

# Системы мобильной связи

Лекция № 14

*Тема: Канальное кодирование в стандарте GSM. Принцип кодирования канала трафика. Турбокоды: определение, принцип формирования кодовой комбинации.*

# ***Канальное кодирование в стандарте GSM***

**Кодер канала** - второй (и последний) элемент цифрового участка передающего тракта. Он следует после кодера речи и предшествует модулятору, осуществляющему перенос информационного сигнала на несущую частоту.

**Основная задача кодера канала** - *помехоустойчивое кодирование сигнала речи, т.е. такое его кодирование, которое позволяет обнаруживать и в значительной мере исправлять ошибки, возникающие при распространении сигнала по радиоканалу от передатчика к приемнику.*

Помехоустойчивое кодирование осуществляется за счет введения в состав передаваемого сигнала довольно большого объема избыточной (контрольной) информации. В английской терминологии такое кодирование носит наименование *Forward Error Correcting coding (FEC coding)*, т.е. кодирование с упреждающей коррекцией ошибок, или кодирование с коррекцией ошибок на проходе.

# ***Канальное кодирование в стандарте GSM***

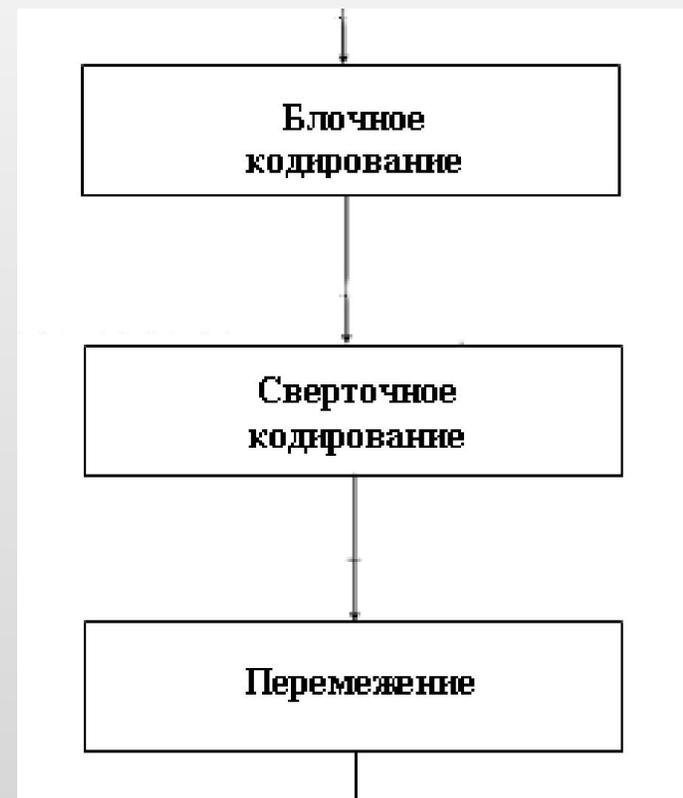
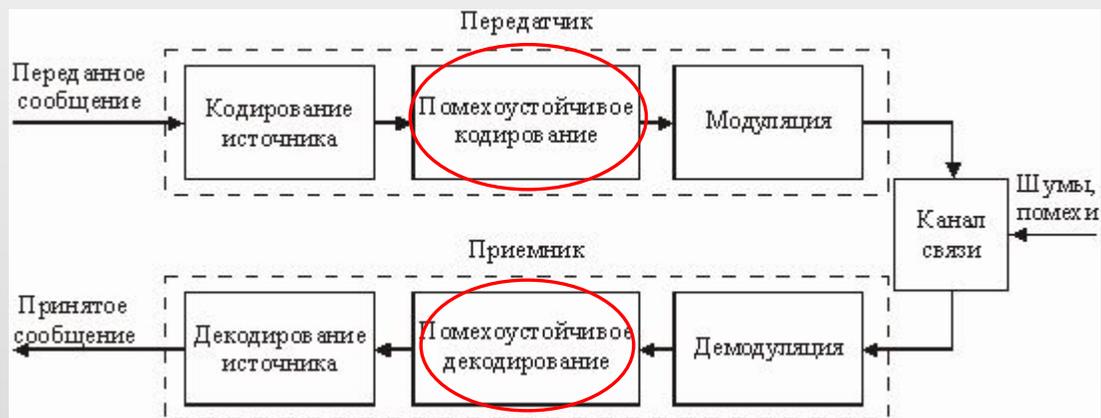
**В сотовой связи помехоустойчивое кодирование реализуется в виде трех процедур:**

- 1. блочного кодирования (block coding);*
- 2. сверточного кодирования (convolutional coding);*
- 3. перемежения (interleaving).*

**Кроме того, кодер канала выполняет еще ряд функций:**

- добавляет управляющую информацию, которая, в свою очередь, также подвергается помехоустойчивому кодированию;*
- упаковывает подготовленную к передаче информацию и сжимает ее во времени;*
- осуществляет шифрование передаваемой информации, если таковое предусмотрено режимом работы аппаратуры.*

# Канальное кодирование в стандарте GSM

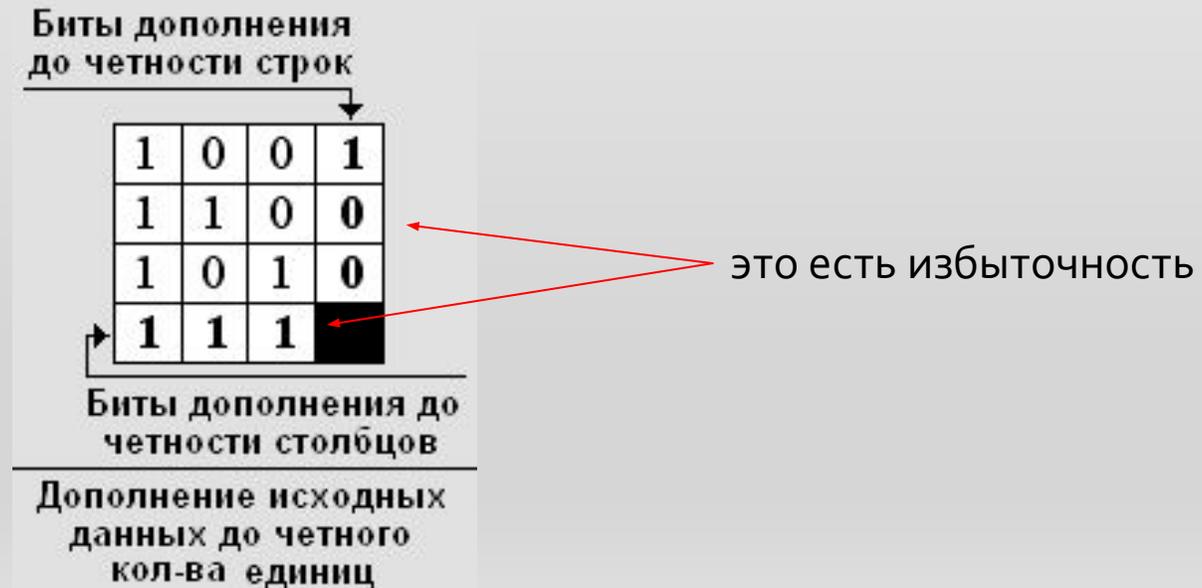


# Блочное кодирование

Избыточное кодирование информации можно разделить на два метода – это блочное кодирование и сверточное.

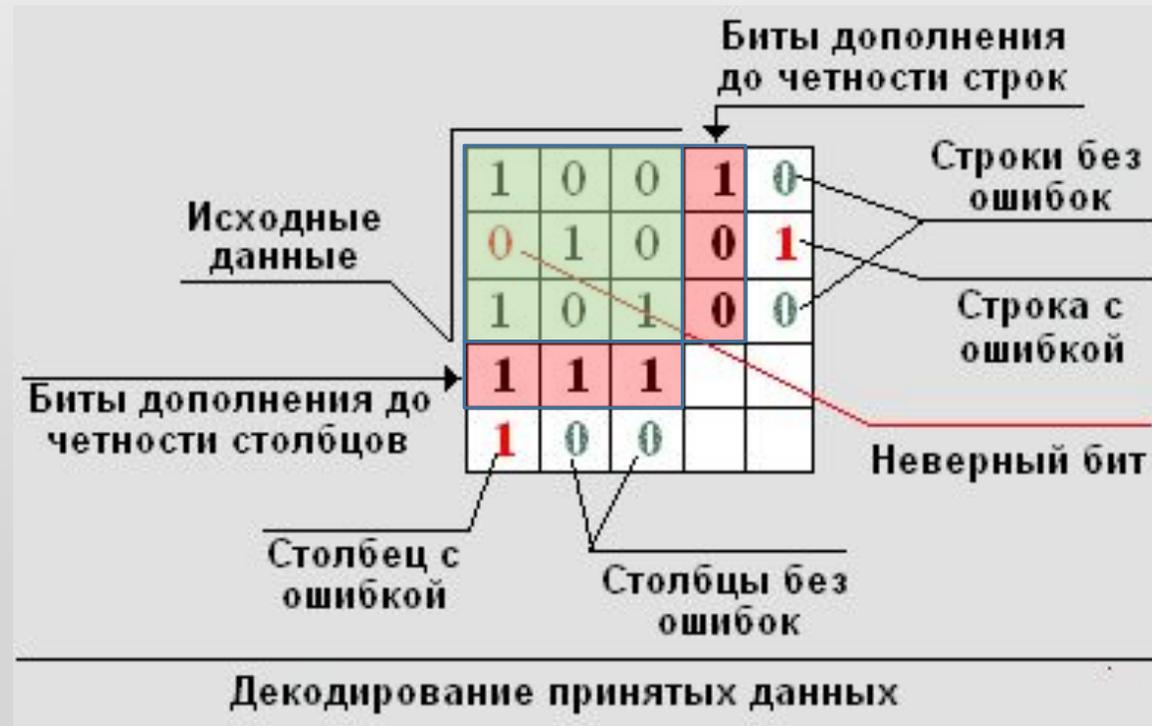
При блочном кодировании информация делится на блоки определенной длины, и каждый блок кодируется отдельно.

Простейшим примером блочного кодирования является дополнение до четности. В этом случае каждый блок делится на группы, которые дополняются одним битом со значением единица или ноль, в зависимости от того четное или нечетное количество единиц в исходной группе.



# Блочное кодирование

Произведем его декодирование, для чего запишем в виде матрицы **4 на 4**, где матрица **3 на 3** - исходные данные, которые надо было передать, остальные элементы матрицы – коды дополнения до четности. Проведя суммирование бит в строках и столбцах, видим, что в первом столбце и второй строке количество единиц - нечетное, а, следовательно, в них содержатся ошибки.



## ***Блочное кодирование***

Блочное кодирование обладает высокой надежностью, простотой реализацией, и получило широкое распространение. Существует большое разнообразие блочных кодов, обладающих различными возможностями по обнаружению и исправлению ошибок, среди которых наиболее известны коды Хемминга.

# Сверточное кодирование

*Сверточные коды работают со всем массивом данных, не деля его на части. В простейшем случае, в исходные данные добавляются проверочные символы, представляющие собой сумму двух символов исходных данных.*

*Рассмотрим для примера работу сверточного кода, в котором проверочные символы вставляются между всеми символами исходного кода и представляют собой сумму двух смежных символов исходного кода.*

Закодируем с помощью этого метода следующую бинарную последовательность данных:

**100 100 110 101**

# Сверточное кодирование

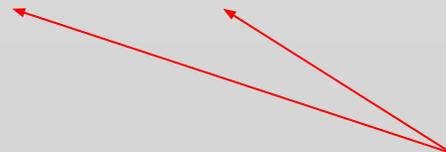
Закодируем с помощью этого метода следующую бинарную последовательность данных:

**100 100 110 101**

Для получения первого проверочного символа сложим первый и второй информационные символы исходной последовательности и запишем результат между ними: **110**. Красным цветом выделен проверочный символ.

Аналогичные действия проводим со всеми остальными информационными битами последовательности. В результате получим следующую закодированную последовательность:

**110 001 110 001 101 101 110 11**

 избыточность

# *Сверточное кодирование*

При приеме данных происходит их декодирование, т.е. суммируются соседние биты исходных данных и сравниваются с их проверочным битом.

Если для двух соседних проверочных битов была зафиксирована ошибка, то общий информационный бит для этих двух проверочных битов - неверен. Для исправления ошибки необходимо заменить его на противоположный.

Если для одного проверочного символа была зафиксирована ошибка, а два соседних проверочных символа ошибку не показали, это означает, что сбой произошел в проверочном символе, а информационные биты корректны.

Для уточнения, где именно произошла ошибка, проверяются соседние пары информационных бит.

**110 001 110 001 101 101 110 11**

# *Перемежение*

Передаваемый через эфир радиосигнал подвергается помехам различных типов. Это могут быть промышленные шумы, атмосферные помехи (например, грозы) и т.п.

**При этом многие ошибки не одиночны по времени, а возникают пачками. Это означает, что длительности воздействующего мешающего сигнала достаточно для возникновения ошибок в нескольких подряд идущих битах.**

Главная опасность такого вида помех заключается в том, что применяемые способы защиты от помех обычно могут распознать и исправить **не более одной ошибки**. Пачечные ошибки эти виды защиты не определяют, что соответственно может привести к ухудшению качества связи.

# Перемежение

Для борьбы с пачечными ошибками в сотовой связи применяется так называемый Interleaving или перемежение.

*Суть его заключается в том, что перед передачей в эфир биты переставляются местами.*

Например, вместо последовательности «**1, 2, 3, 4, 5, 6 ...**» создается последовательность: «**5, 3, 6, 1, 4, 2 ...**».

Причем одна и та же схема перемежения, обычно, накладывается как маска и применяется циклически к цифровому потоку.

*После приема сигнала последовательность подвергается обратной перестановке, чтобы получить исходный сигнал. В случае, если на сигнал будет воздействовать пачечная помеха, например, на подряд идущие биты **3, 6 и 1**, то после восстановления исходного потока эти биты окажутся не рядом стоящими и к ним уже можно будет применить стандартные алгоритмы защиты от ошибок.*

# Перемежение

Очевидно, что чем меньше отрезок сигнала, т.е. чем короче кадр по времени будет подвержен перемежению, тем более коротким пачечным ошибкам он может противостоять.

Однако чем более длительный отрезок сигнала будет вовлечен в перемежение, тем больше это потребует производственных возможностей и может потребовать дополнительных временных затрат и привести к задержкам сигнала.

Поэтому на практике выбирают золотую середину: берут достаточно длительный кадр для перемежения, что бы можно было противостоять пачечным ошибкам достаточно часто встречающимся в радиоэфире.

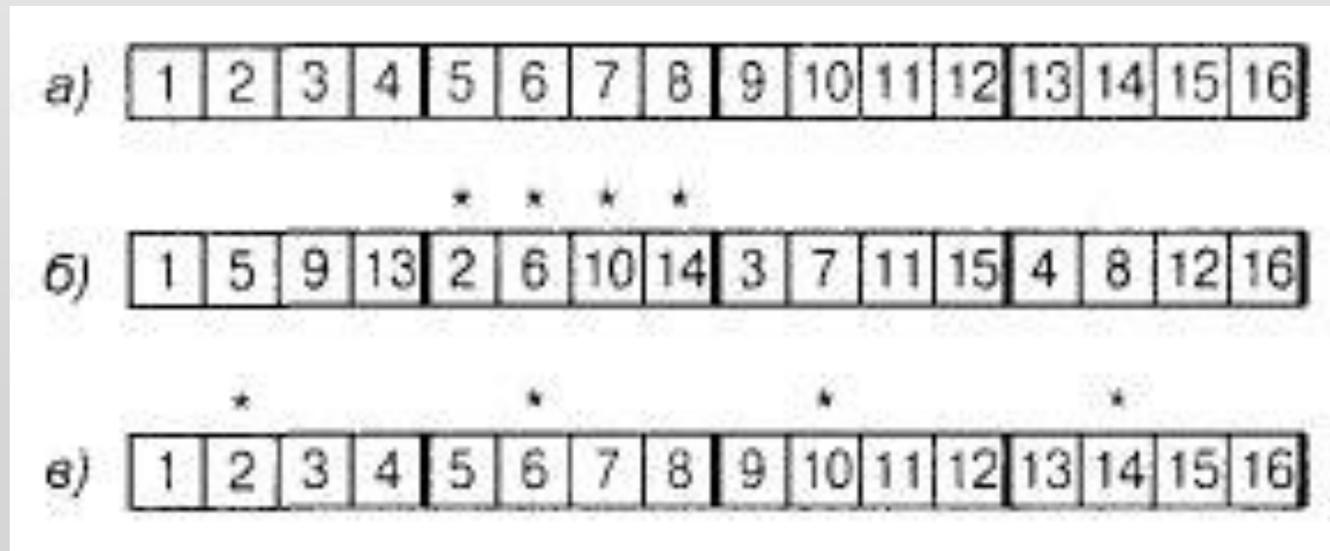


# Перемежение

Пример перемежения и деперемежения. Исходный цифровой сигнал представляет собой последовательность 4-разрядных двоичных слов, передаваемых бит за битом. Перемежение выполняется в пределах каждого 4 слов, т.е. в пределах отрезка цифрового сигнала, содержащего 16 бит.

Числа показывают номера битов в этом отрезке. В результате перемежения биты переставляются.

Биты, искаженные действием пакетной ошибки, отмечены звездочками. В результате деперемежения восстанавливается исходный порядок битов и искаженные биты рассредотачиваются.



# Турбокоды

**Турбо-код** — параллельный каскадный блочный систематический код, способный исправлять ошибки, возникающие при передаче цифровой информации по каналу связи с шумами. Синонимом турбо-кода является известный в теории кодирования термин — каскадный код.

Турбо-код состоит из каскада параллельно соединённых систематических кодов. Эти составляющие называются компонентными кодами. В качестве компонентных кодов могут использоваться свёрточные коды, коды Хемминга, Рида — Соломона, Боуза — Чоудхури — Хоквингема и другие.

В зависимости от выбора компонентного кода турбо-коды делятся на свёрточные турбо-коды и блочные коды-произведения.

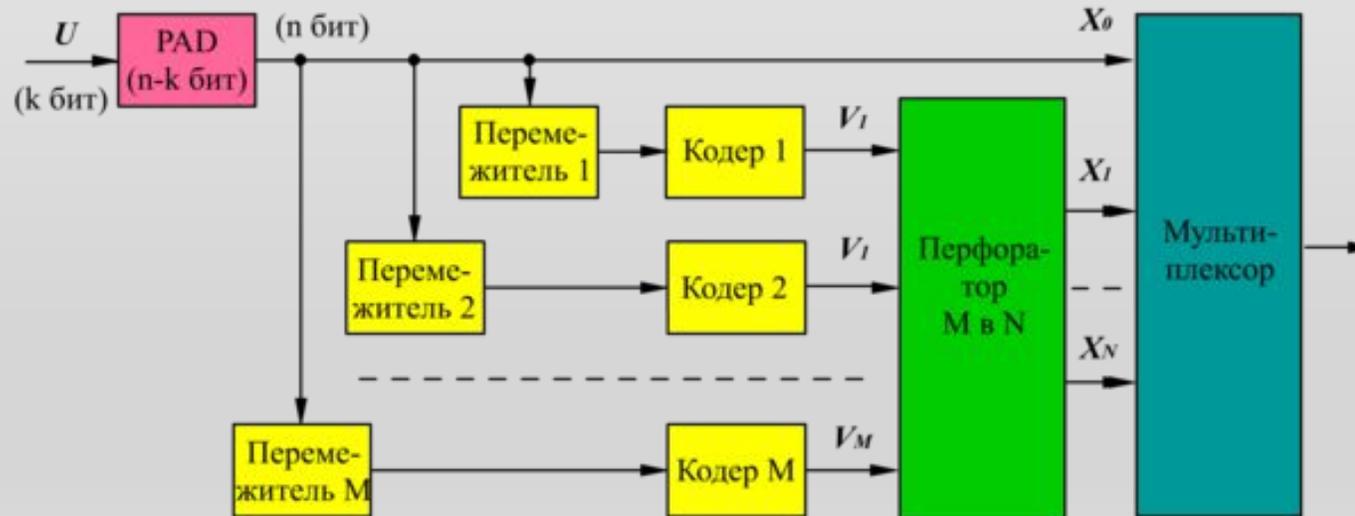
*Турбо-коды были разработаны в 1993 году и являются классом высокоэффективных помехоустойчивых кодов с коррекцией ошибок, используются в электротехнике и цифровой связи, а также нашли своё применение в спутниковой связи и в других областях, в которых необходимо достижение максимальной скорости передачи данных по каналу связи с шумами в ограниченной полосе частот.*

# Турбокоды

## Кодирование:

Сначала на вход формирователя пакетов поступает блок данных длиной  $K$  бит. В формирователе пакетов к данным прибавляется ещё  $(N-K)$  дополнительных бит служебной информации, соответствующих используемому стандарту формирования пакета и включающих в себя символы его начала и окончания. То есть получается пакет состоящий из  $N$  бит.

Далее эта последовательность бит поступает параллельно на  $M$  ветвей содержащих последовательно соединённые перемежитель и компонентный кодер. Таким образом пакет данных используется в качестве входных данных сразу всеми компонентными кодерами.



# Турбокоды

Задача перемежителя — преобразовать входную последовательность так, чтобы комбинации бит пакета входных данных соответствующие кодовым словам с низким весом (**весом** называется число ненулевых бит кодового слова) на выходе первого кодера, были преобразованы в комбинации, дающие кодовые слова с высоким весом на выходах остальных кодеров. Таким образом кодеры получают на выходе кодовые слова с различными весами.

При кодировании формируются кодовые слова так, чтобы получалось максимально возможное среднее расстояние между ними (расстоянием между двумя кодовыми словами называется число бит, в которых они различаются). Из-за того что кодовые блоки формируются из почти независимых частей, на выходе турбокодера среднее расстояние между кодовыми словами больше, чем минимальное расстояние для каждого компонентного кодера, а следовательно растёт эффективность кодирования.

