

Федеральное агентство по образованию  
Государственное образовательное учреждение  
высшего профессионального образования  
Владимирский государственный университет

В. И. БЫКОВ

УСТРОЙСТВА ПРЕОБРАЗОВАНИЯ  
И ОБРАБОТКИ ИНФОРМАЦИИ В СИСТЕМАХ  
ПОДВИЖНОЙ СВЯЗИ

Учебное пособие

Владимир 2009

УДК 621.396.93  
ББК 32.84  
Б95

Рецензенты:

Доктор технических наук, профессор  
зав. кафедрой радиотехники и радиосистем  
Владимирского государственного университета  
*О. Р. Никитин*

Кандидат технических наук, профессор  
зав. кафедрой информатики, информационных технологий  
и компьютерной техники  
Владимирского государственного гуманитарного университета  
*Ю. А. Медведев*

Печатается по решению редакционного совета  
Владимирского государственного университета

**Быков, В. И.**

Б95      Устройства преобразования и обработки информации в системах подвижной связи : учеб. пособие / В. И. Быков ; Владим. гос. ун-т. – Владимир : Изд-во Владим. гос. ун-та, 2009. – 92 с.  
ISBN 978-5-89368-973-0

Содержит описание мобильных терминалов и устройств обработки и преобразования информации, используемых в них. Подробно рассмотрены устройства АЦП и ЦАП, обеспечивающие преобразование речевых сигналов. Описываются схемы и устройства кодирования, декодирования и перемежения цифровых сигналов. Рассмотрены интерфейсы человек-машина, используемые в мобильных телефонах: клавиатура, дисплей, микрофон, телефон. Кратко описаны интерфейсы связи мобильного терминала с другими устройствами: Bluetooth, USB, инфракрасный порт.

Предназначено для студентов 4-го курса специальности 210402 – средства связи с подвижными объектами дневной формы обучения при изучении дисциплины «Устройства преобразования и обработки информации в системах подвижной связи»

Ил. 52. Табл. 11. Библиогр.: 10 назв.

УДК 621.396.93  
ББК 32.84

ISBN 978-5-89368-973-0

© Владимирский государственный университет, 2009

## ПРЕДИСЛОВИЕ

В настоящее время основным направлением развития радиосвязи является мобильная связь. Важное устройство систем связи с подвижными объектами (СПС) – мобильный терминал.

Предлагаемое учебное пособие содержит описание мобильного терминала, принципы работы его элементов и узлов. Основное внимание уделено терминалам сотовой связи, т.е. мобильным телефонам и их разновидностям. Рассматриваются устройства преобразования речевого сигнала в цифровой код и обратно, кодирование сигнала с помощью кодеков различного вида. Важным этапом обработки сигнала в канале являются кодирование и перемежение. Эти устройства рассматриваются с точки зрения их технической (схемной) реализации.

Важны в мобильных терминалах и телефонах устройства, обеспечивающие интерфейсы с человеком: клавиатура, дисплей, микрофон, телефон. Описаны интерфейсы связи мобильного терминала с другими устройствами: *BlueTooth*, *USB*, инфракрасный порт. Кратко рассмотрены дополнительные устройства сотовых телефонов – фотокамера, интерфейс *Blue Tooth* и др.

Изучение устройств преобразования и обработки информации в СПС является важной и необходимой составляющей при подготовке студентов по специальности 210402 – средства связи с подвижными объектами дневной формы обучения. Для нее в федеральном компоненте Государственного образовательного стандарта предусмотрено изучение дисциплины «Устройства преобразования и обработки информации в системах подвижной связи». Поэтому предлагаемое учебное пособие будет необходимо студентам специальности 210402, так как другой отечественной литературы, охватывающей все эти вопросы, нет.

## **ВВЕДЕНИЕ**

Мобильная связь в настоящее время развивается очень быстро. Совершенствуются принципы работы, устройства базовых станций и прежде всего мобильных терминалов. Это связано с тем, что расширяется их функциональность и возрастают требования к их параметрам. Мобильные терминалы очень разнообразны, что связано со множеством видов мобильной связи (пейджинговая, сотовая и т.д.) и расширением их функций.

В связи с этим увеличиваются состав и сложность устройств преобразования и обработки информации. Назначение этих устройств – прием и передача сигналов, обработка сигналов с целью повышения их качества и преобразование их к виду, воспринимаемому человеком. Мобильные терминалы (МТ) и телефоны являются массовыми и потому их изучение и совершенствование должны быть в центре внимания специалистов по средствам связи с подвижными объектами.

### **Характеристики терминального оборудования мобильных станций**

1. Технические характеристики:
  - А. Внешние: диагональ экрана, разрешение экрана, разрешение камеры.
  - Б. Внутренние: мощность передатчика, чувствительность приемника, вид процессора и его характеристики, объем ОЗУ и ПЗУ, наличие и вид разъемов расширения, наличие беспроводных интерфейсов, вид операционной системы.
2. Эксплуатационные характеристики:
  - А. Общие: габаритные размеры, масса, энергопотребление, емкость аккумулятора и его вид, время работы в режимах ожидания и разговора, надежность, срок службы (гарантийный и номинальный), условия эксплуатации.

- Б. Требования живучести и стойкости, определяющие механические и климатические (температуру и влажность) условия эксплуатации и хранения, при которых обеспечивается сохранность параметров МТ.
- В. Нормируемые стыковочные параметры: мощность передатчика, чувствительность приемника.
- 3. Экономические характеристики:
  - А. Стоимость эксплуатации.
  - Б. Цена нового мобильного телефона.
- 4. Качественные характеристики:
  - А. Объем адресной книги.
  - Б. Вид антенны.
  - В. Дизайн мобильного телефона.
  - Г. Наличие русификации.
  - Д. Громкость звонка.
  - Е. Вид меню.
  - Ж. Наличие дополнительных функций: вибровозвонк, таймер разговора, часы, игры, встроенная громкая связь, диктофон, камера, *GPRS*, *WAP*-браузер.

### **Классификация МТ**

- 1. По виду:
  - А. Сотовый телефон.
  - Б. Смартфон.
  - В. КПК.
  - Г. *GPS*-навигатор.
- 2. По классу телефона:
  - А. Начального уровня (бюджетные, эконом-класса).
  - Б. Среднего класса.
  - В. Аппараты, защищенные от внешних воздействий.
  - Г. Бизнес-класса.
  - Д. Имидж-класса.

# 1. МОБИЛЬНЫЕ СТАНЦИИ

## 1.1. Функциональная схема типичной системы цифровой связи

Мобильные станции в пределах каждого класса модели в определенном стандарте сотовой связи различаются между собой не только количеством сервисных услуг, но и параметрами приемо-передающих трактов, конструкцией, внешним видом, блоком управления и т.п.

Функциональная схема (рис. 1.1) иллюстрирует распространение сигнала и этапы его обработки в типичной системе цифровой связи.

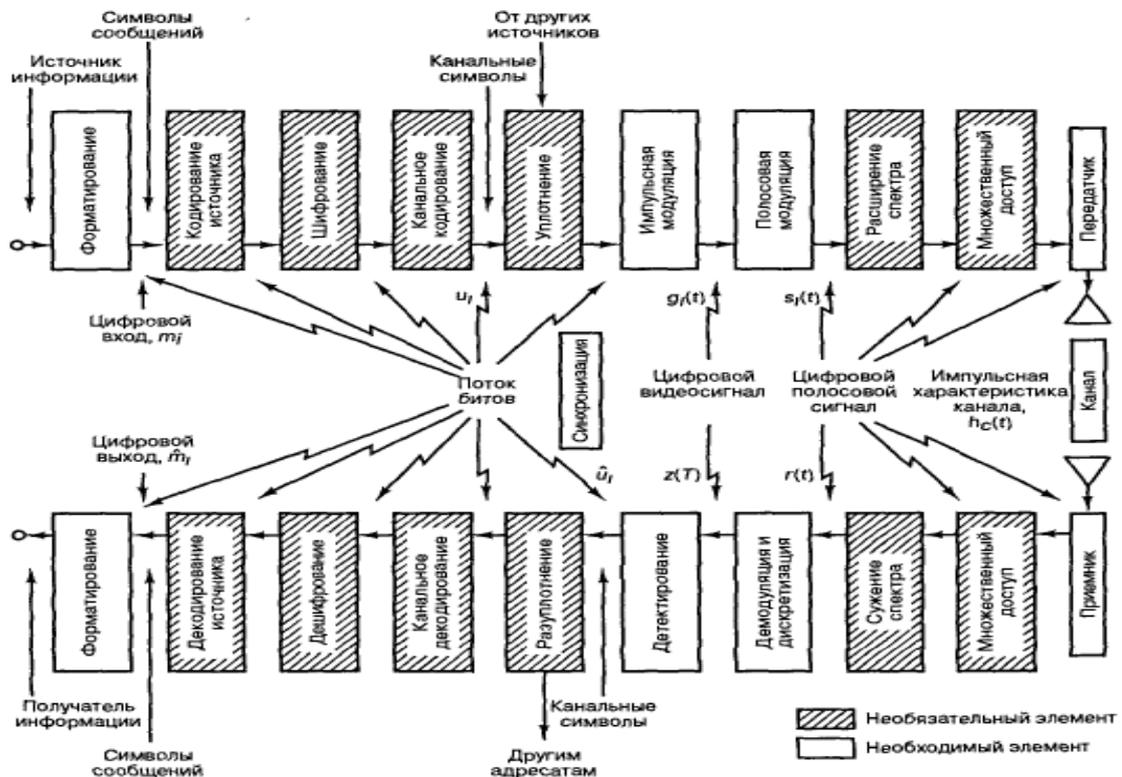


Рис. 1.1. Функциональная схема типичной системы цифровой связи

Верхние блоки – форматирование, кодирование источника, шифрование, канальное кодирование, уплотнение, импульсная модуляция, полосовая модуляция, расширение спектра и множественный доступ – отражают преобразования сигнала на пути от источника к передатчику, нижние блоки диаграммы – преобразования сигнала на пути от приемника к получателю информации.

Детальное описание блоков (см. рис. 1.1) представлено на рис. 1.2.



Рис. 1.2. Основные преобразования сигнала

## 1.2. Структурные схемы мобильной станции

Структурные схемы мобильной станции содержат ряд блоков (рис. 1.3).

На мировом рынке существует большое количество мобильных станций, однако все они имеют следующие типовые блоки:

- А. Блок управления: клавиатура, дисплей, микрофон и телефон.

- Б. Приемопередающий блок.
- В. Антенный блок с коммутатором каналов и диапазонов.
- Г. Логический блок: цифровой сигнальный процессор (*DSP, CPU*) с ОЗУ и ПЗУ, эквалайзер, каналный и речевой кодеки и др.
- Д. Идентификационный пользовательский модуль – *SIM*-карта.

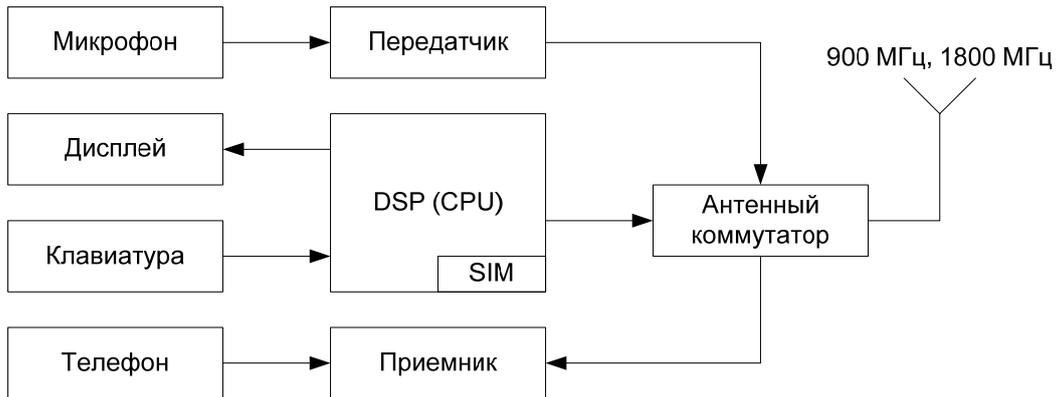


Рис. 1.3. Упрощенная структурная схема мобильной станции

Детальная структурная схема мобильной станции (рис. 1.4) состоит из следующих основных блоков.

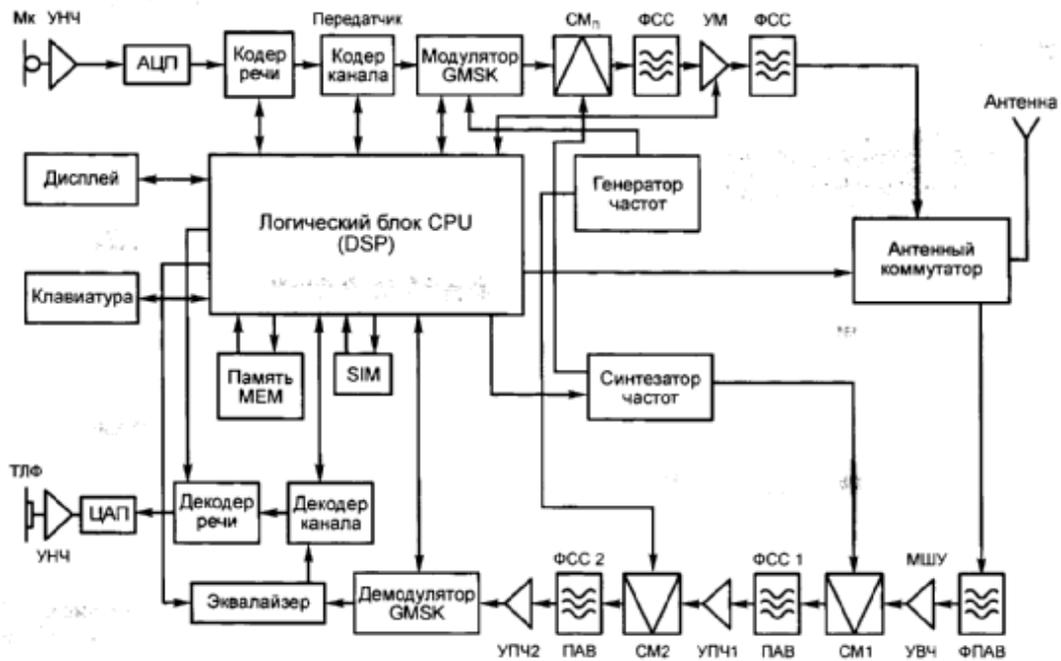


Рис. 1.4. Детальная структурная схема мобильной станции

Антенный блок включает:

- Собственно антенну.
- Антенный переключатель – электронный коммутатор, управляемый из *CPU* и подключающий вход антенны либо на выход передатчика, либо на вход приемника.

Передатчик:

- Речевой сигнал, преобразованный микрофоном ( $M_k$ ) в электрический аналоговый сигнал и усиленный усилителем низкой частоты (УНЧ), поступает в АЦП, на выходе которого речевой сигнал преобразуется в цифровую форму.

- Кодер речи осуществляет кодирование речевого сигнала, т.е. преобразование сигнала по определенным законам с целью сокращения объема информации, передаваемой по каналу связи.

- Кодер канала добавляет в цифровой сигнал дополнительную информацию, предназначенную для защиты от ошибок при передаче сигнала по линии связи; с той же целью информация подвергается перемежению; кроме этого кодер канала вводит в состав передаваемого сигнала информацию управления, поступающую от *CPU*.

- Фазовый модулятор реализует гауссовскую манипуляцию с минимальным частотным сдвигом, т.е. перенос цифрового сигнала в область радиочастотного канала.

- Смеситель (СМп) предназначен для переноса сигнала на выходе фазового модулятора с частотой  $f_{ФМ}$  в диапазон частот 890 – 915 МГц под действием сигнала, поступающего с синтезатора частот, управляемого из *CPU*.

- Фильтр сосредоточенной селекции (ФСС) обычно выполняется на кремниевом фильтре, настраивается на одну из выбираемых *CPU* несущих частот и реализует фильтрацию высших гармоник для уменьшения межканальных влияний.

- Усилитель мощности (УМ) усиливает высокочастотный сигнал с выхода ФСС и под действием команд из *CPU* меняет коэффициент усиления.

- Высокочастотный сигнал с выхода УМ через ФСС поступает в антенный коммутатор (АК) и в случае передачи речевого сигнала *CPU* переключает АК в режим передачи.

Приемник:

- Высокочастотный сигнал в диапазоне 935 – 960 МГц принимается антенной и через АК поступает в высокочастотный фильтр (обычно керамический), усиливается в высокочастотном малошумящем усилителе (МШУ).

- Первый смеситель СМ1 позволяет сдвинуть сигнал в более низкочастотную область, при этом на второй вход СМ1 гетеродинный сигнал подается с синтезатора частот (управляемого от *CPU*), ФСС1 выделяет сигнал на первой промежуточной частоте (ПЧ), и далее этот сигнал усиливается в тракте усилителей УПЧ1.

- Сигнал первой ПЧ поступает на вход второго смесителя СМ2, далее ФСС2 на ПАВ выделяет сигнал второй промежуточной частоты, который усиливается УПЧ2 и поступает в блок демодуляции.

- Блок демодуляции осуществляет вначале демодуляцию сигнала в фазовом демодуляторе (ФДМ), при этом на вход канального эквалайзера он поступает в виде цифрового видеосигнала.

- Канальный эквалайзер предназначен для компенсации той разности хода между составляющими радиолучами при многолучевом распространении радиоволн, которая приводит к межсимвольной интерференции. Далее цифровой сигнал попадает в декодеры канала и речевого сигнала.

- Декодер канала реализует процесс, обратный кодированию, и с учетом закодированной в кодере канала управляющей информации и речевого закодированного сигнала декодирует весь поток информации.

- Декодер речи декодирует цифровой поток речевой информации для дальнейшего преобразования ее из цифровой формы в аналоговую с помощью АЦП.

- Далее электрический сигнал аналоговой речевой информации подается на телефон.

*CPU* является управляющим устройством, он имеет свою оперативную и постоянную память (MEM).

Синтезатор частот (СЧ) является задающим генератором колебаний высокой частоты для передачи информации по радиоканалу. Синтезатор используется в работе как передающего устройства, так и приемного, переключаясь в разные области выделенной полосы частот для передачи и приема.

*SIM*-карта придается к МС в виде съемного модуля, взаимодействует с *CPU* и определяет процедуру аутентификации мобильной станции.

### ***Контрольные вопросы***

1. Стандарты систем подвижной связи (СПС). Системные требования к терминальному оборудованию.

2. Технические параметры и характеристики терминального оборудования и мобильных станций (МС) СПС.

3. Эксплуатационные и экономические параметры и характеристики терминального оборудования и мобильных станций СПС.

4. Качественные характеристики абонентских устройств. Дополнительные функции. Классификация МТ.

5. Структурные преобразования в СПС.

6. Структурная схема МС (МТ).

## 2. ПРЕОБРАЗОВАНИЕ РЕЧЕВОЙ ИНФОРМАЦИИ

### 2.1. Виды информации

Источник информации может быть *аналоговым* или *дискретным*. Выход аналогового источника может иметь любое значение из непрерывного диапазона амплитуд, тогда как выход источника дискретной информации – значения из конечного множества амплитуд. Источники аналоговой информации преобразуются в источники цифровой информации посредством дискретизации и квантования.

Сигналы можно классифицировать на детерминированные (при отсутствии неопределенности относительно их значения в любой момент времени) и случайные (в противном случае).

Определенный сигнал можно отнести либо к энергетическому, либо к мощностному. Энергетический сигнал имеет конечную энергию, но нулевую среднюю мощность, тогда как мощностной сигнал – нулевую среднюю мощность, но бесконечную энергию. Периодические и случайные сигналы выражаются через мощность, а сигналы, являющиеся детерминированными и непериодическими, – через энергию.

Информация делится на символьную, аудиоинформацию, видеоинформацию. Символьная информация делится на числовую и текстовую.

В табл. 2.1 приведены основные характеристики информации.

Таблица 2.1

Символьная информация		Аудиоинформация	Видеоинформация
Числовая	Текстовая		
Разрядность Пределы представления числа (диапазон)	Длина строки Разрядность символа	Частотная характеристика (зависимость уровня сигнала от частоты).	Цветовой диапазон (отличия двух оттенков)

Символьная информация		Аудиоинформация	Видеоинформация
Числовая	Текстовая		
	Длина алфавита	Для речи 300 – 3400 Гц Динамический диапазон D – отношение самого громкого звука к самому слабому Длительность сигнала (10 – 100 мс)	Динамический диапазон (градация яркости) Частотный диапазон (частота смены информации)

## 2.2. Форматирование и передача видеосигнала в цифровой связи

На рис. 2.1 представлена упрощенная функциональная схема типичной системы цифровой связи. Данные, уже имеющие цифровой формат, могут пропускать этап форматирования. Текстовая информация преобразовывается в двоичные цифры с помощью кодера. Аналоговая информация форматируется с использованием трех отдельных процессов: дискретизации, квантования и кодирования. Во всех случаях после формирования получается последовательность двоичных цифр.

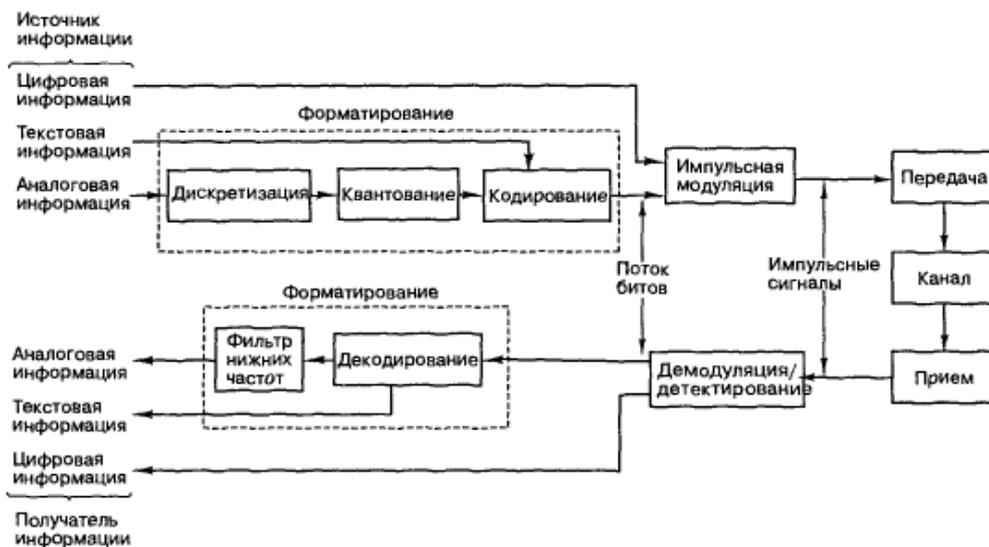


Рис. 2.1. Упрощенная функциональная схема системы цифровой связи

Преобразование потока битов в последовательность импульсных сигналов происходит в блоке «Импульсная модуляция». На выходе модулятора получим последовательность видеопульсов, характеристики которых соответствуют характеристикам цифр, поданных на вход. После передачи по каналу импульсные сигналы восстанавливаются (демодулируются) и проходят этап детектирования; цель последнего этапа, форматирования, – восстановление (с определенной степенью точности) исходной информации.

Процессы дискретизации и квантования сигнала осуществляются аналого-цифровыми преобразователями (АЦП), обратные процессы - цифроаналоговыми (ЦАП).

### **ЦАП**

Основным параметром ЦАП является относительная разрешающая способность  $\Delta = 1/(2^n - 1)$ , характеризующаяся величиной, обратной количеству уровней  $K=2^n - 1$ , которые может выдать ЦАП на выходе.

Виды ЦАП:

- с резистивными матрицами;
- безматричные.

Матричные ЦАП делятся на подвиды:

- с набором резисторов;
- матрицами типа  $R-2R$ .

### **АЦП**

Характеристики АЦП: точность, разрешающая способность, быстродействие, определяемое временем отсчета

$$T_{\text{отс}} = 1/(1,1 \cdot f_{\text{пр}}).$$

### Классификация АЦП

1. Параллельные.
2. Последовательные:

- А) последовательного приближения;
- Б) последовательного счета;
- В) следящие;
- Г) интегрирующие: однотоктные; многотоктные; сиг-ма-дельта; преобразователи напряжение - частота.

3. Последовательно-параллельные:

- А) многотоктные;
- Б) многоступенчатые;
- В) конвейерные.

Более детально ЦАП и АЦП рассмотрены в [1].

### 2.3. Кодирование речи

Кодирование речи осуществляется на основе двух подходов:

- представление временной формы сигнала;
- спектральное представление сигнала.

#### Импульсно-кодовая модуляция (ИКМ)

Речевой сигнал описывается амплитудной характеристикой (рис. 2.2), частотной характеристикой, временем нарастания звука. При преобразовании речевого сигнала в цифровую форму используются частота дискретизации и количество уровней квантования, что обуславливает скорость передачи  $X = 8 \text{ кбайт/с} = 64 \text{ кбит/с}$ .

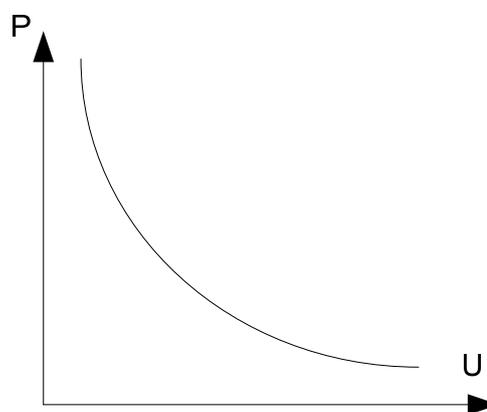


Рис. 2.2. Амплитудная характеристика речи

Частотная характеристика речи: 300 – 3400 Гц.

Время нарастания звука  $t = 20 \text{ мс}$ .

Частота дискретизации  $F_n = 6800 – 8000 \text{ отсчетов/с}$ .

Количество уровней квантования  $N_{\text{кв}} = 256$  при  $K = 8$  бит = 1 байт. При преобразовании речевого сигнала в цифровую форму используют характеристики сжатия (рис. 2.3).

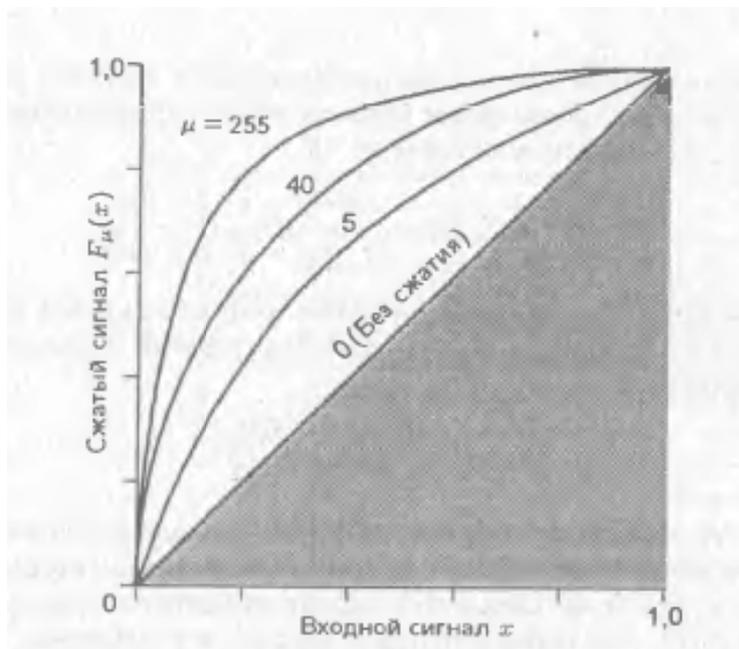


Рис. 2.3. Характеристики сжатия, используемые в цифровых ИКМ-сетях Северной Америки и Японии

Характеристика, соответствующая  $\mu$ -закону, для нормированного диапазона кодирования  $\pm 1$  определяется выражением

$$F_{\mu}(x) = \sin(x) \times \ln(1+\mu|x|)/\ln(1+\mu), \quad -1 \leq x \leq 1.$$

Для малых значений  $x$  характеристика  $F(x)$  приближается к линейной функции, а для больших — к логарифмической.

Стандартным законом сжатия в Европе является  $A$ -закон, который определяется следующим образом:

$$F_A(x) = \sin(x) \cdot (1 + \ln A|x|) / (1 + \ln A), \quad 1/A \leq |x| \leq 1$$

$$F_A(x) = \sin(x) \cdot A|x| / (1 + \ln A), \quad 0 \leq |x| \leq 1/A$$

Характеристика  $F_A(x)$  — действительно логарифмическая функция при  $|x| > 1/A$  и действительно линейная функция при  $|x| < 1/A$ .

Итак, в системах телефонии сигналы речи, модулированные в полосе тональных частот, ограничены верхней частотой 3,4 кГц. Чтобы преобразовать этот аналоговый сигнал в цифровой ИКМ поток данных, применяется дискретизация с частотой  $f = 8000$  отсчетов/с. Каждый отсчет фиксируется на одном из 256 уровней квантования. Для этого количества уровней квантования требуется 8 информационных битов. Таким образом, один канал тональной частоты будет иметь скорость передачи 64 кбит/с.

## Дифференциальная импульсно-кодовая модуляция (ДИКМ)

В ИКМ каждый отсчет кодируется независимо от других. Однако у многих источников сигнала при стробировании со скоростью Найквиста или быстрее проявляется значительная корреляция между последовательными отсчетами. Другими словами, изменения амплитуды между последовательными отсчетами в среднем относительно малы. Относительно простые решения получаются при кодировании разности между последовательными отсчетами, а не самих отсчетов. Поскольку можно ожидать, что разность между отсчетами сигнала меньше, чем действительные значения отсчетов, то потребуется меньшее число бит для представления разностного сигнала. Это и есть ДИКМ.

Простейшие схемы ИКМ и ДИКМ представлены на рис. 2.4, 2.5.

В схеме (рис. 2.6) предсказатель встроен в цепь обратной связи, охватывающей квантователь. Вход предсказателя обозначен  $\tilde{x}_n$ . Он представляет сигнальный отсчет  $x_n$ , видоизмененный процессом квантования, а выход предсказателя равен

$$\hat{x}_n = \sum_{i=1}^p a_i \tilde{x}_{n-i}.$$



Рис. 2.4. Простейшая схема ИКМ

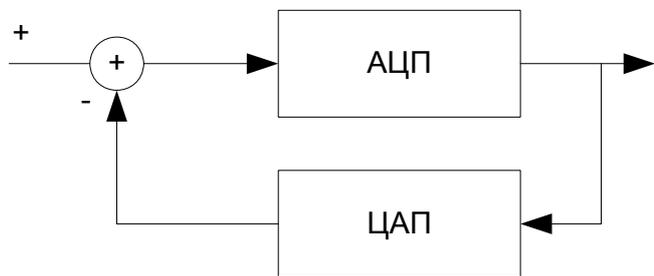


Рис. 2.5. Простейшая схема ДИКМ

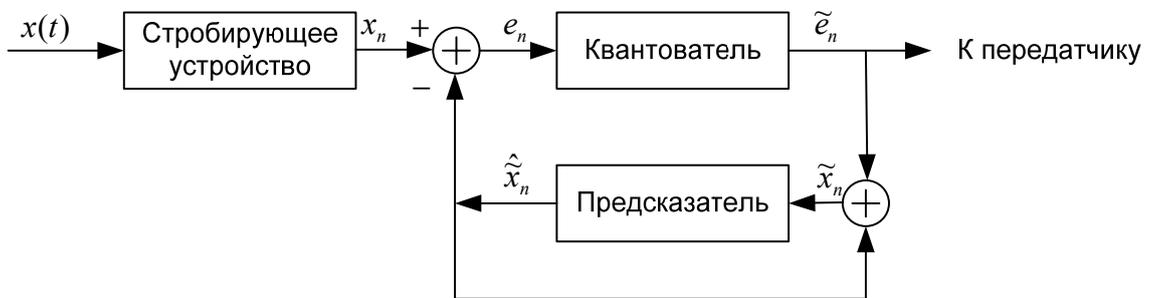


Рис. 2.6. Схема кодера речи ДИКМ

Разность  $e_n = x_n - \hat{x}_n$  является входом квантователя, а  $\tilde{e}_n$  обозначает его выход. Величина квантованной ошибки предсказания  $\tilde{e}_n$  кодируется последовательностью двоичных символов и передается через канал в пункт приема. Квантованная ошибка  $\tilde{e}_n$  также суммируется с предсказанной величиной  $\hat{x}$ , чтобы получить  $\tilde{x}_n$ .

Коэффициенты  $a_i$  выбираются таким образом, чтобы минимизировать некоторую функцию ошибки между  $x_n$  и  $\tilde{x}_n$ . Математически и практически подходящей функцией является среднеквадратическая ошибка (СКО).

$$\varepsilon_p = E(e_n^2) = E[(x_n - \sum_{i=1}^p a_i x_{n-i})^2]$$

Предполагая выход источника стационарным, можно выразить функцию СКО в виде:

$$\varepsilon_p = \phi(0) - 2 \sum_{i=1}^p a_i \phi(i) + \sum_{i=1}^p \sum_{j=1}^p a_i a_j \phi(i-j)$$

где  $\phi(m)$  – автокорреляционная функция последовательности отсчетов сигнала  $x_n$ . Если  $\phi(m)$  априори неизвестна, то она может быть определена по реализации отсчетов  $\{x_n\}$  посредством соотношения

$$\hat{\phi}(j) = \frac{1}{N} \sum_{i=1}^{N-j} x_i x_{i-j}, \quad j = 1, 2, \dots, p$$

На рис. 2.7. и 2.8 представлены схемы декодера речи.

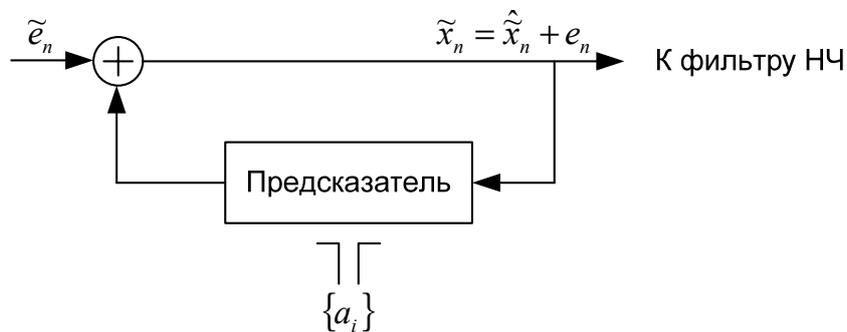


Рис. 2.7. Декодер речи ДИКМ

Улучшение качества оценки (предсказанная величина отсчетов сигнала) можно получить включением в оценку линейно отфильтрованных последних значений квантованной ошибки.

Конкретно оценку можно выразить так:

$$\hat{x}_n = \sum_{i=1}^p a_i \tilde{x}_{n-i} + \sum_{i=1}^m b_i \tilde{e}_{n-i},$$

где  $\{b_i\}$  – коэффициенты фильтра для квантованной последовательности ошибок  $\tilde{e}_n$ .

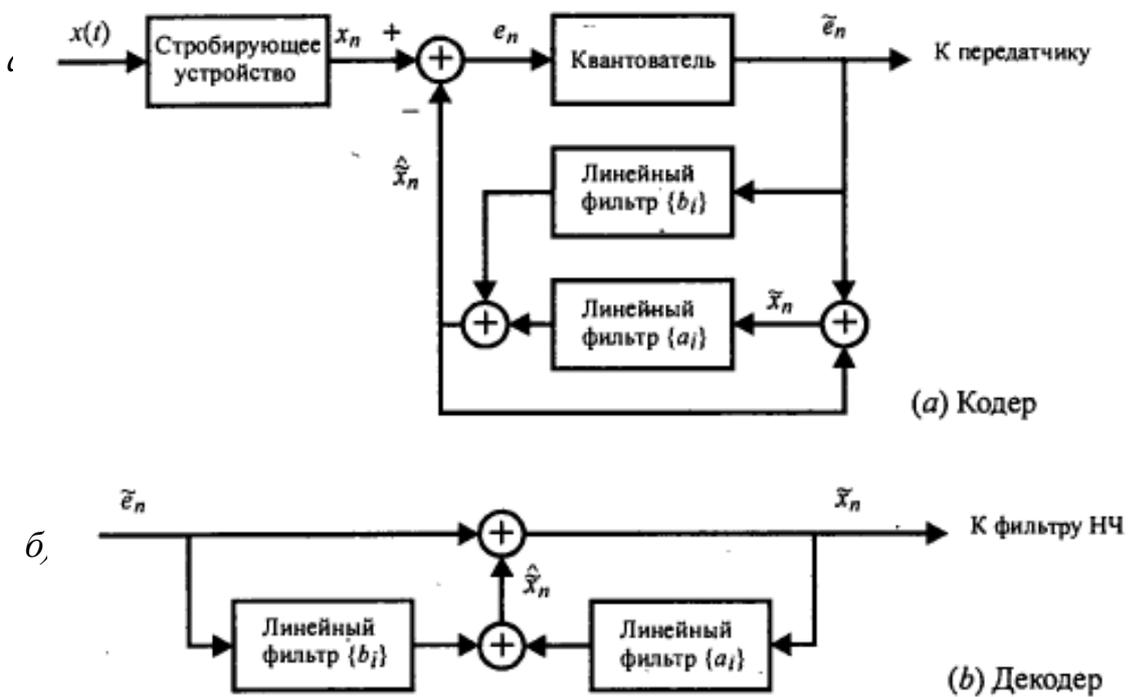


Рис. 2.8. Кодер (а) и декодер (б) модифицированной ДИКМ

### Дельта-модуляция (ДМ)

Дельта-модуляцию можно рассматривать как простейшую форму ДИКМ, в которой используется двухуровневый (1-битовый) квантователь в соединении с фиксированным предсказателем первого порядка (рис. 2.9).

Предсказанное значение  $x_n$  в действительности является предыдущим отсчетом  $x_{n-1}$ , измененным шумом квантования  $q_{n-1}$ . Разностное уравнение определяет интегратор со входом  $\tilde{e}_n$ .



Рис. 2.9. Кодер (а) и декодер (б) при дельта-модуляции (ДМ)

Следовательно, эквивалентной реализацией предсказателя первого порядка является интегратор с входом, равным квантованному сигналу ошибки  $\tilde{e}_n$ . В общем случае квантованный сигнал ошибки масштабируется некоторой величиной, скажем  $\Delta$ , которая называется размером шага. Для достаточной точности кодирования скорость стробирования при дельта-модуляции должна быть высокой, более 5 частот Найквиста. Эквивалентная схема показана на рис. 2.10.

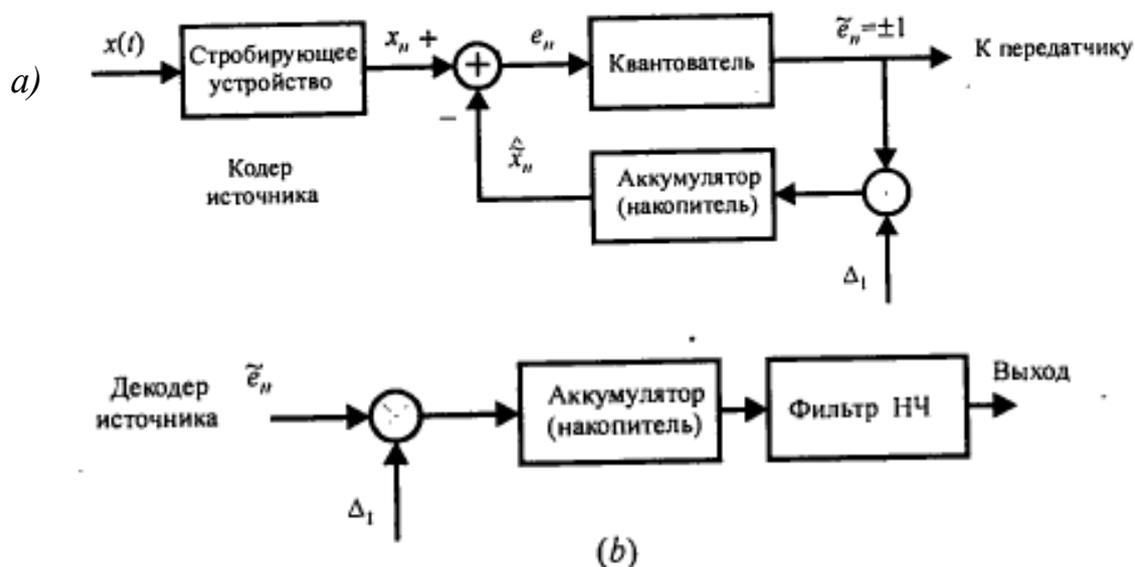


Рис. 2.10. Кодер (а) и декодер (б) при упрощенной дельта-модуляции

При заданной скорости стробирования характеристика качества ДМ кодера ограничена двумя видами искажения (рис. 2.11). Одно называется перегрузкой по наклону. Оно вызывается размером шага  $\Delta$ , который слишком мал, чтобы следить за сигналом, имеющим крутой наклон. Второй тип искажений, названный шумом дробления, возникает от использования размера шага, который слишком велик для сигнала, имеющего малый наклон. Необходимость минимизации обоих видов искажений приводит к противоречивым требованиям для выбора размера шага  $\Delta$ . Решение сводится к выбору  $\Delta$ , который минимизирует сумму средних квадратов от двух видов искажений.



Рис. 2.11. Виды искажений при ДМ

В технической литературе были исследованы и описаны несколько других вариантов адаптивного кодирования ДМ. Один из них называют дельта-модуляцией с непрерывно меняющимся наклоном (ДМ НМН) (рис. 2.12). В ДМ НМН адаптивный параметр размера шага можно выразить так:

$$\Delta_n = \alpha \Delta_{n-1} + k_1,$$

если  $\tilde{e}_n, \tilde{e}_{n-1}$  и  $\tilde{e}_{n-2}$  имеют одинаковые знаки; в противном случае

$$\Delta_n = \alpha \Delta_{n-1} + k_2.$$

Параметры  $\alpha, k_1, k_2$  выбираются так, что  $0 < \alpha \leq 1$  и  $k_1 \gg k_2$ .

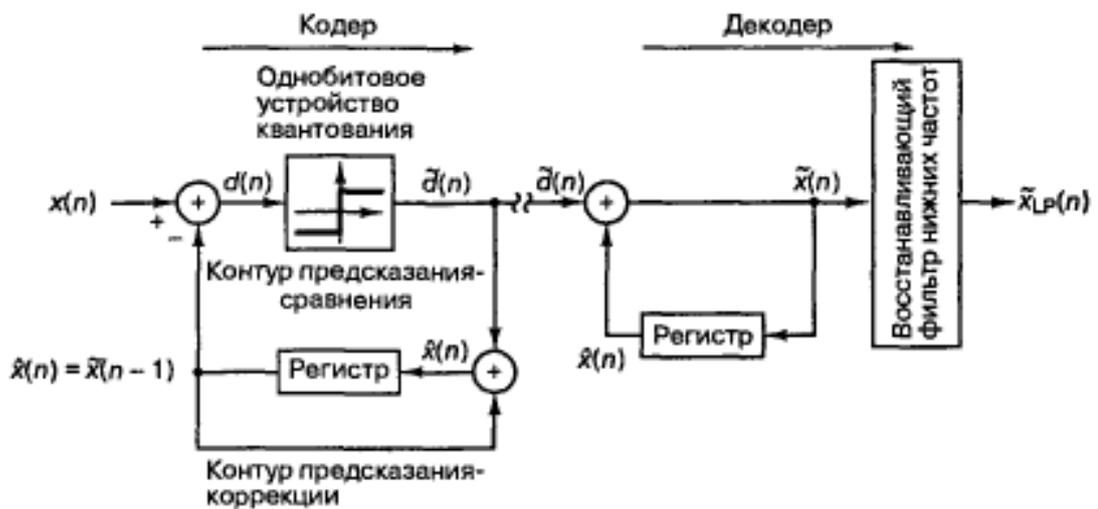


Рис. 2.12. Блок-схема модифицированного одноотводного линейного предсказателя (дельта-модулятора)

Корреляцию поступающих на модулятор выборочных данных можно усилить посредством предварительной фильтрации данных интегратором и компенсацией этой фильтрации с помощью выходного фильтра - дифференциатора. Эта структура изображена на рис. 2.13, где интеграторы, дифференциатор и задержка выражены в терминах z-преобразования.

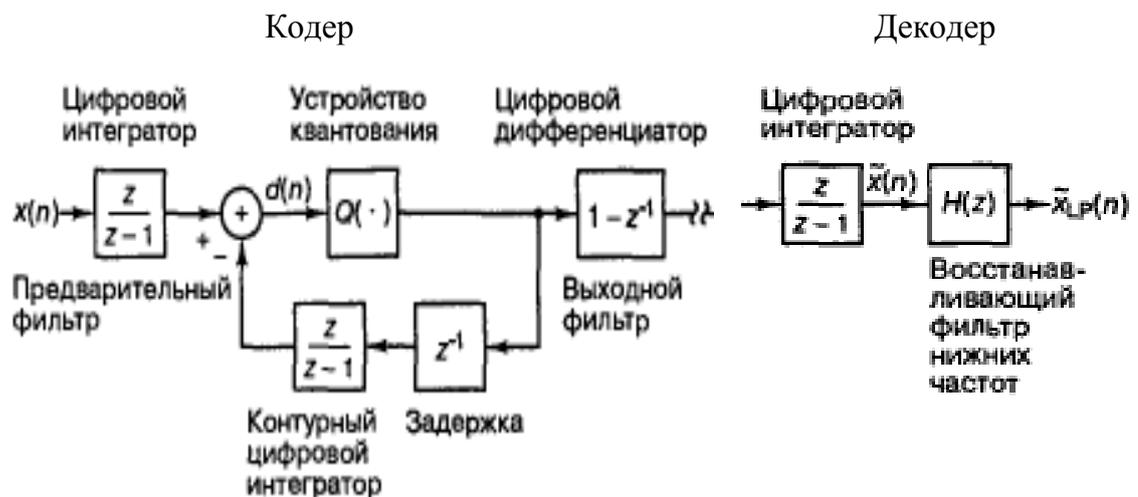


Рис. 2.13. Блок-схема модифицированного одноотводного линейного предсказателя на основе z-преобразования

## Сигма-дельта-модуляция

Для получения выигрыша можно перегруппировать блоки прохождения сигнала. На вход кодера поступают сигналы с выходов двух цифровых интеграторов, которые затем суммируются и вводятся в контур квантования. Первая модификация состоит в том, чтобы использовать один цифровой интегратор, сдвигая два интегратора через суммирующее устройство в кодер. Вторая модификация состоит в том, что выходной фильтр-дифференциатор может быть сдвинут в декодер, что делает ненужным цифровой интегратор на входе в декодер. Таким образом декодер трансформируется в восстанавливающий фильтр нижних частот. Полученная упрощенная схема модифицированной системы ДИКМ, названная сигма-дельта-модулятором, содержит интегратор (сигма) и модулятор ДИКМ (дельта) (рис. 2.14).

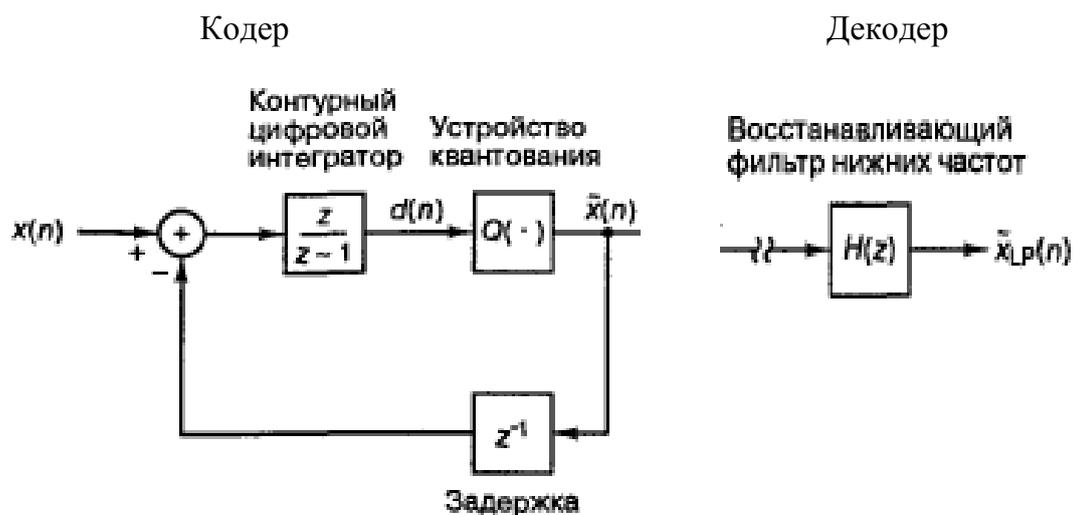


Рис. 2.14. Сигма-дельта-модулятор

Предшествующая ошибка, образованная как разность между входом и выходом устройства квантования, помещается в регистр запаздывания для использования в качестве оценки следующей ошибки квантования. Схему прохождения сигнала можно нарисовать так, чтобы акцентировать внимание на двух входах (сигнал и шум квантования) и на двух контурах (включающий уст-

ройство квантования и не включающий его). Эта форма изображена на рис. 2.15 и является общепринятой для точного изображения участка обратной связи цифрового интегратора.

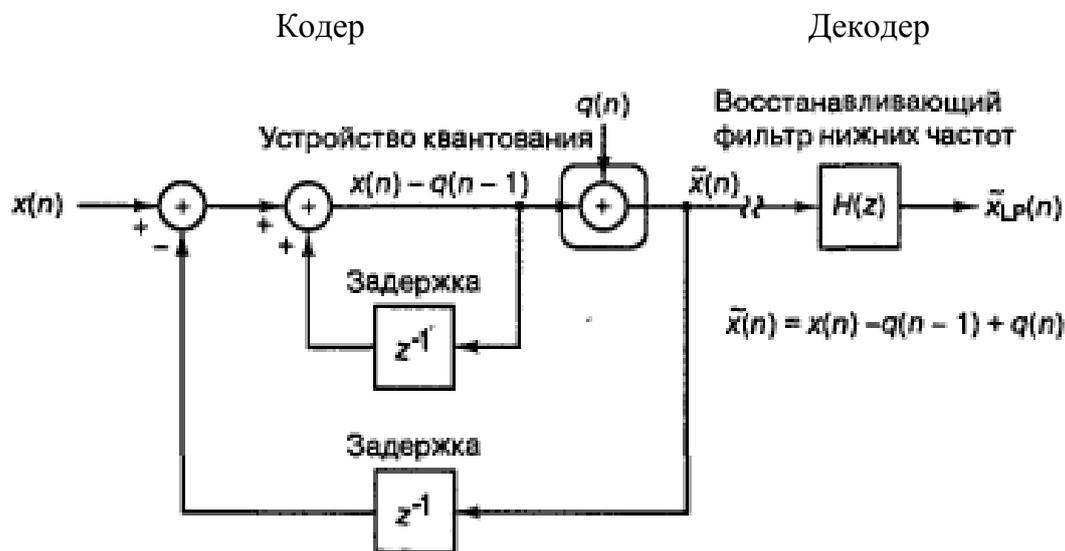


Рис. 2.15. Устройство квантования с обратной связью по шуму, изображенное как сигма-дельта-модулятор

Применение адаптивных методов снижает шум квантования и увеличивает динамический диапазон дельта-модуляторов. Одна из первых реализаций идеи дельта-модулятора с адаптивным изменением размера шага, выполненная в виде БИС, приведена в [2].

### Спектральное кодирование сигнала. Линейное кодирование с предсказанием

Спектральное представление сигнала речи основано на разделении спектра на полосы нижних и верхних частот, а полос – на подполоски. Адаптивные предсказатели были созданы для предсказания или создания хороших оценок входного сигнала. В адаптивной форме предсказываемые коэффициенты вычисляются как побочная информация на основе периодического изучения входных данных. Затем разность между входом и предсказанием передается получателю для разрешения ошибки предсказания. Линейные кодеры с

предсказанием (*LPC*) (рис. 2.16) являются естественным расширением *N*-отводных кодеров с предсказанием.

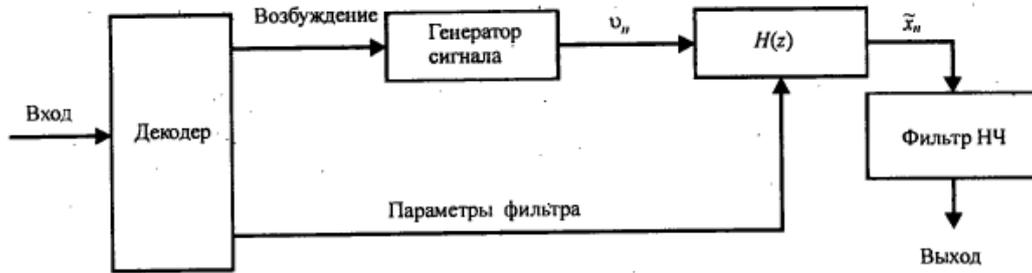


Рис. 2.16. Блок-схема синтезатора сигнала (декодера источника) для ЛПК системы

Если коэффициенты фильтра периодически вычисляются с помощью оптимального алгоритма, предсказание является настолько хорошим, что информации об ошибке предсказания, которую нужно передавать приемнику, не существует. Вместо того чтобы передавать эти ошибки предсказания, система *LPC* передает коэффициенты фильтра и озвученное/неозвученное руководство к действию для фильтра. Таким образом, единственными данными, посланными в *LPC*, является высококачественная побочная информация классического адаптивного алгоритма. Модель *LPC* для синтеза голоса изображена на рис. 2.17. Побочная информация - коэффициенты *LPC* и параметры активации, соответствующие каждому интервалу 20 мс.

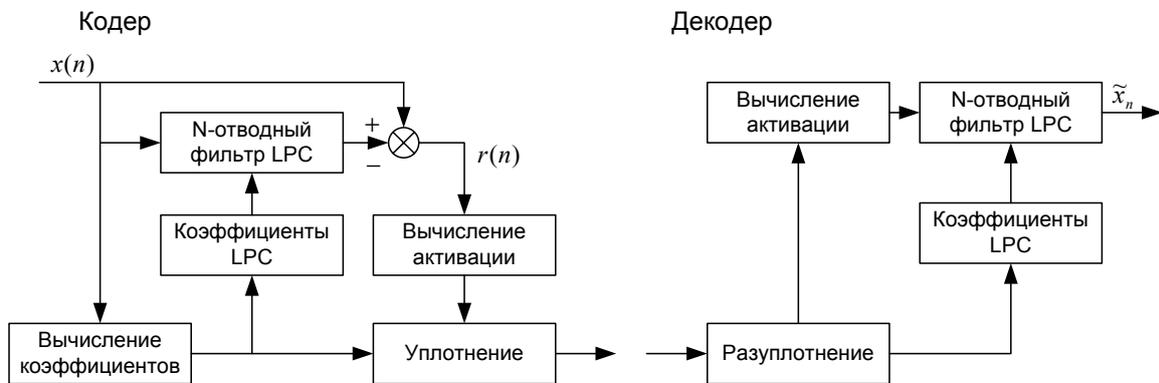


Рис. 2.17. Модель *LPC* для синтеза голоса



ближается к качеству речи при радиовещании и в музыкальных сигналах. Данный кодер использует дополнительные фильтры нижних и верхних частот для отделения входной полосы частот в 7 кГц. После этого речь дискретизируется с частотой в 16 кГц в более высокую и более низкую подполосы, каждая из которых выбирается с частотой 8 кГц. Функции обоих фильтров и операция повторной дискретизации реализованы в цифровом фильтре, известном как квадратурный зеркальный фильтр. Независимые кодеры АДИКМ обрабатывают временные ряды сокращенных полос частот от двух фильтров и выдают скорости в 48 и 16 Кбит/с соответственно на выходе низкой и высокой полос. Блочная диаграмма широкополосного аудиокодера, работающего со скоростью 64 Кбит/с, изображена на рис. 2.19.

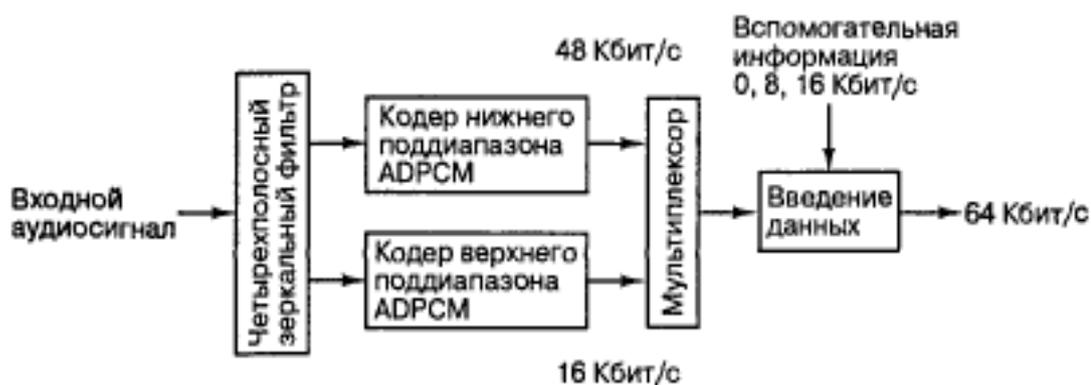


Рис. 2.19. Блочная диаграмма широкополосного аудиокодера

Эти кодеры представляют собой модифицированную версию речевых кодеров АДИКМ МККТТ *G.721*, которые используют фильтры с обратным предсказанием, основанные на закодированном разностном сигнале. Отбрасывание младшего бита коэффициентов предсказывающего фильтра позволяет этому кодеру работать как со скоростями 56 и 48 Кбит/с, так и с номинальной скоростью 64 Кбит/с. При сниженной скорости передачи битов система связи может присваивать неиспользованные биты вспомогательному потоку данных, который передается со скоростью

8 и 16 Кбит/с, если канал поддерживает фиксированную выходную скорость в 64 Кбит/с. Предсказатель использует структуру с 6 нулями и 2 полюсами.

### **Некоторые реализации экономичных кодеков**

В вокодере с возбуждением остатком и линейным предсказанием (*RELP*) выделенный остаток экономично кодируется и передается. В *RELP*-вокодерах фильтр речевого тракта действует как механизм формирования шума для любого белого шума квантования, генерируемого при кодировании остатка. В ряде экспериментов доказано, что для получения хорошего качества связи достаточна передача только части спектра остатка (от 0 до 1000 Гц). Эта процедура раскрывается в [2, с. 57].

Речевой кодек цифровой сотовой системы стандарта США IS-54 использует специфицированный алгоритм, являющийся вариантом *CELP*, и называется кодированием с линейным предсказанием и возбуждением векторной суммой – *VSELP*. В этом алгоритме используется кодовый словарь. Кодовый словарь имеет predetermined structure, в границах которой требуется произвести некоторые вычисления, чтобы процесс поиска в словаре мог быть значительно сокращен.

На структурной схеме цифрового декодера речи показаны различные параметры, которые должны быть определены и закодированы речевым кодером. Речевой декодер использует два кодовых словаря *VSELP* возбуждения. Каждый имеет собственный коэффициент передачи. Оба сигнала возбуждения из кодовых словарей умножаются на соответствующие коэффициенты передачи и суммируются, чтобы создать возбуждение объединенного кодового словаря. Скорость передачи этого кодека  $f = 7,95$  Кбит/с. Основные параметры, которые должны быть определены и закодированы речевым кодером, приведены в табл. 2.2. Структурная схема цифрового декодера речи *VSELP* приведена в [2, с. 63, рис. 2.3.1].

Таблица 2.2

Параметр	Значение параметра
Частота дискретизации	8 кГц
Длина кадра $N_F$	160 отсчетов (20 мс)
Длина субкадра $N$	40 отсчетов (5 мс)
Порядок краткосрочного предсказателя $N_P$	10
Число отводов для долговременного предсказания	1
Число бит в 1-м кодовом слове (число базисных векторов) $M1$	7
Число бит во 2-м кодовом слове (число базисных векторов) $M2$	7

В интервале времени, соответствующем речевому кадру (20 мс), передаются коды параметров, содержащие 159 бит, которые распределяются следующим образом (табл. 2.3).

Таблица 2.3

Параметр алгоритма	Значение параметра
Кратковременные коэффициенты фильтра $\alpha_i$	38 бит на кадр
Энергия кадра $R_0$	5 бит на кадр
Задержка $L$	7 бит на субкадр (28 бит на кадр)
Кодовые слова $I, H$	(7+7) бит на субкадр (56 бит на кадр)
Коэффициенты усиления	8 бит на субкадр (32 бита на кадр)
Первый коэффициент отражения $LPC1$	6 бит

## Кодеки для европейских систем *CT-2* и *DECT*

В качестве алгоритма кодирования речи для стандартов бесшнурового телефона второго поколения *CT-2* и цифровой европейской бесшнуровой системы связи *DECT*, предусматривающих унифицированный интерфейс, определен алгоритм АДИКМ, формирующий поток со скоростью 32 кбит/с. Базовые функциональные схемы этих кодеков приведены на рис. 2.20.

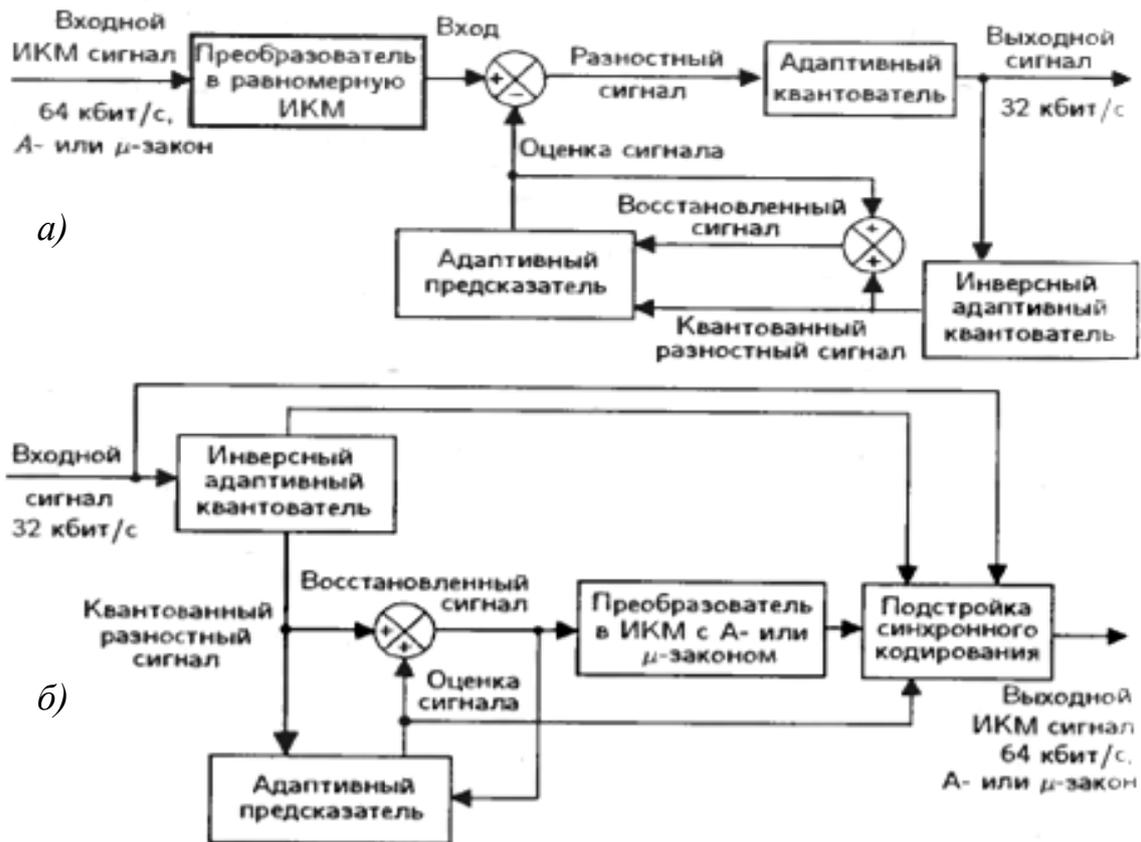


Рис. 2.20. Кодер (а) и декодер (б) АДИКМ системы связи *DECT*

## Кодек стандарта *GSM*

В стандарте *GSM* в системе кодирования речевых сигналов используется метод линейного предсказания с возбуждением регулярной последовательностью импульсов и долговременным предсказанием, т. е. метод *RPE-LTP* [5]. Схема приведена на рис. 2.21 и содержит следующие блоки.

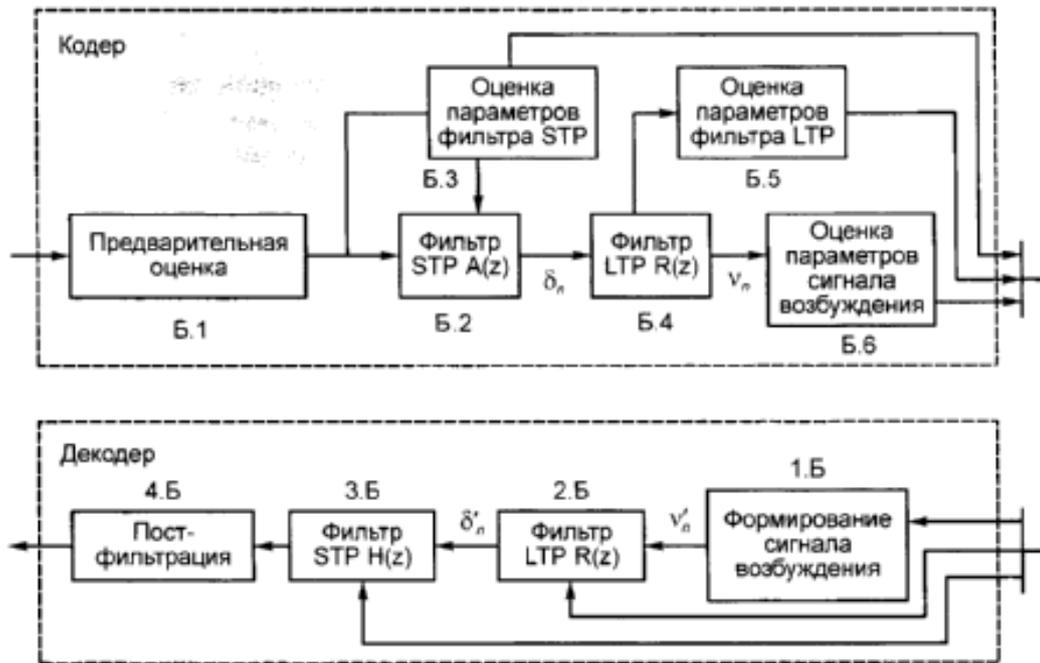


Рис. 2.21. Упрощенная блок-схема кодека стандарта GSM

### Кодер:

Кодер состоит из основных блоков Б.1 ... Б.6.

**Б1 – блок предварительной обработки** – осуществляет следующие функции:

- Предсказание входного цифрового сигнала  $S_n$  (после АЦП) при помощи цифрового фильтра, подчеркивающего верхние частоты.
- Разделение входного сигнала  $S_n$  на сегменты по 160 выборок (20 мс).
- Взвешивание каждого сегмента окном Хемминга («косинус на пьедестале» – амплитуда сигнала плавно спадает от центра окна к краям).

**Б2 – фильтр-анализатор STP:** с выхода блока Б1 сигнал фильтруется решетчатым фильтром-анализатором кратковременного линейного предсказания и по его выходному сигналу – остатку предсказания STP  $\delta_n$  – оцениваются параметры долговременного предсказания LTP: коэффициент предсказания  $g$  и временная задержка  $d$  в блоке Б5.

**Б3 – блок оценки параметров фильтра  $STP$ :** для каждого сегмента (длительностью 20 мс) оцениваются параметры фильтра-анализатора  $STP$  – 8 коэффициентов частичной корреляции  $k_i$  ( $i = 1 \dots 8$ ), при порядке предсказания  $M = 8$  ( $i = 1 \dots M$ ), которые для передачи по каналу связи преобразуются в логарифмические отношения площадей  $LAR$   $r_i$ , причем для функции логарифма используется кусочно-линейная аппроксимация.

**Б4 – фильтр-анализатор  $LTP$ :** сигнал остатка предсказания  $\delta_n$  фильтруется фильтром-анализатором  $LTP$  долговременного линейного предсказания, при этом формируется остаток предсказания  $LTP$  –  $v_n$ .

**Б5 – блок оценки параметров фильтра  $LTP$ :** в блоке по остатку  $STP$  –  $\delta_n$  оцениваются параметры долговременного предсказания  $g$  и  $d$ . При этом выборочный сегмент остатка кратковременного предсказания, имеющий 160 выборок, разделяется на четыре подсегмента по 40 выборок в каждом, и параметры  $g$  и  $d$  оцениваются для каждого подсегмента в отдельности, причем для оценки величины временной задержки  $d$  для текущего подсегмента используется скользящий подсегмент из 40 выборок, перемещающийся в пределах предшествующих 128 выборок сигнала остатка предсказания  $\delta_n$ .

**Б6 – блок оценки параметров сигнала возбуждения:** выходной сигнал фильтра-анализатора  $LTP$  – остаток предсказания  $v_n$  – фильтруется сглаживающим фильтром, и по нему формируются параметры 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 интервалы дискретизации сигнала после АЦП, т. е. исходного сигнала) и имеющих различные амплитуды. Для формирования сигнала возбуждения 40 импульсов подсегмента сглаженного остатка  $v_n$  обрабатываются следующим образом:

- Последний сороковой импульс отбрасывается.
- Первые 39 импульсов разбиваются на три последовательности:
  - 1) в первой последовательности – импульсы 1, 4, ..., 37;
  - 2) во второй – импульсы 2, 5 ... 38;
  - 3) 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. задается номер последовательности, выбранной в качестве сигнала возбуждения для данного подсегмента.

Таким образом, выходная информация кодера речевого сигнала для одного сегмента речи (длительностью 20 мс) включает:

- параметры фильтра-анализатора *STP* – 8 коэффициентов логарифмического отношения площадей *LAR*  $r_i$  ( $i = 1 \dots 8$ ) – один набор на весь сегмент;
- параметры фильтра-анализатора *LTP* – коэффициент долговременного предсказания  $g$  и временную задержку  $d$  для каждого из четырех подсегментов;
- параметры сигнала возбуждения – номер последовательности  $n$ , максимальная амплитуда  $B$ , нормированные амплитуды  $b_i$  ( $1 \dots 13$ ) импульсов последовательности – для каждого из четырех подсегментов.

Число бит, отводимых на кодирование передаваемых параметров, приведено в табл. 2.4. Всего для одного сегмента речи (длительностью 20 мс) передается 260 бит информации, т. е. кодер речи осуществляет сжатие информации почти в 5 раз ( $1280/260 = 4,92$ , где  $1280 = 160 \cdot 8$  – число бит предсказания).

Таблица 2.4

Передаваемые параметры	Число бит	Примечания
Параметры фильтра <i>STP</i> ( <i>LAR</i> $r_i$ ( $i = 1 \dots 8$ ))	36	$r_1, r_2$ – по 6 бит $r_3, r_4$ – по 5 бит $r_5, r_6$ – по 4 бита $r_7, r_8$ – по 3 бита
Параметры фильтра <i>LTP</i> (коэффициент $g$ и время задержки $d$ для каждого из подсегментов)	36	$g$ – 2 бита $d$ – 7 бит
Параметры сигнала возбуждения ( $n$ – номер последовательности, $B$ – максимальная амплитуда, $b_i$ – амплитуды импульсов ( $i = 1 \dots 13$ ) для каждого из четырех подсегментов)	188	$n$ – 2 бита $B$ – 6 бит $B_i$ – 3 бита
Всего за 20 мс в сегменте	260	–

### Декодер

Рассмотрим принцип декодирования информации, поступающей по каналу связи в соответствии со схемой.

**1Б – блок формирования сигнала возбуждения**, используя принятые параметры сигнала возбуждения, восстанавливает 13-импульсную последовательность сигнала возбуждения для каждого подсегмента сигнала речи, включая амплитуды импульсов и их расположение во времени, т. е. сигнал  $v'_n$  как остаток долговременного предсказания поступает на вход блока 2Б.

**2Б – блок фильтра-синтезатора *LTP***. Сформированный таким образом сигнал возбуждения  $v'_n$  фильтруется фильтром-синтезатором долговременного предсказания *LTP*, на выходе которого получается восстановленный остаток кратковременного предсказания  $\delta'_n$ . Следует отметить, что на второй вход фильтра *LTP* поступают сигналы, передающие параметры долговременного линейного предсказания, а именно коэффициент долговременного предсказания  $g$  и временная задержка  $d$ .

**3Б – блок фильтра-синтезатора STP.** Сигнал – остаток кратковременного предсказания  $\delta'_n$  – поступает на решетчатый фильтр-синтезатор STP, который реализует фильтрацию сигнала  $\delta'_n$ , причем параметры фильтра предварительно преобразуются из LAR  $r_i$  в коэффициенты частичной корреляции  $k_i$ .

**4Б – блок фильтрации.** В блоке выходной сигнал с фильтра-синтезатора STP фильтруется цифровым фильтром, восстанавливающим амплитудные соотношения частотных составляющих речевого сигнала, т. е. в кодере блок Б1 вносит компенсирующие предискажения, а в блоке 4Б декодера восстанавливается цифровой сигнал речи.

Таким образом, после декодирования получается цифровой речевой сигнал  $S'_n$ , соответствующий входному цифровому сигналу  $S_n$ .

## 2.4. Обзор методов кодирования речевых сигналов

Передача речевых сигналов по телефонным линиям, радиоканалам и спутниковым каналам составляет наибольшую часть наших ежедневных связей. Полезно сравнить эффективность методов кодирования через битовую скорость, требуемую для передачи сигнала речи. Предполагается, что сигнал речи ограничен полосой частот 200 ... 3200 Гц и стробируется с номинальной скоростью 8000 отсчетов/с для всех кодеров, исключая ДМ, для которой скорость стробирования равна битовой скорости.

В табл. 2.5 приведены средние характеристики методов кодирования, описанные выше.

Таблица 2.5

Метод кодирования	Квантователь	Кодер	Скорость передачи, бит/с
ИКМ	Линейный	12 бит	96000
ЛогИКМ	Логарифмический	7 ... 8 бит	56000 – 64000
ДИКМ	Логарифмический	4 ... 6 бит	32000 – 48000
АДИКМ	Адаптивный	3 ... 4 бит	24000 – 32000
ДМ	Двоичный	1 бит	32000 – 64000
АДМ	Адаптивный двоичный	1 бит	16000 – 32000
ЛКП	–	–	2400 – 4800

В табл. 2.6 приведены сравнительные характеристики методов обработки речи, применяемые в кодеках.

Таблица 2.6

Тип кодека	Требуемая скорость передачи, кбит/с	Пороговая вероятность ошибки на бит (BER)	Субъективное качество	Потребление мощности	Параметр сложности
ИКМ	64	$10^{-4}$	Междугородная связь	Низкое	Простой
АДИКМ	10 ... 40	$10^{-3} \dots 10^{-4}$	Почти междугородная связь	Низкое	Простой
Адаптивная ДМ	10 ... 40	$10^{-2}$	Почти междугородная связь	Низкое	Простой
Вокодер	1 ... 15	$10^{-2}$	Хорошее	Высокое	Сложный
LPC	1 ... 15	$10^{-2}$	Хорошее	Высокое	Сложный

### ***Контрольные вопросы***

1. Типы и виды информации в СПС. Символьная информация и ее разновидности. Аудиоинформация, ее виды, характеристики и параметры.

2. Графическая и видеоинформация, характеристики и параметры; способы представления.

3. Устройства дискретизации и квантования сигнала. ЦАП и АЦП. Назначение и применение. ЦАП: параметры и характеристики; классификация.

4. АЦП. Параметры и характеристики. Классификация.
5. Импульсно-кодовая модуляция (*PCM*).
6. Дифференциальная импульсно-кодовая модуляция.
7. Дельта-модуляция.
8. Сигма-дельта-модуляция. Сигма-дельта-модулятор как перегруппированный дельта-модулятор.
9. Сигма-дельта-модулятор как устройство квантования с обратной связью по шуму. Сигма-дельта-цифро-аналоговый преобразователь.
10. Адаптивное предсказание – прямая адаптация.
11. Адаптивное предсказание – синтетическое / аналитическое кодирование.
12. Адаптивное предсказание – линейное кодирование.
13. Вокодеры. Анализ и синтез речевого сигнала в вокодере.
14. Речевой кодек *ADPCM (G.726)*.
15. Широкополосный кодек *QFM-ADPCM (64 кбит/с) (G.722)*.
16. Кодеки *CELP (VSELP)* и *RELP*. Их особенности.
17. Кодер речи *ADPCM DECT*.
18. Декодер речи *ADPCM DECT*.
19. Кодер *RPE-LTP GSM*.
20. Декодер *RPE-LTP GSM*.

## 3. КОДИРОВАНИЕ КАНАЛА

### 3.1. Назначение канального кодирования

Канальное кодирование представляет собой класс преобразований сигнала, выполняемых для повышения качества связи. В результате сигнал становится менее уязвимым к таким эффектам ухудшения качества передачи, как шум, помехи и замирания. Канальное кодирование можно считать способом приведения параметров системы к желаемому компромиссу (т.е. соотношению между достоверностью передачи и шириной полосы пропускания или мощностью и шириной полосы пропускания).

Кодер канала следует после кодера речи и предшествует модулятору. Главной его задачей является помехоустойчивое кодирование речевого сигнала, т. е. такое кодирование, которое позволяет обнаруживать и в значительной мере исправлять ошибки, возникающие при распространении сигналов по радиоканалу от передатчика к приемнику, например от передатчика *MS* к приемнику *BTS*, и наоборот.

Помехоустойчивое кодирование осуществляется за счет введения в состав передаваемого сигнала достаточно большого объема избыточной информации; при этом может реализоваться кодирование с упреждающей коррекцией ошибок — *FEC coding* (*Forward Error Correcting coding*).

Тему канального кодирования можно условно разделить на два раздела: кодирование сигнала и структурирование последовательности. Рассмотрим кодирование сигнала, которое означает преобразование сигнала в некий «улучшенный сигнал», позволяющий сделать процесс детектирования менее подверженным ошибкам.

Наиболее популярные коды сигнала называются ортогональными и биортогональными.

### 3.2. Ортогональные коды

Набор однобитовых данных можно преобразовать с помощью ортогональных кодовых слов, состоящих из двух разрядов каждое, которые описываются строками показанной ниже матрицы, называемой матрицей Адамара.

<u>Набор данных</u>	<u>Набор ортогональных кодовых слов</u>
0	$\mathbf{H}_1 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$
1	

Для кодирования набора двухбитовых данных упомянутый выше набор следует расширить по горизонтали и вертикали, что дает матрицу  $\mathbf{H}_2$ .

<u>Набор данных</u>	<u>Набор ортогональных кодовых слов</u>
0 0	$\mathbf{H}_2 = \begin{bmatrix} 0 & 0 & \vdots & 0 & 0 \\ 0 & 1 & \vdots & 0 & 1 \\ \dots & \dots & \vdots & \dots & \dots \\ 0 & 0 & \vdots & 1 & 1 \\ 0 & 1 & \vdots & 1 & 0 \end{bmatrix} = \begin{bmatrix} \mathbf{H}_1 & \mathbf{H}_1 \\ \mathbf{H}_1 & \mathbf{H}_1 \end{bmatrix}$
0 1	
1 0	
1 1	

Вообще, для набора  $k$ -битовых данных из матрицы  $\mathbf{H}_{k-1}$  можно построить набор кодовых слов  $\mathbf{H}_k$  с размерами  $2^k \times 2^k$ , который называется матрицей Адамара.

$$\mathbf{H}_k = \begin{bmatrix} \mathbf{H}_{k-1} & \mathbf{H}_{k-1} \\ \mathbf{H}_{k-1} & \mathbf{H}_{k-1} \end{bmatrix}.$$

Каждая пара слов в каждом наборе кодовых слов  $\mathbf{H}_1, \mathbf{H}_2, \mathbf{H}_3, \dots, \mathbf{H}_k, \dots$  содержит одинаковое количество совпадающих и несовпадающих разрядов. Поэтому и каждый из этих наборов ортогонален.

### 3.3. Биортогональные коды

Биортогональный набор сигналов, состоящий из  $M$  сигналов или кодовых слов, получается из ортогонального набора, состоящего из  $M/2$  сигналов, путем дополнения последнего отрицанием каждого сигнала:

$$\mathbf{V}_k = \begin{bmatrix} \mathbf{H}_{k-1} \\ \mathbf{H}_{k-1} \end{bmatrix}.$$

Например, набор 3-битовых данных можно преобразовать в биортогональный набор кодовых слов следующим образом:

<u>Набор данных</u>	<u>Набор ортогональных кодовых слов</u>
0 0 0	$\mathbf{V}_3 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ \dots & \dots & \dots & \dots \\ 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix}$
0 0 1	
0 1 0	
0 1 1	
1 0 0	
1 0 1	
1 1 0	
1 1 1	

В действительности биортогональный набор состоит из двух ортогональных кодов, таких, что для каждого кодового слова в одном наборе имеется антиподное ему слово в другом. Биортогональный набор состоит из комбинации ортогональных и антиподных сигналов. Одно из преимуществ биортогональных кодов перед ортогональными заключается в том, что при передаче аналогичной информации размер кодового слова биортогональных кодов вдвое меньше размера кодового слова ортогональных кодов. Следовательно, при использовании биортогональных кодов требования к полосе пропускания вдвое слабее. Поскольку антиподные векторы сигналов имеют лучшие пространственные характеристики, чем ортогональные, очевидно, что биортогональные коды лучше ортогональных.

### 3.4. Структурированные последовательности

Метод структурированных последовательностей – это преобразование последовательности данных в новую, улучшенную последовательность, обладающую структурной избыточностью

(которая вмещает избыточные биты). Эти избыточные разряды служат для определения и исправления ошибок. На выходе процедуры кодирования получается закодированный (формой сигнала или структурированной последовательностью) сигнал, имеющий лучшие пространственные характеристики, чем некодированный.

Структурированные последовательности делятся на 3 категории: блочные (блоковые), сверточные (древовидные) и турбокоды.

### 3.5. Модели каналов

1. Дискретный канал без памяти характеризуется дискретным входным алфавитом, дискретным выходным алфавитом и набором условных вероятностей  $P(j|i)$  ( $1 \leq i \leq M$ ,  $1 \leq j \leq Q$ ), где  $i$  представляет  $M$ -арный входной символ модулятора;  $j$  –  $Q$ -арный выходной символ демодулятора, а  $P(j|i)$  – это вероятность приема символа  $j$  при переданном символе  $i$ . Каждый выходной символ канала зависит только от соответствующего входного символа, так что для данной входной последовательности  $U = u_1, u_2, u_3, \dots, u_m, \dots, u_N$  условную вероятность соответствующей выходной последовательности  $Z = z_1, z_2, z_3, \dots, z_m, \dots, z_N$  можно записать следующим образом:

$$P(Z|U) = \prod_{m=1}^N P(z_m|u_m).$$

Если же канал имеет память (т.е. в пакете данных имеются помехи или канал подвергается воздействию замирания), условную вероятность последовательности  $Z$  нужно выразить как совместную вероятность всех элементов последовательности.

2. Двоичный симметричный канал является частным случаем дискретного канала без памяти, входной и выходной алфавиты которого состоят из двоичных элементов 0 и 1. Условные вероятности имеют симметричный вид:  $P(0|1) = P(1|0) = p$ ;  $P(1|1) =$

$= P(0|0) = 1-p$ . Эти уравнения выражают вероятности перехода. Поскольку на выход демодулятора поступают дискретные элементы 0 и 1, говорят, что по отношению к каждому символу демодулятор принимает жесткое решение. Поскольку декодер работает на основе жестких решений, определяемых демодулятором, декодирование в двоичном симметричном канале называется также жестким декодированием.

3. Гауссов канал – канал с дискретным входным и непрерывным выходным алфавитами, лежащими в диапазоне  $(-\infty, \infty)$ . Этот канал добавляет шум ко всем передаваемым символам. Если на выходе демодулятора находится непрерывный алфавит или его квантованное приближение (с более чем двумя квантовыми уровнями), говорят, что демодулятор принимает мягкое решение. Если в системе используется кодирование, демодулятор подает такие квантовые кодовые символы на декодер. Поскольку декодер работает на основе мягких решений, определяемых демодулятором, декодирование в гауссовом канале называется мягким.

### 3.6. Блочное кодирование

Структурная схема на рис. 3.1 поясняет процесс передачи сообщений в системе с блочным кодированием. Источник двоичной информации вырабатывает последовательность символов сообщения со скоростью  $R$  символов/с. Эти символы группируются в блоки длиной  $k$  символов. В каждый блок добавляется  $(n - k)$  избыточных символов и образуется кодовое слово  $(n, k)$  избыточного блочного кода. Эти избыточные символы иногда называют проверочными. Так как каждое слово, содержащее  $k$  символов, переносит только  $k$  бит информации, то скорость передачи информации на выходе кодера равна  $k/n$  бит/симв. Величина  $k/n$  называется кодовой скоростью, или скоростью кодирования. Таким образом, в кодере осуществляется преобразование двоичной информационной последовательности (101101...), содержащей  $k$

бит  $X = (x_1, x_2, \dots, x_k)$ , в двоичное кодовое слово длиной  $n$  бит  $Y = (y_1, y_2, \dots, y_n)$  путем добавления соответствующим образом сформированных проверочных символов. В декодере осуществляется обратная операция – по принятой последовательности  $Z = (z_1, z_2, \dots, z_n)$  определяется наиболее вероятное переданное кодовое слово.

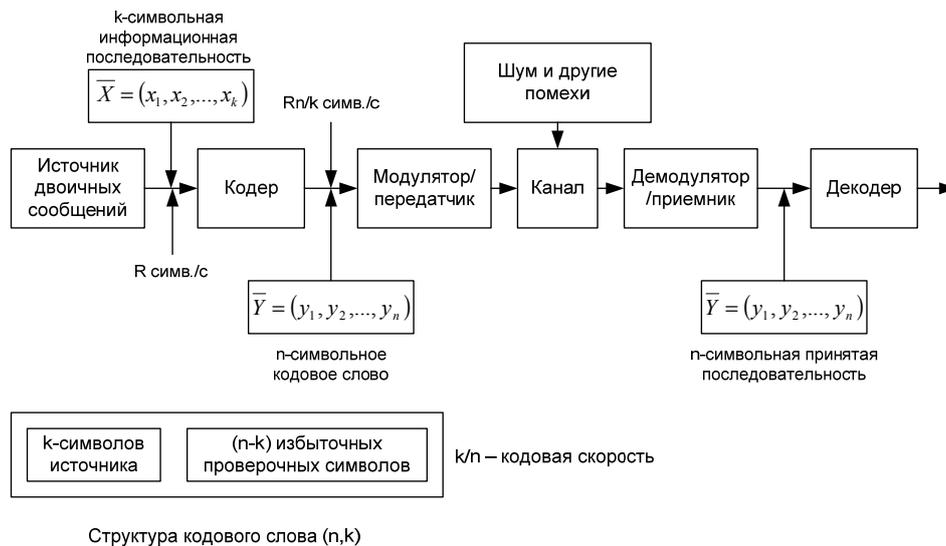


Рис. 3.1. Структурная схема системы с блочным кодированием

Если все переданные слова равновероятны, а канал связи не имеет памяти, то в качестве наиболее вероятного переданного слова выбирается то, которое ближе всего в смысле хеммингова расстояния находится к принятому кодовому слову. Расстояние Хемминга между последовательностями  $Y$  и  $Z$  оценивается как вес (число двоичных единиц) слова, образованного путем посимвольного сложения по модулю 2 последовательностей  $Y$  и  $Z$ .

### Коды с повторением

Наиболее простой блочный код вида  $(n, 1)$ , который позволяет понять концепцию исправления ошибок и оценить характеристики кода, известен как код повторений. В нем значение информационного символа повторяется  $(n-1)$  раз, т.е.  $(n-1)$  проверочных символов являются повторением информационного. Ко-

довая скорость  $1/n$  при достаточно больших  $n$  оказывается крайне низкой. Минимальное расстояние кода равно  $n$ , и при достаточно больших  $n$  коды повторений обладают высокой исправляющей способностью. Поскольку минимальное расстояние равно  $n$ , то кратность исправляемых ошибок в кодовом слове будет составлять  $e=(n-1)/2$ .

### 3.7. Коды с контролем четности

#### Код с одним контрольным битом

Коды с контролем четности для обнаружения или исправления ошибок используют линейные суммы информационных битов, которые называют символами, или битами четности. Код с одним контрольным битом – это прибавление к блоку информационных битов одного контрольного бита. Этот бит может быть равен 0 или 1, причем его значение выбирается так, чтобы сумма всех битов в кодовом слове была четной или нечетной. В операции суммирования используется арифметика по модулю 2 (*XOR*). Если бит четности выбран так, что результат четный, то говорят, что схема имеет положительную четность; если при добавлении бита четности результирующий блок данных является нечетным, то говорят, что он имеет отрицательную четность. В приемном оконечном устройстве производится декодирование, заключающееся в проверке, дают ли нуль суммы принятых битов кодового слова по модулю 2 (положительная четность). Если полученный результат равен 1, то кодовое слово заведомо содержит ошибки. Степень кодирования такого кода можно записать как  $k/(k+1)$ .

#### Прямоугольный код

Прямоугольный код, называемый также композиционным, можно представить в виде параллельной структуры кода, изображенной на рис. 3.2. Код создается следующим образом. Вначале из битов сообщения строятся прямоугольники, состоящие из  $M$  строк и  $N$  столбцов; затем к каждой строке и каждому столбцу

прибавляется бит четности, что в результате дает матрицу с размерами  $(M+1) \times (N+1)$ . Степень кодирования прямоугольного кода может быть записана следующим образом:

$$k/n = M \times N / (M+1) / (N+1).$$

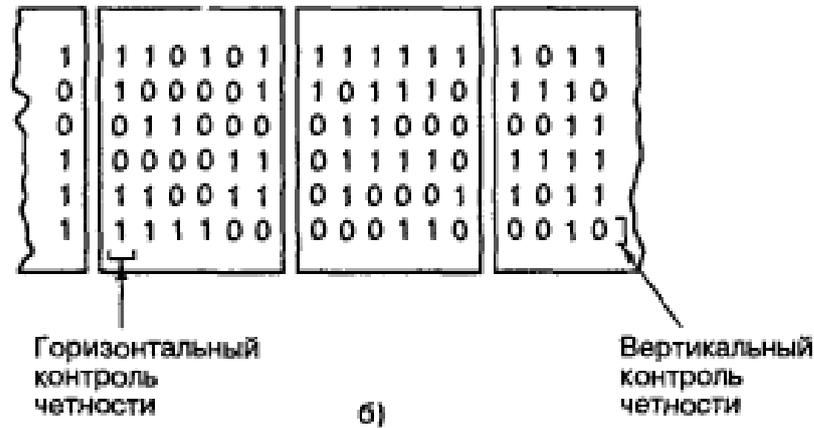


Рис. 3.2. Прямоугольный композиционный код

Любая отдельная ошибка в разряде приведет к нарушению четности в одном из столбцов и в одной из строк матрицы. Следовательно, прямоугольный код может исправить любую единичную ошибку, поскольку расположение такой ошибки однозначно определяется пересечением строки и столбца, в которых была нарушена четность. В примере, показанном на рис. 3.2, размеры информационной матрицы равны  $5 \times 5$ . Следовательно, на рисунке отображен код  $(36,25)$ , способный исправлять единичные ошибки, расположенные в любом из 36 двоичных разрядов. Вычислим для такого блочного кода с коррекцией ошибок вероятность появления неисправленной ошибки, для чего учтем все способы появления ошибки сообщения. Исходя из вероятности наличия  $j$  ошибок в блоке из  $n$  символов, можно записать вероятность ошибки сообщения, называемой также блочной ошибкой, или ошибочным словом, для кода, который может исправить модели ошибок, состоящие из  $t$  или менее ошибочных битов:

$$P_M = \sum_{j=i+1}^n \binom{n}{j} p^j (1-p)^{n-j}.$$

Здесь  $p$  – вероятность получения ошибочного канального символа. В примере код может исправить все однобитовые ошибки ( $t = 1$ ) в прямоугольном блочном коде, состоящем из  $n = 36$  бит. Следовательно суммирование в уравнении начинается с  $j = 2$ .

$$P_M = \sum_{j=2}^{36} \binom{36}{j} p^j (1-p)^{36-j}.$$

При малом  $p$  наибольший вклад дает первое слагаемое суммы. Следовательно, для примера с прямоугольным кодом (36,25) можно записать следующее:

$$P_M = \binom{36}{2} p^2 (1-p)^{34}.$$

Точная вероятность битовой ошибки  $P$  зависит от конкретного кода и используемого декодера.

### 3.8. Компромиссы, решаемые при кодировании

Кодирование с коррекцией ошибок можно рассматривать как инструмент, реализующий различные компромиссы системы:

1. Достоверность или полоса пропускания.
2. Мощность или полоса пропускания.
3. Скорость передачи или полоса пропускания.
4. Пропускная способность или ширина полосы пропускания.

В каждом из упомянутых компромиссов предполагались использование традиционного кода с избыточными битами и более быстрая передача сигналов (для систем связи реального времени); следовательно, в каждом случае платой было расширение полосы передачи. В то же время существуют методы коррекции ошибок, называемые решетчатым кодированием, которые не требуют увеличения скорости передачи сигналов или расширения полосы частот для систем связи реального времени.

### 3.9. Линейные блочные коды

Линейные блочные коды – это класс кодов с контролем четности, которые можно описать парой чисел  $(n, k)$ . В процессе кодирования блок из  $k$  символов сообщения (вектор сообщения) преобразуется в больший блок из  $n$  символов кодового слова (кодовый вектор), образованного с использованием элементов данного алфавита. Если алфавит состоит только из двух элементов (0 и 1), код является двоичным и включает двоичные разряды (биты). Сообщения длиной  $k$  бит формируют набор из  $2^k$  последовательностей сообщения, называемых  $k$ -кортежами;  $n$ -битовые блоки могут формировать  $2^n$  последовательности, также именуемые  $n$ -кортежами. Процедура кодирования сопоставляет с каждым из  $2^k$   $k$ -кортежей один из  $2^n$   $n$ -кортежей. Блочные коды представляют взаимно однозначное соответствие, в силу чего  $2^k$   $k$ -кортежей сообщения однозначно отображаются в множество из  $2^n$   $n$ -кортежей кодовых слов; отображение производится согласно таблице соответствия. Для линейных кодов преобразование отображения является линейным.

Множество всех двоичных  $n$ -кортежей ( $V_n$ ) называется векторным пространством на двоичном поле двух элементов (0 и 1). В двоичном поле определены две операции – сложение и умножение, причем результат этих операций принадлежит этому же множеству элементов. Арифметические операции сложения и умножения определяются согласно обычным правилам для алгебраического поля.

Подмножество  $S$  векторного пространства  $V_n$  называется подпространством, если для него выполняются следующие условия:

1. Множеству  $S$  принадлежит нулевой вектор.
2. Сумма любых двух векторов в  $S$  также принадлежит  $S$  (свойство замкнутости).

Рассмотрим пример линейного блочного кода (6,3). Нетрудно убедиться, что восемь кодовых слов в табл. 3.1 образуют  $V_6$ -подпространство (есть нулевой вектор, сумма любых двух кодовых слов дает кодовое слово этого же подпространства).

Таблица 3.1

Вектор сообщения	Кодовое слово
000	000000
100	110100
010	011010
110	101110
001	101001
101	011101
011	110011
111	000111

### Матрица генерации

При больших  $k$  реализация таблицы соответствия кодера становится слишком громоздкой. Для кода (172,92) существует  $2^{92}$  или приблизительно  $5 \cdot 10^{27}$  кодовых векторов. В таких случаях применяют матрицы генерации.

Поскольку множество кодовых слов, составляющих линейный блочный код, является  $k$ -мерным подпространством  $n$ -мерного двоичного векторного пространства ( $k < n$ ), всегда можно найти такое множество  $n$ -кортежей, которое может генерировать все  $2^k$  кодовых слов подпространства. О генерирующем множестве векторов говорят, что оно охватывает подпространство. Наименьшее линейно независимое множество, охватывающее подпространство, называется базисом подпространства, а число векторов в этом базисном множестве является размерностью подпространства. Любое базисное множество  $k$  линейно независимых  $n$ -кортежей  $V_1, V_2, \dots, V_i$  можно использовать для генерации нужных векторов линейного блочного кода, поскольку каждый

вектор кода является линейной комбинацией  $V_1, V_2, \dots, V_i$ . Иными словами, каждое из множества кодовых слов можно представить следующим образом:  $U = m_1V_1 + m_2V_2 + \dots + m_kV_k$ .

Вообще, матрицу генератора можно определить как массив с размерами  $k \times m$ :

$$G = \begin{bmatrix} V_1 \\ V_2 \\ \vdots \\ V_{k1} \end{bmatrix} = \begin{bmatrix} v_{11} & v_{12} & \cdots & v_{1m} \\ v_{21} & v_{22} & \cdots & v_{2m} \\ \vdots & & \cdots & \\ v_{k1} & v_{k2} & \cdots & v_{km} \end{bmatrix}$$

Для примера, рассмотренного выше, матрица имеет следующий вид:

$$G = \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

### 3.10. Систематические линейные блочные коды

Систематический линейный блочный код  $(n, k)$  – это такое отображение  $k$ -мерного вектора сообщения в  $n$ -мерное кодовое слово, в котором часть генерируемой последовательности совмещается с  $k$  символами сообщения. Остальные  $(n-k)$  бит – это биты четности. Матрица генератора систематического линейного блочного кода имеет следующий вид:

$$G = \begin{bmatrix} & \vdots & \\ P & & I_k \\ & \vdots & \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1(n-k)} & 1 & 0 & \cdots & 0 \\ p_{21} & p_{22} & \cdots & p_{2(n-k)} & 0 & 1 & \cdots & 0 \\ \vdots & p_{11} & & & & & \vdots & \\ p_{k1} & p_{k2} & \cdots & p_{k(n-k)} & 0 & 0 & \cdots & 1 \end{bmatrix}$$

Здесь  $P$  – массив четности, входящий в матрицу генератора,  $p_{ij} = (0,1)$ , а  $I_k$  – единичная матрица с размерами  $k \times k$ .

Можно представить каждое кодовое слово в следующем виде:

$$u_1, u_2, \dots, u_n = [m_1, m_2, \dots, m_k] \times \begin{bmatrix} p_{11} & p_{12} & \dots & p_{1(n-k)} & 1 & 0 & \dots & 0 \\ p_{21} & p_{22} & \dots & p_{2(n-k)} & 0 & 1 & \dots & 0 \\ \vdots & p_{11} & & & & & \ddots & \\ p_{k1} & p_{k2} & \dots & p_{k(n-k)} & 0 & 0 & \dots & 1 \end{bmatrix},$$

где

$$u_i = \begin{cases} m_1 p_{1i} + m_2 p_{2i} + \dots + m_k p_{ki} & \text{для } i = 1, \dots, (n-k), \\ m_{i-n+k} & \text{для } i = (n-k+1), \dots, n. \end{cases}$$

Для данного  $k$ -кортежа сообщения  $m = m_1, m_2, \dots, m_k$  и  $k$ -кортежа кодовых векторов  $U = u_1, u_2, \dots, u_k$  систематический кодовый вектор можно записать в следующем виде:

$$U = p_1, p_2, \dots, p_{n-k}, m_1, m_2, \dots, m_k,$$

где  $p_1 = m_1 p_{11} + m_2 p_{21} + \dots + m_k p_{k1}$ ,

$$p_2 = m_1 p_{12} + m_2 p_{22} + \dots + m_k p_{k2},$$

.....

$$p_{n-k} = m_1 p_{1(n-k)} + m_2 p_{2(n-k)} + \dots + m_k p_{k(n-k)}.$$

Для кода (6,3) кодовое слово выглядит следующим образом:

$$\begin{aligned} U &= [m_1, m_2, m_3] \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} = \\ &= \underbrace{m_1 + m_3}_{u_1}, \underbrace{m_1 + m_2}_{u_2}, \underbrace{m_2 + m_3}_{u_3}, \underbrace{m_1}_{u_4}, \underbrace{m_2}_{u_5}, \underbrace{m_3}_{u_6}. \end{aligned}$$

## Проверочная матрица

Определим матрицу  $H$ , именуемую проверочной, которая позволит декодировать полученные векторы. Для каждой матрицы  $(k \times n)$  генератора  $G$  существует матрица  $H$  с размерами  $(n-k) \times n$  такая, что строки матрицы  $G$  ортогональны к строкам матрицы  $H$ . Иными словами,  $GH^T = 0$ , где  $H^T$  – транспонированная матрица  $H$ , а  $0$  – нулевая матрица с размерами  $k \times (n-k)$ . Чтобы матрица  $H$

удовлетворяла требованиям ортогональности систематического кода, ее компоненты записываются в следующем виде:

$$H = \begin{bmatrix} I_{n-k} & \vdots & P^T \end{bmatrix}$$

Следовательно, матрица  $H^T$  имеет следующий вид:

$$H^T = \begin{bmatrix} I_{n-k} \\ \dots \\ P^T \end{bmatrix} = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & & & \\ 0 & 0 & \dots & 1 \\ p_{11} & p_{12} & \dots & p_{1(n-k)} \\ p_{12} & p_{22} & \dots & p_{2(n-k)} \\ \vdots & & & \\ p_{k1} & p_{k2} & \dots & p_{k(n-k)} \end{bmatrix}$$

Матрица позволяет проверять принятые векторы на предмет их принадлежности заданному набору кодовых слов.  $U$  будет кодовым словом, генерируемым матрицей  $G$ , тогда и только тогда, когда  $UH^T = 0$ .

### Контроль с помощью синдромов

Пусть  $r = r_1, r_2, \dots, r_n$  – принятый вектор, полученный после передачи  $U = u_1, u_2, \dots, u_n$ . Тогда  $r$  можно представить в следующем виде:  $r = U + e$ .

Здесь  $e = e_1, e_2, \dots, e_n$  – вектор ошибки, или модель ошибки, внесенная каналом. Всего в пространстве из  $2^n$   $n$ -кортежей существует  $2^n - 1$  возможных ненулевых моделей ошибки. Синдром сигнала  $r$  определяется следующим образом:

$$S = rH^T.$$

Синдром – это результат проверки четности, выполняемой над сигналом  $r$  для определения его принадлежности заданному набору кодовых слов. При положительном результате проверки синдром равен 0. Если  $r$  содержит ошибки, которые можно исправить, то синдром имеет определенное ненулевое значение, что позволяет отметить конкретную модель ошибки.

Пусть передано кодовое слово  $U = 101110$  и принят вектор  $r = 001110$ , т.е. крайний левый бит принят с ошибкой. Найдем вектор синдрома  $S = rH^T$  и покажем, что он равен  $eH^T$ .

$$S = rH^T = [001110] \times \begin{bmatrix} 100 \\ 010 \\ 001 \\ 110 \\ 011 \\ 101 \end{bmatrix} = [1, 1+1, 1+1] = [101]$$

Далее проверим, что синдром искаженного вектора кода равен синдрому модели ошибки, которая вызвала эту ошибку.

$$S = eH^T = [100000]H^T = [100] \text{ (синдром модели ошибки).}$$

### Декодирование с исправлением ошибок

Процедура декодирования с исправлением ошибок состоит из следующих этапов:

1. С помощью уравнения  $S = rH^T$  вычисляется синдром для  $r$ .
2. Определяются образующие элементы класса смежности (модели ошибки)  $e_j$ , синдром которых равен  $rH^T$ .
3. Полагается, что модели ошибки вызываются искажениями в канале.
4. Полученный исправленный вектор, или кодовое слово, определяется как  $U = r + e_j$ . Можно сказать, что в результате вычитания определенных ошибок мы восстановили верное кодовое слово. (В арифметических операциях по модулю 2 операция вычитания равносильна операции сложения).

Рассмотрим схему декодера (рис. 3.3), где реализованы логические элементы исключающего ИЛИ и И, которые позволяют получить результат для любой модели с одним ошибочным битом в коде (6,3). Исходное слово  $U=101110$ , принято слово с ошибкой  $r=001110$ , исправленный результат  $U$  равен исходному.

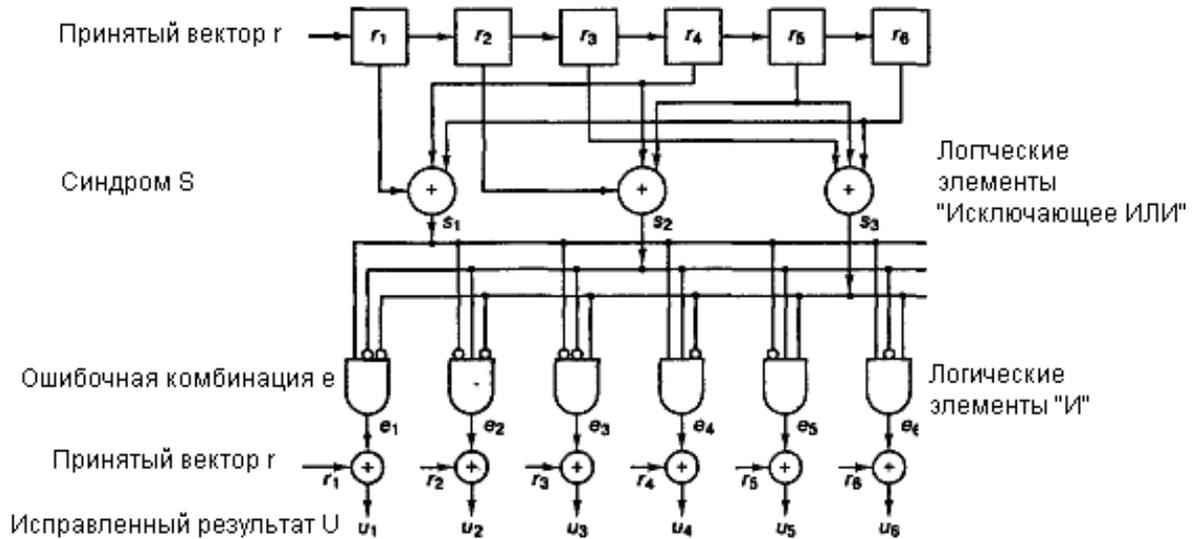


Рис. 3.3. Схема декодера для кода (6,3)

### Оценка возможностей кода

Для больших кодов нормальная матрица позволяет определить важные исходные характеристики, такие как возможные компромиссы между обнаружением и исправлением ошибок и пределы возможностей кода в коррекции ошибок. Одно из таких ограничений, называемое пределом Хэмминга, описывается следующим образом:

$$\text{Количество бит четности: } n - k \geq \log_2 \left[ 1 + \binom{n}{1} + \binom{n}{2} + \dots + \binom{n}{t} \right]$$

$$\text{Количество классов смежности: } 2^{n-k} \geq \left[ 1 + \binom{n}{1} + \binom{n}{2} + \dots + \binom{n}{t} \right].$$

Неравенство определяет нижнюю границу числа  $n-k$  бит четности как функцию возможностей кода в коррекции  $t$ -битовых ошибок.

Рассмотрим факторы, определяющие выбор конкретных значений  $(n, k)$ .

1. Для получения нетривиального соотношения между исправлением и обнаружением ошибок желательно, чтобы код имел

возможности коррекции ошибок по крайней мере с  $t = 2$ . Таким образом, минимальное расстояние  $d_{\min} = 2t + 1 = 5$ .

2. Чтобы кодовая система была нетривиальной, желательно, чтобы количество бит данных было не менее  $k = 2$ . Следовательно, число кодовых слов  $2^k = 4$ .

3. Для реальных прикладных кодов минимальное значение  $n$  выбирается по разным причинам – эффективность полосы пропускания и простота системы. Если при выборе  $n$  используется предел Хэмминга, то  $n$  следует выбирать равным 7. В то же время размерность полученного кода (7,2) не соответствует указанным выше требованиям  $d_{\min} = 2t + 1 = 5$ . Чтобы увидеть это, следует ввести другую верхнюю границу возможностей кода в коррекции  $t$ -битовых ошибок. Эта граница, называемая пределом Плоткина, определяется следующим образом:

$$d_{\min} \leq \frac{n \times 2^{k-1}}{2^k - 1}.$$

В общем случае, линейный код  $(n, k)$  должен удовлетворять всем перечисленным выше условиям, включая возможности коррекции ошибок (или минимальное расстояние).

### 3.11. Блочные циклические коды

Циклические коды – это подкласс линейных кодов, которые удовлетворяют следующим свойствам: если  $C = [c_{n-1} \ c_{n-2} \ \dots \ c_1 \ c_0]$  – кодовое слово циклического кода, тогда  $[c_{n-2} \ c_{n-3} \ \dots \ c_2 \ c_1 \ c_0 \ c_{n-1}]$ , полученное циклическим сдвигом элементов кода  $C$ , также является кодовым словом. Все циклические сдвиги  $C$  образуют кодовые слова. Как следствие циклического свойства, эти коды обладают значительным количеством структурных удобств, которые можно использовать при реализации операций кодирования. Большое количество алгоритмов эффективных кодеров было сделано посредством циклических кодов, что позволило в практических системах связи строить блочные коды большой длины с большим количеством кодовых слов.

При работе с циклическими кодами принято связывать с кодовым словом  $C = [c_{n-1} c_{n-2} \dots c_1 c_0]$  полином  $C(p)$  степени  $\leq n-1$ , определенный таким образом:  $C(p) = c_{n-1}p^{n-1} + c_{n-2}p^{n-2} + c_1p + c_0$ .

Для двоичного кода каждый из коэффициентов полинома является либо нулем, либо единицей. Теперь предположим, мы формируем полином

$$pC(p) = c_{n-1}p^n + c_{n-2}p^{n-1} + c_1p^2 + c_0p.$$

Этот полином не может представить кодовое слово, так как его степень может быть равна  $n$  (если  $c_{n-1} = 1$ ). Однако если мы разделим  $pC(p)$  на  $p^n + 1$ , мы получим

$$pC(p) / p^n + 1 = c_{n-1} + C_1(p) / p^n + 1,$$

где  $C_1(p) = c_{n-1}p^{n-1} + c_{n-3}p^{n-2} + \dots + c_0p + c_{n-1}$ .

Заметим, что полином  $C_1(p)$  представляет кодовое слово  $C_1 = [c_{n-2} c_{n-3} \dots c_2 c_{n-1}]$ , которое как раз образовано из кодового слова  $C$  циклическим сдвигом на одну позицию. Поскольку  $C_1(p)$  представляет собой остаток, полученный делением  $pC(p)$  на  $p^n + 1$ , мы говорим, что  $C_1(p) = pC(p) \bmod (p^n + 1)$ .

Аналогичным образом, если  $C(p)$  представляет кодовое слово в циклическом коде, тогда  $p^i C(p) \bmod (p^n + 1)$  также является кодовым словом циклического кода. Так что можно написать  $p^i C(p) = Q(p)(p^n + 1) + C_i(p)$ , где остаточный полином  $C_i(p)$  представляет кодовое слово циклического кода, а  $Q(p)$  – частное.

Мы можем генерировать циклический  $(n, k)$  код, используя порождающий полином  $g(p)$  степени  $n-k$  с двоичными коэффициентами, который является множителем при факторизации полинома  $p^n + 1$ . Порождающий полином в общем виде можно записать так:

$$g(p) = p^{n-k} + g_{n-k-1}p^{n-k-1} + g_1p + 1.$$

Мы также определяем полином информационного сообщения

$$X(p) = x_{k-1}p^{k-1} + x_{k-2}p^{k-2} + \dots + x_1p + x_0,$$

где  $[x_{k-1} x_{k-2} \dots x_1 x_0]$  определяет  $k$  информационных бит. Ясно, что произведение  $X(p) \cdot g(p)$  – это полином степени меньшей или рав-

ной  $n-1$ , который может представлять кодовое слово. Заметим, что имеется  $2^k$  полиномов  $\{X_i(p)\}$  и, следовательно,  $2^k$  возможных кодовых слов, которые можно формировать при заданном  $g(p)$ .

Из вышесказанного мы видим, что кодовые слова, обладающие циклическими свойствами, можно генерировать умножением  $2^k$  сообщений на уникальный полином степени  $n-k$   $g(p)$ , называемый порождающим полиномом циклического  $(n, k)$  кода, который является множителем при факторизации  $p^n + 1$ . Циклический код, генерируемый указанным образом, занимает пространство  $S_c$  векторного пространства  $S$ . Размерность подпространства  $S_c$  равна  $k$ .

Операции кодирования при создании циклических кодов можно выполнить при помощи линейных сдвигающих регистров с обратной связью, с использованием порождающего полинома.

Итак, генерирование систематического циклического кода включает три ступени, а именно:

- Умножение полинома сообщения  $X(p)$  на  $p^{n-k}$ .
- Деление этого произведения на  $g(p)$ .
- Прибавление остатка  $r(p)$  к  $p^{n-k} X(p)$ .

Из этих трех ступеней только деление является нетривиальным. Деление полинома  $A(p) = p^{n-k} X(p)$  степени  $n-1$  на полином  $g(p) = g_{n-k}p^{n-k} + c_{n-k-1}p^{n-k-1} + \dots + g_1p + g_0$  можно выполнить посредством  $(n-k)$  ячеек регистра сдвига с обратной связью, показанного на рис. 3.4.

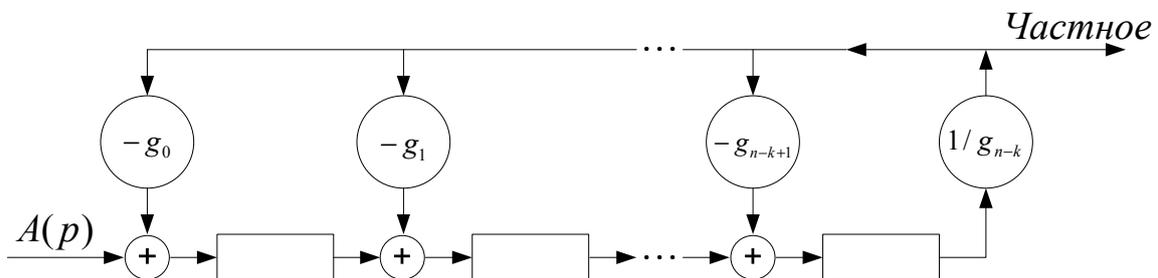


Рис. 3.4. Логическая схема для реализации деления полинома  $A(p)$  на  $g(p)$

Первоначально ячейки сдвига регистра содержат одни нули. Коэффициенты  $A(p)$  поступают и продвигаются по регистру сдвига по одному коэффициенту за такт, начиная с коэффициентов более высокого порядка, т.е. с  $a_{n-1}$ , затем  $a_{n-2}$  и так далее. После  $(n-k-1)$  сдвига, первый ненулевой выход частного равен  $q_1 = (g_{n-k})^{-1}a_{n-1}$ . Последующие выходы генерируются так, как показано на рис. 3.4. Для образования каждого выходного коэффициента линии мы должны вычесть полином  $g(p)$ , умноженный на этот коэффициент, как при обычном «длинном» делении. Это вычитание производится посредством обратной связи. Таким образом, регистр сдвига на рис. 3.4 обеспечивает деление двух полиномов.

В нашем случае  $g_{n-k} = g_0 = 1$ , и для двоичных кодов арифметические операции выполняются по mod 2. Следовательно, операция вычитания сводится к сложению по mod 2. Далее мы будем только интересоваться генерированием проверочных символов для каждого слова, поскольку код систематический. Как следствие, кодер циклического кода принимает вид, показанный на рис. 3.5.

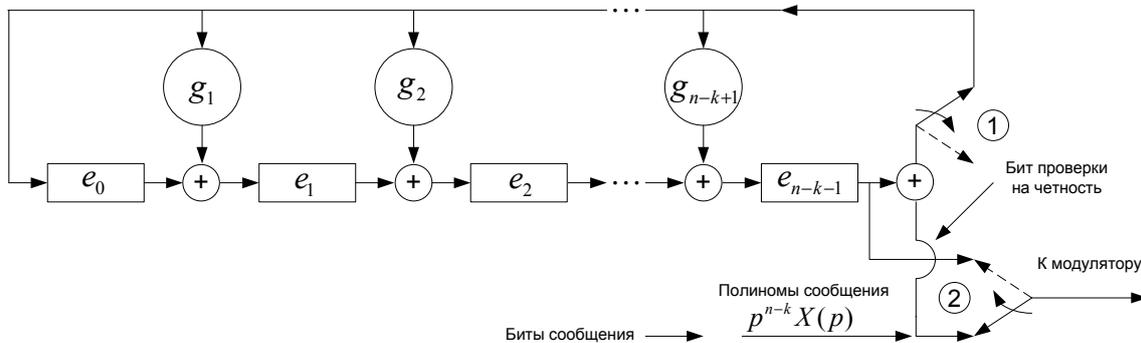


Рис. 3.5. Циклический кодер с использованием порождающего полинома  $g(p)$

Первые  $k$  бит на выходе кодера просто равны  $k$  информационным битам. Эти  $k$  бит одновременно поступают на регистр сдвига, поскольку ключ 1 замкнут. Заметим, что умножение полиномов  $p^{n-k}$  и  $X(p)$  явно не производится. После того как все  $k$

информационных битов попали на вход кодера (и к модулятору), положения двух ключей на рис. 3.5 меняются на обратные. Начиная с этого времени содержимое регистра сдвига просто дает  $n-k$  проверочных символов, которые соответствуют коэффициентам полинома остатка. Эти  $n-k$  битов последовательно отправляются на модулятор.

Структурная схема циклического кодера (7,4) на основе регистра сдвига для кодирования с порождающим полиномом  $g(p) = p^3 + p + 1$  показана на рис. 3.6.

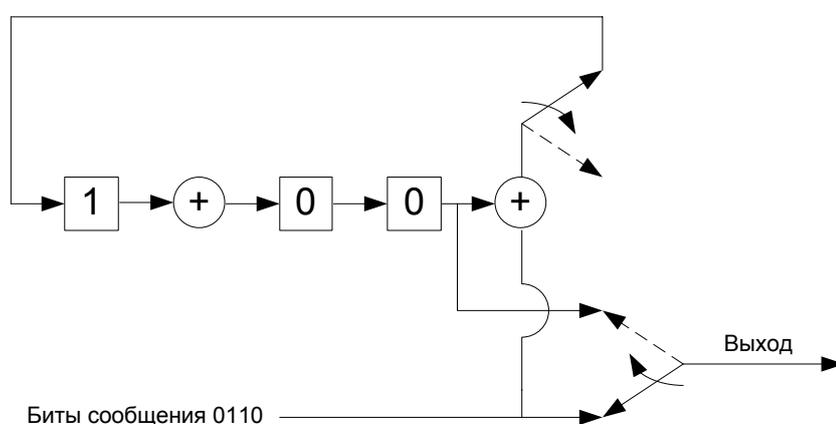


Рис. 3.6. Кодер циклического кода (7,4)

Предположим, что сообщением является цепочка 0110. Содержание сдвигового регистра отображается в табл. 3.2.

Таблица 3.2

Вход	Шаг сдвига	Содержимое регистра		
-	0	0	0	0
0	1	0	0	0
1	2	1	1	0
1	3	1	0	1
0	4	1	0	0

Таким образом, три проверочных символа равны 100.

Декодирование производится на основе вычисления синдрома (рис. 3.7, табл. 3.3). Если вектор синдрома нулевой, считается, что принятый вектор является правильным кодовым словом. Если синдром отличен от нуля, значит, обнаружена ошибка и принятый вектор – это искаженное кодовое слово; данная ошибка исправляется путем прибавления к принятому вектору вектора ошибки.

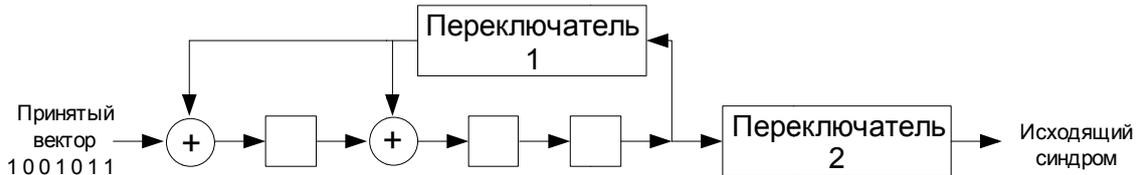


Рис. 3.7. Пример вычисления синдрома с помощью  $(n-k)$ -разрядного регистра сдвига

Таблица 3.3

Входная очередь	Номер сдвига	Содержимое регистра
1001011	0	00
100101	1	100
10010	2	110
1001	3	011
100	4	011
10	5	111
1	6	101
-	7	<b>000</b> Синдром

### 3.12. Варианты блочных кодов

В настоящее время предложено большое число блочных кодов, обладающих различными преимуществами перед обычными. Наиболее известны коды БХЧ, Хэмминга, Рида-Соломона, расширенный код Голея. Эти коды хорошо рассмотрены в [2, 3, 5].

### 3.13. Сверточное кодирование (СК)

Сверточный код описывается тремя целыми числами:  $n$ ,  $k$  и  $K$ . Отношение  $k/n$ , называемое степенью кодирования кода (*code rate*), является мерой добавления избыточности. Целое число  $K$

является параметром, называемым длиной кодового ограничения (*constraint length*); оно указывает число разрядов  $k$ -кортежа в кодирующем регистре сдвига. Важная особенность сверточных кодов состоит в том, что кодер имеет память;  $n$ -кортежи, получаемые при сверточном кодировании, являются функцией не только одного входного  $k$ -кортежа, но и предыдущих  $K-1$  входных  $k$ -кортежей. На практике  $n$  и  $k$  – это небольшие целые числа, а  $K$  изменяется с целью контроля мощности и сложности кода.

Исходное сообщение на входе обозначается последовательностью  $m = m_1, m_2, \dots, m_i, \dots$ , где  $m_i$  – двоичный знак (бит), а  $i$  – индекс времени. Будем предполагать, что все  $m_i$  равновероятно равны единице или нулю и независимы между собой. Будучи независимой, последовательность битов нуждается в некоторой избыточности, т.е. знание о бите  $m_i$  не дает никакой информации о бите  $m_j$  (при  $i \neq j$ ). Кодер (рис. 3.8) преобразует каждую последовательность  $m$  в уникальную последовательность кодовых слов  $U = G(m)$ . Несмотря на то что последовательность  $m$  однозначно определяет последовательность  $U$ , ключевой особенностью сверточных кодов является то, что данный  $k$ -кортеж внутри  $m$  не однозначно определяет связанные с ним  $n$ -кортежи внутри  $U$ , поскольку кодирование каждого из  $k$ -кортежей является функцией не только  $k$ -кортежей, но и предыдущих  $K-1$   $k$ -кортежей. Последовательность  $U$  можно разделить на последовательность кодовых слов:  $U = U_1, U_2, \dots, U_i, \dots$ . Каждое кодовое слово  $U_i$  состоит из двоичных кодовых символов, часто называемых канальными символами, канальными битами, или битами кода; в отличие от битов входного сообщения, кодовые символы не зависят от предыдущих.

В типичных системах связи последовательность кодовых слов  $U$  модулируется сигналом  $s(t)$ . В ходе передачи сигнал искажается шумом, в результате чего получают сигнал  $s'(t)$  и демодулированная последовательность  $Z = Z_1, Z_2, \dots, Z_j, \dots$ . Задача декодера состо-

ит в получении оценки  $m = m_1, m_2, \dots, m_i, \dots$  исходной последовательности сообщения с помощью полученной последовательности  $Z$  и априорных знаний о процедуре кодирования.

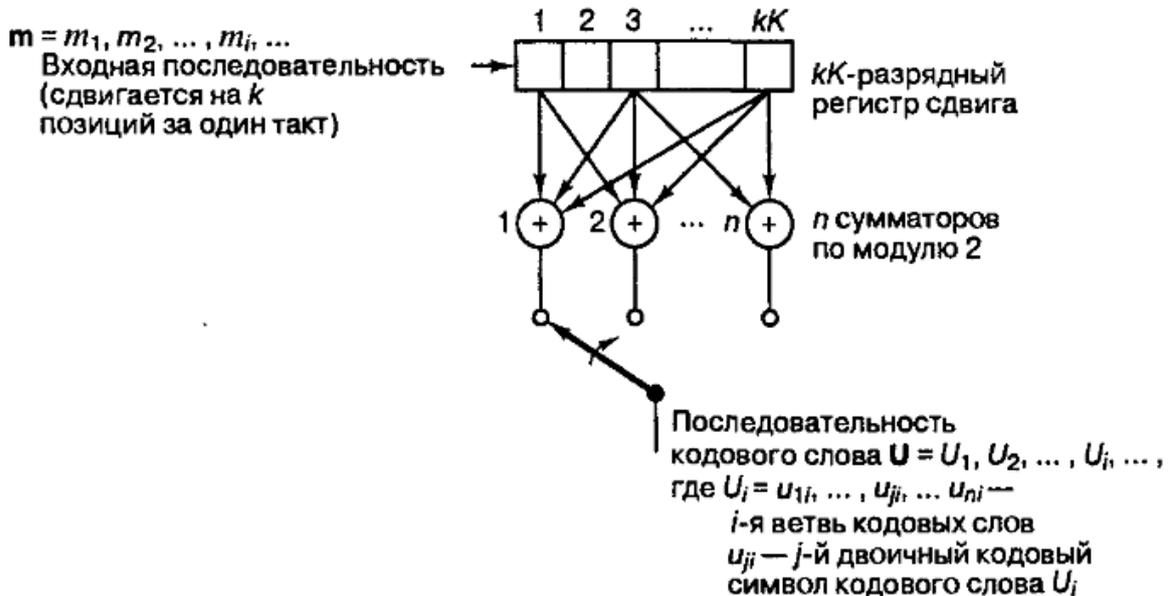


Рис. 3.8. Сверточный кодер с длиной кодового ограничения  $K$  и степенью кодирования  $k/n$

### Систематические и несистематические сверточные коды

Если в последовательности формируемых кодером кодовых символов можно отделить  $r = n - k$  избыточных символов от  $k$  информационных, то код называют систематическим. В систематическом кодере на  $k$  выходах будут информационные последовательности, на остальных  $n - k$  выходах — последовательности проверочных символов, формируемых как линейные комбинации информационных.

Первые  $k$  выходов систематического кодера соединены непосредственно с первыми разрядами регистров. Для кодеров с  $k = 1$  при формировании систематических кодов один из порождающих многочленов либо  $G_1(X) = 1$ , либо  $G_2(X) = 1$ , чтобы информационная последовательность была частью выходной последовательности. Систематические коды позволяют получить на приемной стороне оценку информационных символов, не производя декодирования или какой-либо иной обработки принимаемых символов.

В кодовых несистематических последовательностях информация непосредственно не содержится, они должны строиться так, чтобы при отсутствии ошибок ее можно было легко восстановить в декодере. Достоинством несистематических сверточных кодов является то, что их минимальное свободное расстояние выше, чем у систематических при прочих равных условиях. Это объясняется тем, что систематический сверточный код можно получить из соответствующего несистематического, исключив один из сумматоров по модулю 2, что и приводит к уменьшению минимального свободного расстояния кода.

Однако несистематические сверточные коды в отличие от систематических могут быть катастрофическими. Катастрофическим называется такой сверточный код, для которого конечное число ошибок, приобретенных в канале связи, может привести к бесконечному числу ошибок в декодированных символах.

### Способы представления сверточного кодера

Кодер СК содержит регистр памяти для хранения определенного числа информационных символов и преобразователь информационной последовательности в кодовую последовательность. Процесс кодирования производится непрерывно. Скорость кода  $R=k/n$ , где  $k$  - число информационных символов, одновременно поступающих на вход кодера;  $n$  - число соответствующих им символов на выходе кодера. Схемы простого кодера показаны на рис. 3.9.

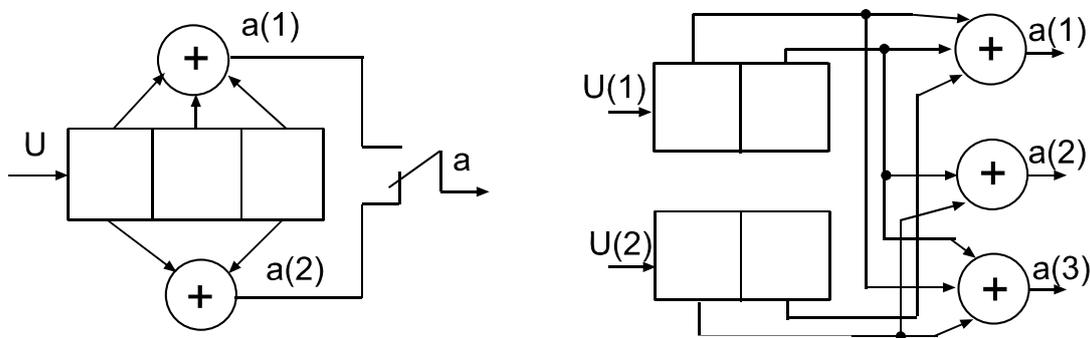
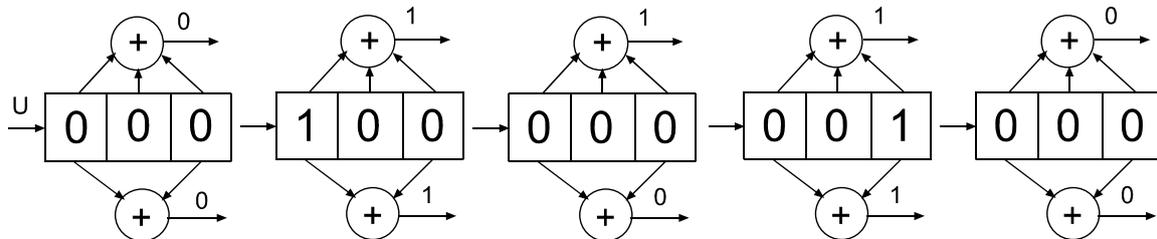


Рис. 3.9. Схемы простых сверточных кодеров

Символы  $a(1)$  и  $a(2)$  на его выходе образуют импульсный отклик  $h = 00111011\ 00\dots$ . Таким образом, если на входе кодера действует произвольная информационная последовательность, то последовательность на его выходе есть сумма по модулю 2 всех импульсных откликов, обусловленных действием смещенных во времени символов 1. Сверточный кодер как автомат с конечным числом состояний может быть описан диаграммой состояний. Диаграмма представляет собой направленный граф и описывает все возможные переходы кодера из одного состояния в другое, а также содержит символы выходов кодера, которые сопровождают эти переходы (см. ниже).



Решетчатая диаграмма является разверткой диаграммы состояний во времени. На решетке состояния показаны узлами, а переходы - соединяющими их линиями. После каждого перехода из одного состояния в другое происходит смещение на один шаг вправо. Решетчатая диаграмма дает наглядное представление всех разрешенных путей, по которым может продвигаться кодер при кодировании. Каждой информационной последовательности на входе кодера соответствует единственный путь по решетке. Построение решетки производится на основе диаграммы состояний. Исходное состояние  $S(1)S(2)=0$ . С поступлением очередного символа  $u=0$  либо 1 возможны переходы в состояния 00 либо 10, обозначаемые ветвями 00 и 11.

Для описания кодера последовательности символов на его входе и выходе представляют с использованием оператора задержки  $D$ :

$$U_{(i)}(D) = U_{(i)0}D^0 + U_{(i)1}D^1 + U_{(i)2}D^2 + \dots,$$

$$U_{(j)}(D) = U_{(j)0}D^0 + U_{(j)1}D^1 + U_{(j)2}D^2 + \dots$$

Здесь индексы в скобках обозначают:  $i$  - номер входа кодера,  $1 \leq i \leq n$ ;  $j$  - номер выхода кодера,  $1 \leq j \leq k$ . Индексы без скобок (0, 1, 2, ...) обозначают дискретные моменты времени.

$$G^{(j)}(D) = g_0^{(j)} + g_1^{(j)}D + g_2^{(j)}D^2 + \dots + g_v^{(j)}D^v.$$

Процесс кодирования может быть представлен как умножение многочлена входной информационной последовательности  $U(D)$  на порождающие многочлены кода  $G^{(j)}(D)$ , которые описывают связи ячеек регистра кодера с его выходами.

Порождающий многочлен представим в виде ряда

$$a^{(j)}(D) = u(D)G^{(j)}(D), 1 \leq j \leq n.$$

СК можно также задавать порождающей матрицей

$$G = \begin{vmatrix} G_0, G_1, G_2, \dots, G_{k-1} & \bullet & \bullet & \bullet \\ \bullet, G_0, G_1, G_2, \dots, G_{k-1} & \bullet & \bullet & \\ \bullet, \bullet, G_0, G_1, G_2, \dots, G_{k-1} & \bullet & & \\ \dots & \dots & \dots & \dots \end{vmatrix}$$

Порождающая матрица состоит из сдвигов базисной порождающей матрицы (верхняя строка матрицы  $G$ ), которая, в свою очередь, состоит из элементарных матриц  $G_i$ ,  $0 \leq i \leq k-1$ , содержащих  $k$  строк и  $n$  столбцов. Элементами этих матриц двоичных кодов являются символы 0 и 1.

Как и при использовании блоковых кодов, процесс кодирования может быть представлен в матричной форме:  $A = UG$ , где  $U$  - полубесконечная матрица входных информационных символов;  $A$  - полубесконечная матрица символов на выходе кодера.

Один из способов реализации кодера заключается в определении  $n$  векторов связи, по одному на каждый из  $n$  сумматоров по модулю 2. Каждый вектор имеет размерность  $K$  и описывает связь регистра сдвига кодера с соответствующим сумматором по модулю 2. Единица на  $i$ -й позиции вектора указывает на то, что соответствующий разряд в регистре сдвига связан с сумматором по модулю 2, а нуль в данной позиции указывает, что связи между разрядом и сумматором по модулю 2 не существует.

Для кодера на рис. 3.10 можно записать вектор связи  $g_1$  для верхних связей, а  $g_2$  — для нижних:  $g_1 = 1\ 1\ 1$ ;  $g_2 = 1\ 0\ 1$ .

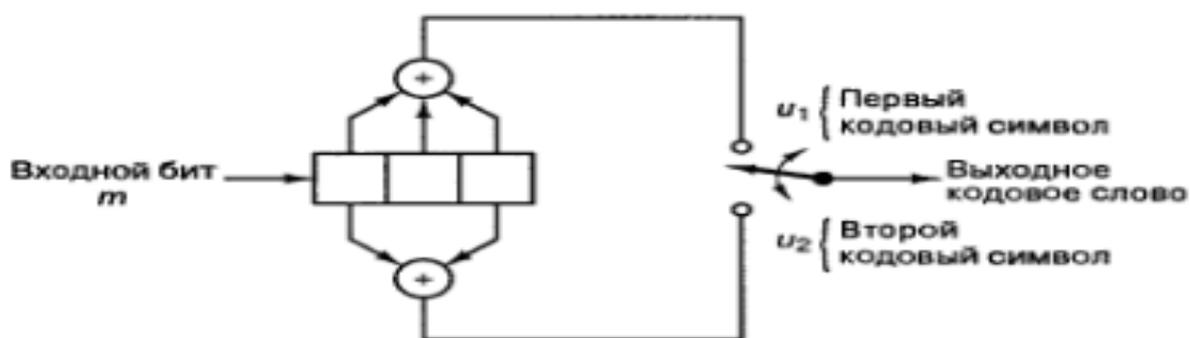


Рис. 3.10. Схема сверточного кодера при  $K=3$  и степени кодирования  $1/2$

В кодере (см. рис. 3.10) имеются 2 сумматора по модулю 2; следовательно, степень кодирования кода  $k/n$  равна  $1/2$ . При каждом поступлении бит смещается в крайний левый разряд, а биты регистра – на одну позицию вправо. Затем коммутатор на выходе дискретизирует выходы всех сумматоров по модулю 2 (т.е. сначала верхний сумматор, затем нижний), в результате формируются пары кодовых символов, образующих кодовое слово, связанное с только что поступившим битом. Это выполняется для каждого входного бита. Выбор связи между сумматорами и разрядами регистра влияет на характеристики кода. Всякое изменение в выборе связей приводит в результате к различным кодам. Связь выбирается и изменяется произвольным образом.

Предположим теперь, что вектор сообщения  $m = 101$  закодирован с использованием сверточного кода и кодера на рис. 3.10. Введены три бита сообщения, по одному в момент времени  $t_1$ ,  $t_2$  и  $t_3$ . Затем для очистки регистра в моменты времени  $t_4$  и  $t_5$  введены  $(K - 1)$ , равные двум нулям, что в результате приводит к смещению конечного участка на всю длину регистра. Последовательность на выходе (рис. 3.11) в моменты времени  $t_1 \dots t_5$  выглядит следующим образом: 11 10 00 10 11, где крайний левый символ представляет первую передачу. Для декодирования сообщения нужна полная последовательность на выходе (включающая кодовые символы). Для удаления сообщения из кодера требуется на единицу меньше нулей, чем имеется разрядов в регистре, или  $K - 1$  очищенных бит. В момент времени  $t_5$  регистр устанавливается в исходное состояние и можно передавать новое сообщение.

		Кодовое слово ветви			
		Содержимое регистра	$u_1$	$u_2$	
		100	1	1	
		010	1	0	
		001	1	1	
Входная последовательность		1 0 0			
Выходная последовательность		11 10 11			
Вход, $m$	Выход				
1	11	10	11		
0		00	00	00	
1			11	10	11
Сумма по модулю 2	11	10	00	10	11

Рис. 3.11. Сигналы кодера

### 3.14. Турбокоды

Схема каскадного кодирования впервые была предложена как метод получения высокоэффективного кода посредством комбинации двух или более компонентных кодов (иногда называемых составными). В результате такие коды могут корректиро-

вать ошибки в значительно более длинных кодах и имеют структуру, которая позволяет относительно легко осуществить декодирование средней сложности. Последовательные каскадные коды часто используются в системах с ограничением мощности, таких как космические зонды. Самая распространенная из этих схем содержит внешний код Рида-Соломона (выполняется первым, убирается последним), который следует за сверточным внутренним кодом (выполняется последним, убирается первым).

Турбокод можно считать обновлением структуры каскадного кодирования с итеративным алгоритмом декодирования связанной кодовой последовательности. Коды образуются посредством компоновки двух или более составных кодов, являющихся разными вариантами чередования одной и той же информационной последовательности. Тогда как для сверточных кодов на финальном этапе декодер выдает жестко декодированные биты (или в более общем случае — декодированные символы), в каскадной схеме, такой как турбокод, для хорошей работы алгоритм декодирования не должен ограничивать себя, подавая на декодеры жесткую схему решений. Для лучшего использования информации, получаемой с каждого декодера, алгоритм декодирования должен применять в первую очередь мягкую схему декодирования, вместо жесткой. Для систем с двумя составными кодами концепция, лежащая в основе турбодекодирования, заключается в том, чтобы передать мягкую схему принятия решений с выхода одного декодера на вход другого и повторять эту процедуру до тех пор, пока не будут получены надежные решения.

### **3.15. Схемы перемежения**

Перемежение (interleaving) представляет собой такое изменение порядка следования символов информационной последовательности, т. е. такую перестановку символов, при которой стоявшие рядом символы оказываются разделенными несколькими другими символами. Перемежение как процедура перестановки

символов предпринимается с целью преобразования групповых ошибок (пакетов ошибок) в одиночные ошибки, с которыми проще бороться с помощью блочного и сверточного кодирования.

### Диагональная схема перемежения

В диагональной схеме перемежения входная информация делится на блоки, а блоки — на субблоки, при этом в выходной последовательности субблоки, например, второй половины предыдущего блока, чередуются с субблоками первой половины следующего блока.

На рис. 3.12 показана схема диагонального перемежения, где каждый блок состоит из 6 субблоков  $a_i$ , второй — из субблоков  $b_i$ , третий — из субблоков  $c_i$ .

Каждый субблок может состоять либо из нескольких символов, либо из одного символа и даже из одного бита. Как видно из рис. 3.12, выходная последовательность субблоков  $b_i$  блока 2 перемежается с второй частью блока 3 и второй частью блока 1.

Приведенная схема диагонального перемежения вносит малую задержку, но расставляет соседние символы лишь через один, т. е. рассредоточение ошибочных символов группы получается сравнительно небольшим.

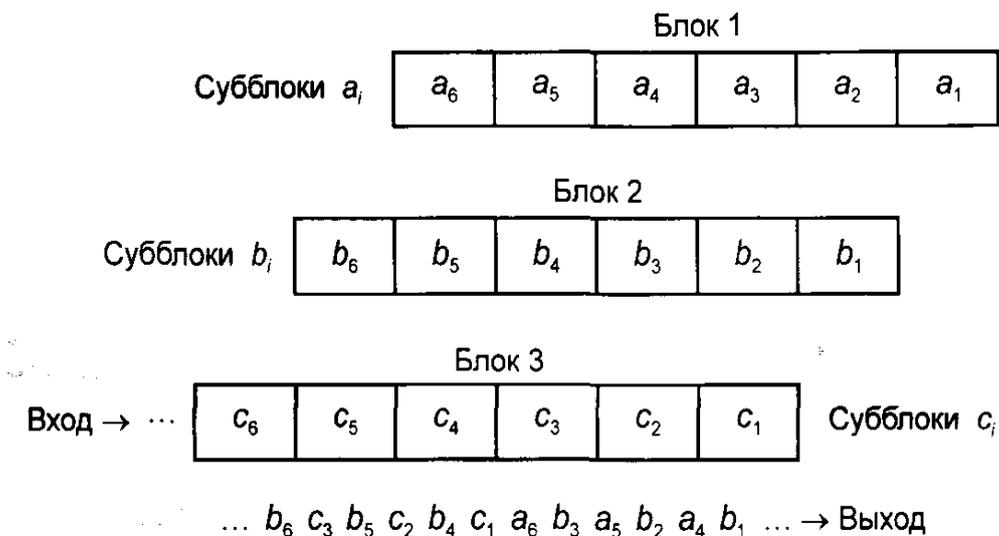


Рис. 3.12. Схема диагонального перемежения

### Блочная схема перемежения

При блочном перемежении входная информация также делится на блоки, по  $n$  субблоков (или символов) в каждом блоке, в выходной последовательности чередуются субблоки  $k_i$  последовательных блоков (рис. 3.13).

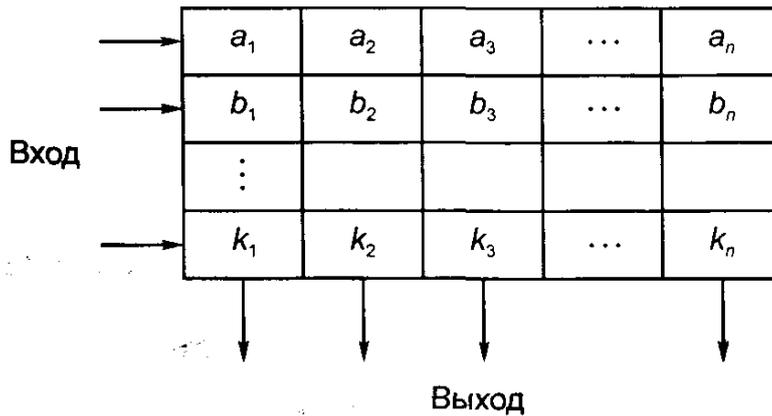


Рис. 3.13. Блочная схема перемежения

Работу этой схемы можно представить в виде:

- записи блоков входной последовательности в качестве строк матрицы с размерами  $k \times n$ ;
- считывания информации, которое производится по столбцам матрицы.

Таким образом, если входная последовательность в этом примере имеет вид:

- $a_1, a_2, \dots, a_n$
- $b_1, b_2, \dots, b_n$
- .....
- $k_1, k_2, \dots, k_n,$

то выходная последовательность будет:

- $a_1, b_2, \dots, k_n$
- $a_1, b_2, \dots, k_n$
- .....
- $a_1, b_2, \dots, k_n$

Субблоки, или символы, в частном случае также могут состоять из одного бита. Схема блочного перемежения вносит большую задержку, чем схема диагонального перемежения, но значительно сильнее рассредоточивает символы группы ошибок.

Общим недостатком обеих рассмотренных схем является жесткая периодичность следования переставленных символов в пределах интервала перемежения.

### 3.16. Структурная схема кодера канала стандарта GSM

В сотовых системах связи помехоустойчивое кодирование выполняется в виде трех процедур:

- блочного кодирования (block coding);
- сверточного кодирования (convolutional coding);
- перемежения (interleaving).

Кроме того, кодек канала выполняет еще ряд функций (рис. 3.14):

- добавляет управляющую информацию, которая, в свою очередь, подвергается помехоустойчивому кодированию;
- упаковывает подготовленную к передаче информацию и сжимает ее во времени;
- осуществляет шифрование передаваемой информации.

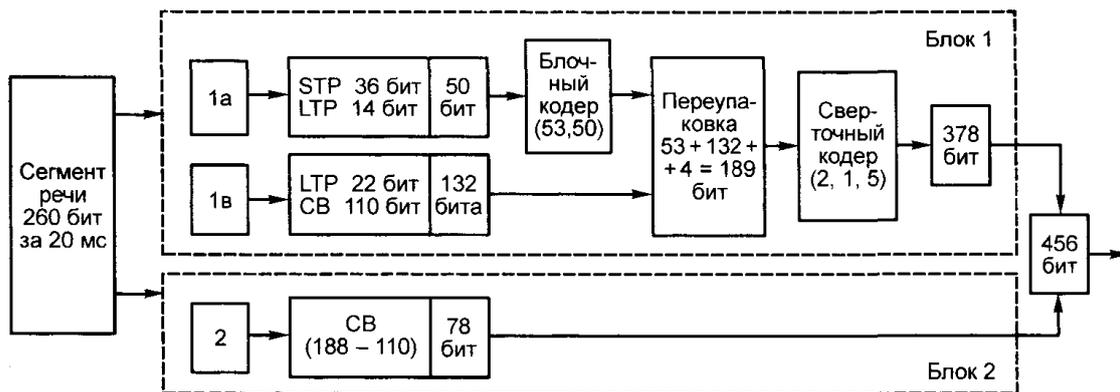


Рис. 3.14. Кодек речевого сигнала стандарта GSM

В кодеке канала кодируются/декодируются как речевая информация, так и информация каналов управления, т. е. информа-

ция каналов трафика *TCH* и каналов управления *CSN*. В то время как в информации канала трафика кодируется лишь часть битов, информация каналов управления кодируется в полном объеме.

Рассмотрим кодирование сегмента речевого сигнала, полученного на выходе кодера речи и имеющего параметр:

- фильтра *STP* — 36 бит;
- фильтра *LTP* — 36 бит;
- сигнала возбуждения (СВ) — 188 бит,

т. е. за 20 мс передаются в сегменте речевого сигнала 260 бит.

В кодере канала 260 бит информации разделяются на 2 класса: в класс 1 включено 182 бита, защищаемых помехоустойчивым кодированием; в класс 2 включены оставшиеся  $260 - 182 = 78$  бит, которые передаются без помехоустойчивого кодирования.

### Схема блочного диагонального перемещения в стандарте *GSM*

В стандарте *GSM* используется достаточно сложная и совершенная схема блочного диагонального перемещения (рис. 3.15).

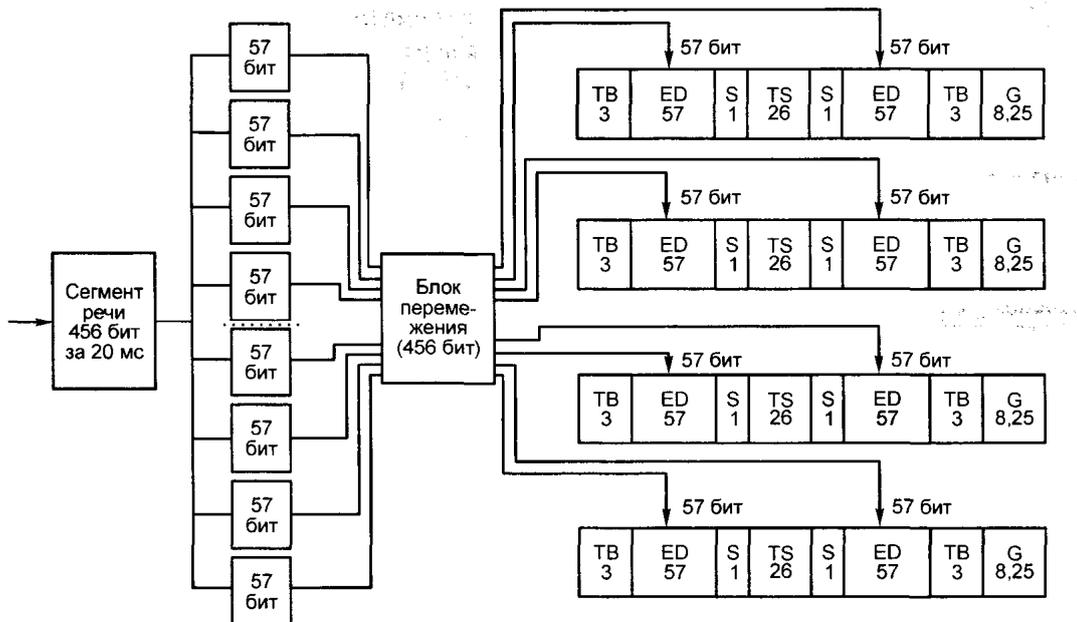


Рис. 3.15. Схема блочного диагонального перемещения

Полученные 456 бит информации одного 20-миллисекундного сегмента речи разбиваются на 8 подсегментов по 57 бит каждый ( $456/8 = 57$ ) (см. рис. 3.15). Далее в блоке перемежения реализуется алгоритм перемежения, обладающий свойствами квазислучайности. Смежные биты исходной последовательности разделяется непостоянным числом бит, при этом после перемежения 57 бит одного подсегмента распределяются между смежными восемью подсегментами таким образом, что смежными с каждым конкретным битом оказываются соответствующие ему по положению биты, отстоящие от него до перестановки на 4 подсегмента, причем на четные и нечетные (после перестановки) битовые позиции подсегмента ставятся биты из смежных сегментов.

Таблица перемежения для речевой последовательности бит имеет вид (табл. 3.4).

Таблица 3.4

Номер кадра	F0	F1	F2	F3	F4	F5	F6	F7	
	$a_0, b_4$	$a_1, b_5$	$a_2, b_6$	$a_3, b_7$	$a_4, b_0$	$a_5, b_1$	$a_6, b_2$	$a_7, b_3$	
Биты речевой последовательности	114	114	114	114	114	114	114	114	биты

После перемежения биты информации одного речевого сегмента распределяются по одноименным слотам (временным интервалам) четырех последовательных кадров канала трафика: два поля по 57 бит в слоте и каждое 57-битовое поле снабжается дополнительным скрытым флажком  $S$ , помечающим информацию речи (рис. 3.16).

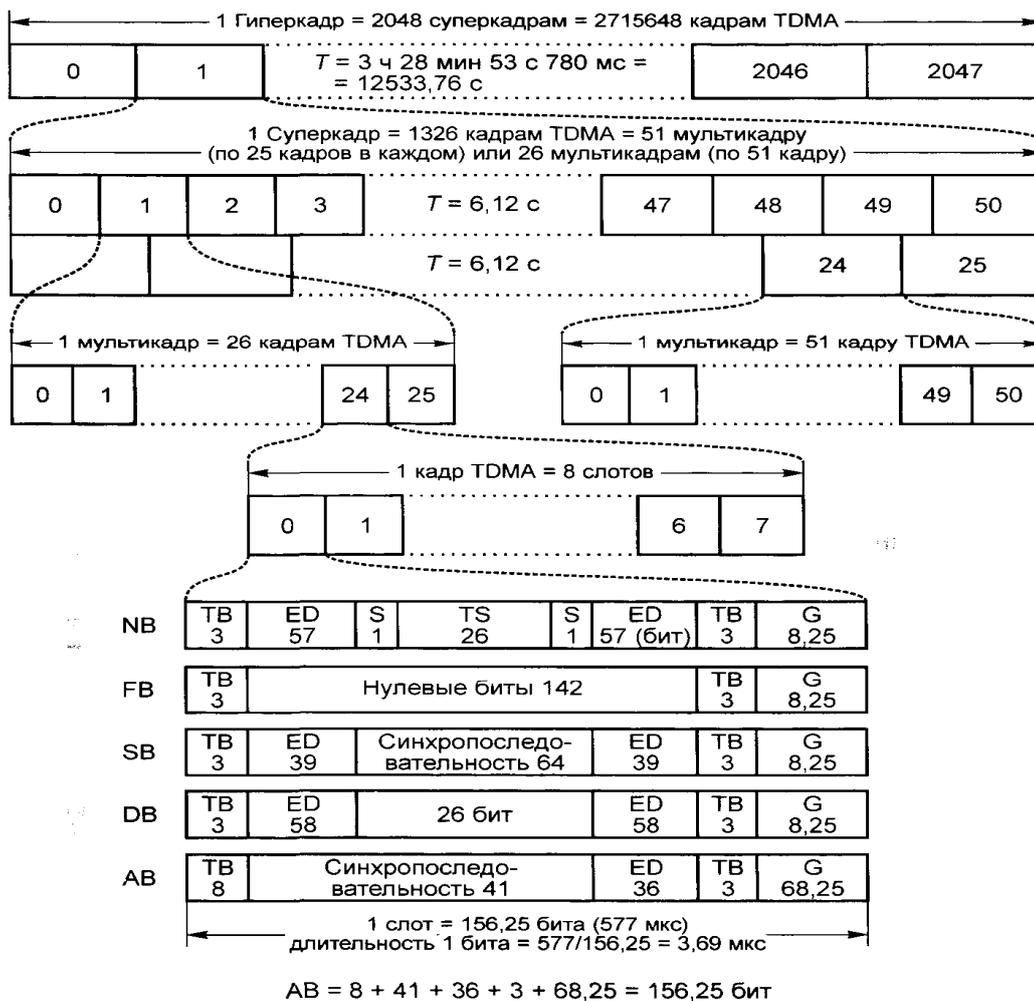


Рис 3.16. Структура кадра и мультикадра в стандарте GSM

### Процесс (алгоритм) декодирования в стандарте GSM:

1. Снимается перемежение, т.е. восстанавливается последовательность битов после шифрования информации.
2. Реализуется дешифрование информации.
3. Осуществляется разделение информационных потоков речи и управляющей информации.
4. Сверточное декодирование информации после эквалайзера, при этом исправляются ошибки в пределах возможностей кода свертки.
5. По коду четности проверяется наличие остаточных ошибок в информации подкласса 1A.
6. Реализуется блочное декодирование речевой информации.

7. Реализуется сверточное и блочное декодирование управляющей информации, которая далее поступает в центральный процессор (*CPU*).

### ***Контрольные вопросы***

1. Кодирование канала. Блочные симметричные линейные коды; коды с контролем четности.
2. Кодирование канала. Способы защиты от ошибок; виды кодов; компромиссы при кодировании с коррекцией ошибок.
3. Кодирование канала. Схема декодера для кода (6,3).
4. Кодирование канала. Циклические коды.
5. Кодирование канала. Коды Хемминга. Расширенный код Голея. Коды БХЧ.
6. Кодирование канала. Декодирование циклического кода (7,4).
7. Кодирование канала. Циклические коды: логическая схема для реализации полиномиального деления.
8. Кодирование канала. Циклические коды: систематическое кодирование с помощью регистра сдвига.
9. Кодирование канала. Циклические коды: обнаружение ошибок с помощью регистра сдвига.
10. Сверточное кодирование.
11. Систематические и несистематические сверточные коды.
12. Модели каналов при сверточном кодировании и методы декодирования при этом.
13. Способы представления сверточного кодера.
14. Схема сверточного кодера при  $K=3$  и степени кодирования  $1/2$ .
15. Диагональная схема перемежения.
16. Блочная схема перемежения.
17. Турбокоды.
18. Структурная схема кодера канала стандарта *GSM*.
19. Схема кодирования сегмента речевого сигнала в стандарте *GSM*.
20. Схема блочного диагонального перемежения в стандарте *GSM*.
21. Структура кадра и мультикадра в стандарте *GSM*.
22. Процесс (алгоритм) декодирования канала в стандарте *GSM*.

## 4. ИНТЕРФЕЙСЫ И ДОПОЛНИТЕЛЬНЫЕ УСТРОЙСТВА МОБИЛЬНЫХ ТЕРМИНАЛОВ

### 4.1. Интерфейс *USB*

*Universal Serial Bus* - универсальная последовательная шина. Является промышленным стандартом. Предназначена для подключения многих устройств, в том числе и мобильных терминалов (телефонов).

#### **Особенности с точки зрения пользователя:**

1. Простота кабельного соединения.
2. Скрытие подробностей электрического подключения от конечного пользователя.
3. Самоидентифицирующиеся периферийные устройства (ПУ), автоматическая связь устройств с драйверами и конфигурирование.
4. Возможность динамического подключения и конфигурирования ПУ.

#### **Скорости:**

*LS (Low Speed)* - низкая - 1,5 Мб/с (*USB1.0*).

*FS (Full Speed)* - полная - 12 Мб/с (*USB1.0*).

*HS (High Speed)* - высокая 480 Мб/с (*USB2.0*).

Структура *USB* и организация шины имеет хостцентрический вид.

Допускается 5 уровней хабов. Все подключенные устройства могут обслуживаться хостом одновременно. Такая структура называется многоярусной звездой: каждый кабельный сегмент соединяет 2 точки (хаб с хабом или хаб с функцией).

#### **Протоколы и уровни взаимодействия между хостом и ПУ**

Протокол включает 3 уровня взаимодействия (рис. 4.1):

1. Верхний – уровень функций. В хосте для уровня функций существует программная часть *Client SW*. Он взаимодействует с функцией *Function* (виртуальное взаимодействие на уровне данных).

*USB-system* взаимодействует виртуально с *USB Logical Device*.

2. Уровень устройств *USB*.

3. Уровень интерфейса шины.

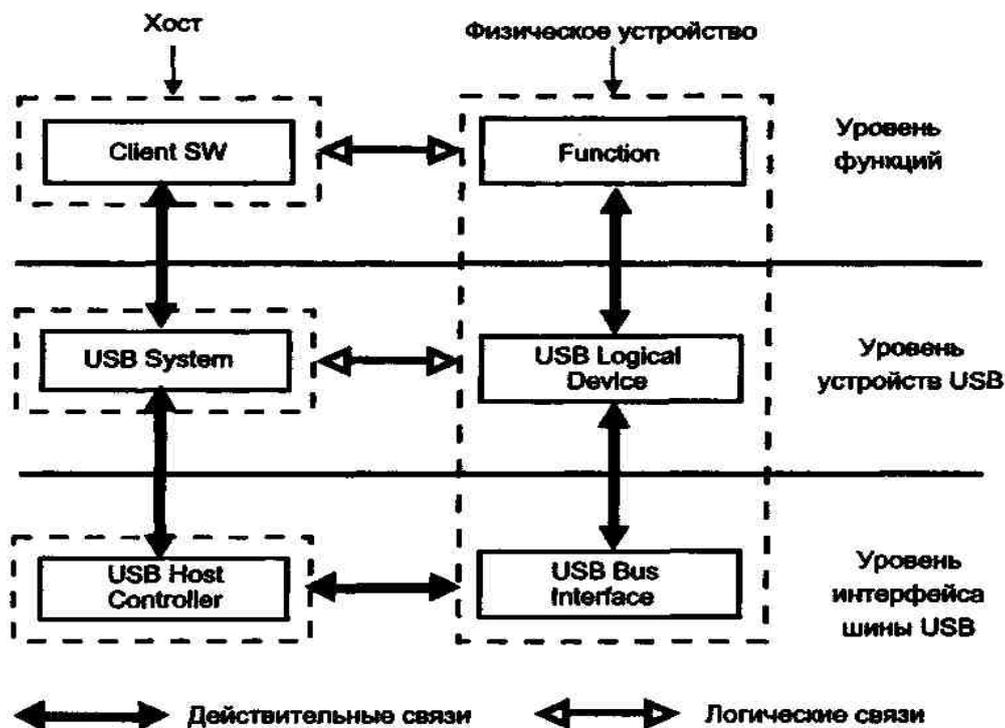


Рис. 4.1. Протоколы и уровни взаимодействия между хостом и ПУ

### Элементы структуры:

Физическое устройство *USB* – устройство на шине, выполняющее функции интерфейса конечного пользователя.

*Client SW* – ПО, соответствующее конкретному устройству, исполняемое на хосте. Может являться составной частью ОС или отдельным ПО (драйвер) устройства.

*USB System SW* – системная поддержка *USB*, не зависящая от конкретных устройств (драйвер шины).

*USB Host Controller* – аппаратно-программные средства для подключения устройств *USB* к хост-компьютеру.

### Физический интерфейс

Физический интерфейс включает в себя 4 провода:

1-й контакт – напряжение  $U_{bus}$  ( $U_{пит} = \pm 5$  В); 2-й контакт –  $D+$ ; 3-й контакт –  $D-$ ; 4-й контакт – земля.

Используются 2 вида соединения: витая пара (максимальная длина 5 м) экранированная; невитая пара – неэкранированная. Различие состоит в качестве и дальности проводов.

Витая пара (рис. 4.2) применяется для подключения полно-скоростных устройств (полная и высокая скорость).

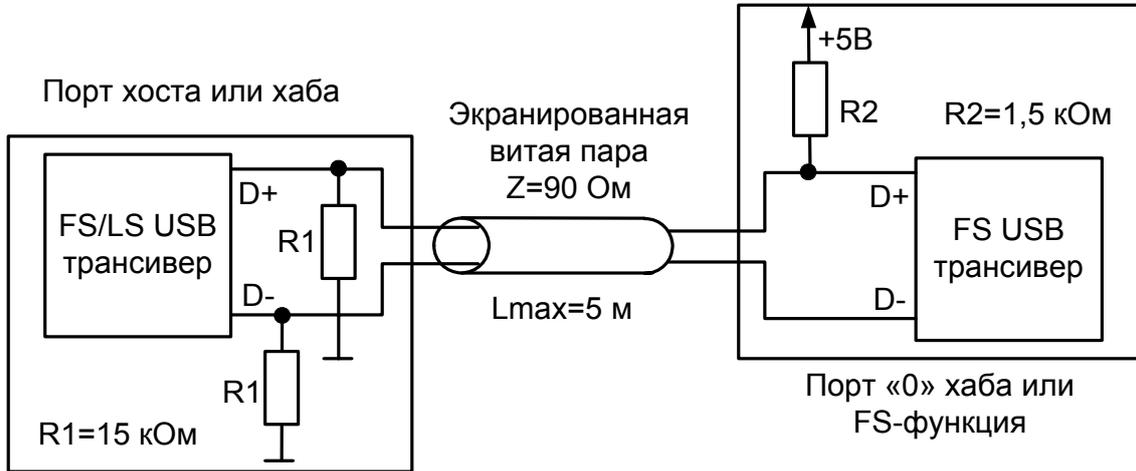


Рис. 4.2. Схема соединения с использованием витой пары

Невитая пара (рис. 4.3). Её длина до 3 м, используется для подключения устройств, работающих на малой скорости, т.е. для LS-функции.

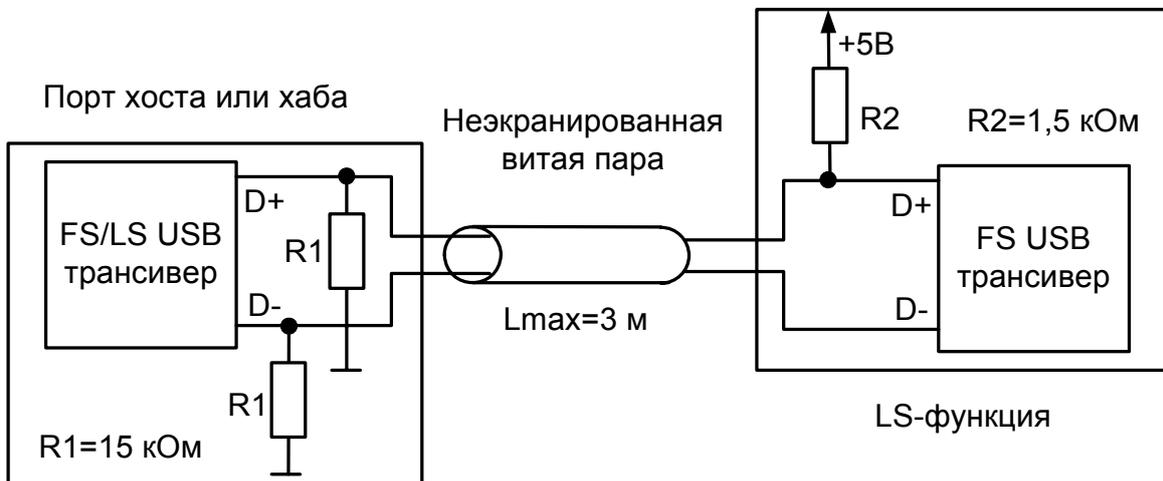


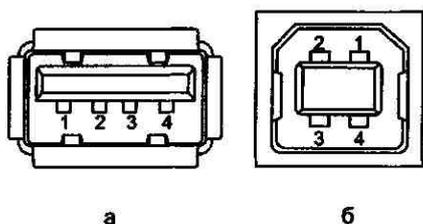
Рис. 4.3. Схема соединения с использованием невитой пары

Кабели 2 типов:

Тип А – для подключения к хабам устройств с неотсоединяемым кабелем (мышь, клавиатура).

Тип *B* – для устройств, у которых кабель может отсоединяться (принтер, сканер).

Разъемы типов *A* и *B* (рис. 4.4) различаются механически, что исключает недопустимые петлевые соединения портов хабов. Четырехконтактные разъемы имеют ключи, исключающие неправильное присоединение.



Конструкция разъемов обеспечивает позднее соединение и раннее отсоединение сигнальных цепей по сравнению с питающими. Для распознавания разъема *USB* на корпусе устройства ставится стандартное символическое обозначение.

Питание устройств *USB* возможно от кабеля (*Bus-Powered Devices*) или от собственного блока питания (*Self-Powered Devices*). Хост обеспечивает питанием непосредственно подключенные к нему ПУ. Каждый хаб, в свою очередь, обеспечивает питание устройств, подключенных к его нисходящим портам. Максимальная потребляемая мощность ограничена: в сумме не более 500 мА (т.е. каждое устройство не более 100 мА). Напряжения сигналов:  $U_{\text{пер}}"0" \leq 0,3$ ;  $U_{\text{пер}}"1" \geq 2,8$ ;  $U_{\text{прм}}"0" < 0,5$ ;  $U_{\text{прм}}"1" < 3,8$  В. Передатчик должен иметь возможность перехода в высокоимпедансное состояние.

Каналы (*pipe*) – модель передачи данных между конечной точкой и хост-контроллером. Могут быть каналы 2 видов: потоки и сообщения, массивы.

Поток может реализовать следующие виды обмена: сплошной, изохронный, прерываний. Сообщение имеет формат, определенный *USB*. Хост посылает запрос к контрольной точке, после которого передает пакет сообщения и информацию о состоянии конечной точки (квитанция о приеме).

## Протокол обмена

Весь обмен происходит кадрами. Кадр состоит из транзакций. Транзакция состоит из пакетов. Пакеты бывают 3 типов: пакет-маркер; пакет данных; пакет-подтверждение (квитанция). На изохронных передачах последний пакет отсутствует.

Параметры кадра:

– *FS, LS*: Длительность 1 мс, частота 1 кГц;

- *HS (USB 2.0)*: Длительность 125 нс; частота 8 кГц.

Каждая транзакция планируется контроллером и идентифицируется по следующим параметрам:

1. Наличие пакета-маркера. Он описывает тип и направление передачи, адрес устройства и номер конечной точки.

2. Источник данных, определяемый маркером. Передает пакет данных или уведомление об отсутствии данных для передачи.

3. После успешного приема пакета источники принимают подтверждение, причем может быть получен отказ от приема данных, если буфер переполнен.

## 4.2. Интерфейс *BlueTooth*

### Общая характеристика

Радиоинтерфейс *BlueTooth* может использоваться в мобильных телефонах для организации режима *Hands Free* (свободные руки) и для подключения к компьютеру и другим устройствам, в том числе к компьютерным сетям. Интерфейс появился в 1998 г. Его поддерживает группа компаний *SIG-2000*.

Интерфейс использует  $f = 2,4 - 2,4835$  ГГц (промышленный научно-технический диапазон - *ISM*).

Скорость передачи:

- 720 Кб/с в асимметричном режиме;

- 420 Кб/с в полудуплексном режиме.

Мощность  $P \leq 10$  мВт.

Длина связи от 10 до 100 м.

Интерфейс регламентируется стандартом IEEE 802.11. Локальные беспроводные сети.

Вся отведенная полоса делится на каналы  $\Delta f = 1\text{МГц}$ . Каждый канал представляет собой псевдослучайную последовательность скачков по 79 – 23 каналам, количество которых зависит от страны. По времени весь канал делится на сегменты (тайм-слоты) длительностью 625 мкс. Скачки происходят синхронно со скоростью 1600 скачков/с. В одной пикосети может быть до 7 подчиненных устройств и одно основное. Подчиненные устройства могут входить в несколько сетей. Через 227 мм происходит переключение. (227 - длина ПСП (всего 10 вариантов)). Частота смены каналов 1600 раз в секунду.

Установка соединения происходит автоматически в несколько этапов.

### Слои протоколов

Интерфейс имеет 4 слоя протоколов (см. таблицу)

Протокольный слой	Протоколы в слое
Корневые протоколы ( <i>Core Protocol</i> )	<i>BaseBand, LMP, L2CAP, S2DP</i>
Протокол с заменой кабеля (кабельное соединение)	<i>RFComm</i>
Протокол управления телефонией	<i>TCS binary, AT-команда</i>
Воспринятые протоколы ( <i>Adapted Protocol</i> )	<i>PPP, UDP/TCP/IP, OBEX, WAP, VCARD, IrMC, WAE</i>

### Временная диаграмма асинхронного режима (рис. 4.5)

Асинхронные пакеты передаются в то время, которое не зарезервировано для синхронных передач. Пакет состоит из 3 частей: 1) 72 бита - код доступа (идентифицирует пакеты и служит для синхронизации процедур запроса); 2) 54 бита - заголовок (состоит из 6 полей, может отсутствовать); 3) 2745 бит – данные.

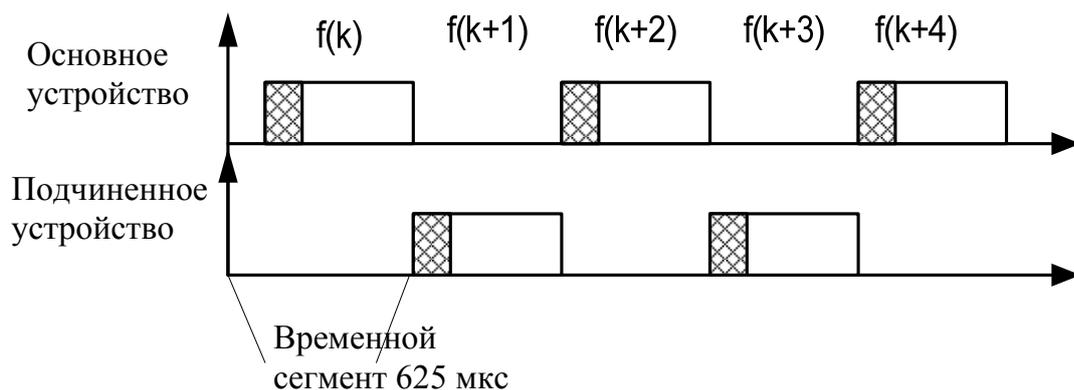


Рис. 4.5. Временная диаграмма асинхронного режима

### Электронная база. Общая структура устройства (рис. 4.6)

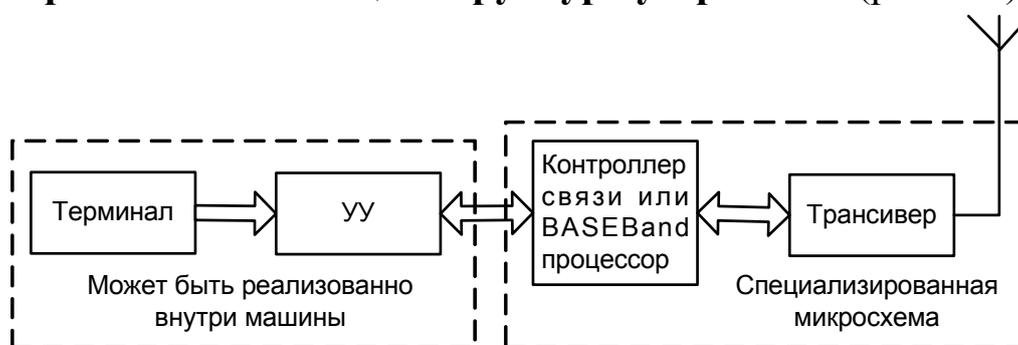


Рис. 4.6. Схема реализации интерфейса *Blue Tooth*

Трансивер преобразует битовый сигнал в радиосигнал. Происходит модуляция радиоволны битового сигнала. Управляющее устройство реализует протоколы верхнего уровня.

### 4.3. Инфракрасный порт

С его помощью можно подключить мобильный терминал (телефон) к компьютеру. Этот последовательный порт поддерживает 3 скорости: низкую  $V=115,2$  Кбит/с; среднюю  $V=1152$  Кбайт/с; высокую  $V=4$  Мб/с.

#### Стандарты:

1. *Sharp - IR*. Режим поддерживает двунаправленную связь, используя *DASK*. Скорости до 38,4 Кбит/с.

2. *IrDA - 1.0 SIR* - двунаправленная связь. Позволяет передавать данные со скоростью 115,2 Кбит/с (низкая скорость). Стандарт фирмы *HP*.

3. *IrDA - 1.1 MIR* - средняя скорость, *FIR* - высокая скорость.

4. Стандарт для бытовой аппаратуры.

### Способы подключения приемопередатчиков ИК

- Внутренние приемопередатчики для низкой скорости подключаются с помощью схемы *UART*.

- Для средней и высокой скорости используется специализированная схема контроллеров, которая обеспечивает прямое обращение к шине (*Pc87109*). Контроллер выполнен в виде карты расширения или интегрируется в системную плату. Приемопередатчик подключается напрямую или через специальный разъем.

- Внешние ИК-адаптеры выполняются с интерфейсом *RS-232C* или с шиной *USB*.

Для организации связи используются диоды (рис. 4.7):

1. Излучающий светодиод излучает в секторе  $30^\circ$  (рис. 4.7, а).

2. Пин-диод (фотодиод) принимает в секторе  $15^\circ$  (рис. 4.7, б).



Рис. 4.7 Типы диодов, используемые в ИК-интерфейсе

Параметры интерфейса: длина надежной связи 1,0 м; тип связи полудуплексный.

### Протоколы

Для организации связей используют протоколы семи уровней.

1. Физический уровень. Его временная диаграмма показана на рис. 4.8. Тактовая частота задается 24 МГц (тактовая частота генератора 1 МГц). Время импульса  $1,63 \text{ мкс} = 3/16$  бита.

В режимах *MIR* и *FIR* есть понятие кадра. Начало и конец кадра отмечаются флагами. Для контроля в *MIR* используется 16-битный *CRC*-код; в *FIR* - 32-битный *CRC*-код (позиционно-импульсный код для любой пары бит).

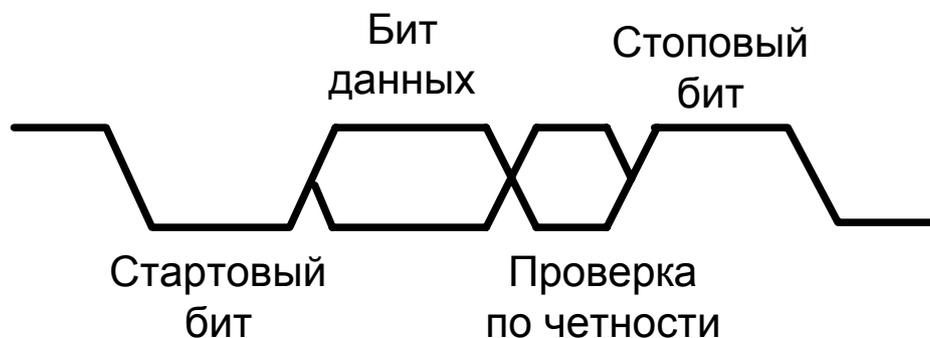


Рис. 4.8. Временная диаграмма ИК-интерфейса

2. *IrLAP* – ИК-модификация протокола *HDFC* для ИК-связи. Протокол описывает процедуру установления, нумерации и закрытия соединения.

3. Протокол установления соединения *IrLMP* - устройство сообщает о своем присутствии. Позволяет обнаруживать сервисы, предоставляемые другими устройствами, проверять потоки данных и выступать в роли мультиплексора.

4. Транспортный протокол *Tiny TP* обслуживает внутренние каналы, производит установку данных в пакеты и сборку данных из пакетов.

5. *IrCOMM* позволяет через ИК-связь эмулировать:

- обычное 3-проводное подключение (аналог *COM*-порта) по протоколу *TXD*, *RDX*, *GND*;
- 9-проводное подключение *RS-232C*.

6. *IrLAN* обеспечивает доступ к локальным сетям *Ethernet*, *TokenRing*. Требуется устройствам с интерфейсом *IrDA*, подключенным к сети.

7. Протокол объектного обмена (высокий уровень) *IrOBEX* определяет команды ввода-вывода (имеет расширение для мобильной связи).

#### 4.4. Устройства отображения информации (УОИ).

##### Дисплей

Различают 2 вида дисплеев: светодиодные и жидкокристаллические (ЖК).

Рассмотрим виды жидких кристаллов:

1. Смектические жидкие кристаллы, в которых длинные молекулы сгруппированы в слои. Между слоями существует слабое взаимодействие, что позволяет им легко сдвигаться относительно друг друга.

2. Нематические жидкие кристаллы, образованные длинными сигарообразными или нитевидными молекулами, которые параллельны, но беспорядочно сдвинуты вдоль своих осей.

3. Холестерические жидкие кристаллы, молекулы которых образуют закрученную структуру.

##### **Виды электрических явлений в кристаллах:**

1. Эффект динамического рассеивания. Под действием напряжения и тока жидкий кристалл становится матовым вследствие нарушения упорядоченности кристалла. Напряжение для достижения этого эффекта составляет 5 – 6 В,  $f$  питания 10 кГц, время включения 50 – 500 мс.

2. Твист-эффект. Эффект заключается в том, что при приложении напряжения молекулы ориентируются вдоль поля и не изменяют поляризацию света.  $U = 9 - 1,5$  В, время переключения 30 – 200 мс.

3. Эффект «гость-хозяин» основан на ориентации молекул носителя вместе с молекулами ЖК.  $U = 2 - 10$  В,  $t = 30 - 500$  мс.

##### **Параметры ЖК-индикаторов:**

1. Контрастность 80 – 100 %.
2. Напряжение 2 – 20 В переменное.
3. Ток 1 – 100 мкА.
4. Частота напряжения питания 10 Гц – 1 МГц.

## **Достоинства и недостатки ЖК**

Достоинства: простота конструкции; низкое энергопотребление; хорошие контраст и совместимость с ИС. Недостатки ЖК: необходимость подсветки; узкий диапазон температур; строго фиксированная разрешающая способность; некоторые пиксели могут выходить из строя.

### **Способы запитки матрицы пикселей**

По способу запитки, т.е. по способу подачи напряжения на активный пиксель, различают пассивные и активные матрицы. В пассивной матрице напряжение на пиксель подается по вертикальной и горизонтальной шинам, переводя пиксель в состояние требуемого уровня светопропускания. Напряжение действует очень коротковременно по отношению к длительности кадра, и поэтому качество изображения получается низким. Достоинство пассивной матрицы – малое число переключающих транзисторов, равное сумме разрешений по горизонтали и вертикали.

В активной матрице напряжение на каждый пиксель подается в течение всего кадра через индивидуальный транзистор, как правило, тонкопленочный. Это существенно повышает качество, но требует большого количества транзисторов, равного произведению разрешений, что ведет к увеличению стоимости. В настоящее время в дисплеях МТ используются, как правило, активные матрицы.

## **4.5. Клавиатура**

Клавиатура служит для набора информации. Она состоит из клавиш, подпружиненных резиновыми колпачками, и трех слоев пленки. На верхний и нижний слои нанесены проводники и контактные площадки под клавишами. Средний слой – разделительный. В нем под каждой клавишей сделано отверстие, обеспечивающее контакт между верхней и нижней пленками при нажатии клавиши. Схема управления (рис. 4.9) состоит из 2 процессоров.

Первый формирует электрический сигнал последовательно на каждой из шин. Второй производит опрос присоединенных к нему шин для обнаружения сигнала, возникающего при замыкании (соединении) горизонтальной шины (см. рис. 4.9) с вертикальной при нажатии соответствующей клавиши.

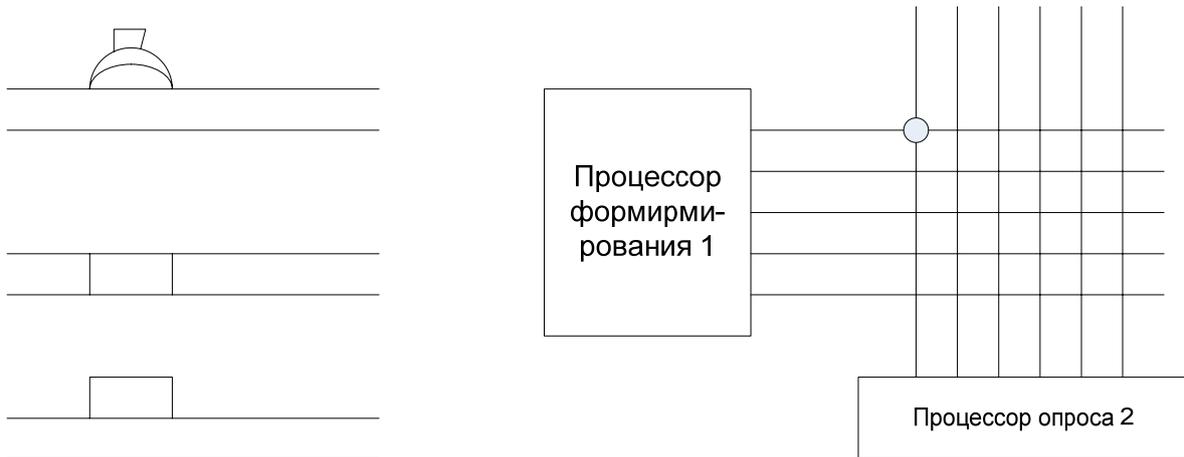


Рис. 4.9. Схема управления клавиатуры

#### 4.6. Фотокамера в смартфонах

Разрешение стандартных фотокамер в смартфонах составляет  $144 \times 176$  или  $128 \times 196$  пикселей (1,3 – 4 мегапикселей). Глубина цвета достигает 24 бит. Один пиксель может отображать один цвет. Применяются палитры RGB (рис. 4.10, а) и CMY (рис. 4.10, б). В RGB преобладает зеленый цвет.

G	R	G	R	G	R
B	G	B	G	B	G
G	R	G	R	G	R
B	G	B	G	B	G
G	R	G	R	G	R
B	G	B	G	B	G

а)

M	Y	M	Y	M	Y
G	G	G	G	G	G
M	Y	M	Y	M	Y
G	G	G	G	G	G
M	Y	M	Y	M	Y
G	G	G	G	G	G

б)

Рис. 4.10. Палитры, используемые в фотокамерах

Структурная схема фотокамеры приведена на рис. 4.11.

Величину  $(r/d)^2$  называют относительным фокусным расстоянием. Свет, прошедший через оптическую систему (линзы), попадает на светочувствительную матрицу. Применяются матрицы двух видов: ПЗС (*CCD*) и КМОП (*CIS*). Далее аналоговый сигнал подается на АЦП, преобразуется в цифровой и записывается в регистр РГ. Блок ЦСП обеспечивает интерполирование и конвертацию в *JPEG*. Видеоинформация хранится в формате *MPEG*.

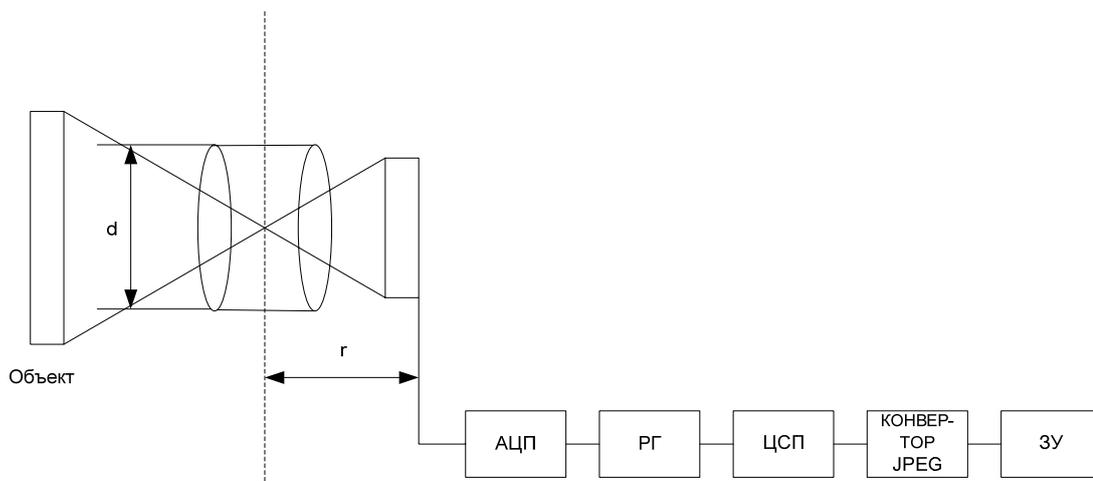


Рис. 4.11. Структурная схема фотокамеры

*MPEG* по структурному принципу очень схож с *JPEG* и даже основывается на этом стандарте, но с одной лишь разницей: в нем используется множество специфических надстроек. Например, для реализации короткого видеофрагмента нам может понадобиться только один полноценный кадр, сжатый в *JPEG*, а все остальное можно достраивать, регистрируя только небольшие изменения относительно исходного. Идея *JPEG*-кодирования проста – изображение по специальному алгоритму разбивается на блоки  $16 \times 16$  пикселей, к которым в дальнейшем применяется дискретное косинусное преобразование. При этом в зависимости от выбранного качества наименее значимые с точки зрения алгоритма цвета убираются (или подменяются более значимыми в

этом блоке), что позволяет значительно уменьшить объем хранимой информации.

Данный алгоритм в обычной жизни хорошо себя проявляет на красочных многоцветных фотографиях и отвратительно – на рисунках с небольшим набором цветов.

### ***Контрольные вопросы***

1. Классификация УОИ. Жидкокристаллический индикатор: виды электрических явлений в нем. Параметры и конструкция ЖК-индикаторов. Типы запитки матрицы.

2. Дисплей. Назначение. Параметры дисплеев на ЖК, преимущества и недостатки. Способы запитки матрицы.

3. Кодирование цифрового видео: *MPEG, JPEG*.

4. Клавиатура и ее клавиши. Схема управления клавиатурой.

5. Фотокамера, ее параметры и назначение. Структурная схема фотокамеры.

## Список рекомендуемой литературы

1. *Гельман, М. М.* Аналого-цифровые преобразователи для информационно-измерительных систем / М. М. Гельман. – М. : Изд-во стандартов, 1989. – 320 с.
2. *Феер, К.* Беспроводная цифровая связь. Методы модуляции и расширения спектра / К. Феер ; под ред. В. И. Журавлёва ; пер. с англ. – М. : Радио и связь, 2000. – 520 с. – ISBN 5-256-01444-7.
3. *Скляр, Б.* Цифровая связь. Теоретические основы и практическое применение / Б. Скляр. – М. : Вильямс, 2004. – 1104 с. – ISBN 5-8459-0497-8.
4. *Галкин, В. А.* Цифровая мобильная радиосвязь : учеб. пособие для вузов / В. А. Галкин. – М. : Горячая линия–Телеком, 2007. – 432 с.
5. *Прокис, Дж.* Цифровая связь / Дж. Прокис ; под ред. Д. Д. Кловского ; пер. с англ. – М. : Радио и связь, 2000. – 800 с. – ISBN 5-256-01434-X.
6. *Попов, В. И.* Основы сотовой связи стандарта GSM / В. И. Попов. – М. : Эко-Трендз, 2005. – 206 с. – ISBN 5-88405-068-2.
7. *Гук, М.* Аппаратные интерфейсы ПК : энцикл. / М. Гук. – СПб. : Питер, 2002. – 528 с. – ISBN 5-94723-180-8.
8. *Он же.* Аппаратные средства IBM PC : энцикл. / М. Гук. – СПб. : Питер, 2003. – 928 с. – ISBN 5-318-00047-9.
9. *Мюллер, С.* Модернизация и ремонт ПК / С. Мюллер. – 12-е изд. – М. : Вильямс, 2001. – 1184 с. – ISBN 5-8459-0167-7.
10. *Волков, Л. Н.* Системы цифровой радиосвязи: базовые методы и характеристики : учеб. пособие / Л. Н. Волков [и др.]. – М. : Эко-Трендз, 2005. – 392 с.

## ОГЛАВЛЕНИЕ

Предисловие.....	3
Введение.....	4
Характеристики терминального оборудования мобильных станций.....	4
Классификация МТ.....	5
<b>1. МОБИЛЬНЫЕ СТАНЦИИ.....</b>	<b>6</b>
1.1. Функциональная схема типичной системы цифровой связи.....	6
1.2. Структурные схемы мобильной станции.....	7
Контрольные вопросы.....	11
<b>2. ПРЕОБРАЗОВАНИЕ РЕЧЕВОЙ ИНФОРМАЦИИ.....</b>	<b>12</b>
2.1. Виды информации.....	12
2.2. Форматирование и передача видеосигнала в цифровой связи.....	13
2.3. Кодирование речи.....	15
2.4. Обзор методов кодирования речевых сигналов.....	35
Контрольные вопросы.....	36
<b>3. КОДИРОВАНИЕ КАНАЛА.....</b>	<b>38</b>
3.1. Назначение канального кодирования.....	38
3.2. Ортогональные коды.....	39
3.3. Биортогональные коды.....	39
3.4. Структурированные последовательности.....	40
3.5. Модели каналов.....	41
3.6. Блочное кодирование.....	42
3.7. Коды с контролем четности.....	44
3.8. Компромиссы, решаемые при кодировании.....	46
3.9. Линейные блочные коды.....	47
3.10. Систематические линейные блочные коды.....	49

3.11. Блочные циклические коды.....	54
3.12. Варианты блочных кодов .....	59
3.13. Сверточное кодирование (СК) .....	59
3.14. Турбокоды.....	66
3.15. Схемы перемежения .....	67
3.16. Структурная схема кодера канала стандарта <i>GSM</i> ....	70
Контрольные вопросы.....	74
<b>4. ИНТЕРФЕЙСЫ И ДОПОЛНИТЕЛЬНЫЕ</b>	
<b>УСТРОЙСТВА МОБИЛЬНЫХ ТЕРМИНАЛОВ .....</b>	<b>75</b>
4.1. Интерфейс <i>USB</i> .....	75
4.2. Интерфейс <i>BlueTooth</i> .....	79
4.3. Инфракрасный порт .....	81
4.4. Устройства отображения информации (УОИ).	
Дисплей .....	84
4.5. Клавиатура .....	85
4.6. Фотокамера в смартфонах.....	86
Контрольные вопросы.....	88
Список рекомендуемой литературы .....	89

Учебное издание

БЫКОВ Валерий Ильич

УСТРОЙСТВА ПРЕОБРАЗОВАНИЯ И ОБРАБОТКИ ИНФОРМАЦИИ  
В СИСТЕМАХ ПОДВИЖНОЙ СВЯЗИ

Учебное пособие

Подписано в печать 11.06.09.

Формат 60x84/16. Усл. печ. л. 5,35. Тираж 100 экз.

Заказ

Издательство

Владимирского государственного университета

600000, Владимир, ул. Горького, 87