 2006 года в мире значилось 18 пилотных проектов по внедрению технологии DMB на основе существующей наземной вещательной инфраструктуры T-DAB. В Европе основные проекты были развернуты в Германии, Франции, Норвегии, Голландии, Финляндии, Англии и Италии. Лидером в этом процессе являлась Германия, где в июне 2006 года заработало 39 передатчиков T-DMB в 12 городах.

В системе T-DMB радиопрограммы и данные передаются стандартным для DAB способом, а для трансляции ТВ используется отдельный потоковый канал MSC, в котором реализована более сложная система инкапсуляции и защиты. Видеопотоки компрессируются в формате H.264/AVC, для аудио используется система компрессии HE-AAC. Сжатые потоки сначала инкапсулируются в транспортные пакеты MPEG-2 и защищаются от помех кодом Рида–Соломона, а затем через перемежатель вводятся в DAB-канал, где дополнительно защищаются сверточным кодом и подвергаются перемежению. Промежуточная инкапсуляция в пакеты MPEG-2 позволяет использовать проверенные механизмы синхронизации разных элементарных потоков, относящихся к одной ТВ-программе, и отработанную схему наложения кода Рида–Соломона.

Европейский профиль T-DMB отличается от корейского в основном отсутствием условного доступа и ограничений на просмотр контента. Что касается британского проекта Movio System, то там подканал MSC используется в пакетном режиме, а компрессированные телевизионные потоки перед размещением в подканале инкапсулируются в IP-пакеты. Особенностью этой системы также

является использование видеокомпресии VC1. Ее выбор, очевидно, обусловлен партнерством с Microsoft, который поставил для проекта не только систему компрессии, но и систему контроля за воспроизведением Windows Media DRM.

Можно встретить довольно много материалов, посвященных сравнению T-DMB и DVB-H. Приверженцы T-DMB говорят о дешевизне развертывания этих сетей на базе существующей транспортной структуры DAB, о возможности более широкого разнесения сот в одночастотных сетях и о более быстром переключении с канала на канал. К преимуществам формата относится и наличие в DAB информационного канала, позволяющего демультиплексору извлекать из принимаемого потока только необходимые составляющие, что упрощает прием. Отмечается также широкий ассортимент DMB-приемников, разработанных для Кореи.

Сторонники DVB-H считают, что сети T-DAB, равно как и частоты, отведенны под эти радиоуслуги, получить затруднительно, что основной ассортимент приемников предназначен для спутниковой версии S-DMB, а эфирные интегрированы с CDMA-телефонами, не актуальными для Европы, где доминируют форматы семейства GSM. Считается существенным преимущество импульсного режима передачи, сберегающего энергию приемников DVB-H. Еще указывается на возможность более широкого, чем в T-DMB, выбора схем модуляции и защиты и, как следствие, более низкие требования к параметру C/N .

Эксперты считают, что эти технологии могут занять разные ниши. Для передачи радио- или небольшого количества ТВ-программ лучше приспособлены каналы DAB, в то время как для трансляции крупных ТВ-пакетов предпочтительнее сети DVB-H.

На рис. 22.4 приведена архитектура видеообслуживания потребителей в системах T-DMB и S-DMB, содержащая пакетный режим стандарта DAB, внешний кодер RS (204, 188) и перемежитель, тракт формирования пакетов MPEG-2 TS, тракт введения синхронизирующих сигналов, область компрессии, содержащую, возможно, детектор объектов, информацию о передаваемой сцене и форме объектов, а также систему композиции и представления аудиовидеосцены.

На рис. 22.5 дана достаточно подробная структурная схема формирования OFDM-сигналов системы DMB, состоящая из двух основных частей: видеомультиплексора и формирователя сигналов стандарта DAB.

Видеомультиплексор объединяет данные формирователей потоков кодированной видео-, аудио- и другой дополнительной информации о передаваемых объектах, а также реализует повышение эффективности кодирования за счет использования кодера Рида–Соломона и блока перемежения данных.

Сформированная таким образом информация подается на стандартный формирователь сигналов стандарта DAB, обеспечивающий опционально шифрование информации, статистическое и сверточное кодирование, временное перемежение битов, мультиплексирование с рядом дополнительных данных и формирование OFDM-сигналов.

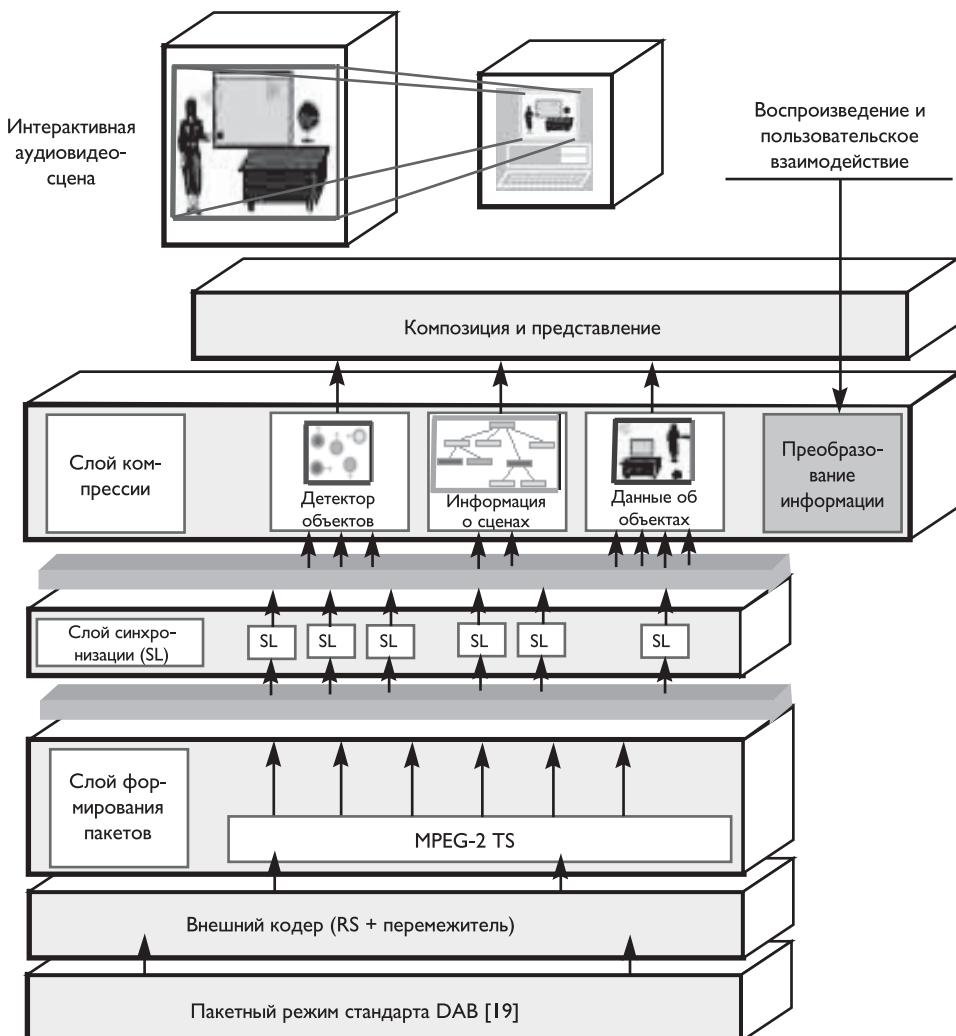


Рис. 22.4. Архитектура видеообслуживания потребителей в системе DMB

22.4. Аудиовизуальная информационная система реального времени РАВИС

Работу подвижных систем связи в ОВЧ-диапазоне в городских условиях с плотной застройкой, в лесистой и горной местностях, в водной акватории определяют несколько характеристик сигнала, к которым относятся: затухание вдоль трассы, распределение амплитуд, многолучевость и спектр задержек, размытие спектра и допплеровский сдвиг.

Прием сигнала в движущемся транспорте, особенно на магистралях, где скорость движения достигает 100 км/час и более, возникает размытие спектра и допплеровский сдвиг радиосигналов.

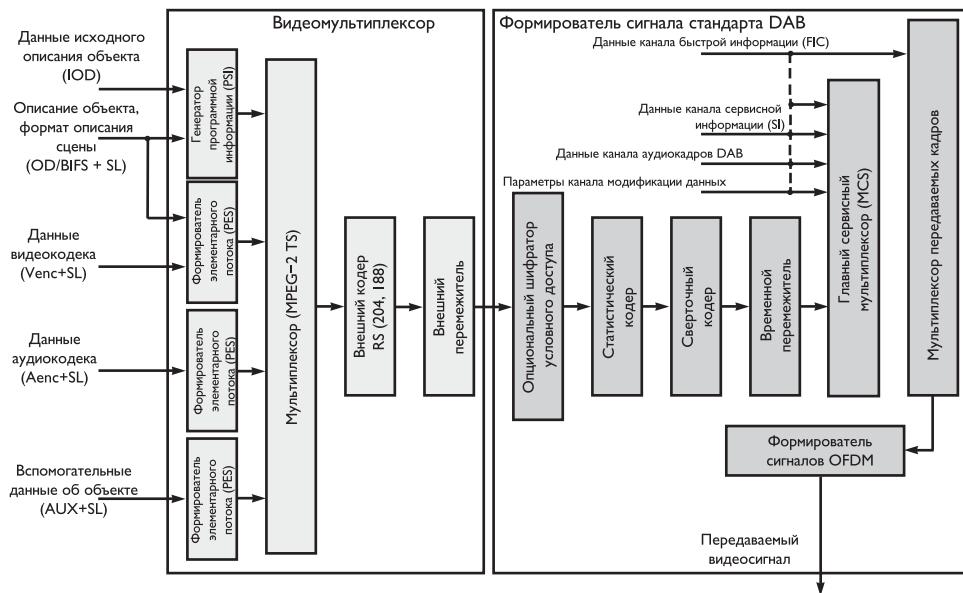


Рис. 22.5. Структурная схема формирования OFDM-сигналов системы DMB

В диапазоне ОВЧ расчетный интегральный масштаб частотной корреляции для многих городских районов современной застройки, согласующийся с экспериментальными данными, составляет 0,8–0,9 МГц.

В связи с этим условия приема сигналов в движущемся транспорте будут комфортными при современной городской застройке, в лесистой и горной местностях, в водной акватории при использовании радиоканала с полосой 0,2–0,25 МГц.

Диапазон частот и полоса спектра радиоканала, используемые для вещания системы РАВИС, позволяет локализовать вещание, т. е. на одной и той же частоте в разных городах передавать различные программы, реализовать одночастотные сети вещания вдоль шоссейных и железнодорожных путей и др. При этом радиус покрытия передатчиком остается достаточно велик для обеспечения приема в отдаленных пунктах, где другим способом невозможно осуществить вещание. При меньших уровнях излучаемого сигнала, чем при FM-вещании, радиус уверенного приема сигнала системы РАВИС может превышать стандартные для FM-диапазона значения до 80–90 км.

Система РАВИС включает как традиционные для радиовещания, так и совершенно новые сервисы:

- высококачественный стереофонический и многоканальный звук;
- видео со стереофоническим звуковым сопровождением на нескольких языках;
- текстовые сообщения, как связанные, так и не связанные со звуковыми или видеопрограммами;
- электронная программа передач;
- передача статических изображений и слайд-шоу;
- локальная дорожная информация, информация о погоде и пр.;

- передача данных (широковещательная, групповая, индивидуальная);
- передача данных с повышенной надежностью;
- организация систем оповещения ГО и ЧС;
- служба service following, совместимая с другими системами вещания, как аналоговыми (FM RDS, AMSS), так и цифровыми (T-DAB, DRM).

Внедрение аудиовизуальной информационной системы РАВИС связано с решением современных тенденций цифровизации телекоммуникационных систем, в частности, сети мобильного радиовещания взамен аналоговых систем УКВ-ЧМ- и FM-вещания в выделенных для этих целей в России диапазонах частот 66–74 и 87,5–108 МГц.

Необходимость и эффективность внедрения цифровых систем, определенные в «Концепции развития телерадиовещания в Российской Федерации на 2008–2015 годы» (Одобрена распоряжением Правительства 29.11.2007 г. № 1700-р) и в ФЦП «Развитие телерадиовещания на 2009–2015 годы» (Утверждена Председателем Правительства 21.09.2009 г. № 1349-р), связаны с требованиями:

- существенного повышения эффективности использования частотного ресурса;
- создания эффективных эфирных систем оповещения при чрезвычайных ситуациях;
- реализации одночастотных сетей мобильного вещания вдоль трассейных и железнодорожных путей в России;
- существенного снижения энергопотребления радиопередающими средствами мобильного мультимедийного вещания;
- достижения на современном этапе развития видео- и аудио- информационных систем нового уровня качества воспроизведения изображений и звука;
- удовлетворения постоянно возрастающих запросов на частотные присвоения систем связи без пересмотра частотных планов;
- развития цифровых систем передачи видео- и звуковых данных при существенном увеличении объема передаваемой сопутствующей информации;
- создания высоконадежных систем передачи информации с эффективной системой защиты от несанкционированного приема;
- интегрирования России в общемировую систему телекоммуникаций с учетом тенденций перехода к мультимедийным системам многопрограммного телерадиовещания в глобальном масштабе.

Приведем основные конкурентные преимущества системы РАВИС для всех категорий потребителей.

Для слушателей открываются такие возможности, как:

- прием высококачественного звукового вещания (стереофонического и многоканального);
- получение новых мультимедийных сервисов (видео, текст, дополнительные данные, электронная программа передач и т. д.);
- простая настройка на станцию по названию, жанру и пр.

Для производителей внедрение системы РАВИС означает:

- массовую замену старых аналоговых приемников;
- модернизацию передающих средств при сохранении инфраструктуры;
- общий рост потенциала рынка передающих и приемных средств.

Для вещателей применение системы РАВИС обеспечивает:

- снижение потребляемой мощности, приходящейся на одну программу;
- расширение зоны охвата вещанием;
- увеличение количества и качества программ, спектра предлагаемых потребителям сервисов;
- возможность адаптации к потребностям как больших, так и малых вещателей.

Для регуляторных органов использование системы РАВИС означает:

- повышение эффективности использования спектра;
- возможность сохранения частотных распределений, простых процедур лицензирования;
- координацию в рамках международно признанной системы.

Аналоги систем, реализующих возможности системы РАВИС, в мировой практике отсутствуют.

Важнейшим стимулом реализации системы РАВИС является утверждение отечественного стандарта ГОСТ Р 54309-2011 [9.25], в котором определены параметры формирования кадровой структуры, канального кодирования и модуляции для системы цифрового наземного узкополосного радиовещания в ОВЧ диапазоне.

Система РАВИС признана на международном уровне. По результатам исследований были разработаны 8 вкладов России в Международный союз электросвязи и 9 вкладов в Комитет по электронным средствам связи (ЕСС) Европейской конференции администраций почт и электросвязи (СЕРТ) [9.26–9.42].

Первое в мировой практике предложение по реализации подобной системы было изложено в патенте Российской Федерации [9.43], в котором в соответствии с описанием предусматривалась возможность трансляции информационного телевидения, содержащего динамические цветные изображения и стереозвуковое сопровождение. Результаты создания модели системы и ее натурных испытаний впервые были изложены в журнале «Broadcasting. Телевидение и радиовещание» в 2005 г. [9.44].

На основании вкладов России были разработаны международные документы [9.45–9.51]. Результаты разработки системы РАВИС изложены в [9.52–9.58].

Система РАВИС позволяет в короткие сроки развернуть единую национальную систему мобильного телевещания. Низкая стоимость строительства инфраструктуры, возможность использования существующих радиочастот позволит обеспечить мобильным телевизионным вещанием практически всю территорию России.

Основное применение системы РАВИС — обеспечение телерадиовещания в транспорте. При поддержке государства и, прежде всего, государственных телерадиокомпаний, в короткие сроки телевещательные компании могут получить дополнительную телеаудиторию.

Система РАВИС имеет три канала — канал основного сервиса (КОС), низкоскоростной канал (НКС) передачи сигналов звукового оповещения и надежный канал данных (НКД) передачи сигналов текстового оповещения. В состав

аппаратуры входят три основные составляющие: кодер источника, передатчик и приемник.

Кодер источника

В зависимости от реализуемых функций кодера источника на его входы поступают множество потоков аудиоданных, а также данные звукового и текстового оповещения (рис. 22.6а), либо вместо потоков аудиоданных — видеинформация с несколькими каналами звукового сопровождения (рис. 22.6б). В дополнение к этим основным данным возможна передача сопроводительной дополнительной информации, например титров, прогноза погоды, аналога RDS-данных при FM-вещании и т. п. (на рис. 22.6 не показано).

Аудиоданные кодируются по стандарту MPEG-4 НЕ-ААС, видеоданные по стандарту H.264 / MPEG-4 AVC. Количество потоков в каждом из вариантов определяется возможной скоростью передачи информации через канал связи и требуемым качеством сервиса для каждого из элементарных потоков (аудио или видео).

Передатчик

Передатчик системы РАВИС построен по общепринятой в настоящее время схеме COFDM-модуляции, сочетающей высокую спектральную эффективность с хорошей помехозащищенностью как от гауссовского шума, так и от других помех, таких как многолучевое распространение сигнала и эффекта Допплера.

Стандарт РАВИС [9.25] устанавливает основные принципы построения системы для цифрового наземного узкополосного вещания РАВИС и определяет процессы формирования кадровой структуры, методов канального кодирования, мультиплексирования и модуляции системы.

Одним из ключевых условий при разработке системы РАВИС было сохранение частотных распределений, т. е. полоса радиоканала в системе РАВИС должна укладываться в полосу радиоканала аналогового ЧМ-вещания. Поэтому максимальная ширина полосы радиоканала РАВИС составляет 250 кГц. Предусмотрены также более узкополосные варианты 200 и 100 кГц. Отличительная особенность системы — интервал между несущими не зависит от ширины полосы радиоканала, при изменении полосы меняется количество несущих.

Кроме ширины полосы канала предусмотрено несколько возможных значений параметров канального кодирования и модуляции системы РАВИС, что позволяет организовывать вещание в различных условиях окружающей среды и помеховой обстановки, обеспечивая вещателю выбор между зоной покрытия и скоростью передачи данных в системе. Эти параметры позволяют также организовывать вещание в одночастотных сетях, например вдоль автомобильных трасс.

В системе предусмотрена возможность формирования помимо логического канала основного сервиса (КОС) двух дополнительных каналов — канала низкоскоростных данных с повышенной надежностью передачи (НСК, пропускная способность около 12 кбитов/с) и канала высоконадежной передачи данных (НКД, пропускная способность около 5 кбитов/с). Эти логические каналы могут использоваться, например, в системах оповещения о чрезвычайных ситуациях и т. д.

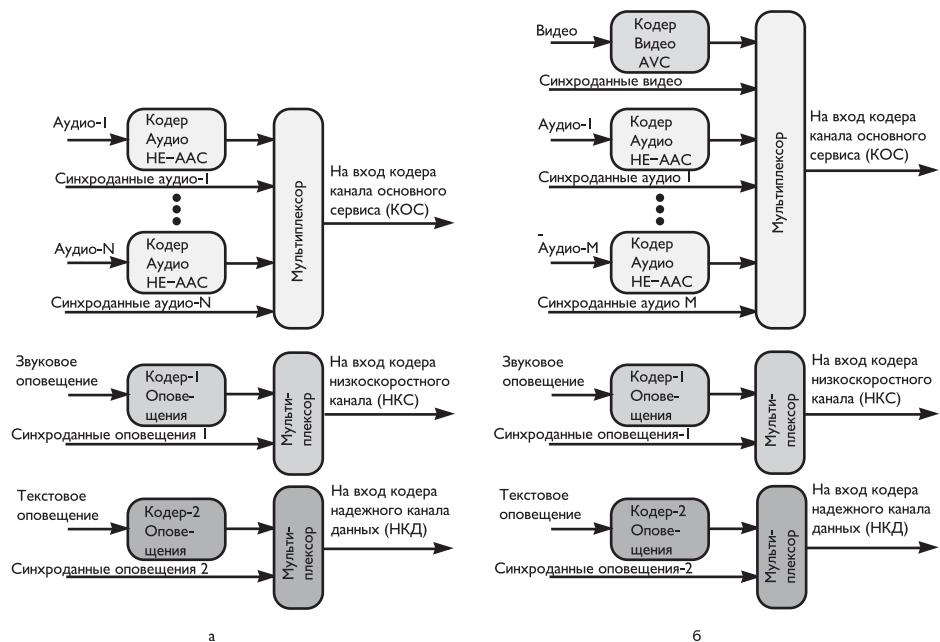


Рис. 22.6. Упрощенные структурные схемы кодера источника системы РАВИС

Данные в логических каналах передаются порциями — кадрами данных (КД). Размер кадра данных в логическом канале КОС зависит от полосы радиоканала, скорости канального кодирования и присутствия или отсутствия дополнительных каналов данных. Допустимые скорости канального кодирования — $1/2, 2/3, 3/4$; допустимые типы модуляции КОС — QPSK, 16-QAM, 64-QAM. Скорости потоков данных в одном радиоканале для всех комбинаций модуляционного созвездия и скорости помехоустойчивого кодирования приведены в табл. 22.9 (скорость потока данных в канале КОС при отсутствии каналов НСК и НКД).

Таблица 22.9. Скорости цифровых потоков в одном радиоканале

Тип созвездия	Скорость кода	Скорость потока данных, кбитов/с		
		Канал 100 кГц	Канал 200 кГц	Канал 250 кГц
QPSK	1/2	80	160	200
	2/3	100	210	270
	3/4	120	240	300
16-QAM	1/2	150	320	400
	2/3	210	420	530
	3/4	230	470	600
64-QAM	1/2	230	470	600
	2/3	310	630	800
	3/4	350	710	900

Размер кадра данных в логическом канале НСК — 656 битов, скорость канального кодирования — $1/2$, тип модуляции — QPSK. Данные канала передают на 32 информационных несущих. Размер кадра данных в логическом канале

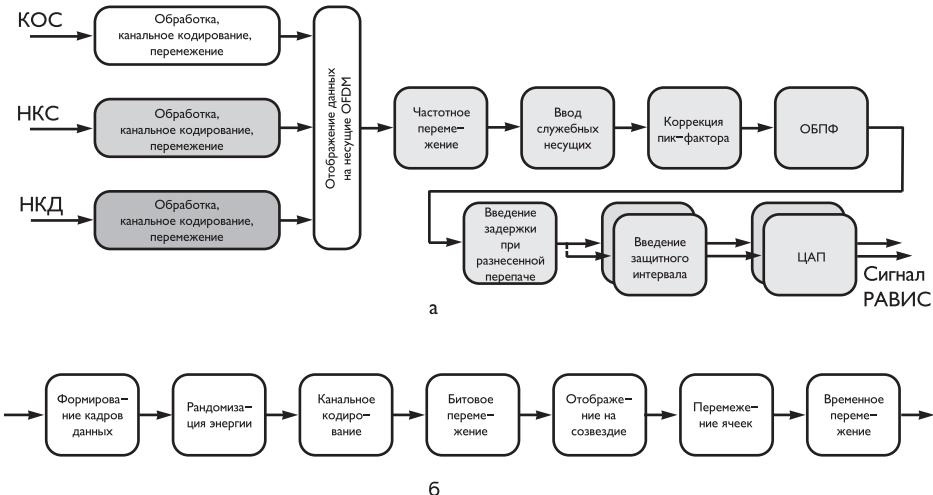


Рис. 22.7. Структурная схема передающей части РАВИС (а) и составные части блока обработки, канального кодирования и перемежения входных данных (б)

НКД — 533 бита, скорость канального кодирования — 1/2, тип модуляции — BPSK. Данные канала передают на 26 информационных несущих. При отсутствии логических каналов НСК и/или НКД их несущие используются для канала КОС.

Интервал между несущими составляет 4000/9 Гц и не зависит от ширины полосы радиоканала. Длительность полезной части символа OFDM составляет 2,25 мс. Длительность защитного интервала — 1/8. Кадр OFDM состоит из 41 символа. Структурная схема передающей части системы РАВИС представлена на рис. 22.7а. Составные части блока обработки, канального кодирования и перемежения входных данных в каждом из трех каналов системы РАВИС изображены на рис. 22.7б.

Кодирование источника

На входы каналов данных КОС, НСК и НКД подаются двоичные данные, переносящие различную информацию. В первую очередь система РАВИС предназначена для передачи видео- и звуковой информации. Эта информация предварительно подвергается эффективному кодированию с целью устранения как статистической избыточности, так и избыточности восприятия (визуальной, звуковой).

В настоящее время наиболее перспективными для использования в системе РАВИС являются звуковой кодер HE-AAC [9.59] (включая кодирование SBR, PS, MPEG Surround [9.60]) и видеокодер H.264/AVC [9.61]. Кодер HE-AAC позволяет передавать в потоке от 32 кбита/с высококачественный стереозвук, а в потоке от 84 кбитов/с — многоканальный звук формата 5.1. Кодер H.264/AVC обеспечивает видео высокого качества формата CIF 25 кадров/с при потоке менее 500 кбитов/с.

Закодированные данные источников мультиплексируют с использованием различных форматов, включая пакеты постоянной (в том числе транспортный поток

MPEG-2 [9.62]) и переменной длины. Таким образом, система РАВИС может служить эффективной транспортной средой для любых типов данных.

Общая структура канального кодирования и модуляции

Система канального кодирования и OFDM-модуляции — это функциональный блок, выполняющий адаптацию данных от выхода кодера источника сигнала к характеристикам канала. Потоки данных всех логических каналов (КОС, НСК, СКД) проходят следующие стадии обработки:

- формирование кадров данных (КД);
- рандомизацию распределения энергии КД;
- внешнее кодирование КД (блочный код Бозе–Чоудури–Хоквингема, БЧХ);
- внутреннее кодирование КД (блочный код LDPC — код с малой плотностью проверок на четность);
- битовое перемежение помехозащищенных кадров данных (ПКД);
- отображение битов на ячейки модуляционного созвездия;
- перемежение ячеек в рамках одного или нескольких ПКД;
- временное перемежение в рамках одного или нескольких ПКД;
- отображение данных всех логических каналов на ячейки OFDM;
- частотное перемежение и ввод служебных несущих;
- коррекцию пик-фактора (оционально);
- обратное быстрое преобразование Фурье (ОБПФ);
- введение защитного интервала и формирование полного сигнала OFDM.

Блок адаптации входного потока играет роль интерфейса и предоставляет возможность подключения к кодеру канала любого входного потока со скоростью, не больше заданной. Кроме того, блок адаптации входного потока формирует элементарные КД для дальнейшей обработки кодером канала.

Блок рандомизации осуществляет процедуру рассеивания энергии двоично-го цифрового потока, превращая длинные последовательности идущих подряд нулей или единиц в псевдослучайную двоичную последовательность. Данная опера-ция обеспечивает энергетический выигрыш при дальнейшем формировании радиосигнала (выигрыш по пик-фактору сигнала).

Блок помехозащитного кодирования включает в себя каскад из двух помехо-устойчивых кодеров: БЧХ и LDPC. Выходная длина ПКД после канального ко-дирования для канала КОС зависит от полосы радиоканала, а также от наличия или отсутствия каналов НСК и НКД. Длина ПКД для канала НСК составляет 1312 битов, а для канала НКД — 1066 битов.

Битовый перемежитель предназначен для ослабления влияния пакетных би-товых ошибок. Битовый перемежитель работает в пределах одного ПКД.

Блок модуляции осуществляет отображение последовательности групп битов (1 — BPSK, 2 — QPSK, 4 — 16-QAM, 6 — 64-QAM) на точки выбранного сигналь-ного созвездия, формируя ячейки данных.

Перемежитель ячеек представляет собой частотный перемежитель, обеспечи-вающий дополнительную устойчивость сигнала к частотным замираниям в ка-нале.

Временной перемежитель предназначен для глубокого перемежения одного или нескольких ПКД для защиты от временных замираний канала. Временном^й перемежитель присутствует только в канале КОС.

Мультиплексор потоков формирует из поступающих на его вход модулированных ячеек данных различных каналов OFDM-символы. При этом каждый канал отображают на определенное число несущих. Канал НСК отображают на 32 несущие, канал НКД — на 26 несущих. Число несущих канала КОС зависит от используемой полосы радиочастотного канала и наличия или отсутствия дополнительных каналов данных — НСК и НКД. После отображения данных всех каналов на несущие OFDM осуществляется частотное перемежение, в результате которого низкоскоростные надежные потоки передаются на различных несущих в различных символах OFDM, что повышает устойчивость системы передачи данных к частотно-селективным замираниям. Кадр OFDM состоит из 41 символа OFDM. Размер ПКД в логическом канале данных равен произведению количества несущих OFDM, соответствующих логическому каналу данных, на число символов в кадре OFDM.

Далее в мультиплексированный поток OFDM-символов вводятся пилотные несущие и несущие параметров передачи сигнала (ППС), обеспечивающие на стороне приема возможности синхронизации, коррекции канальных искажений и передачи дополнительной информации, в том числе о параметрах модуляции и канального кодирования, наличия различных каналов данных и пр.

Коррекция пик-фактора сигнала является необязательной, но рекомендуемой. В приложении к стандарту приведен один из возможных алгоритмов коррекции пик-фактора (метод активного расширения созвездия).

На выходе ОБПФ формируют временной дискретизированный сигнал, который после введения защитного интервала поступает на вход блока ЦАП.

При использовании разнесенной передачи до ввода защитного интервала осуществляется введение задержки сигнала.

Формирование кадров данных и рандомизация распределения энергии

Размер кадра данных КД зависит от параметров канального кодирования и равен размеру некодированного блока БЧХ-кода K_{bch} . КД состоит из заголовка, полезной нагрузки и заполнения.

В качестве полезной нагрузки могут быть использованы данные следующего типа (формата):

- транспортные пакеты MPEG-2 (ТП),
- пакетные данные, а именно пакеты фиксированной или переменной длины,
- пакеты протокола универсальной инкапсуляции потока (GSE) [9.63],
- неструктурированный поток данных (НСПД).

Заголовок КД состоит из следующих полей:

- поле битовых флагов, указывающих тип потока данных, присутствие временной метки первого пользовательского пакета данных в заголовке и/или перед каждым пользовательским пакетом, наличие пакетов переменной длины, наличие порядкового номера кадра данных;

- размер пользовательских пакетов данных (для пакетов постоянной длины, кроме ТП);
- размер поля данных (полезной нагрузки) КД;
- сдвиг начала синхронизации — первого полного пользовательского пакета (отсутствует в режиме НСПД);
- временная метка первого полного пользовательского пакета (при установке соответствующего флага в поле битовых флагов);
- порядковый номер КД (при установке соответствующего флага в поле битовых флагов);
- контрольная сумма CRC-8 полей заголовка.

Полезная нагрузка, представляющая собой пользовательские данные, следует после заголовка. Ее размер записан в поле длины данных заголовка, этот размер не может превышать значения, равного размеру КД минус размер заголовка.

Заполнение следует после полезной нагрузки, если ее размер плюс размер заголовка меньше размера КД; заполнение производится нулями.

Размер заголовка кадра данных — от 4 до 12 байтов, он зависит от типа данных полезной нагрузки, переменной или фиксированной длины пакетов, наличия временной метки пакетов, наличия номера КД.

Кадры данных рандомизируются (скремблируются) путем выполнения операции XOR с псевдослучайной двоичной последовательностью (ПСДП). Загрузка инициализирующей последовательности ПСДП проводится при начале обработки каждого КД.

Подсистема канального кодирования

Подсистема канального кодирования выполняет внешнее кодирование БЧХ, внутреннее кодирование LDPC и побитовое перемежение. Входной поток состоит из КД, а выходной поток — из ПКД.

Каждый КД, содержащий K_{bch} битов, обрабатывается подсистемой канального (помехозащитного) кодирования для формирования ПКД, содержащего N_{ldpc} битов. $N_{bch} - K_{bch}$ проверочных битов систематического внешнего кода БЧХ должны быть присоединены в конце КД, размер КД таким образом увеличивается до $N_{bch} = K_{ldpc}$ битов. $N_{ldpc} - K_{ldpc}$ проверочных битов внутреннего кодера LDPC должны быть присоединены после проверочных битов БЧХ.

Для генерации ПКД код БЧХ (N_{bch}, K_{bch}), корректирующий t ошибок, должен быть применен к каждому КД. Для получения порождающего полинома $g(x)$ кода коррекции t ошибок кодера БЧХ следует перемножить первые t примитивных полиномов, приведенных в стандарте и соответствующих N_{bch} .

Параметры канального кодирования и размеры кадров данных для всех каналов передачи данных приведены в табл. 22.10.

Кодер LDPC обрабатывает выход внешнего кодирования как блок информации размером K_{ldpc} битов, и систематически кодирует его в кодовое слово размером N_{ldpc} .

Матрица H , используемая для расчета проверочных битов кода LDPC, имеет размерность $[K_{ldpc}, M_{ldpc}]$ и содержит относительно малое число ненулевых элементов $h_{n,m} = 1$, нерегулярно распределенных по строкам и столбцам. В стандарте приведен алгоритм генерации матрицы H для всех возможных значений N_{ldpc} .

Таблица 22.10. Параметры канального кодирования системы РАВИС

Примерная скорость кода	Некодированный блок БЧХ K_{bch}	Кодированный блок БЧХ N_{bch}	Некодированный блок LDPC K_{ldpc}	Коррекция t ошибок БЧХ-кода	$N_{bch} - K_{bch}$	Кодированный блок LDPC N_{ldpc} (число несущих в канале данных)	
Канал основного сервиса, КОС, радиоканал 100 кГц							
1/2	3904	4024	10	120	130	8 036 КОС (196 несущих)	
2/3	5232	5362		120			
3/4	5896	6026		130			
1/2	3368	3488		120	130	6 970 КОС (170 несущих) + НКД	
2/3	4520	4650		130			
3/4	5096	5226		120	130	6 724 КОС (164 несущих) + НСК	
1/2	3248	3368		130			
2/3	4352	4482		120	130	5 658 КОС (138 несущих) + НСК + НКД	
3/4	4912	5042		130			
1/2	2712	2832		130			
2/3	3656	3776		120	130	16 400 КОС (400 несущих)	
3/4	4112	4242		130			
Канал основного сервиса, КОС, радиоканал 200 кГц							
1/2	8056	8196	10	140	130	15 334 КОС (374 несущие) + НКД	
2/3	10792	10932		140			
3/4	12160	12300		140			
1/2	7536	7666		130	140	15 088 КОС (368 несущих) + НСК	
2/3	10088	10228		130			
3/4	11360	11500		140	130	14 022 КОС (342 несущие) + НСК + НКД	
1/2	7416	7546		140			
2/3	9920	10060		140			
3/4	11176	11316		140			
1/2	6880	7010		130	140	20 664 КОС (504 несущие)	
2/3	9208	9348		140			
3/4	10376	10516		140			
Канал основного сервиса, КОС, радиоканал 250 кГц							
1/2	10192	10332	10	140	140	19 598 КОС (478 несущих) + НКД	
2/3	13640	13780		140			
3/4	15360	15500		140			
1/2	9664	9804		140	140	19 352 КОС (472 несущие) + НСК	
2/3	12928	13068		140			
3/4	14560	14700		140			
1/2	9536	9676		140	140	18 286 КОС (446 несущих) + НСК + НКД	
2/3	12760	12900		140			
3/4	14376	14516		140			
1/2	9008	9148		140			
2/3	12048	12188		140			
3/4	13576	13716		140			
Низкоскоростной канал, НСК							
1/2	592	652	6	60	1 312 (32 несущие)		
Надежный канал данных, НКД							
1/2	472	532	6	60	1 066 (26 несущих)		

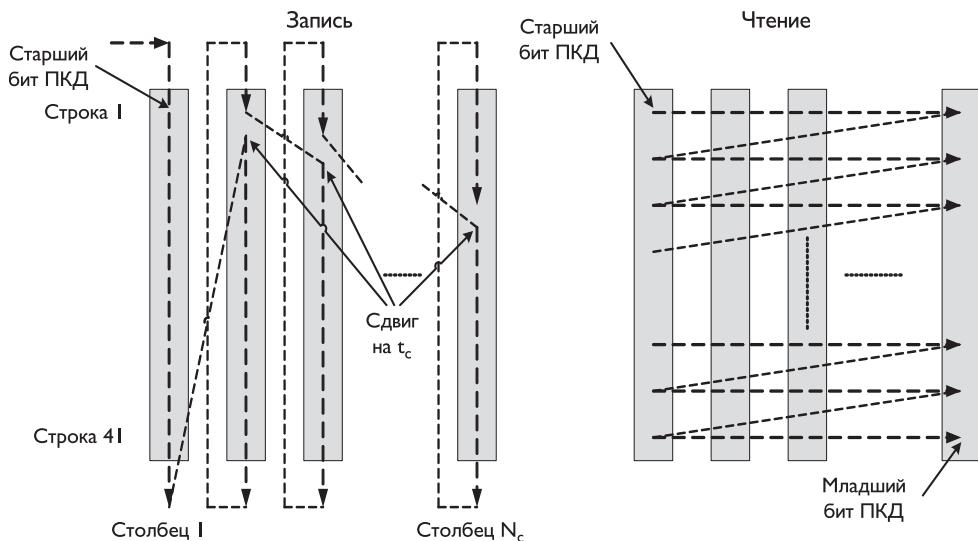


Рис. 22.8. Схема битового перемежения

Выход кодера LDPC подвергается побитовому перемежению, которое представляет собой перемежение со сдвигом начала столбцов. При этом биты данных, поступающие от кодера LDPC, последовательно записываются в перемежитель по столбцам и последовательночитываются по строкам (старший бит заголовка ПКД считывается первым), как показано на рис. 22.8. Запись стартовой позиции каждого столбца сдвигается на величину t_c , заданную в стандарте таблично. Число строк побитового перемежителя $N_r = 41$, число столбцов $N_c = N_{ldpc} \text{div} N_r$.

Отображение битов на ячейки, формирование модуляционных созвездий

Один или несколько ПКД должны быть преобразованы в кодированный и модулированный блок помехозащищенных данных (БПД). Для этого поток входных битов сначала разделяется на кодовые слова ячеек, а затем эти слова отображаются на модуляционное созвездие. Эффективное число битов на ячейку модуляции η_{MOD} составляет 6 для модуляции 64-QAM, 4 — для 16-QAM, 2 — для QPSK и 1 — для BPSK. БПД состоит из η_{MOD} ПКД, таким образом, число ячеек выходных данных в БПД равно N_{ldpc} .

Разделение последовательности битов на модулирующие значения ячеек осуществляется следующим образом: битовый поток, поступающий от битового перемежителя, разделяется (демультиплексируется) на $N_{\text{substreams}} = \eta_{\text{MOD}}$ подпотоков.

Каждое кодовое слово ячеек, поступающее от демодулятора, должно быть отображено на точку созвездия одного из типов: BPSK, QPSK, 16-QAM или 64-QAM — для получения точки созвездия z_q до нормализации. При этом используется отображение Грея. Коэффициент нормализации равен 1 для BPSK, $1/\sqrt{2}$ для QPSK, $1/\sqrt{10}$ для 16-QAM и $1/\sqrt{42}$ для 64-QAM.

Перемежение ячеек, временное перемежение

Псевдослучайный перемежитель ячеек должен равномерно распределять ячейки в БПД, чтобы обеспечить в приемнике декорреляцию канальных искажений, и сдвигать последовательность перемежения в БПД на протяжении одного блока временного перемежения.

Вход перемежителя ячеек ($g_{r,0}, g_{r,1}, \dots, g_{r,N_{\text{cells}}-1}$) должен представлять собой ячейки данных БПД с индексом r , сформированного с помощью созвездия, r обозначает увеличивающийся индекс БПД внутри блока временного перемежения (БВП). Этот индекс сбрасывается в нуль в начале каждого БВП. Выход перемежителя ячеек должен представлять собой вектор ($d_{r,0}, d_{r,1}, \dots, d_{r,N_{\text{cells}}-1}$), определенный выражением:

$$d_{r,L_r(q)} = g_{r,q} \text{ для каждого } q = 0, 1, \dots, N_{\text{cells}} - 1,$$

где N_{cells} — число выходных ячеек данных на БПД (равное N_{ldpc}) и $L_r(q)$ — функция перестановки, применяемая к r -му БПД в рамках БВП. Функция $L_r(q)$ определена следующим образом: $L_r(q) = (q \cdot K_r) \bmod N_{\text{cells}}$, значения K_r заданы в стандарте таблично.

БВП состоит из одного или нескольких БПД. Его отображают на целое число (один или несколько) кадров OFDM. Длительность БВП N_T , выраженная в кадрах OFDM, передается в параметрах передачи сигнала (ППС) и составляет от 1 до 6.

Временной перемежитель представляет собой строчно-столбцовый блоковый перемежитель, аналогичный битовому перемежителю, но без сдвига начала столбцов. Число столбцов $N_{Tc} = 41$ в перемежителе равно числу символов в кадре OFDM. Данные ячейки записывают в столбцы, а считывают построчно. Число строк временного перемежителя $N_{Tr} = (N_{\text{cells}} \cdot N_T) \bmod N_{Tc}$.

Общая схема OFDM-модуляции

После блоков перемежения цифровые потоки логических каналов данных КОС, НСК и НКД поступают на блок отображения данных на несущие OFDM. Общее число и распределение несущие для различных полос радиоканала приведено в табл. 22.11.

При OFDM-модуляции цифровые потоки логических каналов данных разбиваются на блоки длиной $L = 41$ символ комплексной модуляции z_q , причем каждый такой символ имеет длину η_{MOD} битов. Далее в соответствии с кодом Грэя осуществляется отображение z_q на информационные несущие OFDM-символа в соответствии с выбранным вариантом модуляции несущих — BPSK, QPSK, 16-QAM или 64-QAM.

После отображения данных логических каналов на информационные несущие осуществляется частотное перемежение, которое делает распределение логических каналов по несущим псевдослучайным, что позволяет повысить устойчивость системы передачи данных к селективным замираниям.

Помимо информационных несущих в состав OFDM-символа вводятся также пилотные несущие и несущие ППС. Пилотные несущие передаются на повышенном уровне мощности. Фазы пилотных несущих задаются в соответствии с опорной псевдослучайной двоичной последовательностью равными 0° либо 180° .

Пилотные несущие вводятся в состав OFDM-символа с целью обеспечения на принимающей стороне возможности синхронизации сигнала и оценки характеристи-

Таблица 22.11. Распределение несущих при различных полосах радиоканала

Полоса радиоканала Δf , кГц	Общее число несущих OFDM K_{total}	Число несущих					
		Пилотных	ППС	КОС	НСК	НКД	
100	215	15 (8 рассеянных, 7 повторяющихся)	4	196	—	—	
				170	—	26	
				164	32	—	
				138	32	26	
				400	—	—	
				374	—	26	
				368	32	—	
				342	32	26	
				504	—	—	
				478	—	26	
200	439	35 (22 рассеянных, 13 повторяющихся)		472	32	—	
				446	32	26	
250	553	45 (28 рассеянных, 17 повторяющихся)		—	—	—	
				—	—	—	
				—	—	—	
				—	—	—	
				—	—	—	
				—	—	—	
				—	—	—	
				—	—	—	
				—	—	—	
				—	—	—	

стик канала передачи. Пилотные несущие составляют около 9% от общего числа несущих.

Несущие ППС вводятся с целью непосредственной передачи информации о ключевых параметрах передаваемого сигнала (наличие логических каналов данных, скорость LDPC-кода, тип QAM-созвездия, длина защитного интервала и пр.). В каждом символе OFDM передаются четыре несущих ППС.

OFDM-символ, полученный после введения пилотных несущих и несущих ППС, подвергается обратному преобразованию Фурье, которое преобразовывает символ из частотной области во временную. Затем в начало символа вводится защитный интервал длительностью T_G . В защитный интервал копируются последние $N_{\text{symb}} \cdot T_G / T_U$ отсчетов полезной части символа (циклический префикс), где N_{symb} — длина полезной части символа (в отсчетах), T_U — длительность полезного символа (в секундах). В системе предусмотрен защитный интервал с длительностью $T_G = T_U / 8$.

Полученная последовательность временных отчетов ограничивается по уровню, масштабируется и квантуется в соответствии с разрядностью ЦАП. Перед обратным преобразованием Фурье может проводиться операция, направленная на снижение пик-фактора сигнала.

При использовании разнесенной передачи перед введением защитного интервала блок внесения задержки формирует несколько сигналов с необходимыми временными задержками.

Кадровая структура OFDM

Передаваемый сигнал должен быть организован в виде последовательности кадров. Кадр имеет длительность T_F и состоит из $L = 41$ OFDM-символов. OFDM-символ имеет длительность T_S и состоит из защитного интервала длительностью T_G и полезной части длительностью T_U ($T_S = T_U + T_G$).

OFDM-символы содержат полезную информацию (о данных) и опорную информацию (пилотные и ППС-несущие). Символы в кадре OFDM пронумерованы

от 0 до 40. Ключевые параметры блока OFDM-модуляции системы приведены в табл. 22.12.

Таблица 22.12. Ключевые параметры модуляции OFDM системы РАВИС

Параметр	Значение		
Полоса радиоканала Δf , кГц	100	200	250
Число несущих K_{total}	215	439	553
Длительность полезной части символа T_U , мкс	2250		
Расстояние между соседними несущими $1/T_U$, Гц	$\frac{4000}{9} = 444\frac{4}{9}$		
Расстояние между крайними несущими спектра $(K_{\text{total}} - 1)/T_U$ кГц (примерное значение)	95,1	194,7	245,3
Длительность защитного интервала T_G , мкс	281,25		
Длительность символа $T_S = T_U + T_G$, мкс	2531,25		
Длительность кадра OFDM, с	103,8 (41 символ)		

Математическая модель сигнала $s(t)$ описывается выражением:

$$s(t) = \operatorname{Re} \left\{ e^{j \cdot 2\pi \cdot f_c \cdot t} \cdot \sum_{m=0}^{\infty} \sum_{l=0}^{L-1} \sum_{k=K_{\min}}^{K_{\max}} C_{m,l,k} \cdot \Psi_{m,l,k}(t) \right\},$$

где

$$\Psi_{m,l,k}(t) = \begin{cases} e^{j \cdot 2\pi \cdot \frac{k'}{T_U} \cdot (t - T_G - l \cdot T_S - L \cdot m \cdot T_S)}, & (l + L \cdot m) \cdot T_S \leq t \leq (l + L \cdot m + 1) \cdot T_S, \\ 0, & \text{иначе,} \end{cases}$$

K_{\min} — нижний предел индекса несущей, $K_{\min} = 0$; K_{\max} — верхний предел индекса несущей, $K_{\max} = K_{\text{total}} - 1$; L — число символов OFDM в кадре, $L = 41$; k — индекс несущей; k' — индекс несущей относительно центральной частоты, $k' = k - (K_{\max} + K_{\min})/2$; l — индекс символа OFDM в кадре; m — индекс передаваемого кадра; f_c — центральная частота спектра сигнала; $C_{m,l,k}$ — комплексный коэффициент, модулирующий k -ю несущую l -го символа в m -ом кадре OFDM.

С целью нормирования выходного уровня сигнала в системе предусмотрены коэффициенты нормализации величин $C_{m,l,k}$. Применение этих коэффициентов нормирует к единице суммарную мощность модуляционного алфавита информационных несущих сигналов.

Частотное перемежение

Частотное перемежение, которое выполняется после отображения данных логических каналов на информационные несущие, делает распределение логических каналов по несущим псевдослучайным. Такое распределение позволяет повысить устойчивость низкоскоростных каналов данных НСК и НКД к селективным за-мираниям.

Модулированные ячейки данных каналов НСК и НКД отображаются псевдослучайным образом на информационные несущие OFDM с индексами $k' = \pm 1, \dots, \pm 106$ относительно центральной несущей радиоканала. Отображение задано в стандарте таблично.

После отображения данных каналов НСК и НКД модулированные ячейки данных канала КОС отображаются на информационные несущие с индексами

$k = K_{\min} + 1, \dots, K_{\max} - 1$ последовательно, за исключением информационных несущих, уже занятых каналами НСК и НКД.

Пилотные несущие

Назначение пилотных несущих — передача дополнительной (опорной) информации, с помощью которой на приемной стороне осуществляется синхронизация сигнала, а также оцениваются характеристики канала передачи. Пилотные несущие передаются на повышенном уровне мощности. В системе используются два типа пилотных несущих: повторяющиеся и рассеянные. Положение повторяющихся пилотных несущих в спектре сигнала не зависит от номера символа в кадре, а положение рассеянных — зависит. Фазы пилотных несущих могут принимать значения 0° или 180° в соответствии с опорной псевдослучайной двоичной последовательностью (ПСДП). Значение 0 соответствует фазе 0° , значение 1 соответствует фазе 180° . Инициализация ПСДП осуществляется в начале каждого символа OFDM. Мощность пилотных несущих составляет $16/9$, в то время как средняя мощность информационных несущих и несущих ППС в соответствии с коэффициентом нормирования равна 1.

Индексы рассеянных пилотных несущих определены в стандарте таблично. Если совместить центральные несущие при различных полосах радиоканала, то пилотные несущие более узкого радиоканала окажутся подмножеством пилотных несущих более широкого радиоканала. При полосе радиоканала 100 кГц используется 8 рассеянных пилотных несущих, при полосе 200 кГц — 22 несущие, при полосе 250 кГц — 28 несущих.

По маске пилотных несущих формируется пять типов символов, которые последовательно повторяются (от нулевого до четвертого, начиная с начала кадра OFDM). Так как в кадре OFDM содержится 41 символ, то на границе кадра соседствуют два символа нулевого типа.

Индексы повторяющихся пилотных несущих также определены в стандарте таблично и так же, как и рассеянные пилотные несущие, повторяющиеся пилотные несущие более узкого радиоканала являются подмножеством пилотных несущих более широкого радиоканала. При полосе радиоканала 100 кГц используется 7 повторяющихся пилотных несущих, при полосе 200 кГц — 13 несущих, при полосе 250 кГц — 17 несущих. Повторяющиеся пилотные несущие расположены симметрично относительно центральной несущей канала, кроме того, повторяющимися пилотными несущими являются центральная несущая и крайние несущие для всех трех типов полос радиоканала (т. е. несущие с индексами $0, \pm 107, \pm 219, \pm 276$ относительно центральной несущей).

Несущие ППС

Несущие ППС предназначены для передачи параметров сигнала, связанных с режимом передачи. Информация ППС передается параллельно на четырех несущих. Каждая несущая ППС в одном и том же символе передает один и тот же дифференциальный кодированный информационный бит. Индексы этих несущих относительно центральной несущей радиоканала равны ± 27 и ± 81 для всех вариантов ширины полосы радиоканала, т. е. все они расположены в пределах самой узкой полосы канала 100 кГц.

Несущие ППС обеспечивают передачу следующей информации:

- номер версии системы (три бита);
- тип модуляционного созвездия КОС (два бита);
- скорость канального кодирования КОС (три бита);
- число кадров временного перемежения КОС (три бита);
- индекс текущего кадра временного перемежения КОС (три бита);
- битовые флаги наличия или отсутствия логических каналов НСК и НКД (два бита);
- ширина полосы радиоканала (два бита).

Инициализация фазы ППС-несущих осуществляется для каждого кадра OFDM. Для инициализации фазы ППС-несущих первого символа кадра используется также опорная ПСДП, что и для пилотных несущих.

Блок данных ППС соответствует одному кадру OFDM и содержит 41 бит, которые определены следующим образом:

- 27 информационных битов (в текущей версии системы используются 18, остальные зарезервированы для дальнейших расширений);
- 14 избыточных битов для защиты от ошибок (сокращенный код БЧХ (41, 27, $t = 2$)).

Несущие ППС передаются с уровнем мощности, равным 1, с использованием разностно-фазовой модуляции (DBPSK) и с инициализацией в начале каждого кадра OFDM.

Скорость полезного информационного потока

Скорость полезного битового потока различных логических каналов данных R_i (КОС, НСК, НКД) определяется по формуле:

$$R_i = \frac{K_{\text{inf } i} \eta_{\text{MOD}} R_{\text{FEC}} R_{DF}}{T_U (1 + T_G/T_U)},$$

где $K_{\text{inf } i}$ — число информационных несущих логического канала i ; η_{MOD} — число битов информации, передаваемых на одной несущей; $R_{\text{FEC}} = K_{\text{bch}}/N_{\text{ldpc}}$ — скорость помехоустойчивого кодирования, равная отношению числа битов в КД к числу битов в ПКД; R_{DF} — доля полезной информации в КД, равная отношению числа битов пользовательских данных к общему размеру КД; T_U — длительность полезной части символа; T_G — длительность защитного интервала.

В подтверждение того факта, что систему РАВИС можно использовать при вещании без изменения частотных распределений, на рис. 22.9 приведены спектральные маски передачи РАВИС, а также спектральная маска передачи аналогового FM вещания в соответствии с европейским стандартом [9.64].

Приемник

На вход приемника (рис. 22.10) поступает радиосигнал диапазона 65,8–74 МГц или 87,5–108 МГц. Две антенны обеспечивают реализацию метода разнесенного приема. Блок усиления высокой частоты и переноса спектра в низкочастотную область (УВЧ + ПЧ) обеспечивает также цифро-аналоговое преобразование сигнала.

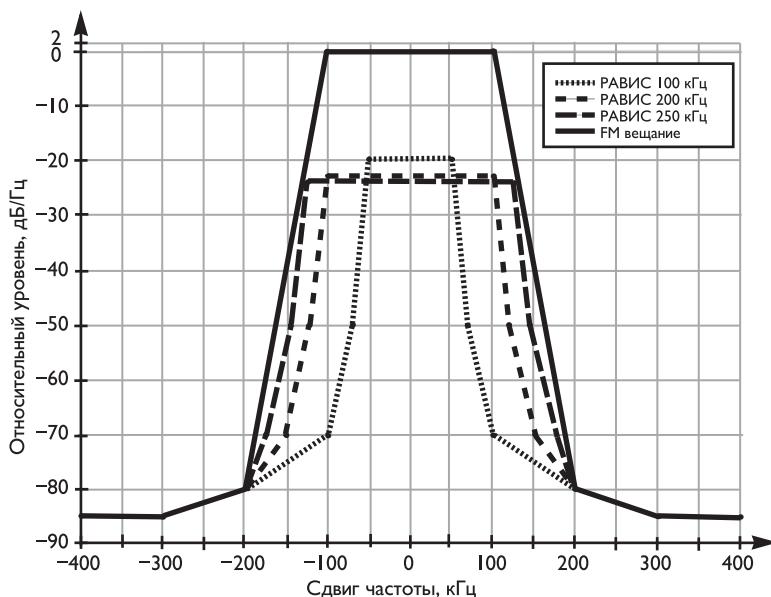


Рис. 22.9. Спектральные маски систем вещания РАВИС и FM

Блок OFDM-синхронизации осуществляет синхронизацию сигнала с помощью введенных в состав сигнала пилотных несущих. Работа данного модуля осуществляется как во временной, так и в частотной области. На приведенной упрощенной схеме приемника модуль быстрого преобразования Фурье (БПФ) следует непосредственно за модулем синхронизации, однако это верно лишь для алгоритмов с синхронизацией во временном пространстве (при частотной синхронизации БПФ предшествует блоку OFDM-синхронизации).

На вход коммутатора разнесенного приема поступает два цифровых потока. В данном модуле осуществляется адаптивное мультиплексирование двух сигналов, обеспечивающее выигрыш по отношению сигнал/шум относительно приема на одну антенну (без разнесенного приема).

Демультиплексор потоков выделяет из синхронизированного потока OFDM-символов три потока, соответствующие каналам основного сервиса, надежного речевого доступа и данных надежного доступа. Каждый из этих потоков переносится отдельным набором несущих OFDM-символа.

Каждый из трех перечисленных потоков поступает на QAM-детектор. Канал основного сервиса имеет 3 режима детектирования: QPSK, 16-QAM, 64-QAM. Канал надежного речевого доступа детектируется только как QPSK, а канал данных надежного доступа — как BPSK.

На вход декодера канала основного сервиса поступает поток ячеек, из которых представляет собой демодулированное значение соответствующей несущей.

Блоковый деперемежитель осуществляет перестановку блоков данных и является, по сути, временным деперемежителем, защищающим сигнал от временных замираний канала.

Деперемежитель ячеек выполняет функцию частотного деперемежителя, защищая сигнал от частотных замираний канала.

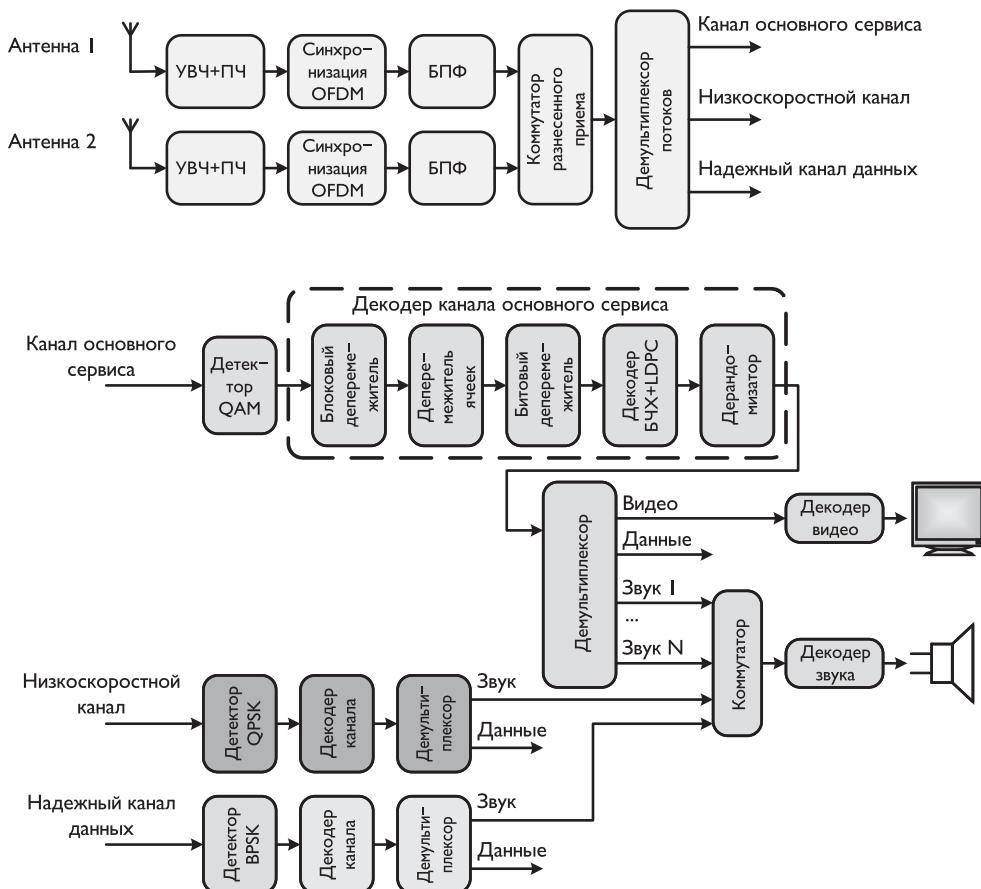


Рис. 22.10. Упрощенная структурная схема приемника системы РАВИС

Битовый деперемежитель служит (вместе с перемежителем на стороне кодера) для ослабления влияния пакетных битовых ошибок.

Блок декодирования БЧХ и LDPC предназначен для устранения ошибок канала.

Блок дерандомизации осуществляет обратное преобразование относительно рандомизатора кодера.

Демультиплексор канала основного сервиса осуществляет выделение потоков видео и аудио в канале основного сервиса на уровне потока данных.

Аудиокоммутатор предназначен для выбора воспроизводимой звуковой программы.

Декодеры канала надежного речевого доступа и канала данных надежного доступа аналогичны описанному выше декодеру канала основного доступа, за исключением блокового деперемежителя, отсутствующего в данных декодерах. Главной конкурентной технологией цифрового мобильного телевещания РАВИС является стандарт DVB-H.

Развитие телевещания в DVB-H продвигается довольно медленно. К примеру, наиболее коммерчески успешными считаются проекты мобильного телевещания в Италии, где услуга предоставляется с лета 2006 г. При этом общее число абонентов к лету 2010 г. достигло 600 тысяч, что для страны с населением в 38 миллионов человек и с практически 100%-м проникновением мобильной связи вряд ли можно назвать впечатляющим.

Такие результаты можно объяснить несколькими причинами.

Во-первых, стоимость строительства инфраструктуры мобильного телевещания остается достаточно высокой. Расчетные значения зоны покрытия протокола DVB-H составляют 7–8 км от передатчика, а в условиях плотной городской застройки значительно меньше. В связи с этим для обеспечения уверенного приема телевещания необходимо развертывание большого количества точек передачи сигнала. Например, для развертывания мобильного телевещания DVB-H в Москве потребовалось развернуть 35 базовых станций общей стоимостью в 15–20 миллионов долларов.

Во-вторых, существует проблема радиочастотного планирования, которая в Европе будет решена, когда произойдет полный переход на цифровое вещание.

В-третьих, европейский потребитель в своей массе довольно «прохладно» относится к услуге мобильного телевидения. Такое отношение обусловлено прежде всего тем, что смотреть обычные телепередачи на маленьком экране мобильного телефона некомфортно, а специального мобильного контента пока недостаточно. Как следствие, производители не спешат с выпуском телефонов со встроенными DVB-H-приемниками.

Другой конкурентной технологией цифрового мобильного радиовещания является система DRM+ (DRM Mode E), разработку которой осуществляет консорциум DRM в составе более 100 организаций из 34 стран (см. раздел 22.5). Эта система более узкополосная и обеспечивает передачу в одном канале не более 4 стереофонических звуковых программ, но не позволяет транслировать видеопрограммы достаточно хорошего качества.

При разработке отечественного стандарта [9.25] и соответствующей аппаратуре была реализована «нестандартная» полоса пропускания радиоканала в 100 кГц, обеспечивающая возможности, определенные стандартом DRM+.

В 2009–2010 гг. были проведены натурные испытания модели системы РАВИС в городских условиях в плотной застройкой в г. Москве, а также в условиях горной местности в г. Сочи. Приемник системы РАВИС был размещен в движущейся легковой автомашине, при этом прием осуществлялся с помощью обычной штыревой антенны, расположенной на крыше автомобиля. Тестирования системы РАВИС было весьма успешным, что позволяет в короткие сроки развернуть единую национальную систему мобильного телерадиовещания. Низкая стоимость строительства инфраструктуры, возможность использования существующих радиочастот позволит обеспечить мобильным телерадиовещанием практически всю территорию России.

22.5. Цифровое радиовещание DRM+ в ОВЧ-диапазоне частот

Стандарт DRM [9.65] разработан для длинно-, средне- и коротковолновых диапазонов частот — ниже 30 МГц. Чтобы удовлетворить новым запросам, в марте 2005 года участники DRM-консорциума и всемирного форума DAB приняли совместное решение по распространению цифрового вещания на диапазон до 108 МГц [9.66], который до настоящего времени используется для аналогового стереовещания. В дальнейшем проекты по реализации этого решения получили название DRM+.

Основной идеей создания стандарта DRM+ является расширение стандарта системы DRM на диапазон частот до 108 МГц и его использование преимущественно для передачи аудиоинформации [9.67–9.71]. В связи с этим к стандартным техническим требованиям добавляются дополнительные:

- ограниченное количество видов сервиса в пределах одного мультиплексированного потока позволяет избежать организации сложного сервиса мультиплексирования на стороне передачи;
- возможности транспорта аудиопотоков и данных, допускающие введение новых мультимедийных функций.

Канал передачи данных

Сигналы частот ОВЧ-диапазона характеризуются дифракцией, рассеиванием и отражением электромагнитных волн между передатчиком и приемником. Кроме того, передвижение приемника приводит к многолучевому приему и влиянию эффекта Доплера.

Поскольку детерминированное описание временной изменчивости канала невозможно, требуется применение стохастического подхода. Поэтому была разработана модель WSSUS (Wide Sense Stationary Uncorrelated Scattering — Некоррелируемый разброс, стационарный в широком смысле), которая получила хорошее подтверждение в последнее десятилетие [9.67].

Схема с задержкой в каждой линии передачи является адекватным представлением с учетом того, что отношение между сигналом на входе $e(t)$ и сигналом на выходе $s(t)$ в суммарном виде может быть представлено следующим образом:

$$s(t) = \sum_{k=1}^N \rho_k \cdot c_k(t) \cdot e(t - \tau_k),$$

где N — количество лучей с соответствующей средней мощностью ρ_k и относительной задержкой τ_k .

Изменяющиеся во времени весовые коэффициенты задержки в линии передачи $c_k(t)$ представляют собой комплекснозначные стационарные гауссовые случайные процессы с нулевым средним, с амплитудами, распределенными по законам Релея или Райса, и равномерно распределенными фазами.

Определение различных профилей каналов позволяет выполнить оценку системы в случаях хорошего, типичного и плохого распространения сигнала. Обзор

Таблица 22.13. Профили каналов

№	Название профиля	Скорость v [км/ч]	Макс. допплеровская частота * $f_{D,max}$ [Гц]	Кол-во лучей N	Максимальная задержка τ_{max} [мкс]
1	AWGN (аддитивный белый гауссов шум)	0	0	1	0
2	Городская местность	2 / 60	0,2 / 5,6	9	3
3	Сельская местность	150	13,9	9	3
4	Труднопроходимая местность	60	5,6	9	16
5	Холмистая местность	100	9,3	12	82,7
6	SFN (одночастотная сеть)	150	13,9	7	600

* при центральной частоте несущих 100 МГц

профилей каналов, относящихся к стандарту DRM+, приведен в табл. 22.13. Каждый профиль определяется количеством лучей с их собственными задержками и допплеровскими сдвигами частоты.

Архитектура системы

В основу базовой архитектуры системы DRM+ [9.68] был положен уже существующий стандарт DRM [9.65]. Такой подход способствует разработке приемников и модуляторов и поддерживает быстрое продвижение товара на рынок. Поскольку характеристики канала передачи в диапазоне ОВЧ и на частотах ниже 30 МГц различны, необходимо было произвести новую оценку. Для совместимости с большей частью распределений каналов в ЧМ-диапазонах общая ширина полосы DRM+ была выбрана равной 100 кГц. Чтобы избежать чрезмерного распространения эффектов замирания сигналов в городских условиях, применены новые методы разнесения.

Обзор системы

На рис. 22.11 приведена архитектура системы. Некоторые виды сервиса передачи звука и данных кодируются источником или предварительно закодированы. Для этого применяются MPEG-4 кодеры источника из стандарта DRM. Это аудиокодер AAC и два кодера речи CELP (кодирование с линейным предсказанием) и HVXC (кодирование усиленными гармоническими векторами). Все кодеры могут быть дополнены SBR (спектральное повторение), средством расширения функциональных возможностей, использующим данные огибающей спектра аудиосигнала для расширения полосы этого сигнала.

Два информационных канала, FAC (Канал быстрого доступа) и SDC (Канал описания сервиса), также перешли из стандарта DRM. Они позволяют подтвердить связность мультиплексированных данных.

Канальное кодирование выполняется на основе сверточных кодов и MLC (многоуровневое кодирование) для созвездий высоких порядков. Соответственно, для MSC (канал основного сервиса), содержащего полезные данные, используется два



Рис. 22.11. Архитектура системы DRM+

созвездия: QPSK и 16-QAM. Определены несколько величин кодовой скорости в интервале между $R = 0,25$ (скорость сверточного кода — $1/3$) для QPSK и $R = 0,625$ (скорость сверточного кода — $3/5$) для 16-QAM, что позволяет достичь гибкости в отношении надежности и скорости передачи данных.

Реализованное устройство перемежения повышает эффективность канального кодирования в системе. MSC и два канала передачи данных FAC и SDC сопровождаются опорной пилотной несущей, которая остается постоянной для оценки канала и синхронизации в приемнике. Весь этот мультиплексированный поток передается в генератор OFDM для формирования временного сигнала, который в дальнейшем может быть преобразован в радиочастотный диапазон.

Параметры передачи OFDM

Вставка дополнительного защитного интервала перед каждым символом во временной области позволяет полностью избежать межсимвольной интерференции. Отдельные поднесущие разделены частотными интервалами $\Delta F = 1/T_U$ во избежание интерференции между несущими, где T_U — временной интервал полезной части одного символа OFDM. Он прибавляется к защитному интервалу T_G для определения полной длительности символа $T_S = T_U + T_G$. При определении параметров OFDM учитываются следующие требования:

- $T_G > \tau_{\max}$: соответствует тому, что величина самой длительной задержки профиля канала должна быть меньше величины защитного интервала; при выборе значения $T_G = 250$ мкс это требование выполняется для всех профилей каналов, за исключением профиля 6 табл. 22.13, но наиболее поздние эхосигналы в профиле SFN имеют пренебрежимо малую мощность;
- $T_G < T_U$: для создания эффективной системы величина защитного интервала должна быть значительно меньше, чем длительность полезного символа; при выборе значения $T_U = 2,25$ мс потери мощности, обусловленные защитным интервалом, составляют менее 0,5 дБ;
- $\Delta F \gg f_D$: приближенный подсчет показывает, что расстояние между поднесущими должно быть в 10 раз больше, чем максимальные допплеровские

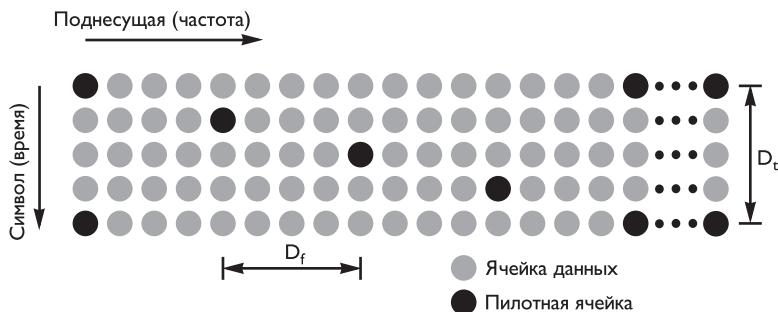


Рис. 22.12. Диаграмма пилотных несущих на частотно-временной сетке

сдвиги частоты, во избежание интерференции между несущими; выбор значения $\Delta F = 444$ Гц позволяет достичь очень высоких скоростей без значительного ухудшения отношения сигнал/шум.

Следует принять во внимание также несколько практических требований. Например, в качестве длины преобразования OFDM (длина ОБПФ) следует выбирать число, которое будет удобным при реализации.

Диаграмма пилотных несущих

На рис. 22.12 показано распределение пилотных несущих на частотно-временной сетке. Диаграмма пилотных несущих адаптирована к каналу передачи для получения хорошей оценки канала в приемнике. Тогда в соответствии с теоремой о дискретизации

$$D_t = \frac{1}{2f_{D,\max} \cdot T_S} \text{ и } D_f = \frac{1}{\tau_{\max} \cdot \Delta F}.$$

Кроме того, следует принимать во внимание определенное пространство для неидеальных фильтров интерполяции канала. При $D_t = 4$ и $D_f = 4$ был найден хороший компромисс между непроизводительными издержками и производительностью сигнала. Выбор этого варианта позволяет приемнику отслеживать канал в том числе и при высокой скорости передачи для всех профилей канала.

Скорости передачи данных

Потребности вещателей в пропускной способности варьируют от одной высококачественной аудиопрограммы до нескольких программ с объемным звуком и дополнительным сервисом передачи данных. Чтобы удовлетворить всем интересам, система должна обеспечивать скорости передачи данных в диапазоне от 40 кбитов/с до 186 кбитов/с. Возможны два адекватных варианта, представленные двумя созвездиями QPSK и 16-QAM с кодовыми скоростями от 0,25 до 0,625.

На рис. 22.13 показана спектральная эффективность системы DRM+ на примере нескольких кодовых скоростей и двух выбранных созвездий. Это означает, что пилотная несущая предоставляет количество битов, которое может быть передано на символ и поднесущую в зависимости от соотношения сигнал/шум

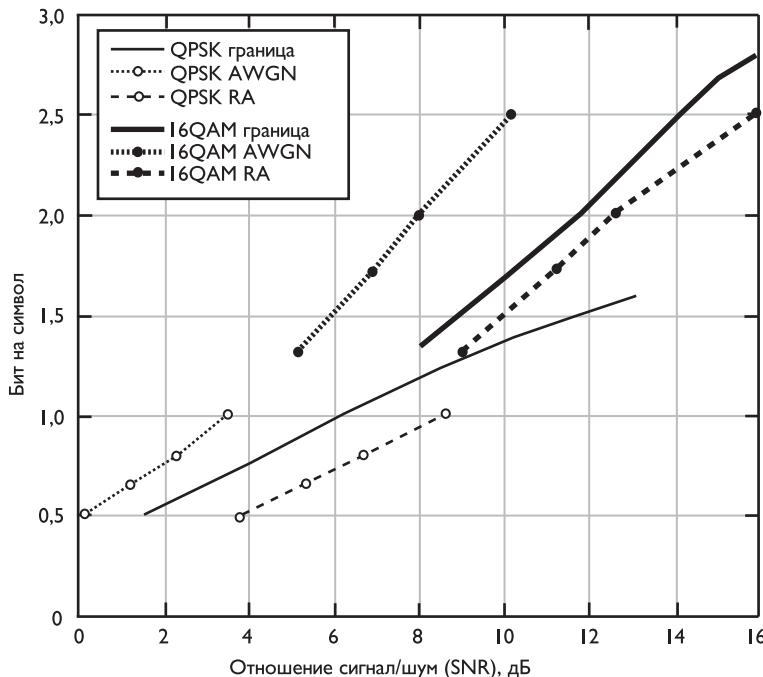


Рис. 22.13. Схема спектральной эффективности системы DRM+

(SNR). Желтая и темно-синяя непрерывные линии соответствуют объединенным границам релеевского канала. Они теоретически показывают максимальную скорость передачи данных при частоте появления ошибочных битов (BER), равной 10^{-4} . Другие линии показывают результаты моделирования для канала с гауссовским белым шумом (AWGN) и для канала при труднопроходимой местности (RA). Можно видеть, что производительность канала RA близка к теоретической производительности релеевского канала и что модель канала AWGN даже превосходит эти пределы. Кроме того, становится очевидным наличие широчайшего диапазона возможных скоростей передачи данных в зависимости от SNR.

На рис. 22.14 приведены зависимости передаваемого цифрового потока от кодовой скорости R при использовании двух созвездий сигналов OFDM.

Разнесение передачи

При проектировании системы стояла сложная задача — достичь хорошей производительности в профиле городской местности канала при низкоскоростной передаче. Она обусловлена проблемой амплитудного замирания, возникающей, когда когерентная полоса канала $1/\tau_{\max}$ уже полосы сигнала. Это означает, что 100 кГц-я когерентная полоса соответствует распространению задержки, равной 10 мкс.

Для области применения системы DRM+ является использование метода циклического разнесения по задержке (CDD, cyclic delay diversity) [9.72]. При этом

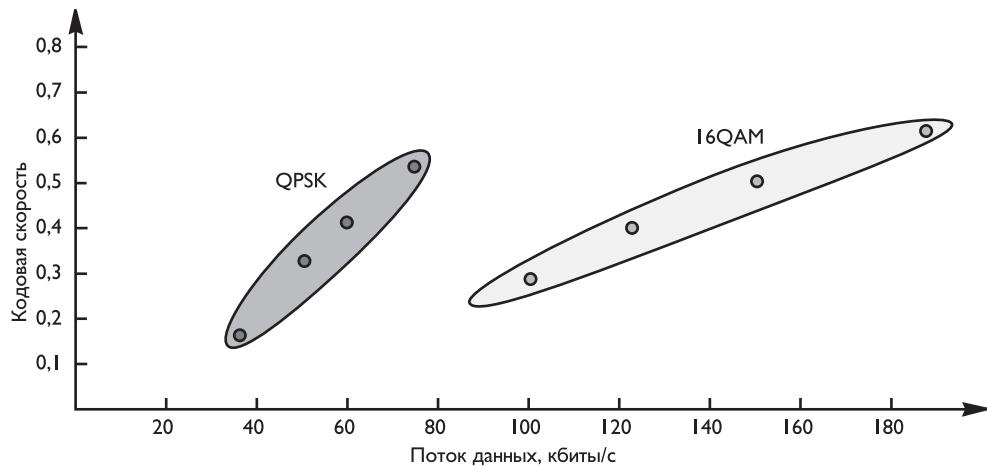


Рис. 22.14. Параметры цифровых потоков при различных кодовых скоростях

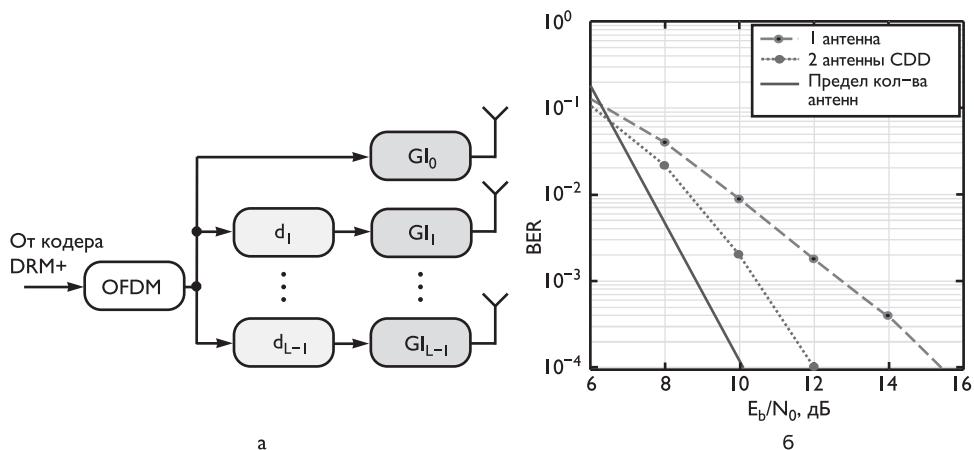


Рис. 22.15. Передатчик OFDM с циклическим разнесением по задержке (CDD) — а) и характеристики его производительности б)

задержка d_l для каждой антенны добавляется между схемой формирования ОБПФ и вставкой защитного интервала (GI), как показано на рис. 22.15а.

Оценка производительности канала определяется изменением зависимости частоты появления ошибочных битов (BER) от отношения энергии полезного бита E_b к плотности шума N_0 . На рис. 22.15б проведено сравнение BER одного передатчика и двух передатчиков CDD, а также предельное изменение этой характеристики при использовании большого количества передатчиков.

ГЛАВА 23

СИСТЕМЫ ВИДЕОКОНФЕРЕНЦСВЯЗИ

Началом массового распространения видеокоммуникаций, наверное, следует считать вывод в декабре 1996 г. на рынок совместного продукта фирм 3Com и PictureTel [9.73]. Начав с относительно простых систем, PictureTel сравнительно быстро вышла на производство терминалов групповой ВКС, поддерживающих многоточечные видеоконференции. Вслед за оборудованием PictureTel появилась разработка ProShare Intel, которая не претендовала на изысканность, однако обладала достаточной функциональностью для организации конференций «точка–точка». Некоторое время спустя фирма Intel отказалась от этого сектора рынка. В течение следующих пяти лет на мировой и российский рынки свою продукцию в области видеоконференцсвязи вывели фирмы Polycom, Sony, VCON, VTEL, Aethra [9.74]. Несколько позже других на рынке появилась норвежская фирма Tandberg, однако на европейском и российском рынках она за короткое время вышла в лидеры как по объемам поставок, так и по ассортименту предлагаемых продуктов [9.75].

23.1. Основные стандарты систем видеоконференцсвязи

Первым стандартом ITU-T, используемым в ВКС, можно считать рекомендацию ITU-T G.711, определяющую способ кодирования звука на основе импульсно-кодовой модуляции, обеспечивающую полосу аудиосигнала примерно 3,5 кГц в канале 64 кбита/с. Первая версия стандарта была принята в декабре 1972 г., т. е. задолго до появления ВКС, в дальнейшем были приняты еще четыре версии, последняя — в ноябре 1988 г. Однако на этом развитие стандарта не остановилось: в 1999, 2000, 2009 гг. были приняты приложения и дополнения.

Первыми стандартами, имеющими непосредственное отношение к передаче видеосигнала, являются H.221 и H.261, принятые ITU-T в 1988 г.

Рекомендация H.221 определяет структуру кадров для телекоммуникационных каналов аудиовизуальных услуг в полосе от 64 до 1920 кбитов/с. Эта рекомендация — одна из самых эволюционирующих среди стандартов ВКС: новые версии принимались в 1990, 1993, 1995, 1997, 1999, 2000 и 2009 гг. В настоящее время действует 8-я версия стандарта.

Рекомендация H.261 стала первым стандартом для видеокодирования в ВКС. Первая версия рекомендации была принята в 1988 г., затем последовали 2-я версия — в 1990 и 3-я — в 1993 гг. Первоначально стандарт определял видеокодек для передачи аудиовидеоданных в канале $r \times 64$ кбита/с. Параметр r может меняться в диапазоне от 1 до 30 для цифровой сети с интеграцией обслуживания

ISDN (Integrated Services Digital Network). Кодирование основано на дискретном косинусном преобразовании (ДКП), предсказании перемещения блоков и использовании кода с переменной длиной. Рекомендация регламентирует алгоритмы кодирования для передачи видео в форматах CIF и QCIF при частоте кадров до 30 кадров/с.

Первым документом ITU-T, определяющим соглашения по стеку протоколов ВКС, стала рекомендация H.320 «Узкополосные видеотелефонные системы и окончное оборудование». Первая версия этого стандарта была принята в 1990 году. Она определяет технические требования к узкополосным видеотелефонным системам и окончному оборудованию в основном для услуг видеоконференций и видеотелефонии. Рекомендация относится к системам ВКС, работающим в се-тях цифровой телефонии ISDN, и характеризует общую конфигурацию системы, состоящей из нескольких элементов, соответствующих некоторым рекомендациями ITU-T, а также определяет режимы связи и типы терминалов, организацию управления соединением, терминальные аспекты и требования к взаимодействию. Новые версии стандарта принимались в 1993, 1996, 1997, 1999 и в марте 2003 г. В настоящее время действует шестая версия рекомендации H.320, реализующая ряд улучшений и разъяснений предыдущей версии, главным образом — описание применений рекомендаций ITU-T G.722.1, H.239, H.241, H.264 и стандарта ISO/IEC 14496-3.

В том же 1990 году одновременно с H.320 была принята первая редакция рекомендации H.242, которая определяет системы установления связи между аудиовизуальными терминалами цифровых каналов до 2 Мбитов/с. Рекомендация касается эксплуатации системы для конференцсвязи между тремя или больше аудиовизуальными терминалами, соответствующими H.320. Стандарт имеет длительную историю развития: новые версии принимались в 1993, 1996, 1997, 1999, 2004 и 2009 гг. В настоящее время действует седьмая версия рекомендации.

Первые три версии связанного с H.242 стандарта H.243, определяющего процедуры установления связи между тремя и более аудиовизуальными окончными устройствами, соответствующими H.320, с использованием цифровых каналов со скоростями до 1920 кбитов/с, приняты синхронно с H.242 в 1993, 1996 и 1997 гг. Следующие версии были приняты в 2000 и 2005 гг. В настоящее время действует пятая версия.

Наибольшей эволюцией из всех стандартов ВКС отличается рекомендация H.245 «Протокол управления для мультимедийной связи». Эта рекомендация определяет синтаксис и семантику информационных сообщений терминалов, а также процедур, используемых для внутриволосного согласования в начале или в течение связи, сообщения, охватывающие возможности приема и передачи, предпочтения по способам (со стороны получателя) передачи сигналов логического канала данных, управления и индикации. Принятые сигнальные процедуры определены для гарантии надежного аудиовизуального и информационного обмена. В ходе развития ВКС коррекция протокольных стандартов была неизбежной: новые функции должны были непротиворечиво увязываться с существовавшими ранее. Поэтому H.245 пересматривался уже 15 раз (в 1998, 2000, 2003 и 2005 по два раза в год). В мае 2011 г. принята действующая 16-я редакция стандарта.

В марте 1993 г. была принята рекомендация H.231. Этот стандарт описывает блоки управления многоточечными видеоконференциями, использующие циф-

ровой канал до 1920 кбитов/с. Рекомендация определяет средства, которыми три или больше аудиовизуальных терминала, соответствующие H.320, могут связаться одновременно по цифровым каналам с постоянной скоростью передачи данных, такая связь определяется как «многоточечный запрос».

С принятием уже первых версий вышеперечисленной группы стандартов был сформирован базис для внедрения и широкого распространения ВКС в сетях цифровой телефонии с дополнительными услугами (ISDN). Устройства, соответствующие этим стандартам, позволяли осуществлять сеансы видеоконференцсвязи в режимах «точка–точка» и «многоточка» с некоторыми возможностями регулировки разрешения и частоты кадров.

Поскольку мультимедийные потоки ВКС несут гораздо больше информации, чем голосовая цифровая телефония, вопросы информационной безопасности для видеоконференцсвязи также явились предметом рассмотрения ITU-T.

В 1993 г. была принята рекомендация H.233 «Система конфиденциальности для аудиовизуальных услуг». Эта рекомендация описывает часть конфиденциальности системы секретной связи, подходящей для использования в узкополосных сетях предоставления аудиовизуальных услуг, соответствующих H.320, H.221, H.230 и H.242. Хотя для такой системы секретной связи требуется алгоритм кодирования, технические требования к алгоритму конфиденциальности не включены: система обслуживает различные алгоритмы и применима к соединению «точка–точка» между терминалами или между терминалом и многоточечным блоком управления (сервером видеоконференцсвязи MCU – Multipoint Control Unit); возможно расширение на многоточечную работу, в которой нет расшифровки в MCU.

Описание механизма конфиденциальности в пределах H.221 определяет сигналы, сопровождающие метод кодирования и процедуры для его использования (алгоритмы независимые). Определенные параметры, применимые к алгоритмам кодирования, зарегистрированные в ISO 9979, изложены в нормативном Приложении А. Следующие версии стандарта были приняты в 1995 и 2002 гг. В настоящее время действительна 3-я версия.

В 1994 году была принята рекомендация H.234 «Управление ключом шифрования и система идентификации для аудиовизуальных услуг». В стандарте описаны три метода управления ключом шифрования, а именно:

- ISO 8732,
- Diffie-Hellman,
- RSA (буквенная аббревиатура от фамилий Rivest, Shamir и Adleman) – криптографический алгоритм с открытым ключом.

Они применимы к кодированию аудиовизуальных сигналов, переданных в цифровой форме, с использованием структуры кадра H.221. Определенные сообщения управления переданы в пределах сигнала управления кодированием (ECS) канала H.221, структура и использование которого определены в H.233.

В том же 1994 г. была принята рекомендация H.281 «Протокол управления камерой удаленного абонента для видеоконференций, использующих H.224». Эта рекомендация описывает, как можно управлять камерой удаленной системы для H.320-терминалов, применяющих H.224, используя LSD, HSD, или MLP как нижний уровень для операций точка–точка и многоточка. С введением этого

стандарта при проведении дистанционных лекций у оператора появилась возможность показывать лектору слушателя, задающего вопросы, а не всю аудиторию целиком.

Поскольку, как сказано выше, видеоконференцсвязь явила технологическим развитием цифровой телефонии, возник вопрос межсетевого взаимодействия.

В 1998 году была утверждена рекомендация H.246 «Интероперабельность терминалов мультимедиа, соответствующих стандартам ITU H-серии и голосовыми терминалами сетей GSTN (коммутируемые линии голосовой телефонии, Generalised Switched Telephone Network или Public Switched Telephone Network — PSTN) и ISDN».

Эта рекомендация описывает шлюзы, которые обеспечивают требуемую трансляцию управления и потоков носителей, чтобы реализовать межсетевое взаимодействие между терминалами, соответствующими различным протоколам. Она поддерживает межсетевое взаимодействие между терминалами мультимедиа H-серии и:

- другими терминалами мультимедиа H-серии,
- речевыми/телефонными терминалами в сетях GSTN или ISDN,
- V.70 терминалами в сетях GSTN,
- многоканальными приложениями на сети GSTN.

В 1996 г. кроме ряда рекомендаций относящихся к связи по каналам цифровой телефонии, была принята первая версия стандарта H.323 «Мультимедийные системы связи на основе пакетов». В данной рекомендации описываются терминалы и другие объекты, обеспечивающие предоставление мультимедийных служб связи в пакетных сетях связи (PBN — Packet-Switched Backbone Network), которые могут не гарантировать необходимое качество обслуживания. Объекты H.323 могут обеспечивать передачу аудио (голоса), видео (подвижных изображений) и/или данных в реальном времени. Поддержка аудио является обязательной, в то время как передача данных и видео является факультативной, но при их обеспечении требуется, чтобы поддерживался такой общий режим работы, при котором терминалы для указанных видов информации могли бы взаимодействовать между собой. В последующие годы стандарт бурно развивался и становился все более значимым для передачи мультимедийного контента, т. к. сети на основе коммутации пакетов, прежде всего, глобальная сеть Интернет, наращивали число пользователей существенно быстрее, чем сети цифровой телефонии. Следующие версии рекомендации H.323 были приняты в 1998, 1999, 2000, 2003, 2006 и в 2009 гг. В настоящее время действует седьмая версия стандарта.

Одновременно с H.323 в 1996 г. была принята рекомендация H.225 «Сигнальные протоколы вызова и пакетизация аудиовизуальных потоков для систем мультимедийной связи, основанных на передаче пакетов». В этом стандарте описываются средства, с помощью которых сигналы аудиовидеоданных и управления объединяются, кодируются и пакетируются для транспортировки между устройствами H.323 по пакетной сети. Рекомендация включает и использование шлюза H.323, который, в свою очередь, может быть соединен с терминалами H.320, H.324 или H.310/H.321.

В 1996 г. была принята рекомендация H.324 «Терминал для мультимедийной связи, использующей низкоскоростные каналы». Этот стандарт определяет



общие методы сжатия аудио- и видеоданных при организации конференций по-средством высокоскоростных (в. 34 и старше) модемов на аналоговой телефонной линии. Несмотря на длительную эволюцию стандарта (последняя, пятая версия была принята в 2009 г.), он не получил широкого применения в системах видеоконференцсвязи в связи с быстрым ростом широкополосных телекоммуникаций.

В 1996 г. была принята рекомендация H.310 «Широкополосные аудиовизуальные системы связи и терминалы». Вторая (и по настоящее время последняя) версия этого стандарта была принята в 1998 г. Поскольку стандарт был ориентирован на применение в сетях асинхронной передачи данных ATM (Asynchronous Transfer Mode), его развитие фактически было остановлено проигрышем технологии ATM в конкурентной борьбе с технологией Gigabit Ethernet.

В том же 1996 г. был принят новый стандарт кодирования видеоизображения H.263 — «Кодирование видеосигнала для низкоскоростной связи». Базовая конфигурация алгоритма кодирования представляет собой объединение межкадрового предсказания для использования временной избыточности и трансформирующее кодирование остаточного сигнала для снижения пространственной избыточности. Кодер источника может работать в пяти стандартных форматах источника изображения: SQCIF, QCIF, CIF, 4CIF и 16CIF с частотой до 60 кадров/с, и, кроме того, может работать с широким диапазоном пользовательских форматов изображения. Декодер имеет возможность выполнять компенсацию движения, что позволяет дополнительно ввести этот метод в кодер. Точность компенсации движения составляет половину пикселя, в отличие от рекомендации H.261 МСЭ-Т, в которой используется точность, равная целому пикселу. Для передаваемых символов используется кодирование с переменной длиной кодовой посылки. Кроме этого базового алгоритма в рекомендацию включено восемнадцать дополнительных режимов, предназначенных для улучшения характеристик компрессии и поддержки дополнительных возможностей. В бинарный поток может быть введена также дополнительная информация, обеспечивающая использование дополнительных возможностей дисплея, или предназначенная для внешнего использования. Вторая версия рекомендации была принята в 1998 г., а третья, действующая по сей день, — в 2005 г. В 1997, 2000, 2001 и 2004 гг. принимались дополнения, которые привели к некоторой путанице для пользователей, т. к. появились обозначения «H.263+», а затем «H.263++», которыми производители подчеркивали, что в их устройствах работают дополнения к стандарту H.263, но в описаниях выделяли как самостоятельные стандарты.

Параллельно с развитием рекомендации H.263 с мая 2003 г. началось внедрение более прогрессивного стандарта кодирования/декодирования видеосигнала H.264 «Усовершенствованное кодирование видео для универсальных аудиовизуальных услуг». Этот стандарт представил собой развитие прежних стандартов кодирования цифровых видеосигналов (H.261, H.262 и H.263) в связи с растущей потребностью передачи видео все более высокого качества с максимально возможным при заданном качестве сжатием (для видеоконференций, записи видео на носители информации, потокового цифрового видео, в т. ч. в сети Интернет и т. п.). Стандарт содержит 7 профилей и 16 уровней для работы с различными приложениями. К настоящему времени стандарт пересматривался четырежды: в 2005, 2007, 2009 и 2010 гг. Действующей является последняя пятая версия.

В 2000 г. была принята рекомендация H.248.1 «Протокол управления шлюзом». В этой рекомендации, с целью обеспечения лучшей масштабируемости,

функция шлюза H.323, определенная в рекомендации H.246, разбивается на функциональные компоненты, а также определяются протоколы, которые используются этими компонентами для связи. Это позволяет обеспечить высокую степень масштабируемости при реализации шлюзов, а также способствует использованию таких возможностей широко распространенных сетей с коммутацией каналов (SCN), как коммутаторы SS7. Кроме того, открывается возможность создания шлюзов H.323 из компонентов нескольких производителей, распределенных среди нескольких физических платформ. Целью данной рекомендации является дополнение возможностей, определяемых в настоящее время для систем H.323, а также она предназначена для обеспечения новых путей выполнения операций, уже поддержанных в рекомендации ITU-T H.323. Вторая версия стандарта была принята в 2002 г., а в настоящее время действует третья версия, принятая в 2005 г.

В 1998–2001 годах была принята группа рекомендаций H.450.XX, регламентирующая организацию предоставления дополнительных услуг в сетях H.323. Благодаря принятию этих стандартов в видеоконференцсвязи появились функции, характерные для голосовой цифровой телефонии: передача вызова на другого абонента, удержание вызова, переадресация вызова, перевод вызова на фиксированный номер, идентификация вызывающего абонента и др.

В 2002–2010 годах была принята еще одна группа рекомендаций H.460.XX, расширявшая возможности видеоконференцсвязи путем введения ряда соглашений для терминалов и серверов в сетях H.323. Из большого количества стандартов этой группы (на сегодня их 25) можно выделить ряд наиболее важных для большинства пользователей: H.460.8 «Запрос альтернативных маршрутов в пределах H.323-систем», H.460.9 «Поддержка сообщений реального времени QoS-мониторинга», обеспечивающего заданное качество обслуживания, H.460.18 «Проход служебной информации (сигналов) сеанса связи в рамках рекомендации H.323 сквозь систему внутренней адресации локальной сети и межсетевые экраны» и H.460.19 «Проход аудио/видео и иной мультимедийной информации сеанса связи в рамках рекомендации H.323 сквозь систему внутренней адресации локальной сети и межсетевые экраны». Использование двух последних стандартов позволяет пользователям разных локальных сетей связываться друг с другом по видео через два слоя внутренней адресации локальной сети (NAT) и межсетевые экраны (Firewall), используя контроллер зоны (Gatekeeper), находящийся во внешней сети Интернет. NAT (Network Address Translation) – это механизм в сетях TCP/IP, позволяющий преобразовывать IP-адреса транзитных пакетов.

В области кодирования звуковых потоков, несмотря на все большее развитие широкополосных сетей, оказались востребованными стандарты, занимающие меньшую полосу, чем G.711.

В 1992 г. был принят стандарт G.728 «Кодирование речи в полосе 16 кбитов/с с использованием линейного предсказание с мультиковдовым управлением». Стандарт не получил дальнейшего развития, если не считать девяти дополнений к первой версии, действующей до настоящего времени.

В 1996 г. были приняты еще две рекомендации, определяющие правила кодирования звука для узкополосных каналов: G.723.1 «Двухскоростной речевой кодер для мультимедийной связи с передачей на скорости 5,3 или 6,3 кбита/с» и G.729 «Кодирование речи в полосе 8 кбитов/с, использующее линейное пред-

сказание с возбуждением по алгебраической кодовой книге». Рекомендация G.723.1 была пересмотрена в 2006 г., рекомендация G.729 — в 2007 г. В настоящее время действуют вторые версии этих стандартов. Использование этих рекомендаций оказалось наиболее актуальным при проведении сеансов многоточечной видеоконференцсвязи с большим количеством участников, когда на каждого абонента может приходиться лишь малая часть общего широкополосного канала связи.

Особое место в развитии ВКС занимают стандарты, связанные с передачей данных в том же потоке, что и мультимедийная информация — группы рекомендаций Т.120 (Т.120–Т.128). Первая редакция базового стандарта группы, Т.120, была принята в 1996 г., вторая, действующая, — в 2007 г.

Вначале Т.120 был интегрирован в Н.323 (первую версию), но это существенно усложняло процедуры установления и поддержания соединений и накладывало ряд ограничений на программное обеспечение терминалов ВКС. Начиная со второй версии Н.323 было проведено разделение стандартов, и Т.120 стал опциональной частью ВКС, осуществляющей вызовы через инструменты Н.323.

Поскольку на ранних стадиях развития ВКС базировалась на персональных компьютерах (например, вставлялась плата функционального расширения, к которой подключалась акустика и видеокамера, загружалось специфическое программное обеспечение), на эти стандарты возлагались большие надежды. В составе одной из первых программ ВКС Microsoft NetMeeting на базе Т.120 предоставлялись функции «белой доски» и совместной работы пользователей над общим документом в офисных и иных приложениях.

На первых порах стандарт поддерживался не только компьютер-ассоциированными терминалами ВКС, но и аппаратно-программными, связанными по локальной сети с «ассистентскими» компьютерами, в которые загружалось специальное ПО (например, WEB-интерфейс Polycom-FX). Однако по мере выпуска рядом фирм-производителей аппаратно-программных терминалов, не связанных с компьютерами, сфера применения группы рекомендаций Т.120 (переведенных в разряд опций) все больше сужалась. Кроме того, жесткие требования в ранних версиях стандартов к идентичности прикладного ПО, в котором идет совместная работа над документом у всех участников распределенной дистанционной работы, снижало количество приверженцев рекомендаций даже среди тех, кто имел физические возможности их применения. Вероятно поэтому, эволюция этой группы стандартов продолжалась с 1993 по 1999 год, а далее в течение семи лет не появлялось новых редакций.

Отдельно следует сказать о рекомендации Н.239. Первая версия этой рекомендации была принята в 2003 г., вторая — в 2005 г. Данный стандарт впервые позволил передавать с терминала одного абонента два или более видеопотоков с раздельной регулировкой параметров для каждого из них.

Так, например, в области дистанционного образования, которым охватывается все большее количество обучаемых на всей планете (особенно в малонаселенных регионах, удаленных от крупных университетских центров), двухпотоковый режим является предпочтительным при чтении лекций, проведении семинаров или демонстраций лабораторных занятий. Получатели контента при этом на первом экране видят лектора, а на втором — его презентацию или демонстрацию лабораторной работы. Для видеопотока с лектором обычно достаточно разрешения CIF при 25 кадрах в секунду, а презентация, в которой могут быть текстовые

фрагменты, требует разрешения не менее 1024×768 , однако для статических слайдов достаточно частоты в 5 кадров/с и менее. Такое разделение качества позволяет проводить лекции и семинары в двухпотоковом режиме даже для тех пользователей, у которых ширина полосы «последней мили» не превышает 768 кбитов/с. Не меньшую роль рекомендация H.239 играет и в телемедицине.

Так при проведении дистанционных консультаций консультант может видеть на одном экране положение руки диагностика с датчиком УЗИ на теле пациента, а на втором — изображение на выходе диагностического прибора. Например, в детской кардиохирургии консультант нередко оценивает степень дисфункции клапанного аппарата сердца маленького пациента при врожденных сочетанных пороках на основании дистанционного эхокардиографического исследования. Это позволяет определить необходимость и сроки оперативного лечения в специализированном стационаре, а также объем предстоящего оперативного вмешательства еще до приезда пациента в клинику.

В табл. 23.1 приведено краткое описание основных рекомендаций ITU-T, используемых в системах видеоконференцсвязи.

Таблица 23.1. Краткая характеристика рекомендаций ITU-T [9.76–9.142]

№	Рек. №	Название/назначение рекомендаций
Серия H: Аудиовизуальные и мультимедийные системы (H Series: Audiovisual and Multimedia Systems)		
1	H.221	Структура кадров для телекоммуникационных каналов аудиовизуальных услуг в полосе от 64 до 1920 кбитов/с.
2	H.222.0	Эта рекомендация/международный стандарт определяет порождающие методы для мультиплексирования мультимедиа, синхронизации и восстановления временной шкалы. Технические требования обеспечивают передачу мультимедиа, основанную на мультиплексировании пакетов, где каждый элементарный поток битов сегментирован в пакетезированный элементарный поток (PES), и затем соответствующие пакеты мультиплексированы в любой из двух типов потоков: программный поток (PS), который является мультиплексным из PES пакетов переменной длины и разработан для использования в безшибочном окружении, или транспортный поток (TS), который состоит из 188-байтных пакетов фиксированной длины, имеет функциональные возможности много-кратного мультиплексирования программы, так же как мультиплексирования различных PES пакетов, и разработан для использования в окружении, допускающем ошибки. Мультимедийная синхронизация и восстановление временной шкалы достигаются с помощью временных меток для системного таймера и представления/декодирования.
3	H.222.1	Эта рекомендация описывает мультиплексирование и синхронизацию мультимедийной информации для аудиовизуальных коммуникаций в ATM-средах. Она определяет H.222.1 программный поток и H.222.1 транспортный поток, выбирая необходимые элементы кодирования порождающих H.222.0 технических требований и добавляя пункты для использования в ATM-средах. Она также охватывает методы для приспособления элементарных потоков.
4	H.224	Протокол управления в реальном времени для односторонних приложений, использующих каналы LSD/HSD/MLP, предусмотренные рекомендациями H.221 и H.243 (напр. управление удаленной камерой или пересылка текстовых сообщений).



Таблица 23.1 (продолжение)

5	H.225.0	Протоколы сигнализации о соединении и пакетирование аудиовизуальных потоков для систем мультимедийной связи, основанных на передаче пакетов. В этом стандарте описываются средства, с помощью которых сигналы аудио-видеоданных и управления объединяются, кодируются и пакетируются для транспортировки между устройствами H.323 по пакетной сети. Рекомендация включает и использование шлюза H.323, который, в свою очередь, может быть соединен с терминалами H.320, H.324 или H.310/H.321.
6	H.231	Блоки управления многоточечными видеоконференциями, использующие цифровой канал до 1920 кбитов/с. Эта рекомендация определяет средства, которыми три или больше аудиовизуальных терминалов, соответствующие H.320, могут связаться одновременно по цифровым каналам с постоянной скоростью передачи данных, такая связь определяется как «многоточечный запрос».
7	H.233	Система конфиденциальности для аудиовизуальных услуг. Эта рекомендация описывает часть конфиденциальности системы секретной связи, подходящей для использования в узкополосных сетях предоставления аудиовизуальных услуг, соответствующих H.320, H.221, H.230 и H.242. Хотя алгоритм кодирования требуется для такой системы секретной связи, технические требования такого алгоритма сюда не включены: система обслуживает более одного конкретного алгоритма. Система конфиденциальности является применимой к соединению «точка-точка» между терминалами или между терминалом и многоточечным блоком управления; это может быть расширено на многоточечную работу, в которой нет расшифровки в MCU. Описание механизма конфиденциальности в пределах H.221 определяет сигналы, сопровождающие метод кодирования и процедуры для использования. Определенные параметры, применимые к алгоритмам кодирования, зарегистрированные в ISO 9979, изложены в нормативном Приложении А.
8	H.234	Управление ключом шифрования и система идентификации для аудиовизуальных услуг. Описаны три метода управления ключом шифрования, а именно: ISO 8732, Diffie-Hellman и RSA. Они применимы к кодированию аудиовизуальных сигналов, переданных в цифровой форме с использованием структуры кадра H.221. Определенные сообщения управления переданы в пределах сигнала управления кодированием (ECS) канала H.221, структура и использование которого определены в H.233.
9	H.239	Распределение приоритетов аудиовизуальных потоков и добавление каналов с независимыми параметрами для терминалов серии H.3xx. В данной рекомендации определяются процедуры использования более одного видеоканала в системах на базе H.3xx и присвоения отдельным каналам маркировки «роли», указывающей требования к обработке канала и роль содержимого канала в вызове. Ролевые маркировки применимы к системам сигнализации, как H.320, так и H.245. Определенные процедуры охватывают управление, индикацию и механизмы обмена возможностями.
10	H.241	Расширенные процедуры видеообработки и управляющие сигналы для терминалов серии H.300. В настоящей рекомендации определяется использование новейших видеокодеков, включая терминалы, соответствующие рекомендациям МСЭ-Т H.310, H.320, H.321, H.322, H.323 и H.324. В ней также определены общие расширенные решения сигнализации, предназначенные для использования во всех видеокодеках терминалов серии H.300.
11	H.242	Системы для установления связи между аудиовизуальными терминалами с использованием цифровых каналов до 2 Мбитов/с. Эта рекомендация касается эксплуатации системы для конференцсвязи между тремя или больше аудиовизуальными терминалами, соответствующими H.320.

Таблица 23.1 (продолжение)

12	H.243	Процедуры для установления связи между тремя и более аудиовизуальными окончечными устройствами с использованием цифровых каналов со скоростями до 1920 кбитов/с. В этой рекомендации описывается работа системы для конференцсвязи между тремя и более аудиовизуальными окончечными устройствами, соответствующими рекомендации МСЭ-Т Н.320.
13	H.245	Протокол управления для мультимедийной связи. Эта рекомендация определяет синтаксис и семантику информационных сообщений терминалов и процедур, чтобы использовать их для внутриполосного согласования в начале или в течение связи. Сообщения, покрывающие принимающие и передающие возможности, так же как предпочтение способа со стороны получателя, передачи сигналов логического канала, управления и индикации. Признанные сигнальные процедуры определены для гарантии надежного аудиовизуального и информационного обмена.
14	H.246	Интероперабельность терминалов мультимедиа, соответствующих стандартам МСЭ Н-серии, и голосовыми терминалами сетей GSTN и ISDN. Эта рекомендация описывает шлюзы, которые обеспечивают требуемую трансляцию управления и потоков носителей, чтобы позволить межсетевое взаимодействие между терминалами, выполняющими различные протоколы. Это поддерживает межсетевое взаимодействие между терминалами мультимедиа Н-серии и: <ul style="list-style-type: none"> - другими терминалами мультимедиа Н-серии, - речевыми/телефонными терминалами в сетях GSTN или ISDN, - V.70 терминалами в сетях GSTN, - многоканальными приложениями на сети GSTN.
15	H.248.1	Протокол управления шлюзом. В этой рекомендации, с целью обеспечения лучшей масштабируемости, функция шлюза Н.323, определенная в рекомендации Н.246, разбивается на функциональные компоненты, а также определяются протоколы, которые используются этими компонентами для связи. Это позволяет обеспечить высокую степень масштабируемости шлюзов, а также способствует использованию таких возможностей широко распространенных сетей с коммутацией каналов (SCN), как коммутаторы SS7. Кроме того, открывается возможность создания шлюзов Н.323 из компонентов нескольких производителей, распределенных среди нескольких физических платформ. Целью данной рекомендации является дополнение возможностей, определяемых в настоящее время для систем Н.323, а также она предназначена для обеспечения новых путей выполнения операций, уже поддержанных в рекомендации МСЭ-Т Н.323.
16	H.261	Видеокодек для передачи аудиовидеоданных в канале $p \times 64$ кбита/с. Значение p может меняться в диапазоне от 1 до 30. Кодирование основано на ДКП, блочном предсказании движения и использовании кодов переменной длины. Рекомендация регламентирует алгоритмы кодирования для передачи видео в форматах CIF и QCIF с частотой до 30 кадров/с.
17	H.263	Кодирование видео для низкоскоростных соединений — метод кодирования, разработанный для Н.324, использующий технологию Н.261 с дополнительными усовершенствованиями. Форматы SQCIF, QCIF, CIF, 4CIF и 16CIF с частотой до 60 кадров/с.
18	H.264	Усовершенствованное кодирование видео — метод кодирования видеосигнала. Стандарт содержит 7 профилей и 16 уровней для работы с различными приложениями. Он рассчитан на различные форматы сигнала изображения — SQCIF (128 × 96) до ТВ сверхвысокой четкости (4096×2304, 16:9).
19	H.281	Протокол управления камерой удаленного абонента для видеоконференций, использующих Н.224. Эта рекомендация описывает, как можно управлять камерой удаленной системы для Н.3xx-терминалов, использующих Н.224, используя LSD, HSD, или MLP как нижний уровень для операций точка-точка и много-точка.

Таблица 23.1 (продолжение)

20	H.310	Широкополосные аудиовизуальные системы связи и терминалы. Эта рекомендация охватывает технические требования для систем и терминалов широкополосных аудиовизуальных служб связи, определенных в рекомендациях серий H.200/AV.100. Определены как односторонние, так и двунаправленные широкополосные аудиовизуальные терминалы H.310. Классификация терминалов H.310 на различные типы базируется на комплексе свойств, связанных с аудиовизуальными особенностями, адаптацией сети и возможностью обмена сигналами. С этими возможностями терминалы H.310 поддерживают широкий диапазон диалоговых и распределенных приложений и услуг.
21	H.320	Узкополосные видеотелефонные системы и оконечное оборудование. Настоящая рекомендация определяет технические требования к узкополосным видеотелефонным системам и оконечному оборудованию в основном для услуг видеоконференций и видеотелефонии. Она описывает общую конфигурацию системы, состоящей из нескольких элементов, которые определяются соответствующими рекомендациями МСЭ-Т, а также определение режимов связи и типов терминалов, организацию управления соединением, терминальные аспекты и требования к взаимодействию. Последняя пересмотренная версия H.320 представляет ряд улучшений и разъяснений предыдущей версии, главным образом — описание применений рекомендаций МСЭ-Т G.722.1, H.239, H.241, H.264 и стандарта ИСО/МЭК 14496-3 в системах H.320.
22	H.321	Адаптация видеотерминалов H.320 к окружению B-ISDN. Эта рекомендация описывает технические требования к приспособлению узкополосных аудиовизуальных терминалов, определенных в рекомендации H.320, к окружению широкополосной ISDN. Терминал, соответствующий этой рекомендации, взаимодействует с тем же самым типом терминалов (т.е. с другими H.321-терминалами), размещенных в сети B-ISDN так же, как с H.320 терминалами, размещенными в сети N-ISDN.
23	H.322	Визуальные телефонные системы и терминальное оборудование для локальных сетей, которые обеспечивают гарантированное качество обслуживания. Эта рекомендация охватывает технические требования для узкополосных визуальных телефонных услуг, определенных в рекомендациях серий H.200/AV.120, в тех ситуациях, где тракт передачи включает одну или более локальных сетей (ЛВС), каждая из которых формируется и управляет, чтобы обеспечить гарантированное качество обслуживания (QoS — Quality Of Service) эквивалентное N-ISDN такое, что не требуется дополнительной защиты или механизмов восстановления свыше тех, которые указаны в рекомендации H.320, которые было необходимо обеспечивать в терминалах.
24	H.323	Мультимедийные системы связи на основе пакетов. В данной рекомендации описываются терминалы и другие объекты, обеспечивающие предоставление мультимедийных служб связи в пакетных сетях связи (PBN), которые могут не обеспечивать гарантированное качество обслуживания. Объекты H.323 могут обеспечивать передачу аудио- (голоса)-, видео- (подвижных изображений)- и/или данных в реальном времени. Поддержка аудио является обязательной, в то время как передача данных и видео являются факультативными, но при их обеспечении требуется, чтобы поддерживался такой общий режим работы, при котором терминалы для указанных видов информации могли бы взаимодействовать между собой.

Таблица 23.1 (продолжение)

25	H.324	Терминал для низкоскоростных мультимедийных сетей связи. Рекомендация ITU-T H.324 описывает терминалы для низкоскоростной мультимедийной системы связи, использующей ITU-T V.34 модемы, работающие по коммутируемой телефонной сети общего пользования (GSTN), пользовательский сетевой интерфейс I.400 для ISDN или соответствующий беспроводной интерфейс для подвижных сетей связи. ITU-T H.324-терминал может передавать голос в реальном масштабе времени, данные и видео или любую комбинацию, включая видеотелефонию.
26	H.331	Многоточечные системы и терминальное оборудование широковещательного типа. Эта рекомендация охватывает технические требования для широковещательного типа многоточечных систем и терминального оборудования, которые позволяют аудиовизуальный сигнал от передающего терминала передать многим принимающим терминалам, используя функцию распределения сигналов коммутаторов ISDN. Используются каналы ISDN (B, H0, составленные из них или H11/H12), обеспечивающие пропускную способность, не превышающую 1920 кбитов/с. Существенная техническая особенность в этой системе — то, что каналы связи не могут быть двунаправленными между терминалом, представляющим информацию, и любым из получающих терминалов, потому что тракт передачи от принимающего терминала до передающего терминала может быть прерван при обмене. Правила связи не могут основываться на переговорах. Этот факт требует рекомендации, отдельной от рекомендации H.320, но основанный на ней и обслуживающей специфические потребности широковещательного типа многоточечных систем так, чтобы терминалы, работающие и для H.320, и для этой рекомендации, были легко осуществлены.
27	H.332	Эта рекомендация описывает терминалы, оборудование и услуги для системы мультимедийной связи по сетям на основе коммутации пакетов. H.332-терминалы и оборудование могут в реальном масштабе времени обмениваться голосом, данными, и видео в любой комбинации. Цель состоит в том, чтобы, работая в пределах H.323-структур, все же обеспечивать хорошую масштабируемость за пределами рекомендации H.323, чтобы дать возможность реализовывать приложения, вовлекающие сотни и тысячи участников. H.332-терминалы могут быть интегрированы в персональные компьютеры или реализованы в форме автономных устройств типа видеотелефонов. Поддержка голоса в них обязательна, в то время как данные и видео являются опциями, но если поддерживаются, то требуется возможность использования определенного синфазного сигнала управления, такого, что все терминалы, поддерживающие эти средства, могли бы взаимодействовать. Подобно рекомендации H.323, эта рекомендация позволяет использовать больше одного канала каждого типа. Другие рекомендации в H.332-ряде включают H.323 визуальное телефонное оборудование для локальных сетей, пакеты H.225.0 и синхронизацию, управление H.245, кодеки видеосигнала H.261 и H.263 и кодеки аудио G.711, G.722, G.728, G.729, G.723.1, ряд протоколов интегральных систем связи Т.120. H.332-терминалы базируются на рекомендации H.323, поэтому они взаимодействуют через шлюзы с H.310-терминалами на B-ISDN, H.320-терминалами на N-ISDN, H.321-терминалами на B-ISDN, H.322-терминалами на гарантированных QoS ЛВС, H.324-терминалами на GSTN и беспроводных сетях, и V.70-терминалах на GSTN.
28	H.450.1	Групповой функциональный протокол для поддержания дополнительных услуг в H.323.
29	H.450.2	Передача вызова на другого абонента в дополнительных услугах для H.323.
30	H.450.3	Переадресация вызова в дополнительных услугах для H.323.
31	H.450.4	Удержание вызова в дополнительных услугах для H.323.

Таблица 23.1 (продолжение)

32	H.450.5	Перевод вызова на фиксированный номер, перехват звонка (перенаправление звонка, инициируемое посторонним абонентом — получателем) в дополнительных услугах для H.323.
33	H.450.6	Ожидание вызова (уведомление абонента о поступлении нового входящего звонка во время разговора с другим абонентом) в дополнительных услугах для H.323.
34	H.450.7	Индикация ожидающего вызова в дополнительных услугах для H.323.
35	H.450.8	Идентификация вызывающего абонента в дополнительных услугах для H.323.
36	H.450.9	Успешное (штатное) завершение вызова (сессии связи) в дополнительных услугах для H.323.
37	H.450.10	Предложение установить сеанс связи.
38	H.450.11	Индикация (сигнал) вмешательства.
39	H.450.12	Сообщения о дополнительных свойствах сети.
40	H.460.1	Руководящие указания для использования общих принципов расширяемости.
41	H.460.4	Определение (обозначение) приоритетов вызовов.
42	H.460.8	Запрос альтернативных маршрутов в пределах H.323-систем.
43	H.460.9	Поддержка сообщений реального времени QoS-мониторинга.
44	H.460.10	Категории соединений в пределах H.323-систем.
45	H.460.11	Отсроченное установление соединения в пределах H.323-систем.
46	H.460.12	Индикатор управления в случае запроса одного ресурса (одной линии) двумя объектами в пределах H.323 систем.
47	H.460.13	Управление передачей вызова звонящим пользователем.
48	H.460.14	Поддержка для многоуровневой очередности и последовательности (MLPP) в пределах H.323-систем.
49	H.460.15	Определение механизма приостановки и восстановления запросов и переназначения каналов в пределах H.323-систем с использованием шлюза (Gatekeeper).
50	H.460.16	Возможность реализации последовательности многократного сообщения.
51	H.460.17	Запрос сигнального соединения с использованием H.225.0 в качестве транспорта для сообщений сервера удаленного доступа.
52	H.460.18	Проход служебной информации (сигналов) сеанса связи в рамках рекомендации H.323 сквозь систему внутренней адресации локальной сети и межсетевые экраны.
53	H.460.19	Проход аудио/видео и иной мультимедийной информации сеанса связи в рамках рекомендации H.323 сквозь систему внутренней адресации локальной сети и межсетевые экраны.
54	H.460.20	Номер вызова (локальный адрес) в пределах/вне H.323-систем.
Серия G: Передающие системы и медиа, цифровые системы и цепи (G Series: Transmission Systems and Media, Digital Systems and Networks)		
55	G.711	Способ кодирования звука, обеспечивающий полосу аудиосигнала примерно 3,5 кГц в канале 64 кбита/с.

Таблица 23.1 (продолжение)

56	G.723.1	Один из базовых кодеков для приложений IP-телефонии. Утвержден МСЭ-Т в ноябре 1995 года. Выбран форумом IMTC как базовый для приложений IP-телефонии. Кодек G.723.1 генерирует кадры длительностью 30 мс, предварительно анализируя 7,5 мс сигнала. Предусмотрено два режима работы: 6,4 кбита/с (кадр имеет размер 189 битов, дополненных до 24 байтов) и 5,3 кбита/с (кадр имеет размер 158 битов, дополненных до 20 байтов). Режим работы может меняться динамически от кадра к кадру. Оба режима обязательны для реализации. Оценка MOS (Mean Opinion Score) составляет 3,9 в режиме 6,3 кбитов/с и 3,7 в режиме 5,3 кбита/с. Кодек специфицирован на основе операций как с плавающей точкой, так и с фиксированной точкой в виде кода на языке Си. Реализация кодека на процессоре с фиксированной точкой требует производительности около 16 MIPS. Кодек G.723.1 имеет детектор речевой активности (VAD) и обеспечивает генерацию комфортного шума на удаленном конце в период молчания. Эти функции специфицированы в приложении А. Параметры фонового шума кодируются очень маленькими кадрами размером 4 байта. Если параметры шума не меняются существенно, передача полностью прекращается.
57	G.728	Кодирование речи в полосе 16 кбитов/с, используя линейное предсказание с мультикововым управлением.
58	G.729	Кодирование речи в полосе 8 кбитов/с, использующее линейное предсказание с возбуждением по алгебраической кодовой книге. Семейство включает кодеки G.729, G.729 Annex A, G.729 Annex B (содержит VAD и генератор комфортного шума). Кодеки G.729 сокращенно называют CS-ACELP Conjugate Structure — Algebraic Code Excited Linear Prediction — сопряженная структура с управляемым алгебраическим кодом линейным предсказанием. Процесс преобразования использует 21,5 MIPS и вносит задержку 15 мс.
Т-серия: терминалы для телематических сервисов (T Series: Terminals for Telematic Services)		
59	T.120	Процедуры пересылки и совместного использования данных и приложений двумя и более пользователями видеоконференции (группа рекомендаций).
60	T.121	Рекомендация МСЭ-Т, описывающая шаблоны общего назначения. Эта рекомендация предоставляет руководство для разработчиков приложений и прикладных протоколов по правильному и эффективному использованию инфраструктуры Т.120. Она предоставляет базовую модель для приложения, которое взаимодействует, используя сервисы Т.120, и определяет базовый прикладной шаблон, задающий правила использования сервисов Т.122 (MCS) и Т.124 (GCC), которые обычно требуются прикладными протоколами, использующими сервисы Т.120. Эта рекомендация предоставляет общую структуру для компонентов прикладного протокола, гарантируя, что отдельные компоненты прикладного протокола могут существовать в той же самой конференции. Компоненты прикладного протокола, которые не следуют операциям, определенным в этой рекомендации, рисуют конфликтовать с другими прикладными протоколами.

Таблица 23.1 (продолжение)

61	T.122	Рекомендации МСЭ-Т, описывающие обслуживание, обеспечивающее связь с многоточечным управлением (MCS) для аудиографической и аудиовизуальной конференционной связи (определения). Эта рекомендация определяет многоточечную службу доставки данных для использования в аудиографических и аудиовизуальных сервисах конференцсвязи. Она предоставляет механизм рассылки всем или подмножеству из группы с одиночной посылкой прimitива и принуждением, если нужно, получения единой последовательности данных всеми пользователями для приложений, поддерживающих многоточечный режим. Она также предоставляет механизм права доступа, позволяющий приложениям управлять ограниченными ресурсами или осуществлять многозадачную сигнализацию и синхронизацию. Эти сервисы предоставляются способом, который является независимым от типа сети, используемой данным соединением.
62	T.123	Рекомендация, описывающая протокол стека для аудиографической и аудиовизуальной конференцсвязи. Рекомендация Т.123 определяет аспекты протоколов данных Т.120-ряда для мультимедийной конференцсвязи, связанные со спецификой сети. Современные сети идентифицируются как цифровая сеть с интеграцией служб (ISDN), сеть передачи данных с коммутацией каналов (CSDN), сеть с коммутацией информационных пакетов (PSDN), коммутируемая телефонная сеть общего пользования (PSTN), широкополосная ISDN (B-ISDN), и ЛВС. Профили связи определены как предоставляющие надежные соединения точка-точка между терминалами и сервером многоточки, между парами терминалов или между парами серверов многоточки (MCU). В некоторых случаях более низкий уровень протокола позволяет мультиплексирование звуковых и видеосигналов в дополнение к соединениям данных. В других случаях, по той же самой или иной сети, могут быть установлены отдельные соединения для передачи звуковых или видеосигналов.
63	T.124	Рекомендация, описывающая общее управление конференционной связью для аудиографических и аудиовизуальных терминалов. Рекомендация Т.124 предоставляет высокоуровневую структуру для управления конференцией и управления мультимедийными терминалами и серверами многоточки. Она охватывает базовое управление конференциями (GCC), т. е. функциями типа установления соединения и завершения конференции, управления списком терминалов, участвующих в конференции, управления списком приложений и возможностями приложений в ходе конференции, регистрации сервисов используемых приложениями, координации проведения конференцией, а также другими разнообразными функциями. Она зависит от сопряженных рекомендаций Т.122, Т.125 (MSC) и Т.123 как части инфраструктуры семейства Т.120.
64	T.125	Эта рекомендация определяет протокол, работающий через иерархию области многоточечной связи. Она определяет формат сообщений протокола и процедур, управляющих их обменом по набору транспортных связей. Цель протокола состоит в том, чтобы осуществить обслуживание многоточечной связи, определенное в соответствии с рекомендацией Т.122.

Таблица 23.1 (окончание)

65	T.126	Рекомендация, описывающая перечень потоколов передачи неподвижных изображений в режиме телеконференционной связи. Рекомендация Т.126 определяет протокол, поддерживающий управление общих многослойных визуальных пространств и многоточечного обмена графической информацией, направленной в эти пространства, включая изображения («твёрдая» и «мягкая» копия), указатели, и заполненные и незаполненные параметрические элементы рисунка (точки, линии, многоугольники и эллипсы). Также включена поддержка визуализации внеполосных видеопотоков в пределах рабочих пространств Т.126. Кроме того, определены «обмены» клавиатурой и устройствами управления курсором для поддержки базового взаимодействия пользователей. Определены элементы протокола для создания визуальных пространств и ссылок на архивированные визуальные пространства, чтобы позволить каталогизировать ранее сохраненные или распределенные графические материалы. Этот протокол использует сервисы, обеспеченные в соответствии с рекомендациями Т.122 (MCS) и Т.124 (GCC) и исполняет руководящие принципы, указанные в рекомендации Т.121 (GAT).
66	T.127	Рекомендация Т.127 определяет протокол для поддержки обмена файлами двоичных данных в пределах диалоговой конференцсвязи или группы, работающей в среде, использующей семейство стандартов Т.120. Она предоставляет механизмы для поддержки синхронного распространения множественных файлов, селективное распространение файлов к подмножеству участников и получения файлов от удаленных сайтов. Также принято соглашение для прямого доступа к удаленному сайту.
67	T.128	Рекомендация Т.128 определяет протокол, который поддерживает разделение приложений при многоточечной конференции. Т.128 протокол поддерживает разделение компьютерных приложений при многоточечной конференции, позволяя видеть программное приложение, выполняемое на одном сайте (компьютере, включенном в сеть) в пределах сеанса связи другим сайтам. Каждый сайт при точно установленных условиях может взять под свой контроль разделенное приложение путем передачи команды отдаленной клавиатуры и устройств управления курсором. Этот стиль разделения приложений не требует и не создает условий для того, чтобы синхронизировать множественные реализации одного и того же приложения на множестве сайтов. Вместо этого это позволяет дистанционно наблюдать и управлять единственным приложением с иллюзией, что приложение является загруженным локально.

Сегодня большая часть видеоконференций проводится по общедоступной сети Интернет с применением рекомендации H.323 и сопряженных стандартов. Как видно из схемы рис. 23.1, в ходе совместной эволюции этой группы стандартов и видеоконференцсвязи последняя развивалась с «креном» в сторону обработки мультимедийного контента и оптимизации функций управления. В то же время, передача и обработка данных в течение ряда лет оставались в стороне от основного пути развития.

На приведенной схеме стандарт V.150 (модем для IP сетей — организация, основные принципы) относится к телефонии в IP-сетях, T.38 (процедуры факсимильных коммуникаций Группы 3 для IP сетей) — к факсимильной связи, и только группа Т.120 — к видеоконференциям.

Однако развитие вычислительной техники и телекоммуникационной инфраструктуры, включая мобильный Интернет, и появление новых программных коммуникационных платформ дали новый виток развития стандартам, связанным с обменом данными.

К 2007 году на рынке видеоконференцсвязи сформировалась новая тенденция. Если раньше передача данных и совместная работа с приложениями были

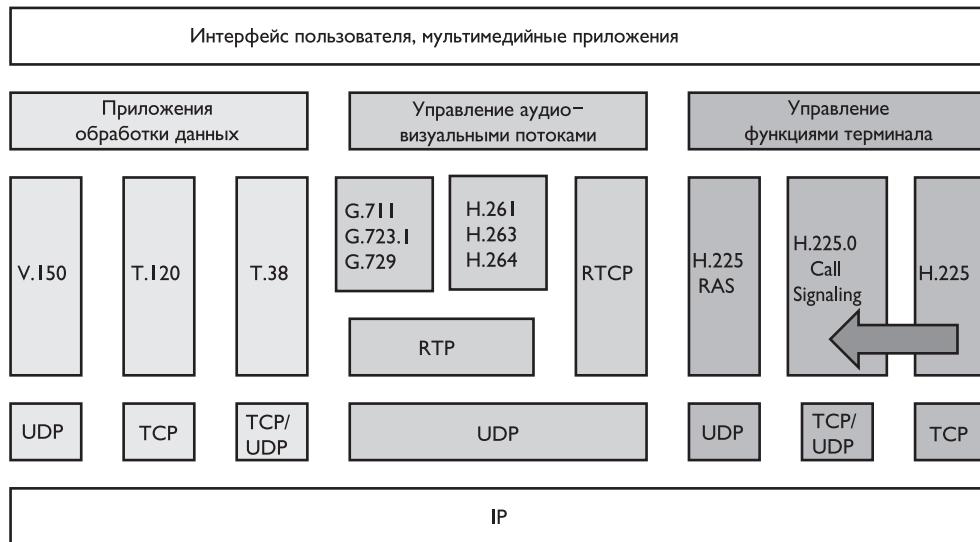


Рис. 23.1. Структура взаимодействия стандартов (стека протоколов) в рамках рекомендации H.323

опциями к собственно «видеотелефонии высокого качества», то теперь появились интегрированные среды для распределенной работы нескольких пользователей в сети, в которых уже видеоконференцсвязь стала одной из удобных опций. Примером такого интегрированного программного продукта для общения сотрудников территориально распределенных предприятий может служить Cisco Unified Communications, объединяющий практически все варианты коммуникаций: голос, видео, обмен текстовыми сообщениями, совместное использование приложений, в том числе и с использованием мобильных платформ и беспроводного доступа в Интернет.

23.2. Возможности технологии видеоконференцсвязи

В России технологии ВКС появились несколько позже, чем в странах Америки и Европы, однако отставание не было фатальным. Так уже в 1997 г. началась реализация телемедицинского проекта «Москва–регионы России», в основе которого — дистанционное консультирование с применением ВКС ведущими столичными специалистами своих коллег из регионов по сложным клиническим случаям [9.143, 9.144]. К настоящему времени проектом, которым руководит Российской ассоциацией телемедицины, охвачено более 130 телемедицинских центров и пунктов в большинстве российских регионов. В 1999 г. в судебной системе Российской Федерации впервые в мировой практике началось внедрение технологий ВКС для проведения судебных процессов. За десять лет в судах общей юрисдикции и СИЗО было установлено более тысячи терминалов ВКС, с помощью которых уже рассмотрено более 350 тысяч уголовных дел [9.145].

Последние годы на российском рынке телекоммуникационных услуг можно охарактеризовать как период бурного роста практического интереса к системам видеоконференцсвязи. Достаточно посмотреть публикации по этой теме в Интернете, чтобы убедиться, что эта технология год за годом все шире внедрялась в бизнес и во все новые сферы общественной жизни [9.146, 9.147].

Однако большинство публикаций или напрямую проплачены производителями и продавцами конкретного оборудования ВКС, что хорошо видно из предпочтений авторов при описании возможных решений, или написаны с позиций анализа рынка.

В то же время точки зрения *потребителя* средств ВКС и *отечественных разработчиков* в этой сфере остаются вне сферы внимания профильной прессы и сетевых информационных ресурсов.

Пользователи реальные и потенциальные

Любая новая техника и технология предназначены для использования определенным кругом организаций и частных пользователей.

Крупные компании, имеющие распределенную структуру и филиалы на разных континентах, безусловно, выигрывают в оперативности управления и экономят средства при широком внедрении ВКС за счет сокращения командировочных расходов и своевременности принятия бизнес-решений [9.148].

Особое значение приобретают ВКС для госструктур. Во-первых, для силовых ведомств, для которых оперативность управления едва ли не синоним успешности выполнения операций [9.149, 9.150].

Во-вторых, для служб, ликвидирующих последствия чрезвычайных ситуаций [9.151, 9.154]. Особое значение приобретают мобильные ВКС, которые помогают МЧС быстро развертывать специализированную инфраструктуру при масштабных авариях или природных катастрофах, что обеспечивает взаимодействие различных служб, работающих в подобных случаях.

В-третьих, для судебной и пенитенциарной систем [9.145]. Применение ВКС при повторном рассмотрении дел заключенных или при рассмотрении новых дел позволяет отказаться от этапирования осужденного из удаленных мест лишения свободы к месту проведения судебного заседания, а также привлечь к участию в нем свидетелей, находящихся в других населенных пунктах вне зависимости от их удаленности [9.152].

В-четвертых, в последнее время значительное количество совещаний центр-регионы проводится высшим руководством страны в режиме ВКС. Губернаторы и другие служащие докладывают президенту или премьеру о состоянии дел в регионах и ходе выполнения поручений руководства [9.153, 9.154].

В-пятых, руководство регионального уровня также переходит к проведению территориально распределенных совещаний вместо вызова подчиненных непосредственно в региональный центр [9.155].

Все более значимую роль играет видеоконференцсвязь в медицине [9.156–9.158]. Телемедицинские консультации, консилиумы, тематические дистанционные циклы повышения квалификации, отдельные лекции проводятся все большим числом ведущих клинических учреждений столицы и региональных центров. Только в Москве сегодня с различной интенсивностью работают более 20 консультирующих телемедицинских центров и пунктов.



Дистанционное образование с применением видеоконференцсвязи внедряется многими вузами страны. В вузах, имеющих филиалы в других городах (регионах), как правило, вначале ВКС используется для совещаний руководящего состава и преподавателей, повышения квалификации и тренингов сотрудников [9.159], а затем — для чтения лекций, проведения семинаров и даже для сдачи зачетов и экзаменов [9.160].

Из многочисленных публикаций в Интернете может сложиться впечатление об успешности внедрения технологий ВКС в мире вообще и в России в частности. Так ли это? Для ответа на этот вопрос достаточно ответить на два других:

- Есть ли у большинства пользователей телефонной связи желание видеть собеседника при разговоре?
- Каково количественное соотношение мобильных телефонов и терминалов видеоконференцсвязи?

На первый вопрос можно ответить коротко — да.

На второй ответить сложнее, т. к. реальные цифры по продажам терминалов ВКС найти не так просто. В то же время очевидно, что это соотношение различается порядками. Даже если отнести к пользователям ВКС всех, кто пользуется «Скайпом» и аналогичными программами, все равно речь будет идти о миллионах. Пользователей же мобильной связи — миллиарды. Если же учитывать только пользователей стандартных аппаратно-программных и программных терминалов ВКС, то соотношение будет не менее чем с пятью нулями.

Отдельно следует сказать о двух направлениях потенциально широкого внедрения технологий видеоконференцсвязи.

Во-первых, это — дистанционное образование. Не секрет, что в нашей стране вузы с сильным преподавательским составом распределены крайне неравномерно. А желание юных граждан из любого населенного пункта получить образование высокого качества вполне законно и должно только приветствоваться. Но не могут же престижные вузы вместить всех желающих, да и не все студенты могут платить за проживание в крупных городах. Наиболее простой выход видится только в распространении дистанционного обучения с применением технологий ВКС.

Во-вторых, это — домашняя телемедицина. Известно, что в развитых странах продолжается старение наций. С каждым годом относительное количество престарелых граждан с множеством хронических заболеваний растет, а относительное количество медицинских работников остается тем же или уменьшается. При сохранении такой тенденции в скором времени поликлиническое звено здравоохранения будет просто парализовано. И снова — выход во внедрении технологий ВКС. Как показывает зарубежный опыт, регулярное общение по видеосвязи медицинского работника среднего звена с пожилыми пациентами (телефатронаж) существенно увеличивает их приверженность к выполнению назначений и эффективность лечения. При этом медработник тратит на каждого пациента в среднем не более пяти минут в день, а нагрузка на поликлинику сокращается в разы (по разным данным от 2 до 5 раз).

Итак, вроде бы у рынка видеоконференцсвязи блестящие перспективы. Но почему же потенциальные пользователи ВКС не становятся реальными?

Структура технических средств ВКС

Минимальное оборудование, необходимое для использования видеоконференцсвязи, это:

- *терминал*, работающий хотя бы по одной из рекомендаций МСЭ — Н.320 или Н.323;
- *модем*, использующий те же рекомендации;
- подключенная «последняя миля» сети Интернет достаточной пропускной способности или канал (каналы) цифровой телефонии (сети ISDN).

Для того чтобы в видеоконференции могли участвовать три человека и более уже необходимо использовать *сервер многоточечной ВКС* (MCU — Multipoint Control Unit), работающий в рамках тех же рекомендаций Н.320 или Н.323.

Многие производители технических средств ВКС заявляют о поддержке их изделиями перспективного протокола SIP, идущего на смену Н.323, однако не удалось найти описания сетей ВКС, реально работающих в России по этому протоколу.

Терминалы ВКС, представленные на современном рынке, это, как правило, моноблочный или модульный комплект из видеокамеры, дисплея и кодирующее-декодирующее-коммутирующего устройства (кодека), выполняющие много полезных функций:

- выбор удаленного абонента (адреса в IP-сети или номера в сети ISDN);
- вызов удаленного абонента;
- ответ на вызов удаленного абонента;
- передача и прием изображения и звука;
- настройка параметров связи (IP-адрес, маски подсети и шлюза, полоса, разрешение и др.);
- работа с адресной книгой (внесение, редактирование и удаление адресов и идентификаторов абонентов);
- выбор источника (источников) локального видео (если терминал позволяет использовать больше одной камеры);
- включения/отключения «картинки в картинке» (PiP);
- выбор положения «картинки в картинке» (PiP) на экране (в одном из углов);
- включение/отключение локального микрофона;
- регулирование громкости входящего звука (динамиков);
- регулирование громкости исходящего звука (чувствительности микрофона);
- визуальная и звуковая индикация событий и ключевых состояний (канала связи, режима записи, отключения микрофона и т. п.);
- переход в режим двух видеопотоков и обратно (если такая функция предусмотрена);
- переход в режим защищенного сеанса связи и обратно;
- переход в «спящий» режим при неактивности более установленного времени (например, 5 минут), активация при поступлении входящего вызова;

- контроль параметров установленного соединения (полоса, скорость, частота кадров, протоколы, потери, джиттер и т. п.) и их отображение по запросу оператора;
- возможность включения/отключения автоматического ответа на входящие вызовы;
- индикация входящих, исходящих, пропущенных (успешных, неуспешных) вызовов.

Для реализации такого многообразия в кодеках встроены специализированные процессоры и «защиты» фирменные программные средства.

Большинство терминалов ВКС оснащены специальными PTZ-камерами (PTZ – pan-tilt-zoom), позволяющими оператору (и/или удаленному абоненту) выбирать и увеличивать объект для отображения из окружающего пространства. В ряде терминалов реализована функция автоматической наводки камеры на говорящего по голосу.

Многие терминалы имеют возможность передавать и/или принимать два видеопотока, причем второй из них – в форматах персональных компьютеров (в соответствии с рекомендациями МСЭ Н.239), однако эта функция, как правило, предоставляется опционально за дополнительную плату значительной величины.

Некоторые терминалы имеют возможность работать из локальной сети сквозь NAT и FireWall в соответствии с рекомендациями МСЭ Н.460.18, Н.460.19.

К названным базовым функциям производители добавляют некоторое количество сервисных возможностей, повышающих привлекательность дорогостоящих изделий.

Модемы, используемые в ВКС, это устройства, осуществляющие функцию оконечного оборудования линии связи, т. е. преобразователя сигналов между терминалом ВКС и канальным оборудованием провайдера.

Наиболее часто в России используются модемы двух типов:

- **ISDN** – модемы для цифровых коммутируемых телефонных линий;
- **DSL** – используются для организации выделенных (некоммутируемых) линий, используя обычную телефонную сеть.

Существенно реже используются беспроводные модемы:

- **Спутниковые** – модемы для сетей INMARSAT и VSAT;
- **3G** – модемы сотовых сетей (например, SkyLink);
- **4G** – модемы беспроводных сетей WiMax и LTE (например, YOTA).

Еще реже используются модемы телевизионных (коаксиальных) кабельных сетей в стандарте **DOCSIS**.

Передача аудио/видеопотоков между абонентами требует использования *каналов* гораздо большей пропускной способности, чем обычные телефонные.

На начальных этапах внедрения ВКС такими каналами служили линии цифровой телефонии ISDN, объединяемые в параллельный канал требуемой пропускной способности в соответствии с рекомендациями МСЭ Н.320. Так, для обмена потоками с разрешением CIF при частоте до 25 кадров в секунду обычно использовались 3 канала BRI (BRI – Basic Rate Interface – обеспечивает пользователю предоставление двух цифровых B-каналов по 64 кбита/с и однополосный канал

сигнализации D со скоростью передачи данных 16 кбитов/с). При этом пользователь фактически монопольно занимал 6 телефонных каналов на время сеанса связи и оплачивал их по действующим тарифам телефонных компаний.

По мере повышения доступности широкополосного доступа к публичному Интернету все чаще применяется видеоконференцсвязь по IP-каналам в соответствии с рекомендациями МСЭ H.323. Пока оба варианта сосуществуют, но, учитывая тенденции развития IP-сетей и неразвитость сетей ISDN в России, можно предположить переход в ближайшем будущем к абсолютному доминированию IP-варианта. На сегодня такое доминирование H.323 в России сдерживается только низкой стабильностью отечественных IP-сетей и нежеланием многих провайдеров улучшать инфраструктуру и поддерживать гарантированное качество (QoS – Quality of Service) обслуживания. Термином QoS в области компьютерных сетей определяют вероятность того, что сеть связи соответствует заданному соглашению о трафике [9.162].

Некоторые производители оборудования для ВКС в последнее время заявляют, что дни «обычной» видеоконференцсвязи сочтены, и в ближайшее время они планируют поставлять только оборудование для HD-видеосвязи [9.163].

Если при сеансах обычной ВКС (CIF, 25 кадров/с) зачастую не хватает пропускной способности каналов связи (с учетом реальной загрузки множества маршрутизаторов по пути прохождения сигнала), то необходимо представить, как должен транспортироваться видеопоток с разрешением, соответствующим телевидению высокой четкости.

Сервер, как правило, представляет собой сложный и дорогой аппаратно-программный комплекс, осуществляющий следующие функции:

- вызов более одного абонента;
- прием вызова более чем от одного абонента;
- декодирование входящих аудио/видеопотоков от каждого из удаленных абонентов;
- коммутация входящих видеопотоков (превращение их в единый поток с разделением экрана в соответствии с заданной раскладкой; в простейшем случае площадь экрана делится пропорционально количеству входящих потоков);
- коммутация входящих аудиопотоков;
- кодирование исходящего скоммутированных видео- и аудиопотоков;
- отправка каждому из удаленных абонентов скоммутированных видео- и аудиопотоков.

Кроме этого серверы некоторых производителей могутварьировать распределение изображений на одном или двух экранах, подключать к видеоконференции аудиоабонентов и т. п.

Существует альтернативная идеология построения серверов ВКС, когда сервер выступает в роли интеллектуального маршрутизатора. При этом резко снижаются требования к производительности процессора сервера, т. к. ко всем участникам конференции просто перенаправляются два потока от тех абонентов, которые ведут диалог в настоящий момент. Естественно, что в этом случае в автоматическом либо ручном варианте определяется очередность участия в дискуссии,

а требования к пропускной способности входящего к серверу канала возрастают пропорционально числу участников конференции.

Аппаратные и программные решения

Все вышесказанное относительно терминалов ВКС касалось технических решений на базе специализированных процессоров (аппаратных кодеков). В то же время на современном телекоммуникационном рынке достаточно предложений по «программным» терминалам ВКС. Кавычки применены, потому что реально, конечно, эти терминалы работают на базе персональных ЭВМ, т. е. используют аппаратные средства компьютеров и их операционную среду. Среди таких решений можно назвать продукцию фирм Polycom [9.164], Tandberg [9.165], VCON [9.166], Microsoft [9.167].

Если исходно программные решения в области ВКС существенно уступали аппаратным (например, программа NetMeeting от Microsoft обладала очень скромными возможностями по сравнению с аппаратными терминалами Polycom FX), то сегодняшние решения во все возрастающей степени стирают грань различий в их возможностях с точки зрения пользователя. Так HD4000 от VCON уже позволяет передавать видео высокой четкости и использовать протокол H.239 для передачи двух видеопотоков, что совсем недавно было прерогативой дорогостоящих групповых аппаратных систем ВКС высокого уровня. До настоящего времени, пожалуй, единственным серьезным преимуществом аппаратных решений является проработка акустической части систем:

- высокоэффективное эхоподавление,
- высокоэффективное шумоподавление,
- автоматическое наведение камеры на говорящего.

В остальном программные системы, с точки зрения пользователя, становятся все привлекательнее:

- они дешевле,
- они гибче (позволяют большую комбинаторику функций и аппаратных узлов),
- они динамичнее (обновления программных решений происходят чаще и не требуют замены аппаратных составляющих),
- они мобильнее (производительный ноутбук, веб-камеру высокого разрешения и аудиогарнитуру всегда можно взять с собой).

Альтернативные сервисы

Потребитель хочет пользоваться определенным набором функций ВКС и совсем не имеет желания много платить за это. Идя навстречу его пожеланиям, был разработан и внедрен в сентябре 2003 г. сетевой комплект сервисов Skype [9.168]. Комплект включает возможности:

- шифрованной голосовой связи между компьютерами, подключенными к сети Интернет,
- шифрованной аудиоконференции,
- пересылки файлов,

- обмена текстовыми сообщениями,
- шифрованной видеоконференцсвязи «точка–точка».

Программа терминала предоставляется бесплатно, а часть сервисов, являющихся платными, стоят дешево. Однако имеются некоторые проблемы [9.168].

Во-первых, Skype не поддерживает протокола H.323, а следовательно, не совместим с серверами и сетями стандартных систем ВКС.

Во-вторых, «в отличие от многих других программ IP-телефонии, для передачи данных Skype использует P2P-архитектуру. Каталог пользователей Skype распределен по компьютерам пользователей сети Skype, что позволяет сети легко масштабироваться до очень больших размеров (в данный момент более 100 миллионов пользователей, 15–20 миллионов онлайн) без дорогой инфраструктуры централизованных серверов. Кроме того, Skype может маршрутизировать звонки через компьютеры других пользователей. Это позволяет соединяться друг с другом пользователям, находящимися за NAT или брандмауэром, однако создает дополнительную нагрузку на компьютеры и каналы пользователей, подключенных к Интернету напрямую».

В-третьих, несмотря на заверения разработчиков о конфиденциальности связи, существуют данные о возможности перехвата сообщений, а использование протокола шифрования AES-256, который не сертифицирован на территории Российской Федерации, ставит российских пользователей вне закона.

В-четвертых, Skype является коммуникационной средой для переноса вирусов, т. к. имеет встроенные эффективные алгоритмы прохождения сквозь NAT и брандмауэры.

В-пятых, Skype используется для кражи конфиденциальной информации с компьютеров пользователей, включая пароли для различных сервисов (например, входа в систему «электронный банк»).

На российском рынке активно продвигается система «Видеопорт» [9.169]. Однако и ей присущи неудобства других проприетарных систем: невозможность связи со стандартными терминалами ВКС, привязка к собственному серверу для расширения функциональности, бесплатность только минимального набора функций. Увеличить же функциональность можно, только начав платить абонентскую плату (вместо разового платежа за лицензию, как это сделано у Sony или Tandberg).

Есть еще ряд бесплатных программ ВКС, не поддерживающих стандарты МСЭ. Некоторые из них упрощены до предела (с точки зрения пользователя), например программа Logitech Vid, поставляемая вместе с веб-камерами высокого разрешения [9.169]. Однако их функциональность, как правило, весьма ограничена, а отсутствие возможности установления связи со стандартным оборудованием ВКС делает их неудобными во многих случаях применения.

Цена решений

Когда речь заходит о широте охвата рынка, то многое определяется двумя фактами. На первом месте, конечно, цена изделия. Выложить от шести до двадцати тысяч долларов за возможность видеть, а не только слышать собеседника, готов далеко не каждый потребитель услуг связи.

Программные терминалы гораздо доступнее по ценам. Так, еще два года назад на российском рынке программный терминал «vPoint HD» фирмы Emblaze-VCON предлагался по ценам от 5200 р. до 18 000 р. в зависимости от подключенного набора функций. То есть **ценовая разница с аппаратными решениями в десятки раз в пользу программных**.

Подавляющему большинству потенциальных пользователей нужны дешевые терминалы ВКС при требуемой достаточно ограниченной их функциональности. Часть потенциальных пользователей готова доплачивать за расширение функциональности, но их существенно меньше. И лишь очень малое количество потенциальных пользователей готово много платить за имиджевые аппаратные решения, причем, как правило, не своими, а государственными или корпоративными деньгами.

Покупателей дорогих аппаратных серверов многоточечной связи на рынке ничтожно мало. Большинство ограничивается серверами многоточки, встроенными в терминал. Ограничение в количестве от 4 до 9 одновременно подключаемых удаленных абонентов (что обеспечивают встроенные MCU), как правило, соответствуют реальным запросам потребителя. Однако при широком внедрении дистанционного образования и домашней телемедицины вопрос о серверах станет узким местом проектов с малым бюджетом. И именно тогда на первый план должны выйти программные серверы. Безусловно, программный сервер на большое число подключений, построенный по коммутирующему принципу, требует аппаратной платформы в виде компьютера очень высокой производительности. В то же время его цена все равно будет ниже, чем у аппаратных решений. Но для большинства применений достаточно маршрутизирующего программного сервера многоточечной ВКС! Тем более что пропускная способность «последней мили» большинства пользователей постоянно растет из-за высокой конкуренции между провайдерами услуг широкополосного доступа в Интернет [9.171].

Программные решения VPhone

Разработка алгоритмов и программ для передачи цифрового видеопотока по относительно узкополосным и нестабильным каналам связи была начата коллективом сотрудников НИИРадио еще в 1996 г. И уже через год разработка была представлена в Женеве вниманию международного сообщества. Программный комплекс под названием «VPhone» оказался интересен для проекта «уличного телевидения» во время избирательной кампании на Украине в 1999 г. Программные терминалы VPhone, установленные на десятках микроавтобусов, использовались для установления видеомостов между избирательным штабом и «улицей» во многих населенных пунктах. Эффект «живого диалога с народом» внес свой вклад в повышение рейтинга президента и его победу на выборах [9.172].

Несколько позже программный комплекс VPhone привлек внимание вузов, работающих в технологиях дистанционного образования: Современной гуманистической академии [9.173] и Российского нового университета [9.175]. Наличие программного сервера многоточечной видеоконференцсвязи, одновременная передача видео/аудиоинформации и текстовых сообщений, возможность передачи двух и более видеопотоков, удобный инструментарий оператора видеосервера — все это позволяет гибко конфигурировать многоточечную видеоконференцию в учебном



Рис. 23.2. Экранная форма программы VPhone: многоточечная ВКС (4 терминала)

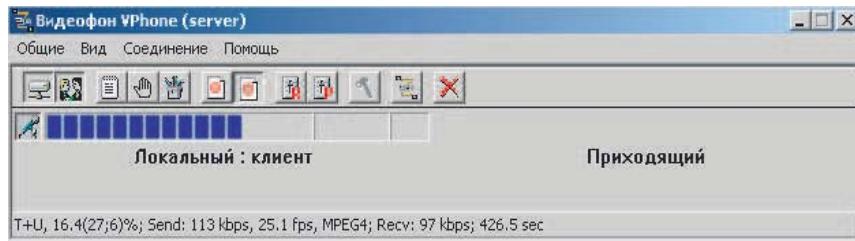
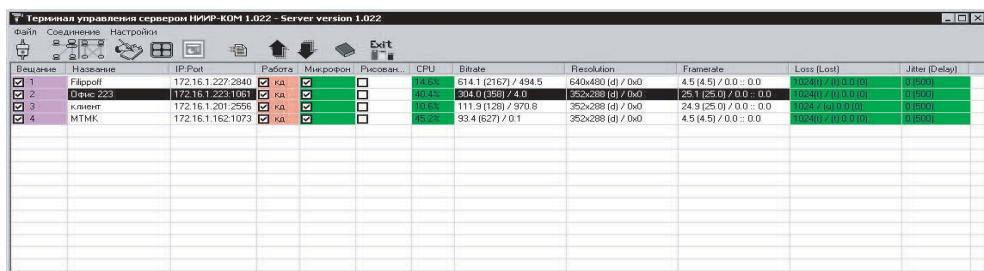


Рис. 23.3. Экранные формы программы VPhone: терминалы управления сервером и клиентом

процессе, охватывая в одном сеансе десятки филиалов и транслируя их слушателям лектора, презентацию и демонстрацию опытов одновременно (рис. 23.2 и 23.3).

Дальнейшее распространение комплекса VPhone сдержала его проприетарность. С целью улучшения работы по нестабильным узкополосным IP-сетям VPhone

был разработан на собственных протоколах, не совпадающих с H.323, а в стране уже все шире применялись стандартные системы, с которыми он был несовместим. В 2007 г. коллектив разработчиков продолжил работу по созданию программных средств видеоконференцсвязи, совместимых с протоколами H.323.

Определяемые стандартами алгоритмы сжатия динамических изображений весьма гибки, что при различных реализациях может обеспечить существенное различие в качестве декодируемой информации при одной и той же степени ее сжатия. Стандарт видеокодирования H.264/AVC постоянно совершенствуется, способен обеспечивать высокое качество воспроизведения динамических изображений при существенно более низких битовых скоростях передачи информации, чем это позволяли стандарты предыдущих поколений.

Программные решения «ДАВ-телефон»

Одним из результатов разработки стал программный терминал ВКС, получивший название «ДАВ-телефон» (данные-аудио-видео-телефония), который представляет собой эффективное средство видеосвязи, обладающее дружественным программным графическим интерфейсом (ПГИ) с развитыми возможностями управления параметрами текущего сеанса и адресной книгой (рис. 23.4–23.8).

Программа «ДАВ-телефон» зарегистрирована в государственном Реестре программ для ЭВМ за № 2010611864.

На рис. 23.4 представлена исходная форма ПГИ, предназначенная для вызова удаленных абонентов по записи в адресной книге, IP-адресу или списку последних соединений. Кнопки внизу предназначены для вызова основных функций меню: настройки, статистики параметров текущего соединения, управления выводом входящих видеопотоков, управления запуском приложений, назначения источников исходящих видеопотоков, управления соединениями. Кнопки и движки на правой вертикальной панели предназначены для: управления громкостью входящего и исходящего звука и его блокирования, вызова виртуальной клавиатуры, переключения из полноэкранного режима в оконный и наоборот, включения/отключения исходящего видеопотока, включения/отключения режима двух видеопотоков, включения/отключения режима «показ всех окон».

На рис. 23.5–23.8 показаны окна настройки параметров конкретных функций и окно редактирования карточки абонента.

Программный терминал видеоконференцсвязи «ДАВ-телефон», работающий в IP-сетях в соответствии с рекомендациями H.323 МСЭ, в чем-то несколько уступает по функциональности аппаратным комплексам ведущих производителей, в чем-то превосходит их, однако имеющихся возможностей вполне достаточно для подавляющего большинства пользователей.

Поскольку различия между программами «ДАВ-телефон» и «VPhone» касаются прежде всего стека протоколов, организующих взаимодействие терминалов между собой и терминалов с сервером, очевидно: ПГИ может быть перенесен в программу «VPhone», что существенно упростит ее применение неподготовленными пользователями. И оба программных продукта, имеющих сходный интерфейс, могут применяться в соответствии со своими особенностями: «VPhone» — в тех случаях, когда требуется устойчивая работа на узкополосных нестабильных сетях и/или проведение многоточечных видеоконференций без дорогостоящих



Рис. 23.4. ПГИ — меню и интерфейс вызовов абонентов

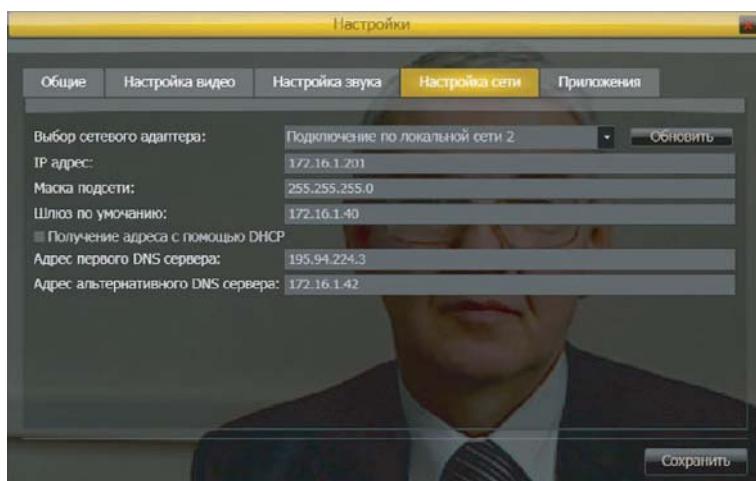


Рис. 23.5. Окно настроек параметров сети

аппаратных видеосерверов, «ДАВ-телекон» — когда нужен бюджетный терминал, совместимый с H.323.

При внимательной оценке реальных потребностей в области видеоконференцсвязи, возникает вопрос: а насколько оправданно стремление к совместимости со стандартом H.323, разработанным исходно для более широкого класса телекоммуникационных услуг и обладающим избыточной сложностью для реализации большинства запросов пользователей ВКС? Не слишком ли высока плата за

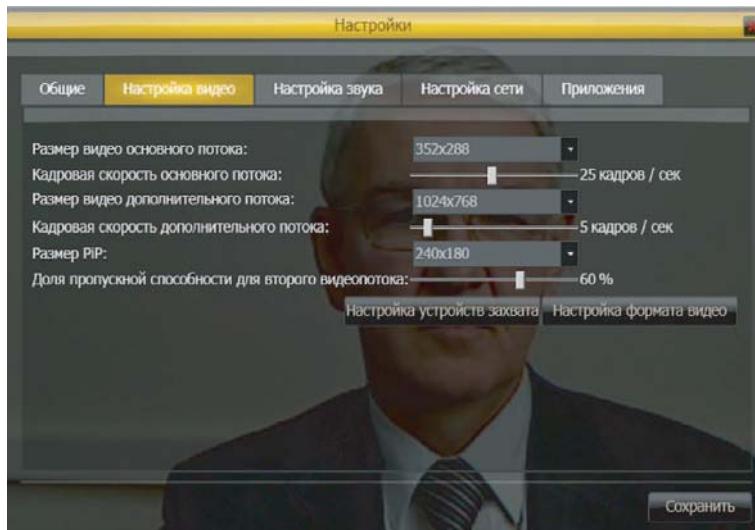


Рис. 23.6. Окно настроек параметров видеопотоков

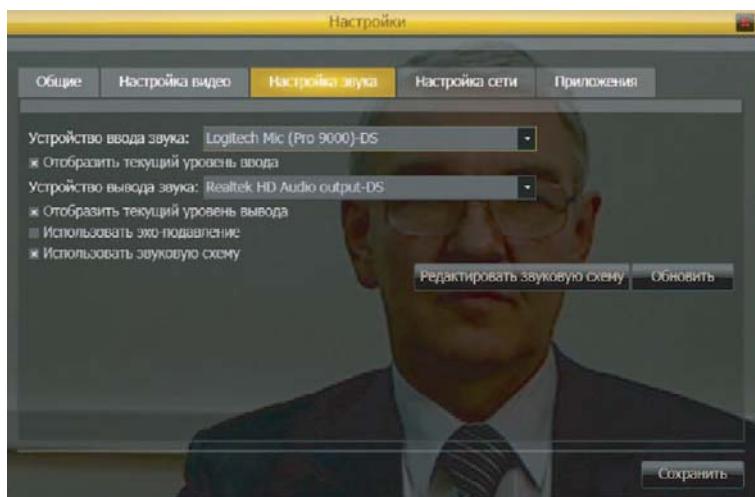


Рис. 23.7. Окно настроек параметров аудиоподсистемы

невостребованную универсальность? Может быть, следует в качестве российского стандарта принять существующую де-факто разработку, позволяющую с приемлемым качеством работать по узкополосным и нестабильным сетям связи, а в широкополосных сетях обеспечивающую качества сравнимое с дорогостоящей стандартизированной аппаратурой?

Что касается тех немногих ситуаций, когда действительно требуется полная совместимость с аппаратурой ведущих мировых производителей, то проблема относительно просто решается путем применения программного шлюза, обеспечивающего перекодирование стека протоколов [9.175].



Рис. 23.8. Режим редактирования карточки абонента.

Сегодня технологиями ВКС пользуется относительно небольшой процент населения, а завтра они станут массовыми. И тогда вопрос о соотношении цена/качество выйдет на первый план. Российские разработчики готовы предложить потребителю необходимые программные средства по разумным ценам. Вопрос только за нормативным обеспечением и решением руководства телекоммуникационной отрасли.

Литература к части IX

- 9.1. ETSI EN 302 304, DVB-H Transmission System for Handheld Terminals. 2004-06.
- 9.2. ETSI TR 102 377, Implementation Guidelines for DVB-H Services (построение сервисов, основанных на DVB-H). 2005-05.
- 9.3. ETSI TR 102 377, Validation Task Force Report (результаты экспериментов с целью подтверждения спецификаций DVB-H). 2008-12.
- 9.4. *Бителева А.* Система вещания DVB-H // Теле-Спутник. 2006. № 6.
<http://www.telesputnik.ru/archive/128/article/86.html>
- 9.5. Стандарт DVB-H. Система мобильного ТВ вещания. Группа компаний «Контур-М».
<http://www.konturm.ru/tech.phpid=dvbw>
- 9.6. ETSI TS 102 470, IP Datacast over DVB-H: PSI/SI, 2009-12.
- 9.7. ETSI A096, IP Datacast over DVB-H: Set of Specifications for Phase 1, 2005-11.
- 9.8. ETSI TR 102, 473 v1.1.1 IP Datacast over DVB-H: Use Cases and Services, 2006-04.
- 9.9. ETSI TR 102, 469 v1.1.1 IP Datacast over DVB-H: Architecture, 2006-05.
- 9.10. ETSI TS 102, 471 v1.2.1 IP Datacast over DVB-H: Electronic Service Guide (ESG), 2006-11.
- 9.11. ETSI A112, Electronic Service Guide (ESG) Implementation Guidelines, 2009-04.
- 9.12. ETSI TS 102 472, IP Datacast over DVB-H: Content Delivery Protocols (CDP), 2005-11.
- 9.13. ETSI A113, Content Delivery Protocols (CDP) Implementation Guidelines, 2009-04.
- 9.14. EBU / CENELEC / ETSI Joint Technical Committee (JTC). ETSI TS 102 589, Forward Link Only Air Interface; Specification for Terrestrial Mobile; Multimedia Multicast. 2009-02.
- 9.15. MediaFLO (QUALCOMM FLO TV\texttrademark) News.
http://www.microsofttranslator.com/bv.aspx?from=en&to=ru&a=http%3A%2F%2Fwww.digitaltvnews.net%2Fcontent%2F%3Fcategory_name%3Dmediaflo
- 9.16. ETSI TS 102 589: «Forward Link Only Air Interface; Specification for Terrestrial Mobile; Multimedia Multicast». MediaFLO.
<http://en.wikipedia.org/wiki/MediaFLO>
- 9.17. *Бителева А.* Системы вещания на портативные терминалы // Теле-Спутник. 2006. 7 (129).
- 9.18. *Жернакова О.* Мобильное вещание в Европе. ТелеМультиМедиа. 30 октября 2007 г.
<http://www.telemultimedia.ru/art.php?id=246>



- 9.19. ETSI EN 300 401 Radio Broadcasting Systems;Digital Audio Broadcasting (DAB) to mobile, portable and fixed receivers. V1.4.1. 2006-01.
- 9.20. ETSI TS 102427. Digital Audio Broadcasting (DAB); Data Broadcasting — MPEG-2 TS streaming. V1.1.1. 2005-07.
- 9.21. ETSI TS 102428. Digital Audio Broadcasting (DAB); DMB video service; User application specification. V1.2.1. 2009-04.
- 9.22. *Варгаузин В.* Цифровое вещание на мобильные терминалы (по материалам конференции CSTB 2006)
<http://www.telemultimedia.ru/art.php?id=111>
- 9.23. *Варгаузин В.* Цифровое вещание на мобильные терминалы (по материалам конференции CSTB 2007)
<http://www.telemultimedia.ru/art.php?id=222>
- 9.24. [1] DAB Digital Radio tutorial. [2] DAB frequencies [3]. DMB - Digital Multimedia Broadcasting.
<http://www.radio-electronics.com/info/broadcast/dab/dmb-digital-multimedia-broadcasting.php>
- 9.25. ГОСТ Р 54309-2011. «Аудиовизуальная информационная система реального времени (РАВИС). Процессы формирования кадровой структуры, канального кодирования и модуляции для системы цифрового наземного узкополосного радиовещания в ОВЧ диапазоне. Технические условия».
- 9.26. ITU-R Doc. 6E/336, 6M/133, 03.03.2006. «Increase of the Band 8 (VHF) Utilization Efficiency».
- 9.27. ITU-R Doc. 6M/150, 14.08.2006. «Technical Report. Digital Mobile Narrow-band Multimedia Broadcasting System AVIS (for draft new Recommendation ITU-R BT. [Doc. 6/270])».
- 9.28. ITU-R Doc. 6A/86, 6B/47, 08.10.2008. «Proposal of preliminary draft new Recommendation ITU-R BS.[XXXX] on service requirements for digital terrestrial sound and multimedia broadcasting in VHF band — Service requirements for digital terrestrial sound and multimedia broadcasting for mobile reception in VHF bands I and II».
- 9.29. ITU-R Doc. 6A/193, 21.04.2009. «Proposal on draft revision of recommendation ITU-R BS.774-2 - Service requirements for digital terrestrial sound and multimedia broadcasting to vehicular, portable and fixed receivers in the VHF/UHF bands».
- 9.30. ITU-R Doc. 6A/226, 22.10.2009. «Предварительный проект новой рекомендации МСЭ-R BS.[MULT-FM] - Расширенные мультимедийные службы для наземного цифрового радиовещания в полосах I и II ОВЧ диапазона». «Proposal for a preliminary draft new recommendation ITU-R BS.[MULT-FM] — Enhanced multimedia services for digital terrestrial broadcasting in VHF Bands I and II».
- 9.31. ITU-R Doc. 6A/345, 14.04.2010. «Proposal toward a draft new recommendation ITU-R BS.[MULT{\-\}FM] - Enhanced multimedia services for digital terrestrial broadcasting in VHF Bands I and II ».



- 9.32. ITU-R Doc. 6B/230, 14.10.2010. «Proposal toward a draft revision of Report ITU-R BT.2049-3 — Broadcasting of multimedia and data applications for mobile reception».
- 9.33. ITU-R Doc. 6A/488, 28.04.2011. «Planning parameters for digital broadcasting system RAVIS».
- 9.34. ECC Doc. FM45(09)135, 27.11.2009. «Digital terrestrial sound and multimedia broadcasting system RAVIS for Bands I and II».
- 9.35. ECC FM(10)014 Annex 1, Comment from Russia, 06.01.2010. «Result of public consultation on draft ECC Report 141 on Band II, Annex 1, Comment from Russia».
- 9.36. ECC Doc. FM45(09)162, 16.03.2010. «Information for inclusion into draft revision of ECC Report 117».
- 9.37. ECC Doc. FM45(09)171, 26.03.2010. «Technical Elements and Parameters for Digital Terrestrial Broadcasting System RAVIS in Band II».
- 9.38. ECC Doc. FM45(10)194, 08.06.2010. «Proposals for Initial Draft Report on Possibilities for Future Delivery of Audio Broadcasting Services».
- 9.39. ECC Doc. FM45(10)195, 08.06.2010. «Technical Elements and Parameters for Digital Terrestrial Broadcasting System RAVIS in Band II».
- 9.40. ECC Doc. FM45(10)229, 29.12.2010. «Technical Elements and Parameters for Digital Terrestrial Broadcasting System RAVIS in Band II».
- 9.41. ECC Doc. FM45(11)252, 15.04.2011. «Working Document on Draft Supplement to ECC Report 141».
- 9.42. ECC Doc. FM45(11)259, 09.06.2011. «Working Document on Draft Supplement to ECC Report 141».
- 9.43. *Дворкович А.В., Дворкович В.П., Зубарев Ю.Б. и др.* Способ трансляции информационного телевидения// Патент России № 2219676 от 20.12.2003.
- 9.44. *Дворкович А.В., Дворкович В.П., Иртлога В.А.*// Новая аудиовизуальная информационная система// Broadcasting. Телевидение и радиовещание. 2005. № 5. С. 52–56.
- 9.45. ITU-R. BT.2049-4 (10/2010). «Broadcasting of multimedia and data applications for mobile reception».
- 9.46. ITU-R. BS.[PLAN_PAR] (Doc. 6/343, 11.05.2011, «Planning parameters for terrestrial digital sound broadcasting systems in VHF bands»).
- 9.47. ITU-R. BT.1892 (05/2011). «Requirements for enhanced multimedia services for digital terrestrial broadcasting in VHF Bands I and II».
- 9.48. ECC 117 (09/2010). «Managing the transition to digital sound broadcasting in the frequency bands below 80 MHz».
- 9.49. ECC 141 (05/2010). «Future possibilities for the digitalisation of band II (87.5–108 MHz)».
- 9.50. ECC 141 (06/2011). «Technical characteristics and parameters of candidate digital terrestrial broadcasting systems for band II».

- 9.51. ECC (06/2011). «Possibilities for future terrestrial delivery of audio broadcasting services».
- 9.52. Дворкович В.П., Дворкович А.В., Иртюга В.А. Цифровые методы повышения эффективности использования ОВЧ-диапазона частот// Цифровая обработка сигналов и ее применение: Докл. 11-й Междун. конф. М., 2009. Вып. XI. 1. С. 10–20.
- 9.53. Дворкович В., Дворкович А. Отвечает ли отечественная система РАВИС потребностям локального радиовещания. Ч. 1, 2// Broadcasting. Телевидение и радиовещание. 2011. № 2, 3. С. 34–38 (№ 2), 42–46 (№ 3).
- 9.54. Дворкович В.П., Дворкович А.В. Российская система цифрового мультимедийного радиовещания// Электросвязь. 2011. № 6. С. 18–24.
- 9.55. Дворкович А.В., Дворкович В.П., Иртюга В.А. Способ мобильного узкополосного цифрового мультимедийного радиовещания // Решение о выдаче патента по заявке № 2010131023/08(043944) от 26.07.2010.
- 9.56. Дворкович А.В., Дворкович В.П. Канальное кодирование и декодирование для системы РАВИС // Свидетельство о государственной регистрации программы для ЭВМ № 2011613912 от 19 мая 2011.
- 9.57. Дворкович А.В., Дворкович В.П. Модулятор и демодулятор системы РАВИС// Свидетельство о государственной регистрации программы для ЭВМ № 2011613911 от 19 мая 2011.
- 9.58. Дворкович А.В., Дворкович В.П. Синхронизация сигнала системы РАВИС// Свидетельство о государственной регистрации программы для ЭВМ № 2011613910 от 19 мая 2011.
- 9.59. ISO/IEC 14496-3 – Information Technology – Coding of Audio-Visual Objects. Part 3: Audio. ISO, 2009. 1416 p.
- 9.60. ISO/IEC 23003-1 – Information Technology – MPEG Audio Technologies. Part 1: MPEG Surround. ISO, 2007. 56 p.
- 9.61. ITU-T Recommendation H.264. Advanced Video Coding for Generic Audiovisual Services (ISO/IEC 14496-10. Information Technology – Coding of Audio-Visual Objects. Part 10: Advanced Video Coding). ITU-T, 03/2009. 666 p.
- 9.62. ISO/IEC 13818-1. Information technology – Generic coding of moving pictures and associated audio information. Part 1: Systems. ISO, 2007. 174 p.
- 9.63. ETSI TS 102 606 V.1.1.1 (2007-10) Digital Video Broadcasting (DVB); Generic Stream Encapsulation (GSE) Protocol. ETSI, 2007. 25 p.
- 9.64. ETSI EN 302 018-2. V1.1.1. Electromagnetic Compatibility and Radio spectrum Matters (ERM); Transmitting equipment for the Frequency Modulated (FM) sound broadcasting service; Part 2: Harmonized EN under article 3.2 of the R&TTE Directive. 2002-07.
- 9.65. ETSI ES 201 980: Digital Radio Mondiale (DRM); System Specification V2.2.1 (2005).
- 9.66. www.drm.org/pdfs/press_release_1.pdf



- 9.67. Höher P. «A statistical discrete-time model for the WSSUS multipath channel»// IEEE Trans. Vehicular Technol. V. 41. No. 4. P. 461-468 (1992).
- 9.68. ETSI ES 201 980 V.3.1.1 (2009-08) Digital Radio Mondiale (DRM); System Specification. ETSI, 2009. 221 p.
- 9.69. ITU-R. Doc. 6E/413-E. Digital Radio Mondiale (DRM).Narrowband digital broadcasting with the DRM System in bands I and II. 14 August 2006.
- 9.70. Heuberger Albert. Digital Broadcasting in Band I and II with DRM+ (Digital Radio Mondiale).Fraunhofer Institut f\"{u}r Integrierte Schaltungen IIS Erlangen. albert.heuberger@iis.fraunhofer.de. 2006
- 9.71. Mayer F., Waal A., Tissen A. Evaluation of Soft Delay Diversity for Digital Broadcast System DRM+. Труды РНТОРЭиС им. А.С. Попова. Серия: Цифровая обработка сигналов и ее применение. Вып. XI-1. М., 2009.
- 9.72. Dammann A., Kaiser S. «Transmit/receive antenna diversity techniques for OFDM systems»// Europ. Trans. Telecomm. Vol. 13. P. 531–538 (2002).
- 9.73. http://www.rol.ru/it/news/96/12/03_22.htm (04.03.2010)
- 9.74. <http://www.polycom.com>, <http://www.sony.ru>, <http://www.emblaze-vcon.ru>, <http://www.vtel.com>, <http://www.aethra.ru> (04.03.2010)
- 9.75. <http://www.tandbergRussia.ru> (04.03.2010)
- 9.76. ITU-T. Rec. H221. Frame structure for a 64 to 1920 kbit/s channel in audiovisual teleservices 2009-03-16 (1990-12)
- 9.77. ITU-T. Rec. H222.0. Information technology – Generic coding of moving pictures and associated audio information: Systems. 2006-05-29. Identical standard: ISO/IEC 13818-1:2007.
- 9.78. ITU-T. Rec. H222.1. Multimedia multiplex and synchronization for audiovisual communication in ATM environments.1996-03-20.
- 9.79. ITU-T. Rec. H224. A real time control protocol for simplex applications using the H.221 LSD/HSD/MLP channels. 2005-01-08.
- 9.80. ITU-T. Rec. H225.0. Call signalling protocols and media stream packetization for packet-based multimedia communication systems. 2009-12-14 (1998-02).
- 9.81. ITU-T. Rec. H231. Multipoint control units for audiovisual systems using digital channels up to 1920 kbit/s. 1997-07-11 (1993-03).
- 9.82. ITU-T. Rec. H233. Confidentiality system for audiovisual services. 2002-11-29 (1993-03).
- 9.83. ITU-T. Rec. H234. Encryption key management and authentication system for audiovisual services. 2002-11-29 (1994-11).
- 9.84. ITU-T. Rec. H239. Role management and additional media channels for H.300-series terminals. 2005-09-13 (2003-07).
- 9.85. ITU-T. Rec. H241. Extended video procedures and control signals for H.300-series terminals. 2006-05-29 (2003-07).
- 9.86. ITU-T. Rec. H242. System for establishing communication between audiovisual terminals using digital channels up to 2 Mbit/s. 2009-03-16 (1990-12).

- 9.87. ITU-T. Rec. H243. Procedures for establishing communication between three or more audiovisual terminals using digital channels up to 1920 kbit/s. 2005-10-07 (1993-03).
- 9.88. ITU-T. Rec. H245. Control protocol for multimedia communication. 2011-05-14 (1996-03).
- 9.89. ITU-T. Rec. H246. Interworking of H-series multimedia terminals with H-series multimedia terminals and voice/voiceband terminals on GSTN, ISDN and PLMN. 2006-05-29.
- 9.90. ITU-T. Rec. H248.1. Gateway control protocol: V. 3. 2005-09-13 (V.1.1. 2002-03).
- 9.91. ITU-T. Rec. H261. Video codec for audiovisual services at p x 64 kbit/s. 1993-03-12 (1990-12).
- 9.92. ITU-T. Rec. H263. Video coding for low bit rate communication. 2005-01-13 (1997-03).
- 9.93. ITU-T. Rec. H264. Advanced video coding for generic audiovisual services. Identical standard:ISO/IEC 14496-10. 2010-03-09 (2003-05).
- 9.94. ITU-T. Rec. H281. A far end camera control protocol for videoconferences using H.224. 1994-11-01.
- 9.95. ITU-T. Rec. H310. Broadband audiovisual communication systems and terminals. 1998-09-25 (1996-11).
- 9.96. ITU-T. Rec. H320. Narrow-band visual telephone systems and terminal equipment 2004-03-15 (1990-12).
- 9.97. ITU-T. Rec. H321. Adaptation of H.320 visual telephone terminals to B-ISDN environments. 1998-02-06 (1996-03).
- 9.98. ITU-T. Rec. H322. Visual telephone systems and terminal equipment for local area networks which provide a guaranteed quality of service. 1996-03-20.
- 9.99. ITU-T. Rec. H323. ITU-T. Rec. H323. Packet-based multimedia communications systems. 2009-12-14 (1998-02).
- 9.100. ITU-T. Rec. H324. Terminal for low bit-rate multimedia communication. 2009-04-29 (1996-03).
- 9.101. ITU-T. Rec. H331. Broadcasting type audiovisual multipoint systems and terminal equipment. 1993-03-12.
- 9.102. ITU-T. Rec. H332. H.323 extended for loosely coupled conferences. 1998-09-25.
- 9.103. ITU-T. Rec. H450.1. Generic functional protocol for the support of supplementary services in H.323. 2011-05-14 (1998-02).
- 9.104. ITU-T. Rec. H450.2. Call transfer supplementary service for H.323. 2011-05-14 (1998-02).
- 9.105. ITU-T. Rec. H450.3. Call diversion supplementary service for ITU-T H.323. 2011-05-14 (1998-02).
- 9.106. ITU-T. Rec. H450.4. Call hold supplementary service for H.323. 1999-05-27.



- 9.107. ITU-T. Rec. H450.5. Call park and call pickup supplementary services for H.323. 1999-05-27.
- 9.108. ITU-T. Rec. H450.6. Call waiting supplementary service for H.323. 1999-05-27.
- 9.109. ITU-T. Rec. H450.7. Message waiting indication supplementary service for H.323. 1999-05-27.
- 9.110. ITU-T. Rec. H450.8. Name identification supplementary service for H.323~. 2000-02-17.
- 9.111. ITU-T. Rec. H450.9. Call completion supplementary services for H.323. 2000-11-17.
- 9.112. ITU-T. Rec. H450.10. Call offering supplementary services for H.323. 2001-03-01.
- 9.113. ITU-T. Rec. H450.11. Call intrusion supplementary service for H.323. 2001-03-01.
- 9.114. ITU-T. Rec. H450.12. Common Information Additional Network Feature for H.323. 2001-07-29.
- 9.115. ITU-T. Rec. H460.1. Guidelines for the use of the generic extensible framework. 2002-03-29.
- 9.116. ITU-T. Rec. H460.4. Call priority designation and country/international network of call origination identification for H.323 priority calls. 2007-01-13 (2002-11).
- 9.117. ITU-T. Rec. H460.8. Querying for alternate routes within H.323 systems. 2002-11-29.
- 9.118. ITU-T. Rec. H460.9. Support for online QoS-monitoring reporting within H.323 systems. 2002-11-29.
- 9.119. ITU-T. Rec. H460.10. Call party category within H.323 systems. 2004-03-15.
- 9.120. ITU-T. Rec. H460.11. Delayed call establishment within H.323 systems. 2004-03-15.
- 9.121. ITU-T. Rec. H460.12. Glare control indicator within H.323 systems. 2004-03-15.
- 9.122. ITU-T. Rec. H460.13. Called user release control within H.323 systems. 2004-03-15.
- 9.123. ITU-T. Rec. H460.14. Support for Multi-Level Precedence and Preemption (MLPP) within H.323 systems. 2004-03-15.
- 9.124. ITU-T. Rec. H460.15. Call signalling transport channel suspension and redirection within H.323 systems. 2004-03-15.
- 9.125. ITU-T. Rec. H460.16. Multiple-message release sequence capability within H.323 systems. 2005-01-08.
- 9.126. ITU-T. Rec. H460.17. Using H.225.0 call signalling connection as transport for H.323 RAS messages. 2005-09-13.
- 9.127. ITU-T. Rec. H460.18. Traversal of H.323 signalling across network address translators and firewalls. 2005-09-13.
- 9.128. ITU-T. Rec. H460.19. Traversal of H.323 media across network address translators and firewalls. 2005-09-13.

- 9.129. ITU-T. Rec. H460.20. Location number within H.323 systems. 2005-09-13.
- 9.130. ITU-T. Rec. G711. Pulse code modulation (PCM) of voice frequencies. 1988-11-25.
- 9.131. ITU-T. Rec. G723-1. Dual rate speech coder for multimedia communications transmitting at 5.3 and 6.3 kbit/s. 2006-05-29 (1996-03).
- 9.132. ITU-T. Rec. G728. Coding of speech at 16 kbit/s using low-delay code excited linear prediction. 1992-09-01.
- 9.133. ITU-T. Rec. G729. Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear prediction (CS-ACELP). 2007-01-13 (1996-03).
- 9.134. ITU-T. Rec. T120. Data protocols for multimedia conferencing. 2007-01-13 (1996-07).
- 9.135. ITU-T. Rec. T121. Generic application template. 1996-07-03.
- 9.136. ITU-T. Rec. T122. Multipoint communication service – Service definition. 1998-02-06 (1993-03).
- 9.137. ITU-T. Rec. T123. Network-specific data protocol stacks for multimedia conferencing. 2007-01-13 (1999-05).
- 9.138. ITU-T. Rec. T124. Generic Conference Control. 2007-01-13 (1995-08).
- 9.139. ITU-T. Rec. T125. Multipoint communication service protocol specification. 1998-02-06 (1994-04).
- 9.140. ITU-T. Rec. T126. Multipoint still image and annotation protocol. 2007-08-29 (1995-08).
- 9.141. ITU-T. Rec. T127. Multipoint binary file transfer protocol. 2007-08-29 (1995-08).
- 9.142. ITU-T. Rec. T128. Multipoint application sharing. 2008-06-13 (1998-02).
- 9.143. Дворкович В.П., Федоров В.Ф. Перспективы развития видеоконференцсвязи в России// Цифровая обработка сигналов. 2010. № 4. С. 57–65.
- 9.144. <http://www.osp.ru/text/print/302/179460.html> (04.03.2010)
- 9.145. http://oblsud.sah.sudrf.ru/modules.php?name=press_dep&op=1&did=283 (04.03.2010)
- 9.146. http://www.muzbazar.ru/Biblioteka/Videokonferencii_i_ih_ispolzovanie.html (04.03.2010)
- 9.147. <http://www.videosvyaz.ru> (04.03.2010)
- 9.148. http://www.tandbergrussia.ru/ind_focus/energy/index.jsp (04.03.2010)
- 9.149. <http://www.pentagonus.ru/publ/3-1-0-669> (05.03.2010)
- 9.150. <http://www.connect.ru/article.asp?id=9552> (05.03.2010).
- 9.151. <http://www.pcweek.ru/themes/detail.php?ID=82777> (05.03.2010)
- 9.152. <http://sanweras.pp.ua/videokonferentssvyaz-2/vpervie-v-rossii-bil-proveden-dopros-svidetelya-s-ispolzovaniem-videokonferentssvyazi-639.html> (05.03.2010)
- 9.153. <http://www.pfo.ru/-id=19837> (05.03.2010)



- 9.154. <http://www.kremlin.ru/news/6148> (05.03.2010)
- 9.155. <http://www.rian.ru/politics/20050610/40504925.html> (06.03.2010)
- 9.156. <http://tele.med.ru/u1.htm> (06.03.2010)
- 9.157. <http://www.telemed.ru> (06.03.2010)
- 9.158. http://www.pmc.ru/Products_and_Services/Telemedicine/GRCCTelemed (09.03.2010)
- 9.159. http://www.rosnou.ru/important/filial_konf (09.03.2010)
- 9.160. http://www.vfmgiu.ru/chto_novogo_v_visshem_obrazovanii_450/videokonferencsvyaz_prob-lemy_reshenie_466/index.html (09.03.2010)
- 9.161. <http://www.connect.ru/newsdetail.asp?id=6543> (09.03.2010)
- 9.162. *Фёдоров В.Ф.* Российские услуги связи с точки зрения телемедицины // В сб. научно-практической конф. «Мобильные телемедицинские комплексы. Домашняя телемедицина». Ростов н/Д, 2005. С. 75–82.
- 9.163. <http://www.cnews.ru/reviews/index.shtml-2008/05/19/300715> (09.03.2010)
- 9.164. <http://www.bs-media.ru/catalogcat/video/1147247225~1148886071> (09.03.2010)
- 9.165. <http://www.tandbergussia.ru/products/pc-video-conferencing.jsp> (09.03.2010)
- 9.166. http://www.emblaze-vcon.ru/solut_nastol.htm (09.03.2010)
- 9.167. <http://office.microsoft.com/en-us/livemeeting/HA102403231033.aspx> (10.03.2010)
- 9.168. <http://ru.wikipedia.org/wiki/Skype> (10.03.2010)
- 9.169. <http://www.videoport.ru> (10.03.2010)
- 9.170. http://www.logitech.com/index.cfm/webcam_communications/&cl=ru,ru (10.03.2010)
- 9.171. <http://telecom.cnews.ru/reviews/index.shtml-2010/03/19/383377> (10.03.2010)
- 9.172. <http://www.telesputnik.ru/archive/64/article/40.html> (11.03.2010)
- 9.173. http://www.muh.ru/stud_practice.htm (11.03.2010)
- 9.174. <http://steelsoft.ru/rosnou/> (11.03.2010)
- 9.175. *Дворкович В.П., Фёдоров В.Ф.* Перспективы развития видеоконференцсвязи в России// Радиочастотный спектр. 2011. № 3. С. 28–37.

Производство книг на заказ
Издательство «Техносфера»
тел. (495) 234-01-10
e-mail: knigi@technosphera.ru

Реклама в книгах:

- модульная
- статьи

Подробная информация о книгах на сайте
www.technosphera.ru

Дворкович Виктор Павлович
Дворкович Александр Викторович

**Цифровые видеоинформационные системы
(теория и практика)**

Компьютерная верстка – С.А. Кулешов

Корректор – А.А. Конькова

Дизайн – А.А. Давыдова

Выпускающий редактор – О.Н. Кулешова

Ответственный за выпуск – С.А. Орлов

Формат 70x100/16. Печать офсетная.

Гарнитура Computer modern LaTeX

Печ.л. 63. Тираж 1000. Зак. №

Бумага офсет. №1, плотность 65 г/м².

Издательство «Техносфера»

Москва, ул. Краснопролетарская, д.16, стр.2

Отпечатано в ООО «Чебоксарская типография № 1»
428000, Чувашская Республика, г. Чебоксары, пр. И. Яковлева, 15.

Тел. +7(8352) 28-77-98, 57-01-87

www.volga-print.ru