



Исключающее ИЛИ
Контроль по четности

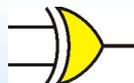
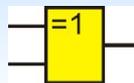
XOR
Parity check

XOR

B	A	Q
0	0	0
0	1	1
1	0	1
1	1	0

XOR

Графическое
обозначение

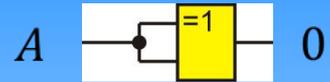


Алгебраическое
обозначение

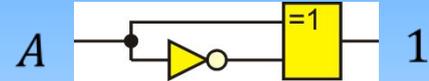
$$Q = A \oplus B$$

XOR

$$A \oplus A = 0$$



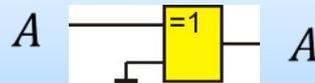
$$A \oplus \bar{A} = 1$$



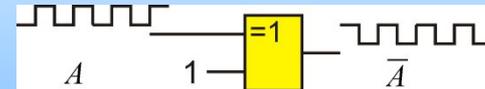
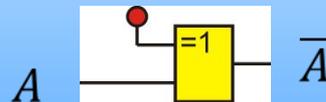
$$A \oplus B = B \oplus A$$

Управляемый инвертор

$$A \oplus 0 = A$$



$$A \oplus 1 = \bar{A}$$



XOR сравнение слов

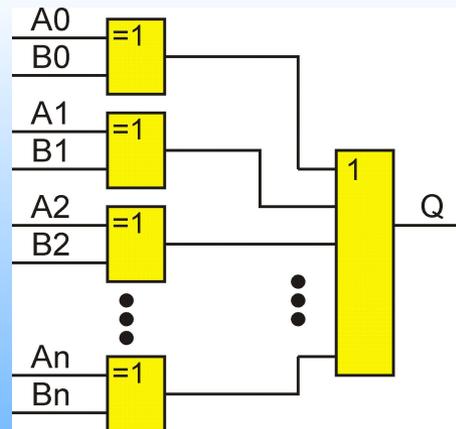
XOR

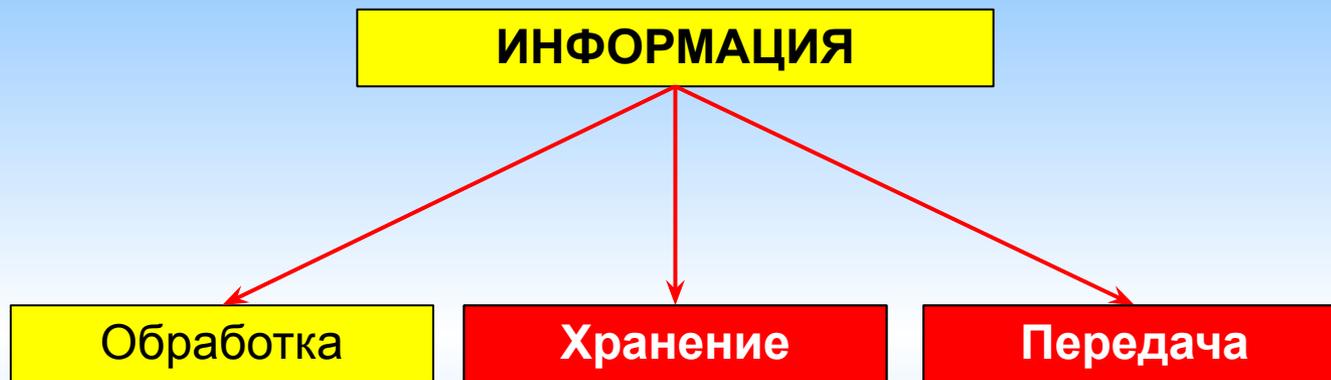
B	A	Q
0	0	0
0	1	1
1	0	1
1	1	0

$A=B \rightarrow Q=0$

$A \neq B \rightarrow Q=1$

Поразрядное сравнение





Везде могут быть ошибки

Работа над ошибками

BER (Bit Error Rate)

System	Error rate (errors/bit)
Wiring of internal circuits	10^{-15}
Memory chips	10^{-14}
Hard disk	10^{-9}
Optical drives	10^{-8}
Coaxial cable	10^{-6}
Optical disk (CD)	10^{-5}
Telephone System	10^{-4}

Table 6.2: Rates of errors for various systems.

Причины возникновения ошибок:

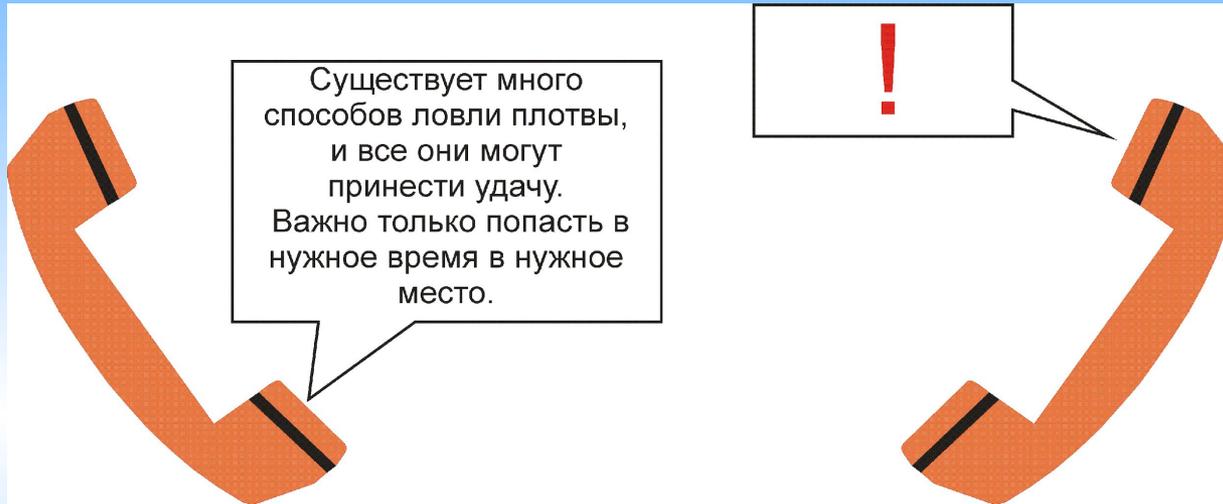
- Внешние воздействия
- Внутренние шумы
- Деградация устройств памяти и каналов связи

Работа над ошибками:

- Обнаружение
- Исправление

Работа над ошибками

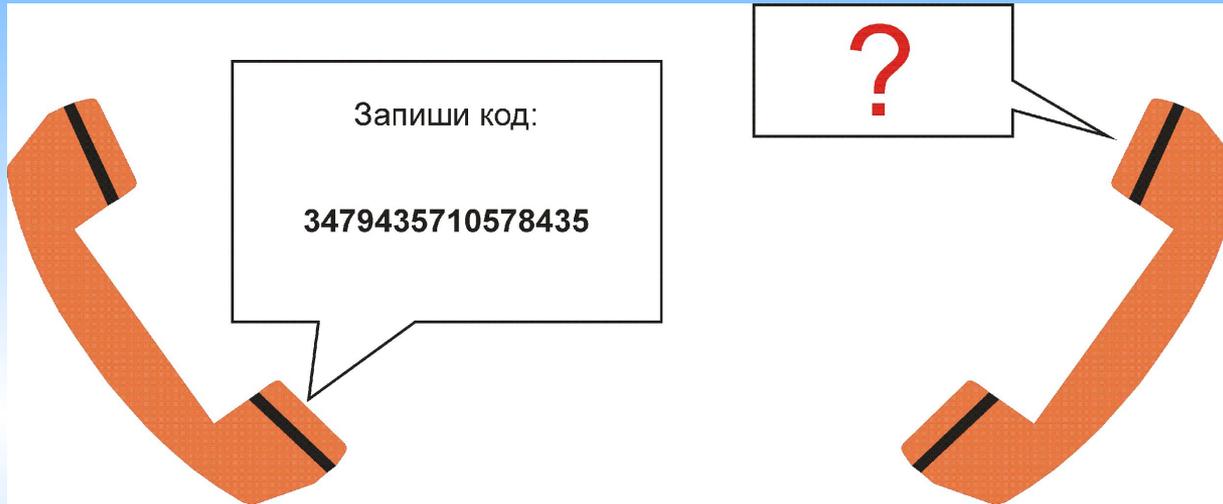
Пример



Можно говорить быстро и долго. Информация воспринимается правильно.

Работа над ошибками

Пример



Нужно говорить медленно и просить подтверждения.
Вероятность ошибок очень велика.

Почему такая разница?

Работа над ошибками

Пример исправления ошибок

Бриатские ученые установили: не важно, как
вы расставляете буквы внутри слова, главное,
чтобы первая и последняя буквы остались
неизменными, текст будет восприниматься
правильно.

Из интернетов

ИЗБЫТОЧНОСТЬ ИНФОРМАЦИИ

Работа над ошибками

Как избыточность может помочь?

Двоичный код

Избыточность отсутствует.

Любое сочетание 0 и 1 имеет право на существование.

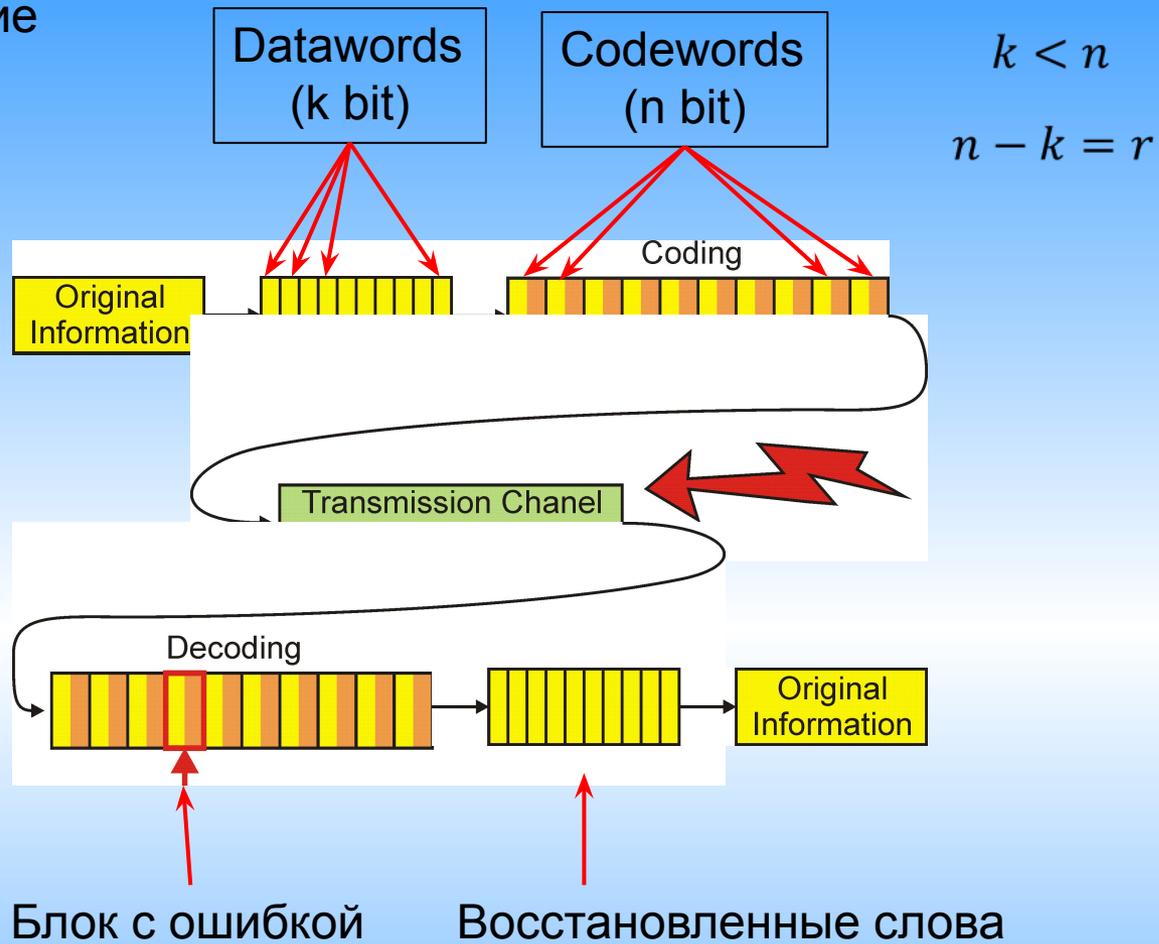
Выявить ошибку невозможно

10	h	2			
		b3	b2	b1	b0
0	0	0	0	0	0
1	1	0	0	0	1
2	2	0	0	1	0
3	3	0	0	1	1
4	4	0	1	0	0
5	5	0	1	0	1
6	6	0	1	1	0
7	7	0	1	1	1
8	8	1	0	0	0
9	9	1	0	0	1
10	A	1	0	1	0
11	B	1	0	1	1
12	C	1	1	0	0
13	D	1	1	0	1
14	E	1	1	1	0
15	F	1	1	1	1

Необходимо придумать такой код у которого не все комбинации допустимы

Кодовое слово

Блоковое кодирование



Относительная избыточность кода = r/n

Скорость кода = k/n

Кодовое слово

Блочное кодирование

1. Все сообщение разбивается на слова.
2. К словам добавляются дополнительные биты и получаются кодовые слова или блоки, как и в человеческом языке.
3. Не все сочетания 0 и 1 в кодовом слове допустимы.
4. Информация передается или хранится в таком закодированном виде.
5. Ошибка в любом из битов (нескольких битах) кодового слова должна приводить к недопустимой комбинации (некодovому слову).
6. При приеме или чтении из памяти исходная информация восстанавливается или сопровождается сообщением об ошибке.

Кодовое слово

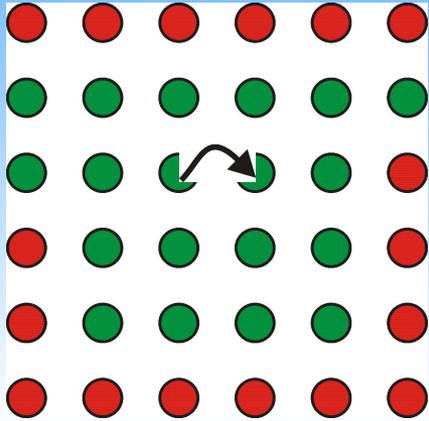
Что такое хороший код?

Требования

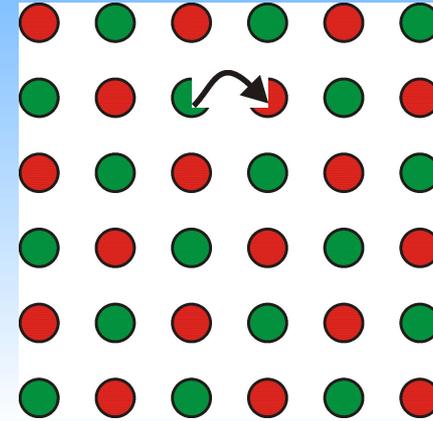
- Возможность обнаружения ошибок
- Возможность коррекции ошибок
- Минимально допустимая избыточность
- Скорость работы кодера и декодера
- Простота реализации

Кодовое слово

Что такое хороший код?



Плохой код



Код лучше



Кодовые слова (codewords)



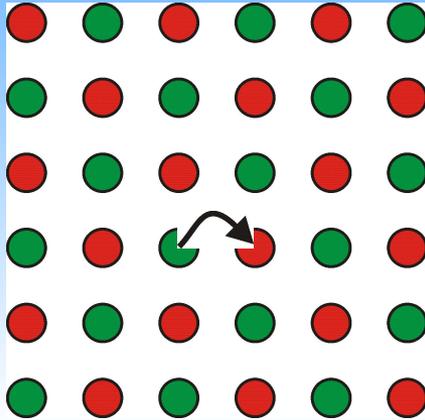
Не кодовые слова (noncodewords)



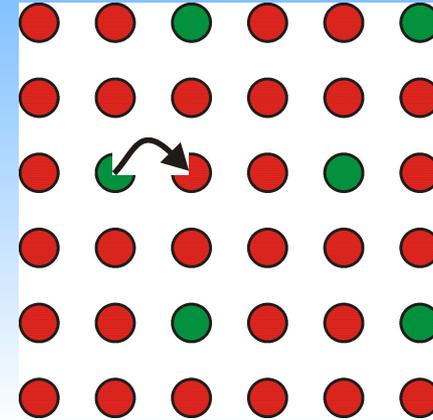
Ошибка (error)

Кодовое слово

Что такое хороший код?



Код лучше



Код еще лучше

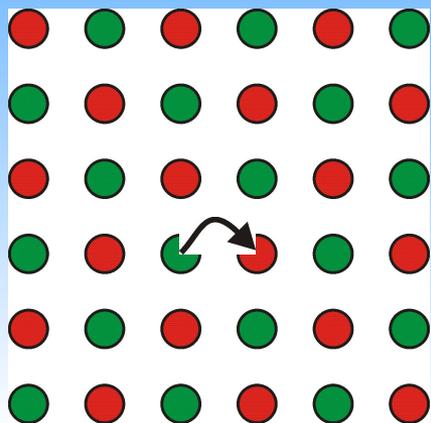
-  Кодовые слова (codewords)
-  Не кодовые слова (noncodewords)
-  Ошибка (error)

Чем выше избыточность, тем надежнее система.

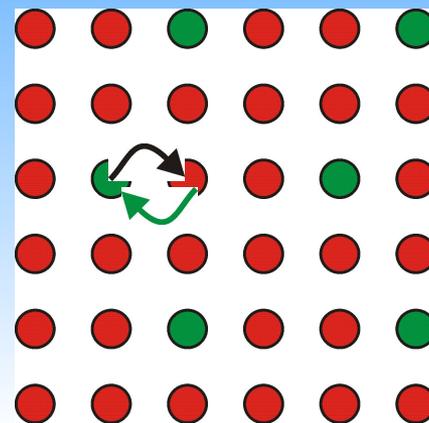
Кодовое слово

Хороший код. Исправление ошибок.

Стратегия наибольшего правдоподобия (Maximum Likelihood Decoding)

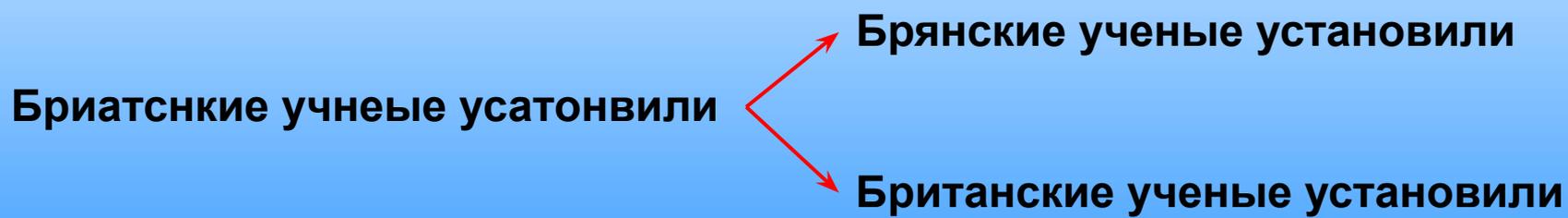


Код лучше



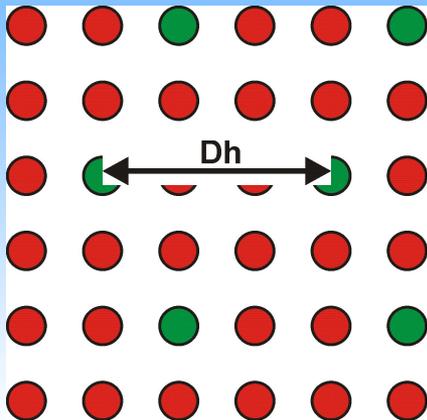
Код еще лучше

Исправление ошибок: некодовое слово превратить в ближайшее кодовое.



Кодовое расстояние

Расстояние или дистанция Хемминга



Кодовое расстояние или расстояние Хемминга D_h расстояние между **ближайшими** кодовыми комбинациями. Оно определяется числом позиций, в которых их двоичные знаки не совпадают.

Чем больше D_h , тем надежнее код, но больше избыточность.

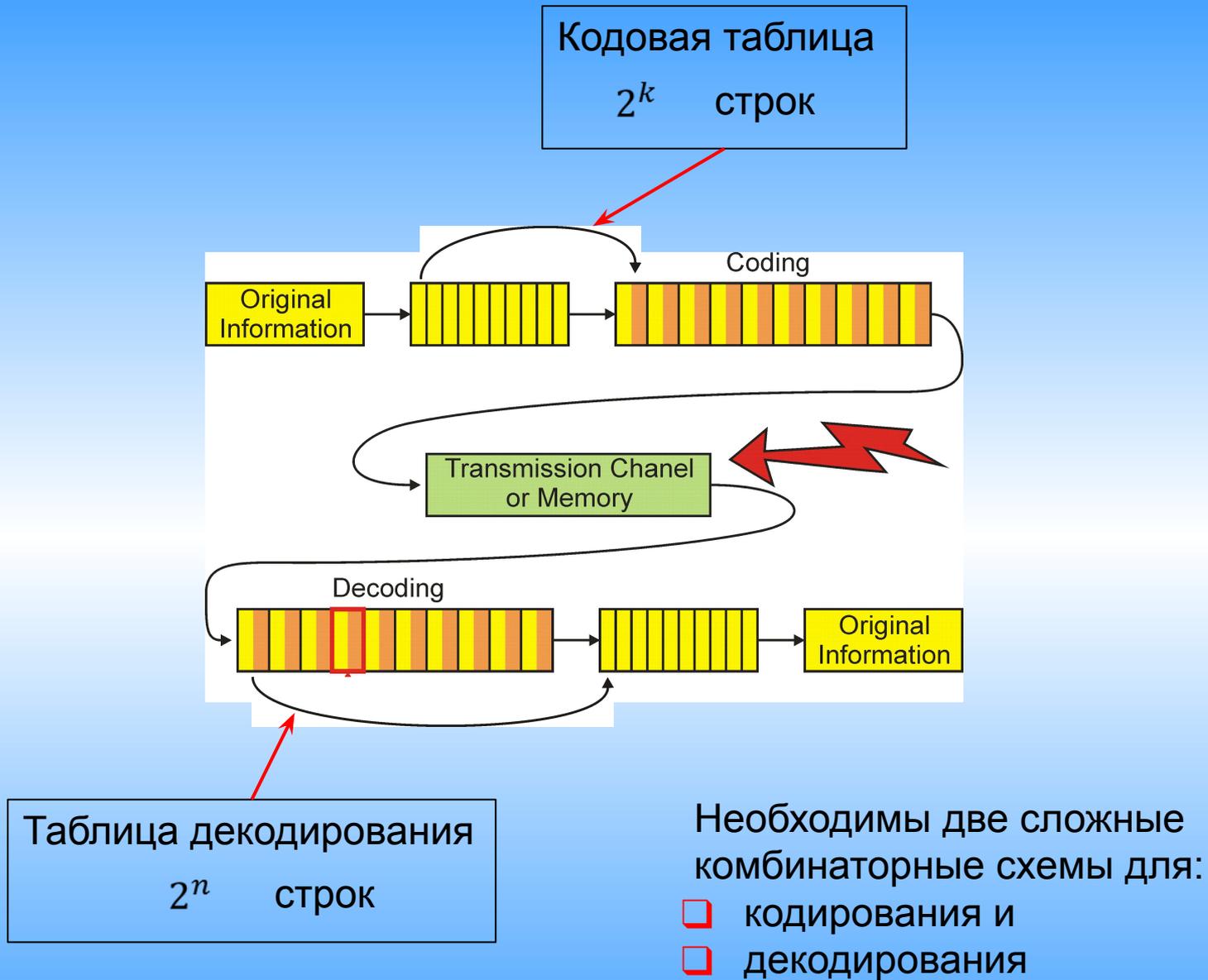
- Кодовые слова (codewords)
- Не кодовые слова (noncodewords)

У простого двоичного кода $D_h=1$.

Избыточности нет – хорошо.

Помехоустойчивости так же нет – плохо.

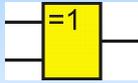
Кодирование



Простота схем кодирования и декодирования

Обнаружение ошибок. Контроль по четности.

Четность.



B	A	Q
0	0	0
0	1	1
1	0	1
1	1	0

$A=B \rightarrow Q=0$

$A \neq B \rightarrow Q=1$

Четное кол-во 1

Нечетное кол-во 1

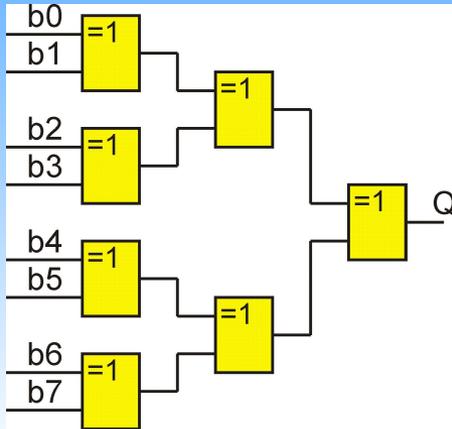
B	A	Q
0	0	0
0	1	1
1	0	1
1	1	0

Четное кол-во 1 $\rightarrow Q=0$

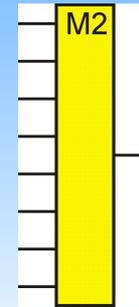
Нечетное кол-во 1 $\rightarrow Q=1$

Обнаружение ошибок. Контроль по четности.

Сумматор по модулю 2.



=



Четное кол-во 1 \rightarrow $Q=0$
Нечетное кол-во 1 \rightarrow $Q=1$

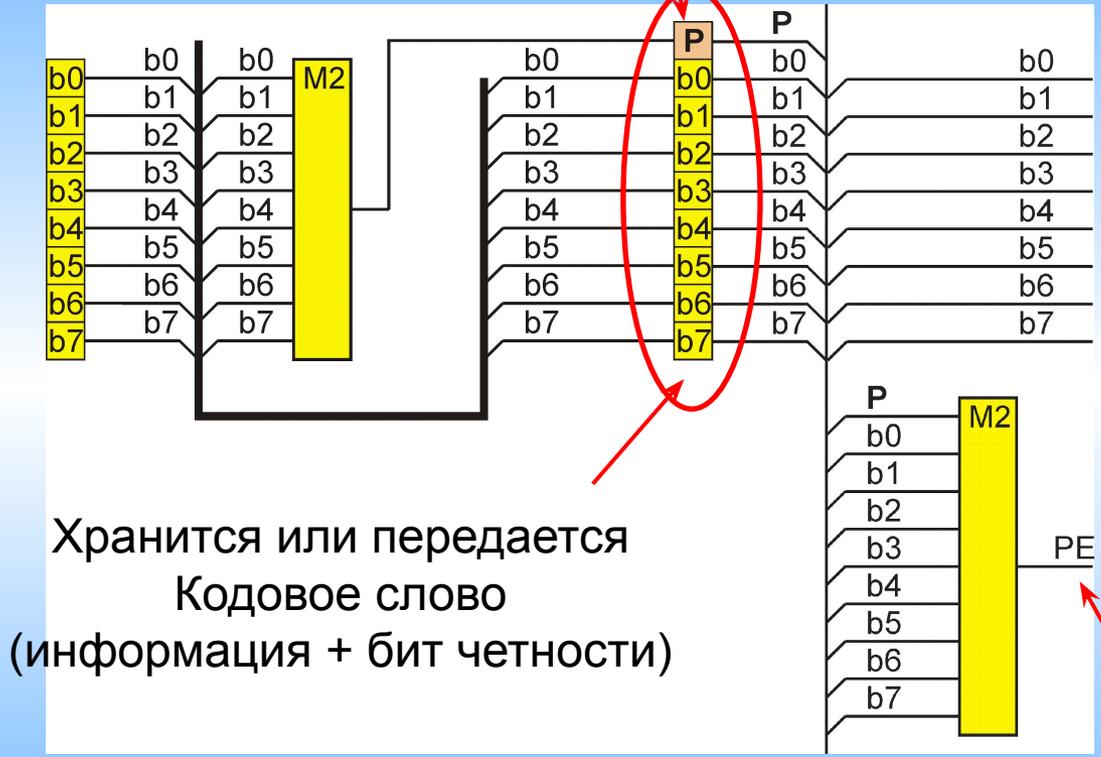
Сумматор по модулю 2
(сумматор без переносов)

Обнаружение ошибок. Контроль по четности.

Добавочный бит четности

Чтение (прием)
кодового слова

Исходная
Информация
(байт)



Хранится или передается
Кодовое слово
(информация + бит четности)

Parity Error

Кодовое слово ВСЕГДА имеет четное количество 1

Синдром ошибки

Обнаружение ошибок. Контроль по четности.



- Кодовые слова (codewords)
- Не кодовые слова (noncodewords)

Дистанция Хемминга $D_h=2$.

Синдром ошибки возникает при ошибке в одном бите.

Двойная ошибка останется незамеченной.

Более точно:
Нечетное количество ошибок будет замечено
Нечетное – нет.

Обнаружение ошибок. Вероятность многократных ошибок.

Синдром ошибки возникает при ошибке в одном бите.

Двойная ошибка останется незамеченной.

System	Error rate (errors/bit)
Wiring of internal circuits	10^{-15}
Memory chips	10^{-14}
Hard disk	10^{-9}
Optical drives	10^{-8}
Coaxial cable	10^{-6}
Optical disk (CD)	10^{-5}
Telephone System	10^{-4}

Table 6.2: Rates of errors for various systems.

Достаточно
условная таблица

Для двух независимых событий

Вероятность двойного события = (вер. события 1) X (вер. события 2)

$$P(1 + 2) = P(1) \times P(2)$$

$$P(\text{одинарная}) = 10^{-5}$$

$$P(\text{двойная}) = 10^{-10}$$

Контроль по четности. Длина слова.



$$\text{Избыточность} = 1/9 \approx 11\%$$

Вероятность одиночной битовой ошибки = $9P$



$$\text{Избыточность} = 1/17 \approx 6\%$$

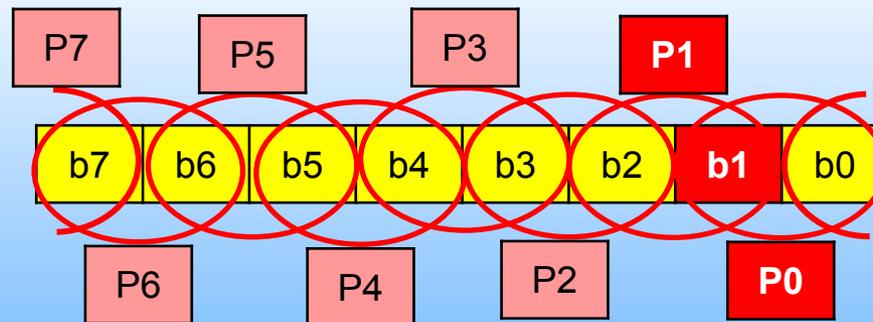
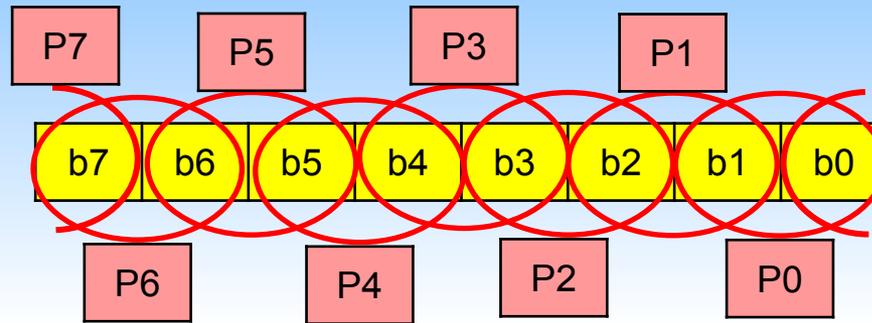
Вероятность одиночной битовой ошибки = $17P$

Контроль по четности. Исправление ошибок.

Идея. Перекрывание (Overlapping).



Простейший контроль по четности



Исправить = инвертировать!

Избыточность = 50%

Контроль по четности. Исправление ошибок.

Идея. Перекрытие (Overlapping).

dataword $k = 8$



codeword $n = 16$



Исправляет **любые** одинарные ошибки

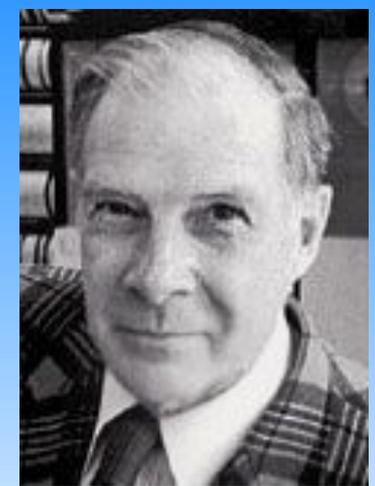
Избыточность (для любой разрядности данных k) = 50%

Это лучше, чем просто повторить передачу

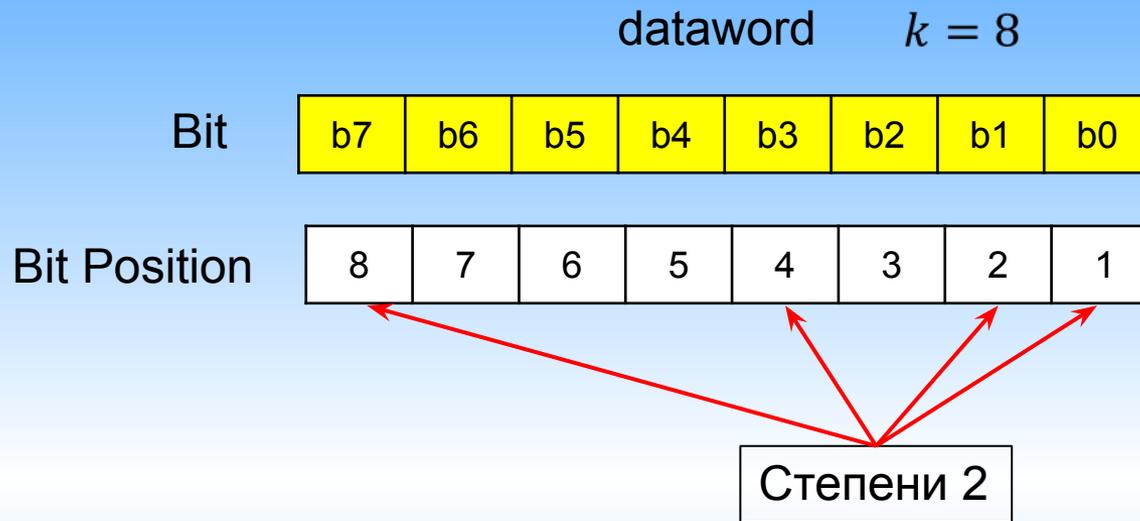
$D_h=3$

Исправляющие коды

Код Хемминга. Идея.



Richard W. Hamming
1915-1998



Опять контроль по четности

12	11	10	9	8	7	6	5	4	3	2	1
b7	b6	b5	b4	P3	b3	b2	b1	P2	b0	P1	P0

Вначале в позиции битов контроля по четности записываются 0

b7	b6	b5	b4	0	b3	b2	b1	0	b0	0	0
----	----	----	----	---	----	----	----	---	----	---	---

Исправляющие коды

Код Хемминга. Формирование кода.

Вычисляются биты контроля по четности.

Bit Position	12	11	10	9	8	7	6	5	4	3	2	1
codeword	b7	b6	b5	b4	0	b3	b2	b1	0	b0	0	0
P0= ⊕		b6		b4		b3		b1		b0		0
P1= ⊕		b6	b5			b3	b2			b0	0	
P2= ⊕	b7					b3	b2	b1	0			
P3= ⊕	b7	b6	b5	b4	0							

Исправляющие коды

Код Хемминга. Формирование кода.

Bit Position	12	11	10	9	8	7	6	5	4	3	2	1
codeword	b7	b6	b5	b4	P3	b3	b2	b1	P2	b0	P1	P0

Это кодовое слово хранится или передается.

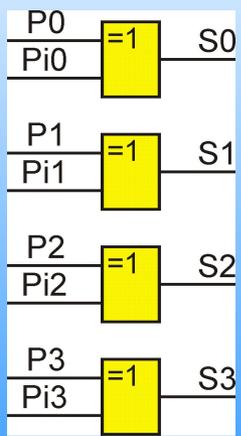
Исправляющие коды

Код Хемминга. Прием или чтение из памяти.

Ошибка

Bit Position	12	11	10	9	8	7	6	5	4	3	2	1
codeword	b7	b6	b5	b4	P3	b3	b2	b1	P2	b0	P1	P0
Pi0= ⊕		b6		b4		b3		b1		b0		P0
Pi1= ⊕		b6	b5			b3	b2			b0	P1	
Pi2= ⊕	b7					b3	b2	b1	P2			
Pi3= ⊕	b7	b6	b5	b4	P3							

$P_{i0} \neq P_0$
 $P_{i1} \neq P_1$
 $P_{i2} \neq P_2$
 $P_{i3} = P_3$



Синдром

S3	S2	S1	S0
0	1	1	1

7

Ошибка в позиции 7

Исправляющие коды

Код Хемминга. Параметры.

$D_h=3$

Избыточность (bit)



Исправляет любую одиночную ошибку

Избыточность %



Работа над ошибками. Исправление ошибок.

Двухмерный контроль по четности. Идея.
Two-Dimensional Parity Check

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
-----	-----	-----	-----	-----	-----	----	----	----	----	----	----	----	----	----	----

b15	b14	b13	b12	Pr3
b11	b10	b9	b8	Pr2
b7	b6	b5	b4	Pr1
b3	b2	b1	b0	Pr0
Pc3	Pc2	Pc1	Pc0	

Биты четности по строкам

Биты четности по столбцам

$$\text{Избыточность} = \frac{8}{24} \approx 33\%$$

Работа над ошибками. Исправление ошибок.

Двухмерный контроль по четности. Идея.
Two-Dimensional Parity Check

b15	b14	b13	b12	Pr3	
b11	b10	b9	b8	Pr2	
b7	b6	b5	b4	Pr1	←
b3	b2	b1	b0	Pr0	
Pc3	Pc2	Pc1	Pc0		

↑

Одиночную ошибку исправляет

Работа над ошибками. Исправление ошибок.

Двухмерный контроль по четности. Идея.
Two-Dimensional Parity Check

b15	b14	b13	b12	Pr3
b11	b10	b9	b8	Pr2
b7	b6	b5	b4	Pr1
b3	b2	b1	b0	Pr0
Pc3	Pc2	Pc1	Pc0	

Двойную ошибку фиксирует

b15	b14	b13	b12	Pr3
b11	b10	b9	b8	Pr2
b7	b6	b5	b4	Pr1
b3	b2	b1	b0	Pr0
Pc3	Pc2	Pc1	Pc0	

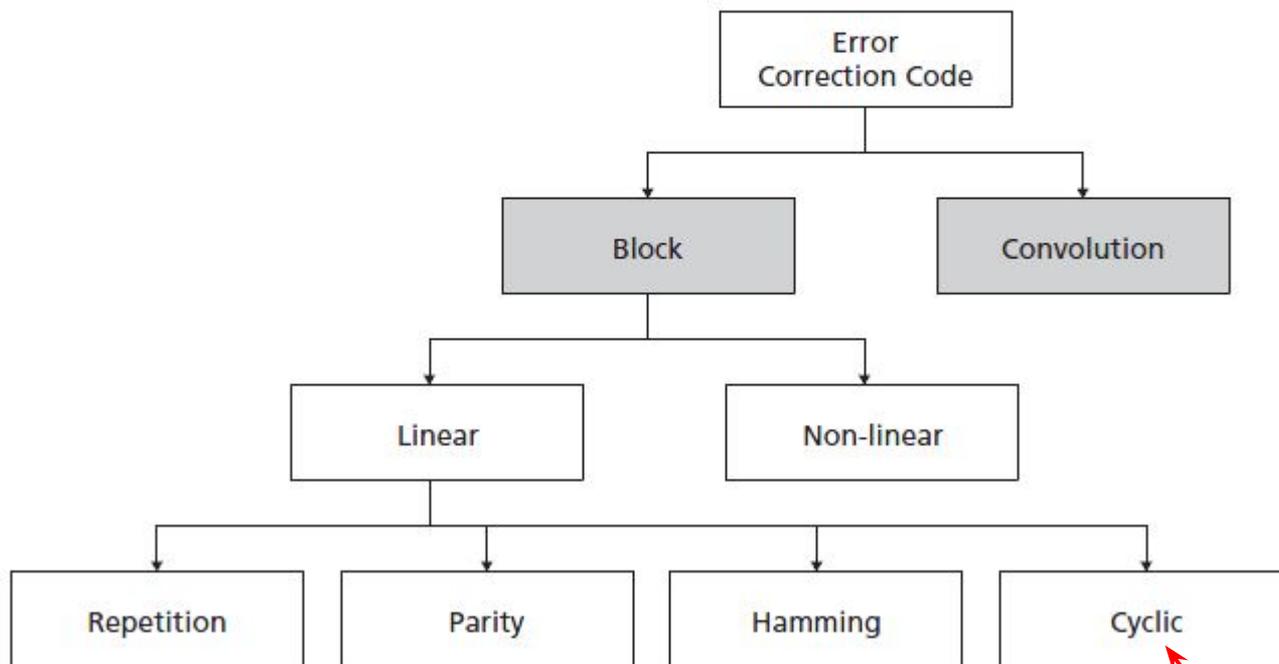
Тройную ошибку фиксирует

b15	b14	b13	b12	Pr3
b11	b10	b9	b8	Pr2
b7	b6	b5	b4	Pr1
b3	b2	b1	b0	Pr0
Pc3	Pc2	Pc1	Pc0	

Четвертную ошибку не обнаруживает



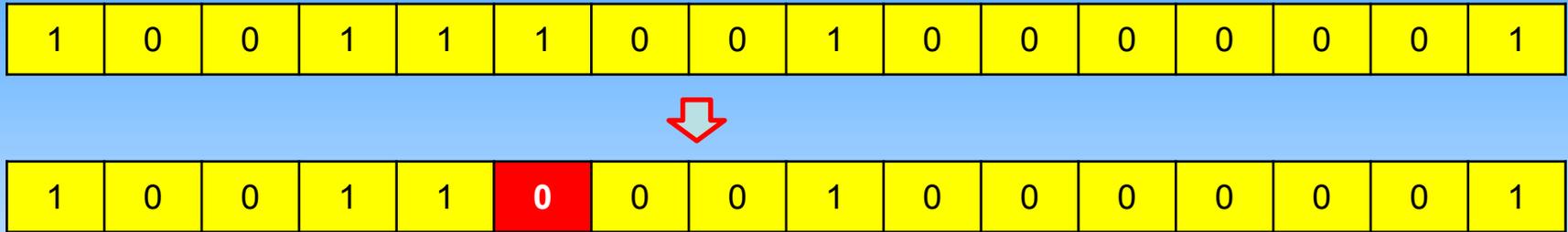
Figure 1: Block Codes



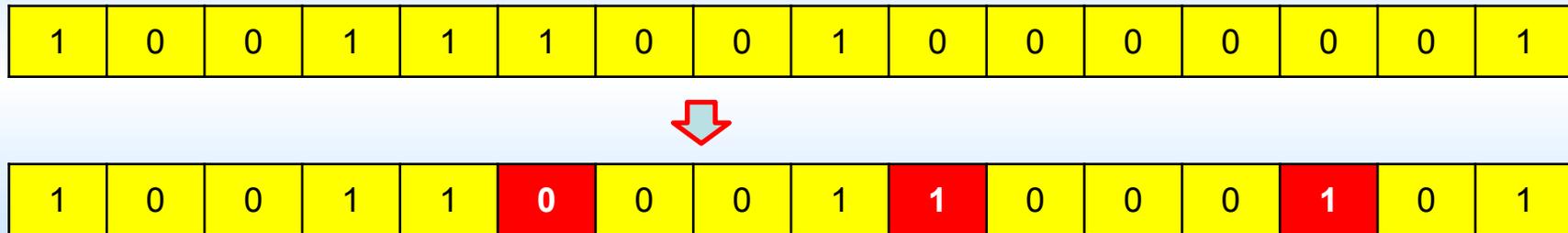
ЭТО ПОТОМ

Types of Errors

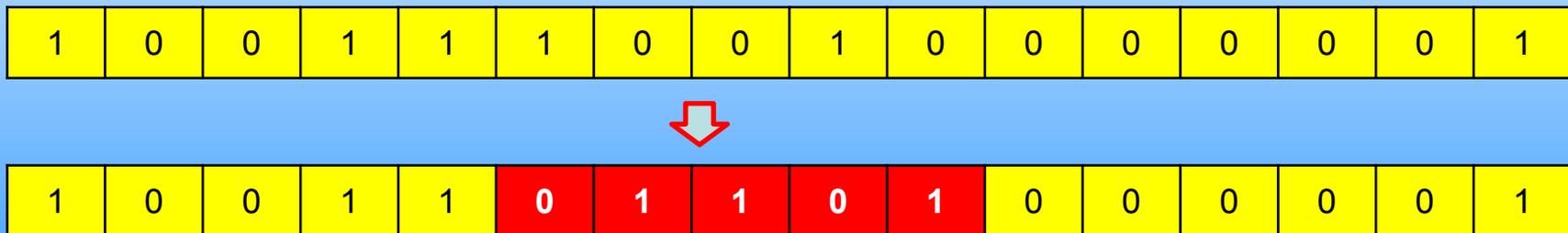
Single-Bit Error:



Multiple-Bit Error:



Burst Error:



Это ошибки, имеющие одну причину

Burst Error

Пример

Burst Error:

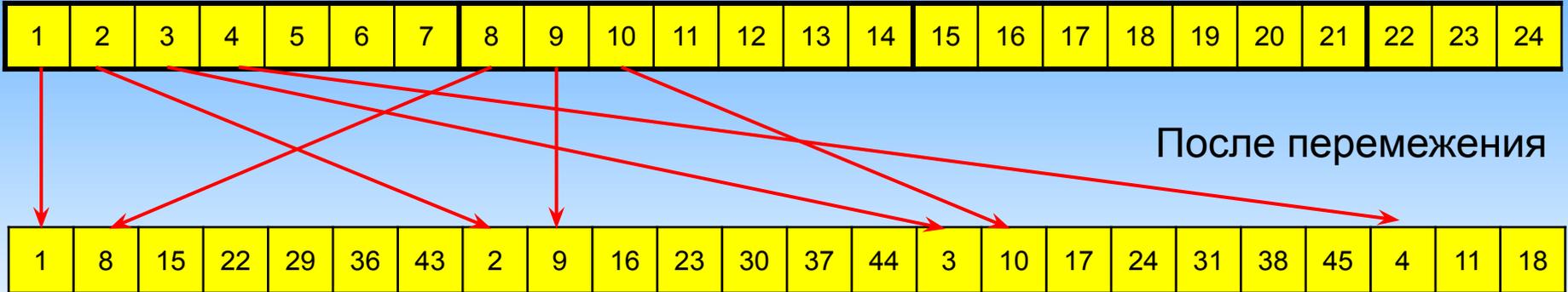


Как исправить?

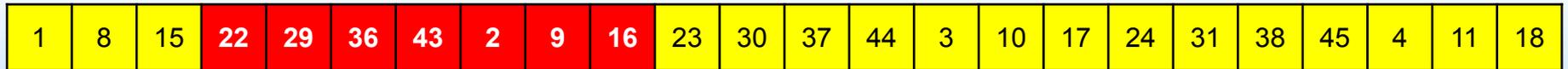
Надо превратить серию последовательных ошибок в набор одиночных.

Interleaving

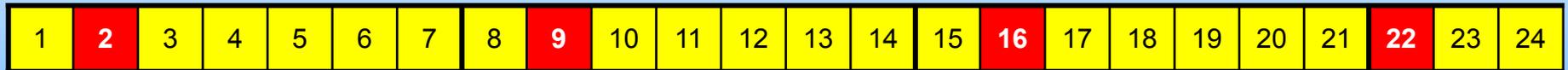
Исходная кодовая последовательность. Хемминг (4,3)



Burst Error



Восстановленная кодовая последовательность с одиночными ошибками



Это легко исправляется