Лекция 5

Тема: помехоустойчивое кодирование

1. Связь корректирующей способности кода с кодовым расстоянием
2. Простейшие помехоустойчивые коды
3. Построение двоичного группового кода
4. Методы обнаружения и исправления ошибок
5. Код Хэминга (с d_{min}=3 и с d_{min}=4)

1. Связь корректирующей способности кода с кодовым расстоянием

- В помехоустойчивых кодах вводится искусственная избыточность, т.е. к k информационным символам добавляется некоторое количество проверочных символов r, которые не несут информации, но позволяют обнаружить или исправить ошибку в принятом сообщении. При блочном кодировании* $n=k+r$.

Связь корректирующей способности кода с кодовым расстоянием

- Рассмотрим связь между d_{min} и корректирующей способностью кода на примере равномерного двоичного кода.

1. Общее число кодовых комбинаций

$$N=m^n=2^n \tag{1}$$

2. Общее число разрешенных комбинаций

$$N_p=2^k \tag{2}$$

3. Общее число запрещенных комбинаций

$$N_з=2^n - 2^k \tag{3}$$

Связь корректирующей способности кода с кодовым расстоянием

- Обнаружение ошибок основано на том, что при искажении в КС разрешенная кодовая комбинация переходит в запрещенную.
- В КСШ могут осуществляться следующие переходы

1. Разрешенная комбинация переходит в произвольную. Число таких переходов

$$N_p \cdot N = 2^k \cdot 2^n \qquad (4)$$

Связь корректирующей способности кода с кодовым расстоянием

2. Разрешенная комбинация переходит в любую другую разрешенную*. Число таких переходов

$$N_p \cdot (N_p - 1) \qquad (5)$$

Доля необнаруживаемых кодом ошибок

$$N_p \cdot (N_p - 1) / N_p \cdot N = (N_p - 1) / N \qquad (6)$$

Связь корректирующей способности кода с кодовым расстоянием

3. Разрешенная комбинация переходит в запрещенную*. Число таких переходов

$$N_p \cdot (N - N_p) \qquad (7)$$

Доля обнаруживаемых ошибок составляет

$$K_o = N_p \cdot (N - N_p) / N_p \cdot N = 1 - N_p / N \qquad (8)$$

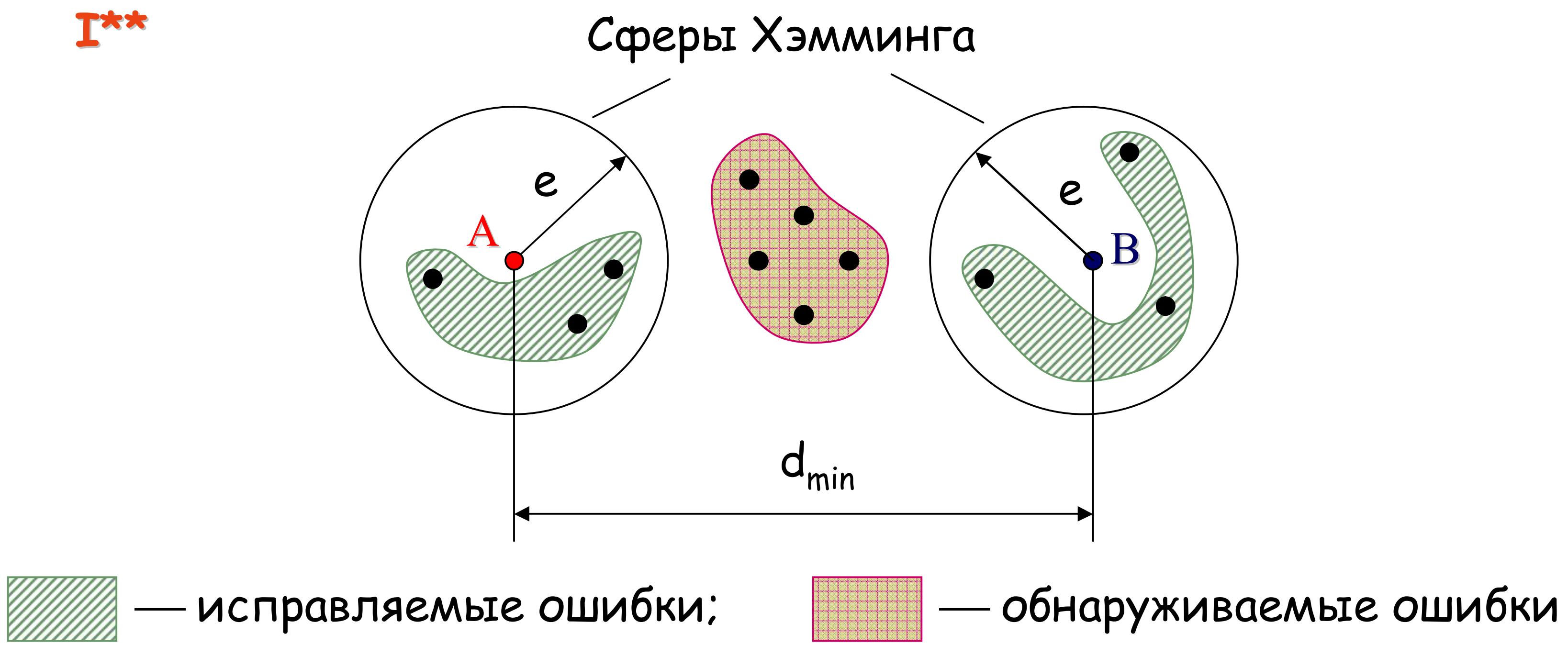
Доля исправляемых ошибок составляет**

$$K_и = (N - N_p) / N_p \cdot (N - N_p) = 1 / N_p \qquad (9)$$

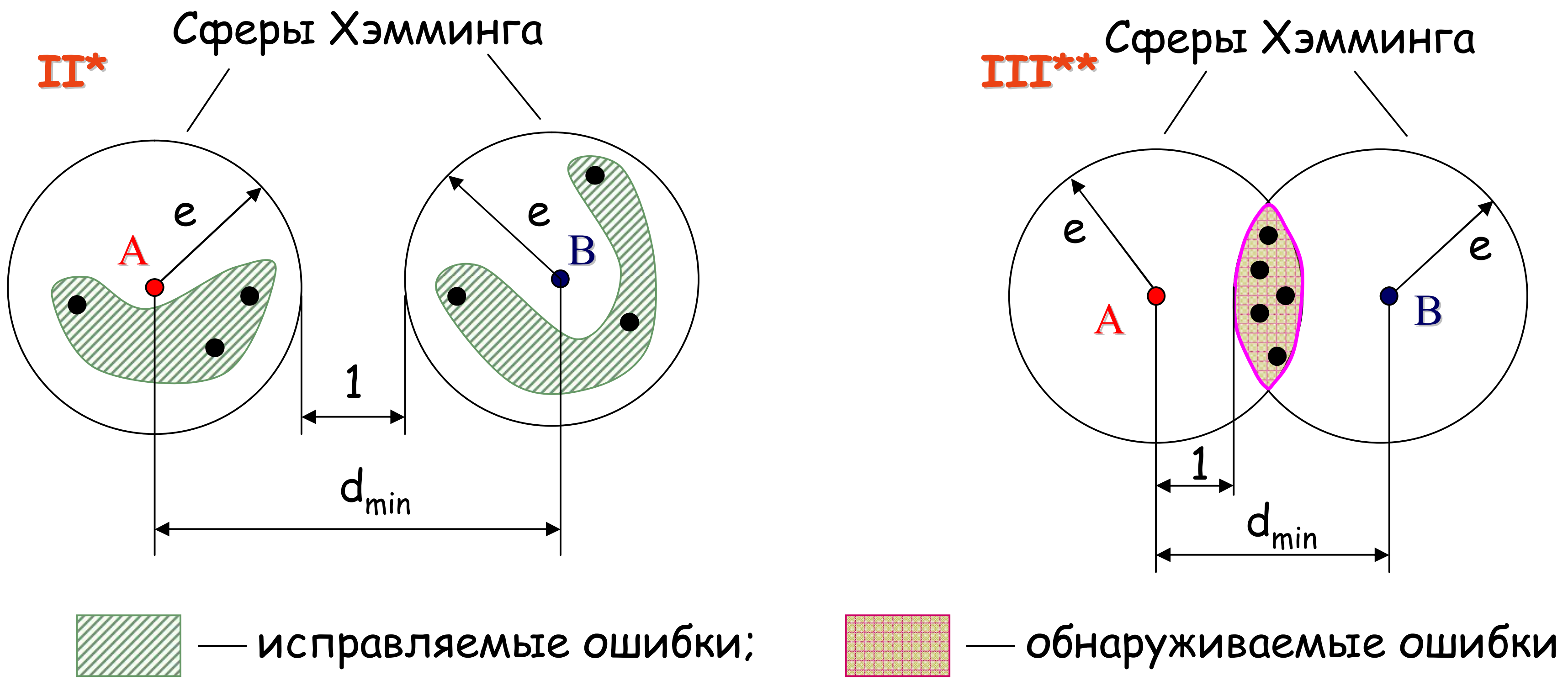
Связь корректирующей способности кода с кодовым расстоянием

- Возможность обнаружения и исправления ошибок связана с минимальным межкодовым расстоянием*.

I**



Связь корректирующей способности кода с кодовым расстоянием



Связь корректирующей способности кода с кодовым расстоянием

- Для **обнаружения** ошибок должно выполняться неравенство*

$$d_{min} \geq t_o + 1 \quad (10)$$

- Для **исправления** ошибок должно выполняться неравенство**

$$d_{min} \geq 2t_и + 1 \quad (11)$$

- Для **обнаружения** ошибок кратности t_o и **исправления** ошибок кратности $t_и$ минимальное кодовое расстояние должно составить

$$d_{min} \geq t_o + t_и + 1^{***} \quad (12)$$

Связь корректирующей способности кода с кодовым расстоянием

- Для определения связи между кратностью исправляемых ошибок, числом проверочных символов, кодовым расстоянием и длиной блока используют границы соответствия

$$r \geq \log_2 \left(1 + \sum_{i=1}^{\frac{(d_{min}-1)}{2}} C_n^i \right);$$

[Хэмминга] (13)

$$r \geq \log_2 \left(1 + \sum_{i=1}^{(d_{min}-1)} C_n^i \right);$$

[Варшамова-Гилберта] (14)

Связь корректирующей способности кода с кодовым расстоянием

- В выражениях (13) и (14) применены следующие обозначения

  i - кратность ошибки;

  C_n^i - число сочетаний из n по i.

- Если $d_{min}=1$, то $r=0$ (т.к. $\Sigma=0$).

Связь корректирующей способности кода с кодовым расстоянием

- Экспериментально установлено, что более точное значение дает граница Варшамова-Гилберта.
- Для кодов с $d_{min}=3$ получено точное соотношение между числом проверочных символов r и длиной кода n

$$r \geq \log_2(n+1)\ . \qquad (15)$$

- Рост избыточности в помехоустойчивых кодах связан с тем, что КС могут обеспечить передачу сообщений с вероятностью поражения каждого элемента $p_э \approx 10^{-6}$ в то время, как требуемые значения вероятности правильного приема могут достигать $p_п \approx 1\text{-}10^{-12}$.

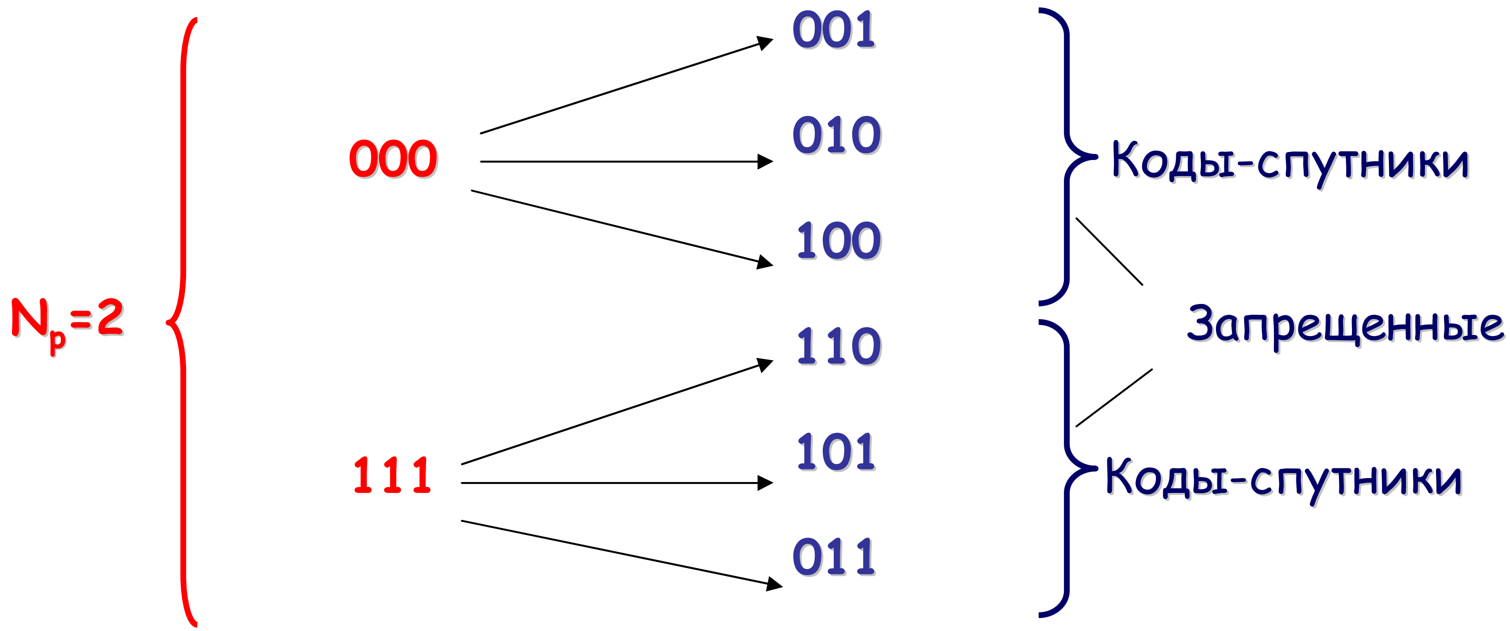
Пример 1

- Построим помехоустойчивый код с $d_{min}=2$ на основе двоичного кода ($n=3$, $N=2^3=8$)

000	000		001	
001	011	$N_p=4;$ $d_{min}=2$	010	$N_з=N-N_p=4;$ $d_{min}=2$
010	101		100	
011	110		111	
100				
101	Разрешенные с четным числом единиц*		Запрещенные с нечетным числом единиц	
110				
111				

Пример 2

- Построим помехоустойчивый код с $d_{min}=3$ на основе двоичного кода ($n=3$, $N=2^3=8$)



Построение кодов-спутников

Выводы

- Для исправления ошибок необходимо каждой разрешенной комбинации поставить в соответствие свое подмножество запрещенных кодовых комбинаций.
- Эти подмножества (коды-спутники) для разных разрешенных комбинаций не должны пересекаться.
- Для построения кода-спутника разрешенную комбинацию складывают по mod2 со всеми возможными векторами ошибок*.
- Каждой разрешенной кодовой комбинации ставится в соответствие столько запрещенных комбинаций, чтобы можно было исправить (и обнаружить) ошибки заданной кратности. Для кодов, исправляющих однократные ошибки это число равно n.

Построение кодов-спутников

(см. Пример 2)

$$\begin{array}{r}000\\ \oplus\,001\\ \hline 001\end{array}\qquad \begin{array}{r}000\\ \oplus\,010\\ \hline 010\end{array}\qquad \begin{array}{r}000\\ \oplus\,100\\ \hline 100\end{array}\quad\Big|\quad \begin{array}{r}111\\ \oplus\,001\\ \hline 110\end{array}\qquad \begin{array}{r}111\\ \oplus\,010\\ \hline 101\end{array}\qquad \begin{array}{r}111\\ \oplus\,100\\ \hline 011\end{array}$$

2. Простейшие (примитивные) помехоустойчивые коды

2.1. Код с проверкой по чету

- При любом числе информационных символов k количество проверочных символов равно 1 ($r=1$), т.е. $n=k+1$.
- Проверочный символ дополняет кодовую комбинацию до четного числа единиц.
- Проверочный разряд располагается после информационной комбинации справа.
- $d_{min}=2$.
- Проверка при декодировании сводится к суммированию по mod2 разрядов принятого кодового слова. Если сумма равна нулю, то считают, что комбинация передана без ошибок. Если сумма равна единице, то в процессе передачи произошла ошибка.
- Код позволяет обнаруживать однократные ошибки и все ошибки нечетной кратности.

Код с проверкой по чету

- Пример (k=7, n=8)

Инормационная комбинация	Количество единиц в информ. комбинации (чет/неч)	Проверочный разряд	Кодовое слово
0000000	чет	**0**	**00000000**
0000001	неч	**1**	**00000011**
0000010	неч	**1**	**00000101**
0000011	чет	**0**	**00000110**
...	...	**...**	**...**
1111100	неч	**1**	**11111001**
1111101	чет	**0**	**11111010**
1111110	чет	**0**	**11111100**
1111111	неч	**1**	**11111111**

2.2. Код с простым повторением

- Эти коды удваиваются при передаче. Проверочные символы дублируют информационные и располагаются справа от них. Например, к информационным символам 1110 добавляют проверочные символы 1110. В результате получают кодовое слово 11101110.
- Скорость кода R=0,5 (т.е. k=r).
- На приемной стороне код делится на две равные половины, которые затем складывают по mod2. Нулевая сумма соотвествует правильной передаче, а ненулевая — передаче с ошибкой

$$\oplus\begin{array}{r}1110\\1110\\\hline 0000\end{array}$$ - правильная передача

Вывод

- Помехозащищенность кода с простым повторением выше, чем у кода с проверкой на четность, т.к. он позволяет обнаруживать все ошибки, за исключением одновременных ошибок в «парных» элементах (стоящих на одинаковых позициях в первой и второй комбинациях).

2.3. Корреляционный код

- При кодировании выполняют замены: $0 \to 01$ и $1 \to 10$.
- Например,

$$1100 \to 10\ 10\ 01\ 01.$$

- На приемной стороне комбинация разбивается на пары, которые затем декодируются. Ошибочными являются комбинации, в которых встречаются пары 00 и 11.
- Скорость кода $R=0,5$ ($k=r$).

Корреляционных код

Вывод

- Характеристики корреляционного кода полностью совпадают с характеристиками кода с простым повторением.
- Различие в помехоустойчивости кодированных сигналов обнаруживается в несимметричных каналах, у которых вероятности переходов $0 \rightarrow 1$ и $1 \rightarrow 0$ различны, а также в каналах, имеющих различные вероятности искажения соседних символов по сравнению с другими возможными искажениями.

2.4. Инверсный код

Кодирование

- Если комбинация имеет четное число единиц, то она просто удваивается как в коде с простым повторением. Если комбинация имеет нечетное число единиц, то вторая (проверочная) комбинация инвертируется.
- Например,

11011 → 11011 11011;

11001 → 11001 00110.

Инверсный код

Декодирование

- На приемной стороне код делится пополам и проверяется на четность первая половина.
- Если в первой половине четное число единиц, то полученные половины складываются по mod2.
- Если в первой половине нечетное число единиц, то вторая половина инвертируется и складывается по mod2 с первой.
- Если полученные суммы равны нулю, то считают, что ошибки передачи отсутствуют.

Инверсный код

Вывод

- Инверсный код позволяет обнаруживать практически все ошибки, за исключением одновременного искажения двух, четырех и т.д. элементов в исходной комбинации и соответствующих им двух, четырех и т.д. элементов в повторяемой комбинации.

3. Построение двоичного группового кода

- Множество всех двоичных слов длины k образует абелеву (коммутативную) группу относительно поразрядного сложения.
- Пусть G — кодирующая $k \times n$–матрица, у которой есть $k \times k$–подматрица с отличным от нуля определителем, например, единичная. Тогда отображение $a \rightarrow aG$ переводит группу всех двоичных слов длины k в группу кодовых слов длины n.
- Блочный код называется групповым, если его кодовые слова образуют группу.
- Если код является групповым, то наименьшее расстояние между двумя кодовыми словами равно наименьшему весу ненулевого слова.

Построение двоичного группового кода

Определения

- **Образующей** (порождающей, производящей, генерирующей) матрицей (**ОМ**) называется матрица, при помощи которой осуществляется построение кода.
- **Проверочной** (**ПМ**) называется матрица, при помощи которой строится система проверок для обнаружения и исправления ошибок.

Построение двоичного группового кода

Правила построения ОМ

- Обозначим a_{ij} (i=1, 2, …, k; j=1, 2, …, k) информационные символы, а b_{kl} (l=1, 2, …, r) — проверочные символы блочного группового кода.
- Тогда ОМ $G_{k \times n}$ имеет k строк и n столбцов.

Построение двоичного группового кода

$$G_{k\times n} = \underbrace{\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1k} \\ a_{21} & a_{22} & \dots & a_{2k} \\ \dots & \dots & \dots & \dots \\ a_{k1} & a_{k2} & \dots & a_{kk} \end{bmatrix}}_{k}\!\!\underbrace{\begin{matrix} b_{11} & b_{12} & \dots & b_{1r} \\ b_{21} & b_{22} & \dots & b_{2r} \\ \dots & \dots & \dots & \dots \\ b_{k1} & b_{k2} & \dots & b_{kr} \end{matrix}}_{r} \qquad (1)$$

Построение двоичного группового кода

- Из (1) видно, что ОМ состоит из двух подматриц: информационной (размер $k \times k$) и проверочной (размер $k \times r$).
- Всего ОМ содержит k разрешенных комбинаций.
- Остальные 2^k-k разрешенные комбинации получают путем суммирования по mod2 строк ОМ.

Построение двоичного группового кода

- Обычно в качестве информационной подматрицы выбирают единичную матрицу, представленную в каноническом виде

$$I_{k\times k} = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 1 \end{bmatrix} \qquad (2)$$

- Тогда ОМ можно записать в следующем виде

$$G_{k\times n}=I_{k\times k},\ C_{k\times r}\ , \qquad (3)$$

где $C_{k\times r}$ — контрольная (проверочная) подматрица.

Построение двоичного группового кода

Требования к $C_{k \times r}$

1. Вес строки д.б. не менее, чем d_{min}**-1**.
2. Вес суммы по mod2 любой пары строк д.б. не менее, чем d_{min}**-2**.

Пример

Пусть $d_{min}=3$, $k=4$, $r=3$, $n=7$.

1. Построение **ОМ**

$$\begin{array}{cc} \begin{array}{ccccccc} & & & & b_1 & b_2 & b_3 \end{array} & \\ G_{4\times 7} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix} & \begin{array}{c} (1) \\ (2) \\ (3) \\ (4) \end{array} \end{array} \qquad (4)$$

номера строк

Пример

2. Вычисление разрешенных комбинаций

ОМ должна обеспечивать получение $2^4=16$ разрешенных комбинаций

[redacted]	Сочетания строк	*Количество комбинаций
	две из четырех	6
	три из четырех	4
	четыре из четырех	1

Пример (вычисление разрешенных комбинаций)

- Просуммируем, например, строки ОМ 1 и 2, а также 1, 2, 3, 4.

$(1) \oplus (2)$

$$
\begin{array}{r}
1000011 \\
\oplus\; 0100110 \\
\hline
\mathbf{1100101}
\end{array}
$$

$(1) \oplus (2) \oplus (3) \oplus (4)$

$$
\begin{array}{r}
1000011 \\
0100110 \\
\oplus\; 0010101 \\
0001111 \\
\hline
\mathbf{1111111}
\end{array}
$$

Получение проверочных уравнений по ОМ

- Для получения проверочных равенств необходимо просуммировать по mod2 все информационные разряды, индексы которых соответствуют номерам строк, содержащих единицы в соответствующем столбце проверочной подматрицы. Тогда из (4) следует

$$\begin{aligned} b_1 &= a_2 \oplus a_3 \oplus a_4 \\ b_2 &= a_1 \oplus a_2 \oplus a_4 \\ b_3 &= a_1 \oplus a_3 \oplus a_4 \end{aligned} \qquad (5)$$

Пример

- Пусть необходимо закодировать комбинацию 1011 ($a_1 = 1, a_2 = 0, a_3 = 1, a_4 = 1$). Тогда, в соответствии с (5) $b_1 = 0$, $b_2 = 0$, $b_3 = 1$, т.е. $1011 \rightarrow 1011001$.*

Проверочная матрица (ПМ)

- ПМ применяют наряду с ОМ.
- ПМ можно записать в виде (6а) или (6б)

$$H_{r\times n} = C^{T}_{r\times k}, I_{r\times r} \qquad (6а)$$

где — $C^{T}_{r\times k}$ транспонированная контрольная подматрица ОМ;

$$H_{r\times n} = \begin{bmatrix} b_{11} & b_{21} & \dots & b_{k1} & 1 & 0 & \dots & 0 \\ b_{12} & b_{22} & \dots & b_{k2} & 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ b_{1r} & b_{2r} & \dots & b_{kr} & 0 & 0 & \dots & 1 \end{bmatrix} \qquad (6б)$$

$$\underbrace{\qquad\qquad\qquad}_{k} \qquad \underbrace{\qquad\qquad\qquad}_{r}$$

Проверочная матрица (ПМ)

- Например, по ОМ (4) можно построить ПМ

$$
\begin{array}{ccccccccc}
 & & a_1 & a_2 & a_3 & a_4 & b_1 & b_2 & b_3 \\
H_{3\times 7} & = & \multicolumn{7}{c}{\begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}}
\end{array}
\qquad (7)
$$

$$
\left\{
\begin{array}{ccccccccc}
 & & & & & & b_1 & b_2 & b_3 \\
G_{4\times 7} & = & \multicolumn{7}{c}{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}}
\end{array}
\right\}
$$

Проверочная матрица (ПМ)

- Например, по ОМ (4) можно построить ПМ

$$
\begin{array}{c c}
 & \begin{array}{ccccccc} a_1 & a_2 & a_3 & a_4 & b_1 & b_2 & b_3 \end{array} \\
H_{3\times 7} = & \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}
\end{array}
\qquad (7)
$$

$$
\left\{
\begin{array}{c c}
 & \begin{array}{ccccccc} & & & & b_1 & b_2 & b_3 \end{array} \\
G_{4\times 7} = & \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}
\end{array}
\right\}
$$

Проверочная матрица (ПМ)

- По (7) можно записать проверочные равенства. Для каждого проверочного символа в соответствующей строке находят информационные символы, значения которых равны единице, и складывают их по mod2. В результате получают выражения (5).
- ОМ и ПМ связаны соотношением

$$GH^{T}=0, \qquad (8)$$

где H^{T} — транспонированная ПМ.

4. Методы обнаружения и исправления ошибок

Различают два основных метода

1. Создание кодов-спутников (КСп)
2. Определение синдрома ошибки (СОш)

1. Создание КСп предполагает, что каждой кодовой комбинации данного кода ставится в соответствие не менее n КСп. Каждый КСп получают суммированием по mod2 разрешенной кодовой комбинации с вектором ошибки.

   Недостаток метода — неоправданные затраты памяти, особенно при больших n.

Создание кодов-спутников

Пример

- Построить КСп для комбинации 11011 двоичного блочного кода, исправляющего однократные ошибки.
- Т.к. вес вектора ошибки $w_e = 1$, то получим 5 комбинаций КСп

$$\begin{array}{r} 11011 \\ \oplus\ \underline{00001} \\ 11010 \end{array} \quad \begin{array}{r} 11011 \\ \oplus\ \underline{00010} \\ 11001 \end{array} \quad \begin{array}{r} 11011 \\ \oplus\ \underline{00100} \\ 11111 \end{array} \quad \begin{array}{r} 11011 \\ \oplus\ \underline{01000} \\ 10011 \end{array} \quad \begin{array}{r} 11011 \\ \oplus\ \underline{10000} \\ 01011 \end{array}$$

- КСп хранят на приемной стороне. Прием любого КСп означает, что передавалась исходная разрешенная комбинация.

Определение синдрома ошибки

2. Синдром ошибки (СОш)

- СОш определяется на приемной стороне путем суммирования по mod2 соответствующего проверочного символа, пришедшего из КС, и проверочного символа, вычисленного по ПМ.
- Количество разрядов СОш равно r.
- Если происходит безошибочная передача, то принятые и расчетные значения проверочных символов совпадают и СОш равен нулю.

Определение синдрома ошибки

- В рассмотренном выше примере r=3, т.е. СОш имеет три разряда*

$$
\begin{aligned}
S_1 &= b_1 \oplus b_1' = a_2 \oplus a_3 \oplus a_4 \oplus b_1' \\
S_2 &= b_2 \oplus b_2' = a_1 \oplus a_2 \oplus a_4 \oplus b_2' \\
S_3 &= b_3 \oplus b_3' = a_1 \oplus a_3 \oplus a_4 \oplus b_3'
\end{aligned}
\qquad (9)
$$

$$
\begin{array}{cc}
 & \begin{array}{ccccccc} a_1 & a_2 & a_3 & a_4 & b_1 & b_2 & b_3 \end{array} \\
H_{3\times7} = & \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}
\end{array}
$$

Определение синдрома ошибки

Пример

- Пусть была передана комбинация $\begin{matrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ a_1 & a_2 & a_3 & a_4 & b_1 & b_2 & b_3 \end{matrix}$

  а принята комбинация $\begin{matrix} 0 & \mathbf{0} & 1 & 1 & 1 & 0 & 0 \\ a_1 & a_2 & a_3 & a_4 & b_1 & b_2 & b_3 \end{matrix}$

- Тогда, $S_1=0 \oplus 1 \oplus 1 \oplus 1=\mathbf{1}$;
  $S_2=0 \oplus 0 \oplus 1 \oplus 0=\mathbf{1}$;
  $S_3=0 \oplus 1 \oplus 1 \oplus 0=\mathbf{0}$.
- Комбинация 110 является комбинацией СОш второго разряда.

Определение синдрома ошибки

Выводы

1. Использование СОш позволяет указать на наличие ошибки и на разряд, в котором произошла ошибка.
2. Каждая комбинация СОш имеет r элементов. Если не учитывать нулевой синдром, то общее число комбинаций, которые может контролировать СОш, составит 2^r-1.
3. Поэтому для одиночных ошибок должно выполняться неравенство

$$2^r - 1 \geq C_n^1 \qquad (10)$$

Определение синдрома ошибки

Выводы (продолжение)

4. Для одиночной и двойной ошибок должно выполняться неравенство

$$2^r - 1 \geq C_n^1 + C_n^2 \qquad (11)$$

5. Общая формула для обнаружения и исправления ошибок кратности i

$$2^r - 1 \geq C_n^1 + C_n^2 + \ldots + C_n^i \qquad (12)$$

5. Код Хэмминга

Связь между количеством информационных и проверочных разрядов

- КХ относится к систематическим блочным кодам. Обнаружение и исправление ошибок происходит с помощью синдрома, который сразу указывает на номер искаженного разряда. Синдромы строят по проверочным матрицам (ПМ). Особенность ПМ КХ состоит в том, что в ней проверочные разряды расположены в столбцах, порядковые номера которых (если читать их слева направо) кратны степеням двойки.
- Т.к. при $d_{min}=3$ КХ исправляет однократные ошибки, то

$$2^r-1\geq n \qquad (1)$$

или

$$r\geq\log_2(n+1) . \qquad (2)$$

Код Хэмминга

Пример (d_{min}=3)

Если $k=4$, $r=3$ и $n=7$, то проверочные символы в ПМ должны располагаться на позициях с номерами 1 (2^0), 2 (2^1) и 4 (2^2).

$$\begin{array}{cc} & \begin{array}{ccccccc} a_1 & a_2 & a_3 & a_4 & a_5 & a_6 & a_7 \end{array} \\ H_{3\times 7} = & \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix} \\ & \begin{array}{ccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 \end{array} \end{array} \qquad (3)$$

Код Хэмминга

- Из (3) видно, что

$$\left.\begin{aligned} a_1 &= a_3 \oplus a_5 \oplus a_7 \\ a_2 &= a_3 \oplus a_6 \oplus a_7 \\ a_4 &= a_5 \oplus a_6 \oplus a_7 \end{aligned}\right\} \qquad (4)$$

- Символы a_1, a_2, a_4 являются проверочными, т.к. встречаются в ПМ только один раз.

Код Хэмминга

- Синдром ошибок

$$\left.\begin{aligned} S_1 &= a_1 \oplus a_3 \oplus a_5 \oplus a_7 \\ S_2 &= a_2 \oplus a_3 \oplus a_6 \oplus a_7 \\ S_3 &= a_4 \oplus a_5 \oplus a_6 \oplus a_7 \end{aligned}\right\} \qquad (5)$$

- Если при декодировании получают нулевое значение синдрома, то считают, что передача данных произошла без ошибок.
- Если значение синдрома не равно нулю, то оно указывает на номер ошибочного разряда (если читать значение синдрома снизу вверх, а номер разряда слева направо).

Код Хэмминга

Пример

- Пусть передается информационная последовательность 1011
- В соответствии с ПМ получим кодовое слово

0110011
1 2 3 4 5 <u>6</u> 7

Код Хэмминга

- Предположим, что ошибка произошла в шестом разряде, т.е. было принято слово

0110001

1 2 3 4 5 <u>6</u> 7

- Рассчитаем значение синдрома по формулам (5)

$$S_1=a_1 \oplus a_3 \oplus a_5 \oplus a_7=0\oplus1\oplus0\oplus1=0$$
$$S_2=a_2 \oplus a_3 \oplus a_6 \oplus a_7=1\oplus1\oplus0\oplus1=1$$
$$S_3=a_4 \oplus a_5 \oplus a_6 \oplus a_7=0\oplus0\oplus0\oplus1=1$$

110→6 (разряду)

Расширенный КХ (d_{min}=4)

- Расширенный КХ получают добавлением к КХ с $d_{min}=3$ одного проверочного разряда, который представляет собой результат суммирования по mod2 всех разрядов кодового слова.
- Например, в рассмотренном примере для КХ с параметрами $k=4$, $r=3$ и $n=7$ добавляют разряд a_8:

$$a_8=a_1\oplus a_2\oplus a_3\oplus a_4\oplus a_5\oplus a_6\oplus a_7 \qquad (6)$$

- Длина кода при таком преобразовании составляет 2^r разрядов, из них $(r+1)$ разрядов являются проверочными.

Код Хэмминга

- ПМ увеличивается на одну строку и один столбец

$$H_{4\times 8} = \left[\begin{array}{ccccccc|c} 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ \hline 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \end{array}\right] \qquad (7)$$

Код Хэмминга

- Дополнительный столбец в матрице $H_{4\times 8}$ получают суммированием по mod2 всех элементов проверочных строк.
- Дополнительную строку получают с учетом проверочного равенства (6)

$$a_8=a_1\oplus a_2\oplus a_3\oplus a_4\oplus a_5\oplus a_6\oplus a_7=$$

$$=\underbrace{a_3\oplus a_5\oplus a_7}_{a_1}\oplus\underbrace{a_3\oplus a_6\oplus a_7}_{a_2}\oplus a_3\oplus\underbrace{a_5\oplus a_6\oplus a_7}_{a_4}\oplus a_5\oplus a_6\oplus a_7,$$

т.е. $a_8=a_3\oplus a_5\oplus a_6$

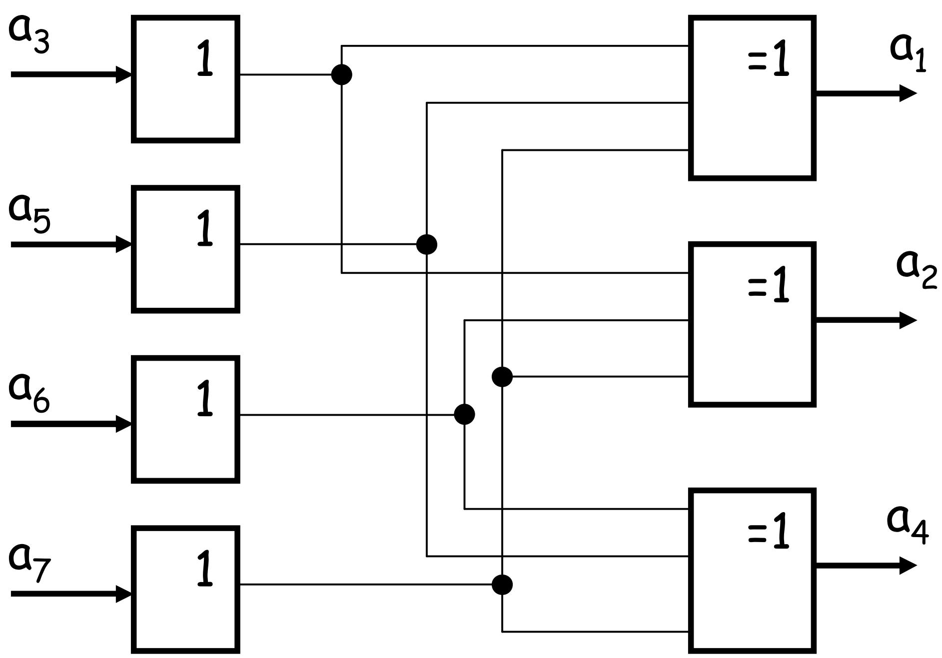
Особенности декодирования расширенного КХ

- Отличие расширенного КХ состоит в том, что для обнаружения двойных ошибок используется дополнительное проверочное равенство (6).
- На приемной стороне анализируется основной синдром и равенство (6)

Синдром	Равенство (6)	Выводы
Не равен нулю	Выполняется	Произошла двойная ошибка
Не равен нулю	Не выполняется	Произошла одиночная ошибка
Равен нулю	Выполняется	Ошибок нет
Равен нулю	Не выполняется	Произошла ошибка более высокой нечетной кратности (3, 5, …)

Техническая реализация КХ

Изобразим фрагмент схемы кодера КХ



Корректирующие возможности КХ

- Количество необнаруживаемых и неисправляемых ошибок КХ вычисляется с помощью полиномов Хэмминга.
- Для КХ с d_{min}=3

$$f(x)=\frac{1}{1+n}\left[(1+x)^n+n(1+x)^{\frac{n-1}{2}}(1-x)^{\frac{n+1}{2}}\right] \qquad (8)$$

Корректирующие возможности КХ

- Для КХ с d_{min}=4

$$f(x)=\frac{1}{2n}\left[(1+x)^{n}+n(1+x)^{\frac{n-1}{2}}(1-x)^{\frac{n+1}{2}}\right] \qquad (9)$$

- В выражениях (8) и (9)

n – длина кодового слова;

x – фиксированная переменная.

Корректирующие возможности КХ

- Коэффициенты полиномов (8) и (9) указывают на количество ошибок сообщений соответствующей кратности, которые код не может обнаружить.

Корректирующие возможности КХ

Пример

- Для КХ с параметрами $d_{min}=3$, $n=7$ получим полином Хэмминга

$$f(x)=\frac{1}{8}\left[(1+x)^7+7(1+x)^3(1-x)^4\right]=x^7+7x^4+7x^3+1$$

Выводы

1. Данный код содержит одно кодовое слово с нулевым весом (w=0), 7 кодовых слов с весом (w=3), 7 кодовых слов с весом 4 (w=4), одно кодовое слово с весом семь (w=7)*.

Корректирующие возможности КХ

Выводы (продолжение)

2. Код не может обнаружить 7 ошибок кратности 4 и 7 ошибок кратности 3.
3. Код обнаруживает все одно-, двух-, пяти- и шестикратные ошибки.

Корректирующие возможности КХ

- Для определения доли необнаруживаемых трех- или четырехкратных ошибок необходимо найти общее количество трех- или четырехкратных ошибок по формуле

$$C_n^i = \frac{n!}{i!(n-i)!}.$$

- Т.е.

$$C_7^3 = C_7^4 = \frac{7 \cdot 6 \cdot 5}{1 \cdot 2 \cdot 3} = 35.$$

Корректирующие возможности КХ

- Доля необнаруживаемых кодом трех- или четырехкратных ошибок составляет:

$$7/35=1/5,$$

т.е. 20%.

- Доля необнаруживаемых кодом семикратных ошибок составляет:

$$\frac{1}{C_7^7}=1$$

т.е. 100%.