

Лекция 7

Тема: Непрерывные коды и составные коды

1. Свёрточный код ($R=0,5$)
2. Принцип каскадного кодирования;
3. Перемежение информации;
4. принципы построения турбо-кодов (ТК)



Непрерывные коды

Определение

- **Непрерывными*** (НК) называются коды, в которых операции кодирования и декодирования производятся непрерывно над последовательностью посылок без деления их на блоки. Т.е. проверочные символы образуются по мере поступления информационных символов в кодирующее устройство, начиная с того момента, когда число информационных символов превысит единицу ($n \geq 2$).



Непрерывные коды

- В НК выделить информационные и проверочные символы можно только в процессе декодирования.
- НК способны обнаруживать и исправлять случайные (одиночные) ошибки и пачки ошибок. Их корректирующая способность зависит от длины выборки.



Сверточные коды

- Наиболее распространенными являются сверточные коды (СК) с избыточностью $R=0,5$ (на один информационный символ приходится один проверочный).
- СК позволяют при вероятности ошибки $p=0,1$ получить вероятность ошибочного приема $p_{оп}=10^{-9}$. Т.е. достоверность повышается в 10^8 раз.



Сверточные коды

- Принцип обнаружения и исправления ошибок аналогичен методу максимального правдоподобия, который применяют для отыскания оптимального пути на графе по критерию минимального веса исследуемых кодовых комбинаций.
- Выбирается тот путь, который имеет минимальные веса, т.е. максимальное сходство с исходной комбинацией.



Сверточные коды

- Для этого кодирующие и декодирующие устройства должны запоминать несколько предыдущих состояний.
- Для кодирования применяют автоматы **Мили** или **Мура**.
- Для декодирования обычно используют алгоритм **Витерби**.

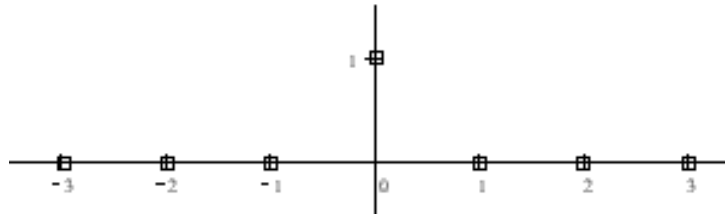


Свертка дискретных сигналов (пояснение)

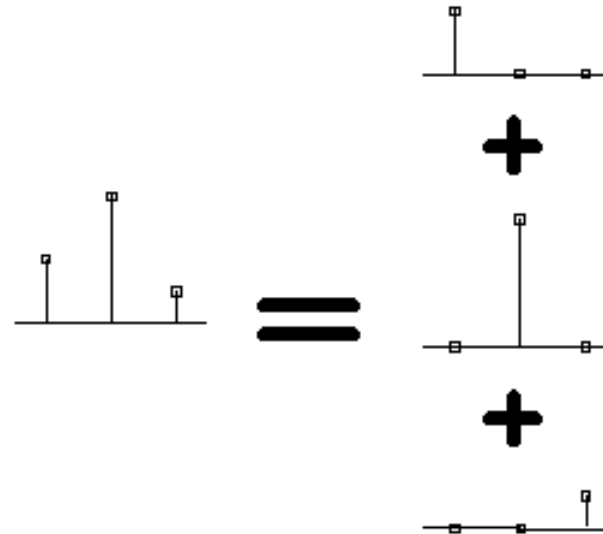
- Сверткой называется математическая операция вычисления отклика линейной системы (ЛС) на произвольный входной сигнал
- Для вычисления свертки необходимо знать входной сигнал и отклик ЛС на единичный импульс (импульсную характеристику или ядро свертки)
- Математически свертку можно представить как произведение полиномов (сигнала и импульсной характеристики)

Импульсная характеристика ЛС

- Единичный импульс $\delta[n]$

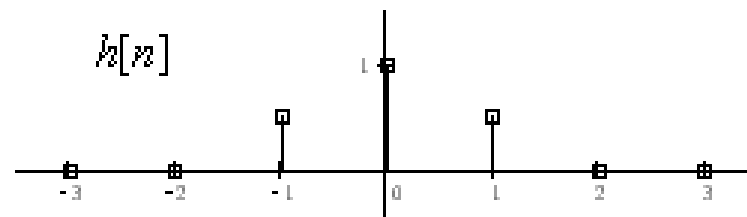
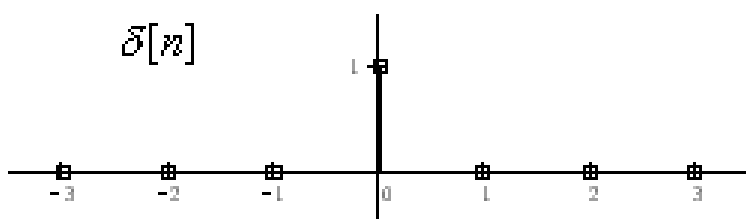


- Дискретный произвольный сигнал можно разложить на взвешенную сумму единичных импульсов



Импульсная характеристика

- Отклик системы на единичный импульс



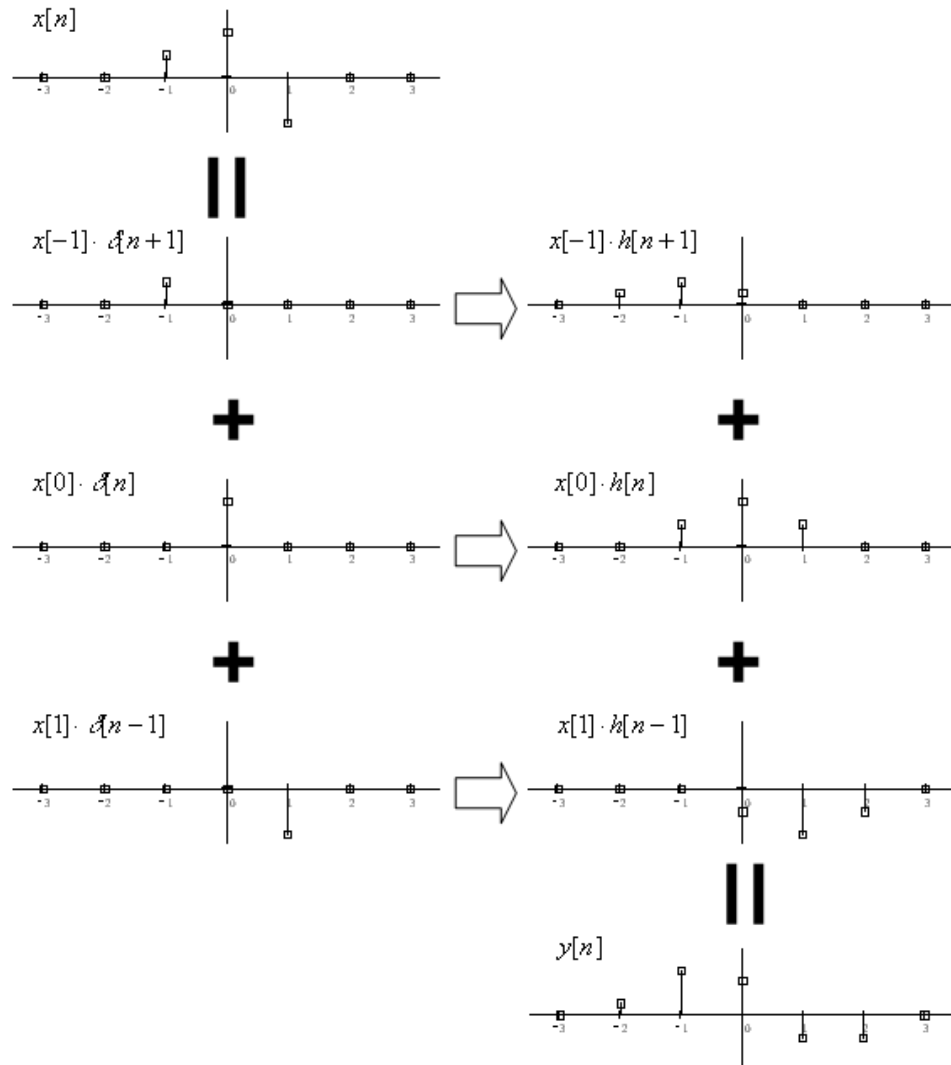
- $h[n]$ - импульсная характеристика системы (импульсный отклик системы)

Пример вычисления отклика линейной системы на произвольный входной сигнал (свертки)

$$y[n] = h[n] * x[n]$$

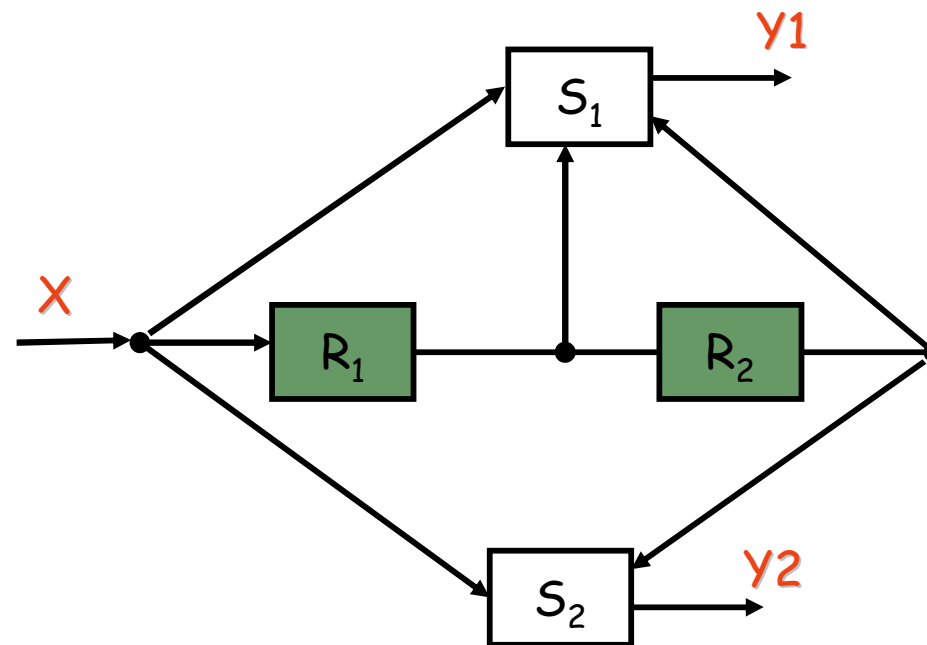
$$y[n] = \sum_{k=-\infty}^{+\infty} x[n-k]h[k]$$

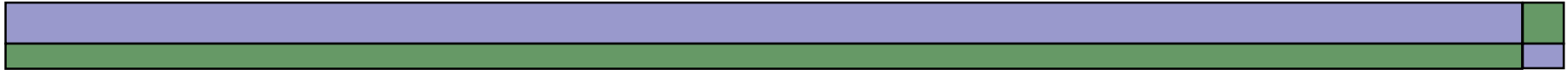
$h[n]$ – ядро свертки



Сверточные коды

- Схема автомата Мура, применяемая для кодирования содержит два сумматора по $\text{mod } 2$ (S_1 и S_2) и два регистра (R_1 и R_2).





Пример

- Пусть на вход **КУ** поступает информационная последовательность **10011101**.
- В начальный момент времени оба регистра сдвига обнулены.

Пример (кодирование)

X	R_1	R_2	S_1	S_2	Y_1	Y_2
1	0	0	$1 \oplus 0 \oplus 0$	$1 \oplus 0$	1	1
0	1	0	$0 \oplus 1 \oplus 0$	$0 \oplus 0$	1	0
0	0	1	$0 \oplus 0 \oplus 1$	$0 \oplus 1$	1	1
1	0	0	$1 \oplus 0 \oplus 0$	$1 \oplus 0$	1	1
1	1	0	$1 \oplus 1 \oplus 0$	$1 \oplus 0$	0	1
1	1	1	$1 \oplus 1 \oplus 1$	$1 \oplus 1$	1	0
0	1	1	$0 \oplus 1 \oplus 1$	$0 \oplus 1$	0	1
1	0	1	$1 \oplus 0 \oplus 1$	$1 \oplus 1$	0	0

\uparrow \uparrow \uparrow \uparrow \uparrow
 X R_1 R_2 X R_2



Пример (кодирование)

- Из таблицы видно, что после кодирования исходной последовательности получили кодовую комбинацию

11 10 11 11 01 10 01 00,
1 0 0 1 1 1 0 1

в которой пары символов, расположенные слева направо, представляют собой последовательные значения выходных сигналов кодера y_1 и y_2 .

Проверим полученный результат

- В рассмотренном примере

$$h_1[x] = x^2 + x + 1$$

$$h_2[x] = x^2 + 1$$

$x \rightarrow 10011101$ соответствует
 $x^7 + x^4 + x^3 + x^2 + 1$

т.е.

$$y_1 = x * h_1[x] \rightarrow 1111010011$$

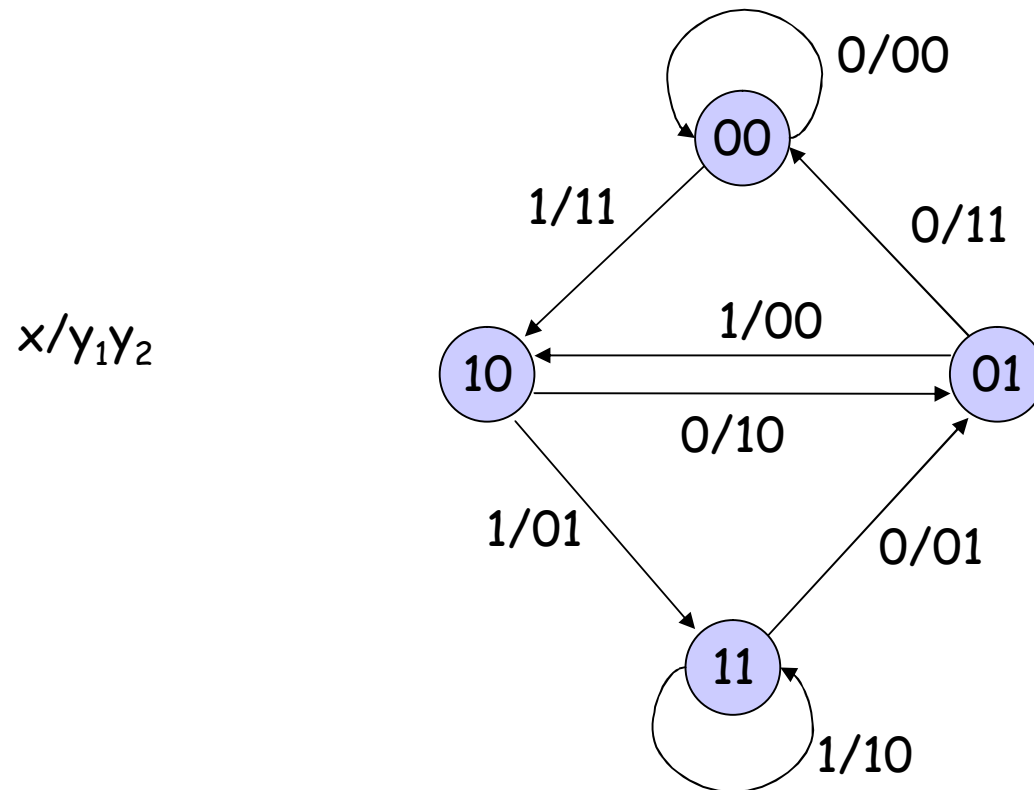
$$y_2 = x * h_2[x] \rightarrow 1011101001$$

$$\begin{array}{r}
 x^7 + \quad \quad \quad x^4 + x^3 + x^2 + \quad 1 \\
 \quad \quad \quad x^2 + x + 1 \\
 \hline
 x^7 + \quad \quad \quad x^4 + x^3 + x^2 + \quad 1 \\
 x^8 + \quad \quad \quad x^5 + x^4 + x^3 + \quad x \\
 x^9 + \quad \quad \quad x^6 + x^5 + x^4 + \quad x^2 \\
 \hline
 y_1 = x^9 + x^8 + x^7 + x^6 + x^4 + x + 1
 \end{array}$$

$$\begin{array}{r}
 x^7 + \quad \quad \quad x^4 + x^3 + x^2 + \quad 1 \\
 \quad \quad \quad x^2 + 1 \\
 \hline
 x^7 + \quad \quad \quad x^4 + x^3 + x^2 + \quad 1 \\
 x^9 + \quad \quad \quad x^6 + x^5 + x^4 + \quad x^2 \\
 \hline
 y_2 = x^9 + x^7 + x^6 + x^5 + x^3 + 1
 \end{array}$$

15

Орграф автомата Мура





Пример (декодирование)

- Для декодирования переданного сообщения необходимо строить дерево или сеть состояний.
- Т.к. кодер запоминает только два предыдущих состояния, то сеть или дерево представляют в виде графа, в вершинах которого отмечают два предыдущих состояния.



Пример (декодирование)

- На дугах отмечают текущие значения входного (информационного) сигнала x на данном шаге.
- Рядом отмечаются соответствующие значения выходных сигналов y_1 и y_2 .
- Для того, чтобы различать входные и выходные сигналы, будем обводить значение входного сигнала рамкой.



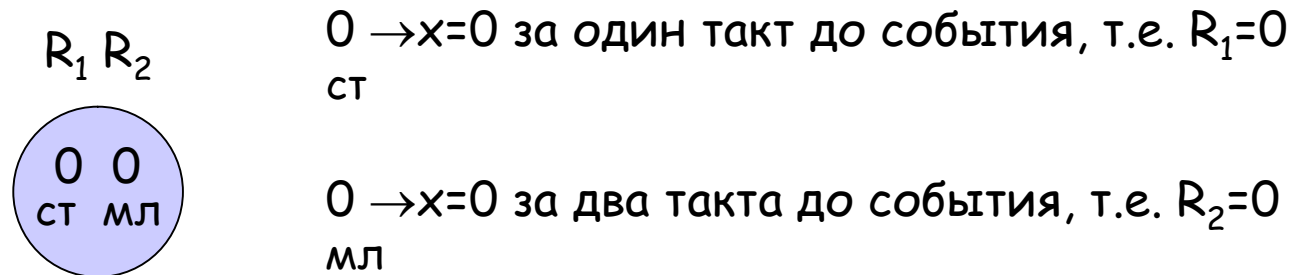


Пример (декодирование)

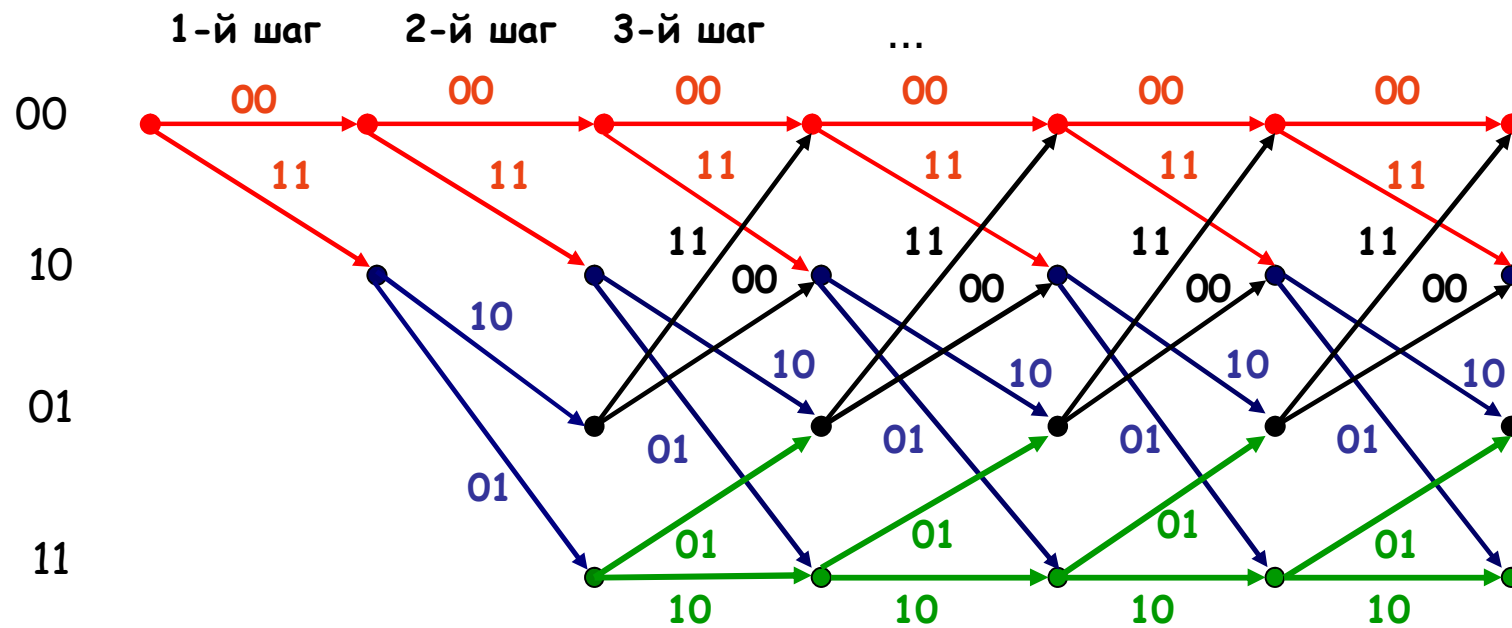
- Из графа видно, что если на вход **x** поступает **0**, то осуществляется переход вверх по дереву, а если **1** - то вниз по дереву.
- При этом состояния автомата определяются значениями входного сигнала за два предыдущих такта.
- Например, если текущее состояние автомата обозначено **00**, то это означает, что значение входной переменной **x** было равно нулю как за один, так и за два такта до рассматриваемого события.



Обозначение состояния автомата на графе



Сеть Витерби*





Алгоритм декодирования Витерби

- Декодирование осуществляется по тому же дереву, фрагмент которого изображен на слайде 14, но в вершинах дерева записываются не состояния автомата, а веса.
- Эти веса получают суммированием по **mod2** пришедших из **КС** пар символов с парами выходных символов, которые должны получаться на данном переходе при кодировании.



Алгоритм декодирования Витерби

- По мере продвижения по дереву подсчитывается текущий вес, а сверху над вершинами проставляется суммарный вес с учетом предыдущих переходов.
- Правильным или «выжившим» считается путь, дающий минимальный суммарный вес (минимальное расхождение с правильной кодовой комбинацией).
- Пройдя «выживший» путь, записывают правильную кодовую комбинацию.



Пример декодирования

- В рассмотренном примере была закодирована комбинация

10011101.

- После кодирования была получена кодовая комбинация

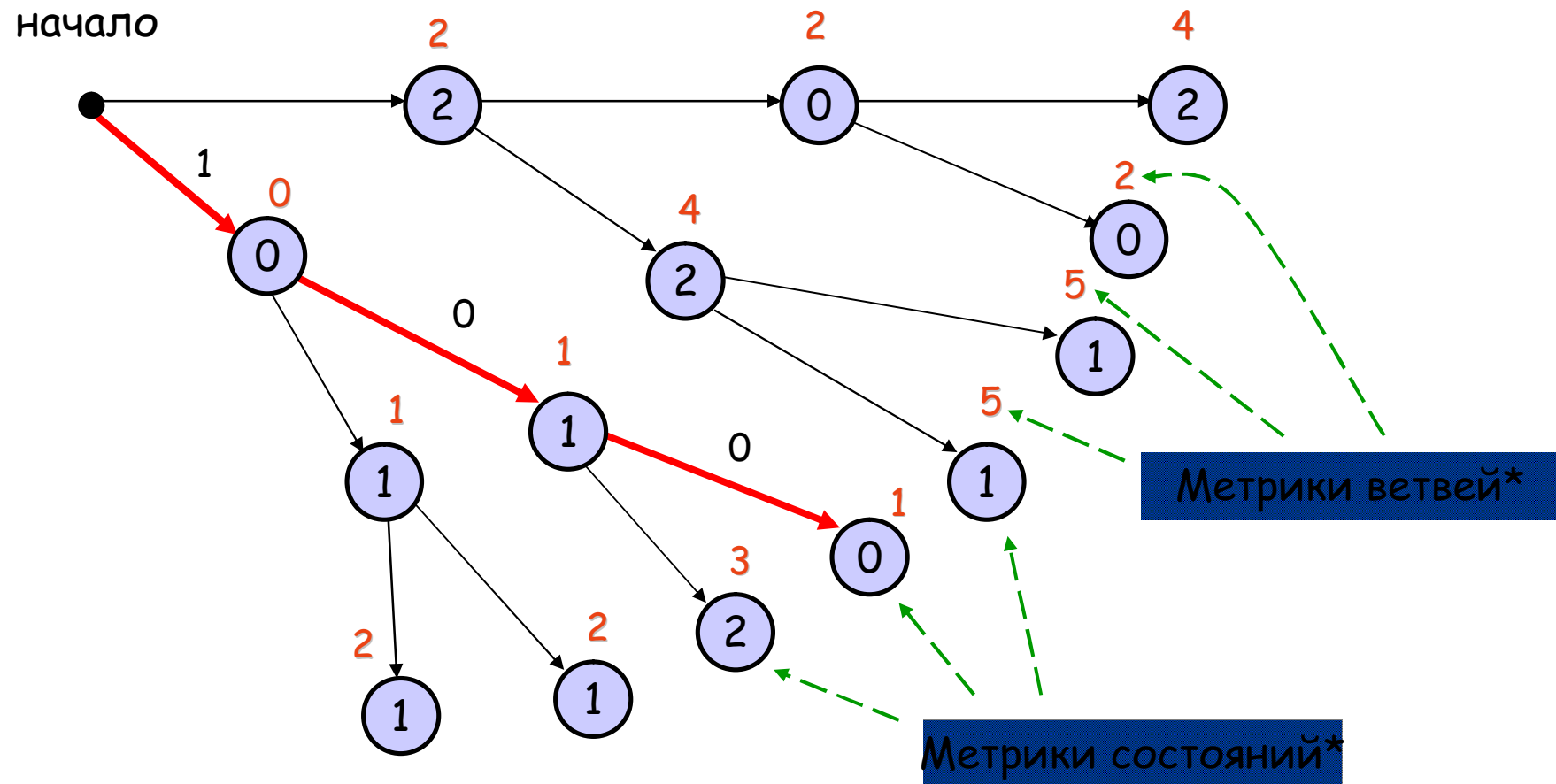
11	10	11	11	01	10	01	00
<u> </u>	<u> </u>	<u> </u>	<u> </u>	<u> </u>	<u> </u>	<u> </u>	<u> </u>
1	0	0	1	1	1	0	1



Пример декодирования

- Рассмотрим процесс декодирования начального фрагмента этой комбинации, полагая, что с ошибкой был принят третий символ, т.е. на вход декодера поступает комбинация 110011... вместо 111011...

Пример декодирования



Пример декодирования

10011101 → 111011... (без ошибок)

110011 (с ошибкой в 3-м символе)

1-й шаг

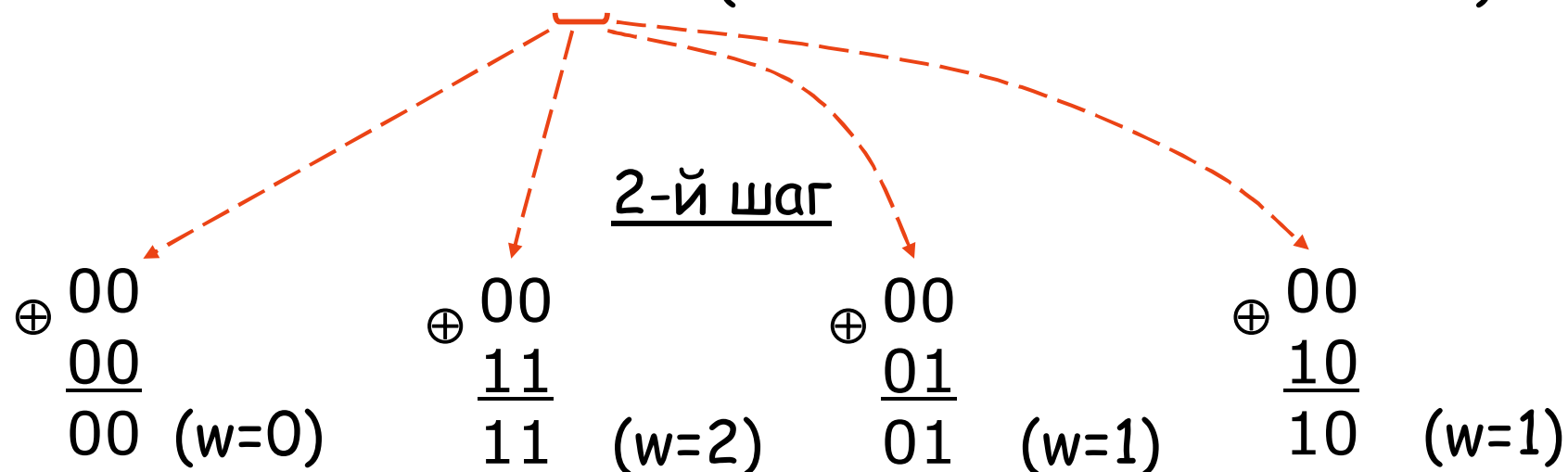
$$\begin{array}{r} \oplus \quad 11 \\ \quad \underline{00} \\ \quad 11 \end{array} \quad (w=2)$$

$$\begin{array}{r} \oplus \quad 11 \\ \quad \underline{11} \\ \quad 00 \end{array} \quad (w=0)$$

Пример декодирования

10011101 → 111011... (без ошибок)

110011 (с ошибкой в 3-м символе)





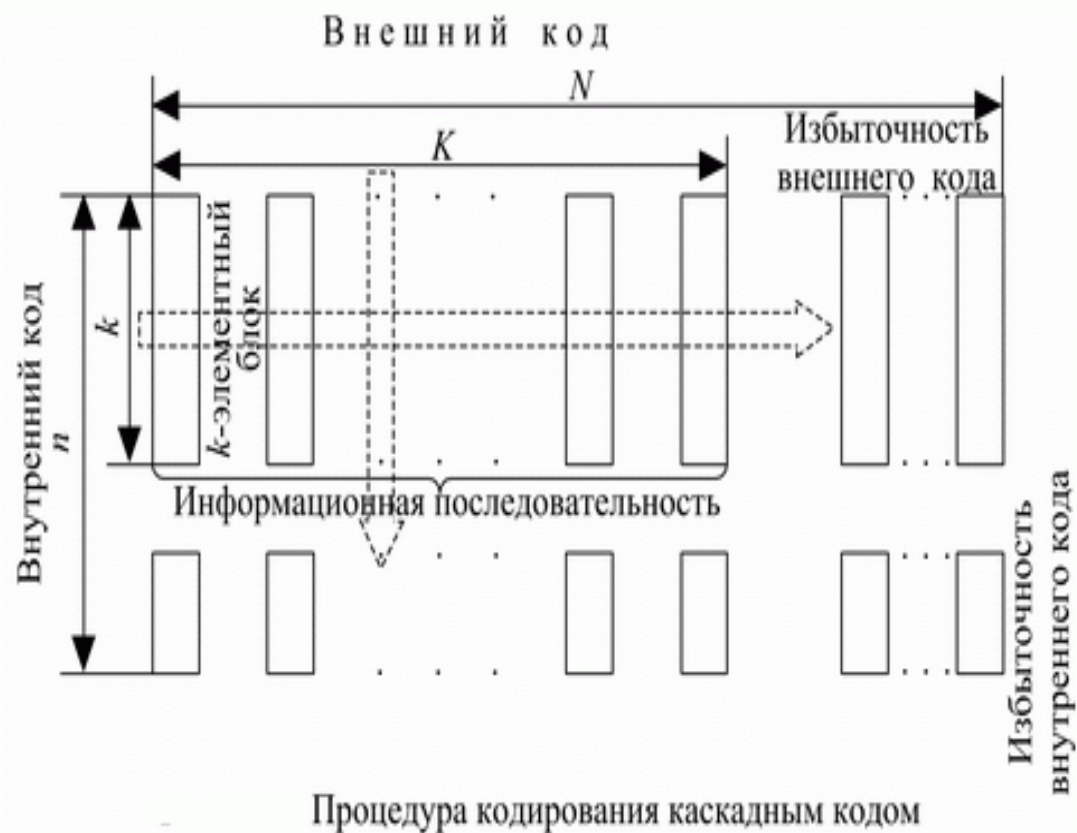
Выводы

1. Если в процессе декодирования получится не один, а несколько «выживших» путей, то это означает, что данная выборка мала для того числа ошибок, которые произошли в **КС**. Выборку следует увеличить.
2. На основе алгоритма Витерби строятся пороговые или многопороговые кодеры и декодеры*.
3. Отбрасывание «невыживших» путей дает экономию времени по сравнению с методом полного перебора.

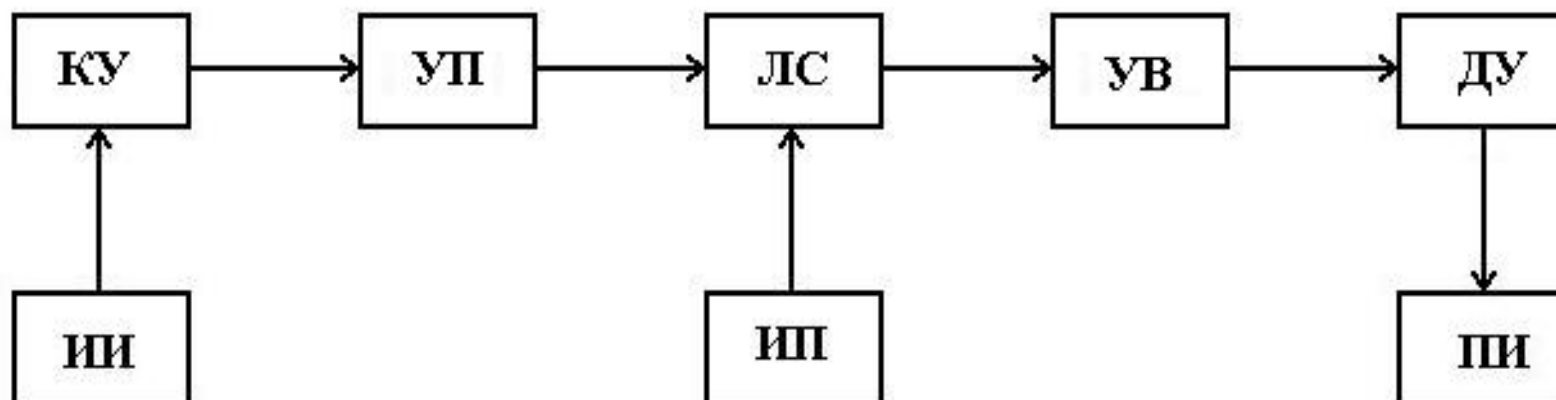
2. Принцип каскадного кодирования



Процедура кодирования



3.Перемежение информации



Периодическое перемежение

- Формируется матрица символов $s_{i,j}$:

$$\begin{pmatrix} s_{0,1} & \dots & s_{0,b} \\ s_{1,1} & \dots & s_{1,b} \\ \dots & \dots & \dots \\ s_{n-1,1} & \dots & s_{n-1,b} \end{pmatrix}$$

- Передача в КС осуществляется построчно:

$$s_{0,1} \dots s_{0,b}, s_{1,1} \dots s_{1,b}, \dots, s_{n-1,1} \dots s_{n-1,b}$$



4.Турбо-код (ТК)*

Определение

ТК - параллельный каскадный блочный систематический код, способный исправлять ошибки, возникающие при передаче цифровой информации по КСШ;

- **Используемые компонентные коды**
сверточные, Хемминга, РС, БЧХ

Различают

- **сверточные ТК** (Turbo Convolutional Codes (**TCC**))
- **блочные коды-произведения** (Turbo Product Codes (TPC))



Преимущества ТК

- Приближаются к границе Шеннона – теоретическому пределу максимальной пропускной способности КСШ;
- позволяют увеличить скорость передачи информации без увеличения мощности передатчика;
- независимость сложности декодирования от длины информационного блока



О границе Шеннона

Всякий КСШ характеризуется своей предельной скоростью передачи информации, называемой пределом Шеннона. При скоростях передачи выше этого предела неизбежны ошибки. Зато снизу к этому пределу можно подойти сколь угодно близко, обеспечивая соответствующим кодированием информации сколь угодно малую вероятность ошибки при любой зашумленности канала.



Недостатки ТК

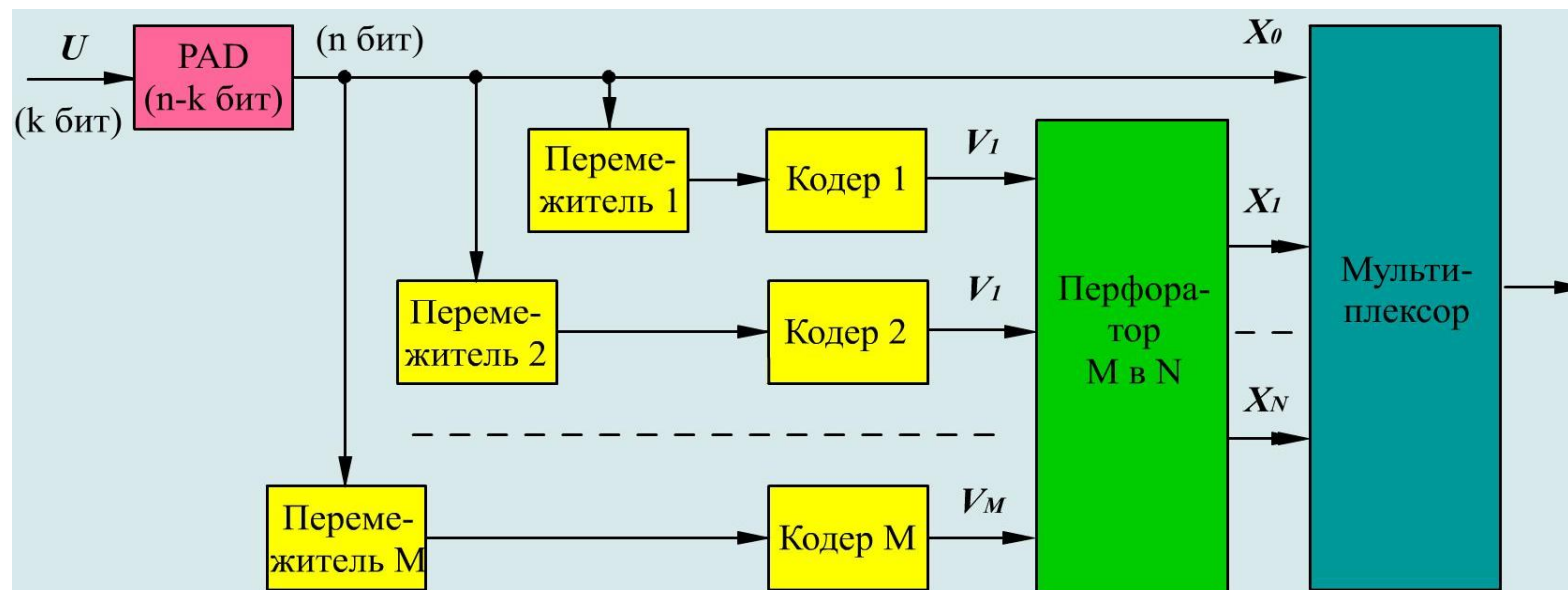
- сложность декодирования и большая задержка;
- недостаток открытого программного обеспечения



Применение ТК

- В системах спутниковой и мобильной связи;
- в системах беспроводного широкополосного доступа;
- в системах цифрового ТВ

Структурная схема М-блочного ТК



PAD (Packet Assembler/Disassembler) —
формирователь пакетов



Кодовая скорость ТК

□ без перфорации —
$$R = \frac{k}{n(M+1)} \quad (1)$$

□ с перфорацией —
$$R = \frac{k}{n(N+1)} \quad (2)$$

($N < M$)

□ При $k > 10000$ —
$$R = \frac{1}{N+1} \quad (3)$$



Декодирование

-
- Анализируются априорная и апостериорная вероятности;
 - На основе алгоритма максимума апостериорной вероятности (**Maximum of A-posteriori Probability** (MAP)) выносится «мягкое» решение
 - Применение MAP позволяет организовать несколько итераций (т.е. изменяя число итераций декодирования, можно адаптировать декодер к текущему состоянию канала передачи и достичь требуемой вероятности ошибки на бит)