Помехоустойчивое кодирование

Определение. ***Помехоустойчивость*** – называется способность системы осуществляющей прием информации в условиях наличия помех в линиях связи.

Определение. ***Помехой*** называется сторонние возмущение, действующее в системе, препятствующее правильному приему сигналов.

Для защиты полезной информации необходимо вводить ***избыточность*** *(смысловая, физическая, статистическая,)*

Коды, позволяющие обнаруживать и исправлять ошибки бывают двух видов: - блоковые коды; - сверточные коды

Коды, которые обеспечивают возможность обнаружения и исправления ошибки, называют ***помехоустойчивыми****.*

Код, содержащий помимо информационных еще и контрольные разряды называется ***систематическим кодом.***

Длина слова ***систематического кода*** *(n) =*
число информационных разрядов (m) + число контрольных разрядов(k)

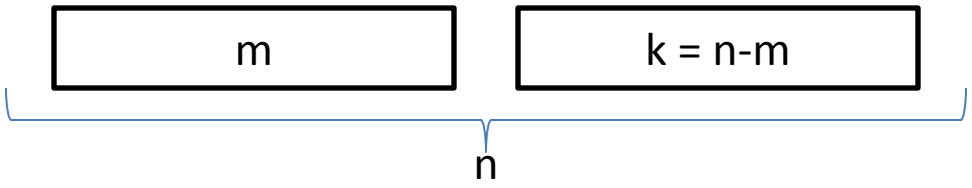


Схема системы связи

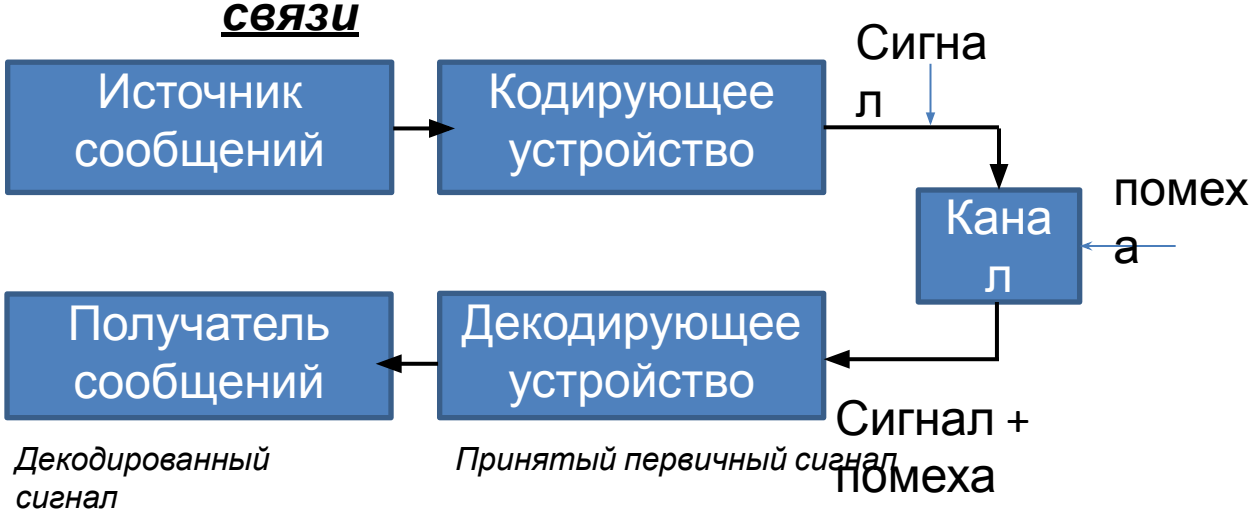


Схема работы кодирующего и декодирующего устройств

Алфавит:

$\Sigma = \{a_1, a_2, \ldots, a_n\}$

$\Sigma' = \{b_1, b_2, \ldots, b_p\}$

Сообщение:

$\alpha = a_{i1}\, a_{i2} \ldots a_{im}$, где $a_{ij} \in \Sigma$

→ Процесс кодирования

→ Процесс декодирования

Кодовая таблица

a_1	$c(a_1) = b_{11} b_{12} .. b_{1r}$
a_2	$c(a_2) = b_{21} b_{22} .. b_{2r}$
..	…
a_k	$c(a_k) = b_{k1} b_{k2} .. b_{kr}$

Закодированное сообщение:

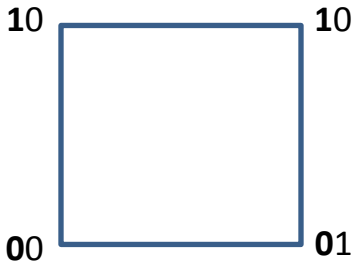
$\beta = c(a_{i1})\, c(a_{i2}) \ldots c(a_{im}) = b_{i1}\, b_{i2} \ldots b_{in}$

Геометрическая интерпретация построения кодовых слов

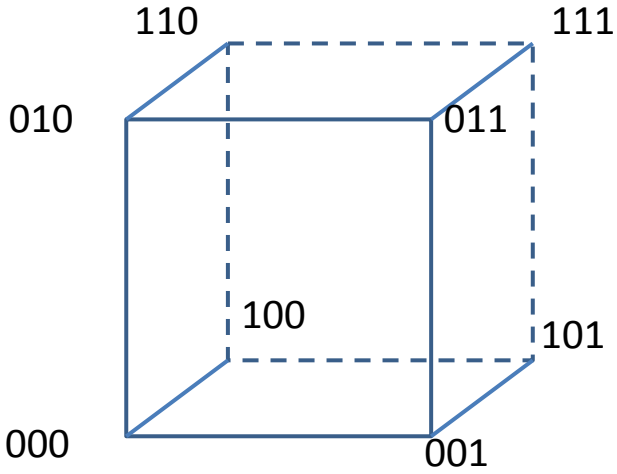
Пусть α= 101011 – вектор в пространстве или точка в пространстве B^n, где n – длина слова (блока); B={0,1}.

Всего точек в пространстве $B^n \rightarrow 2^n$.

Примеры пространств B^n разных размерностей



Пример пространства B^2



Пример пространства B^3

Пример пространства B^1

Код

– это некоторые точки пространства B^n, или некоторое подмножество пространства B^n

Замечания к процедуре построения двоичных кодовых слов:

1. На вход кодирующего устройства поступает последовательность из k информационных двоичных символов. На выходе ей соответствует последовательность из n двоичных символов, причем $n>k$.

2. Всего может быть:

- $\mathbf{2^k}$ различных входных и
- $\mathbf{2^n}$ различных выходных последовательностей.

3. Из общего числа 2^n выходных последовательностей только 2^k последовательностей соответствуют входным. Их называют ***разрешенными кодовыми комбинациями***.

4. Остальные 2^n-2^k возможных выходных последовательностей для передачи не используются. Их называют **запрещенными кодовыми комбинациями**.

Передача сообщений

При передаче сообщения от источника к получателю возможны следующие варианты передачи и получения сообщений:

1. Передача с ошибкой, ошибка обнаружена (отправили $\alpha_i \rightarrow$ получили α_j, $\forall$ **i,j=1..p, i≠j**);
2. Передача без ошибок (отправили $\alpha_i \rightarrow$ получили α_i, $\forall$ **i=1..p**);
3. Передача с ошибкой, ошибка не обнаружена (отправили $\alpha_i \rightarrow$ получили **β**, $\forall$ **i=1..p, β** - не является кодовым словом);

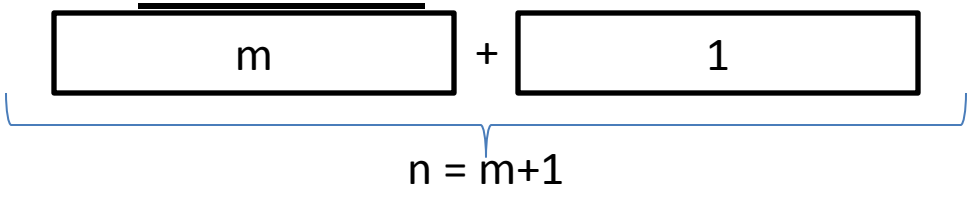
Существуют следующие способы борьбы с ошибками передачи:

- Увеличить расстояние между точками пространства B^n;
- Уменьшить количество кодовых слов.

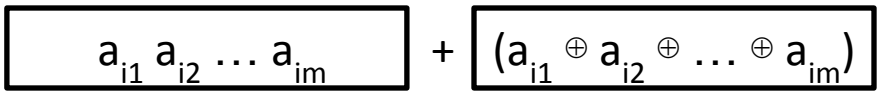
Определение. Код обнаруживает **t** ошибок, если для всякого **r ≤ t r** ошибок переводит кодовое слово в некодовое.

Примеры кодов

Код с проверкой на четность

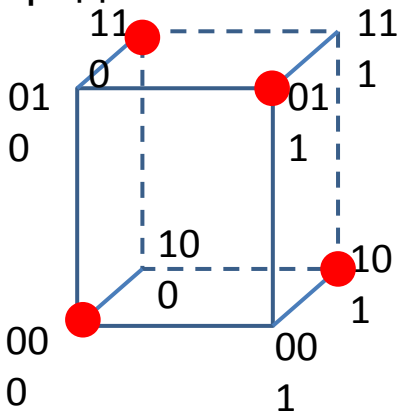


Описание: Код дополняется 1 контрольным разрядом, в который записывается 0 или 1, дополняющие сумму информационных разрядов до четного числа.



Пример кодирования двухразрядных слов:

$00 \rightarrow 000$
$01 \rightarrow 011$
$10 \rightarrow 101$
$11 \rightarrow 110$



Код - тетраэдр

Примеры кодов

Модификация кода с проверкой на четность

Кодируется слово $a_{i1}\ a_{i2} \dots a_{im}$. Длинное слово разбивается на группы по q разрядов и записываются в таблицу размера $(q \times p)$, где p = $\frac{m}{q}$. Контрольные разряды выделяются всем группам по строкам и столбцам.

a_{i1}	a_{i2}	…	a_{iq}	k_{11}
$a_{i(q+1)}$	$a_{i(q+2)}$	…	$a_{i(2q)}$	k_{12}
…	…	…	…	…
$a_{i(q*(p-1)+1)}$	$a_{i(q*(p-1)+2)}$	…	a_{im}	k_{1p}
k_{21}	k_{22}	…	k_{2q}	

Замечания.

1. Увеличение избыточности передаваемых кодов приводит к тому, что появляется возможность не только обнаружить, но и исправить ошибку.
2. Признаком отсутствия искажения в процессе приема-передачи является равенство контрольного разряда нулю

Коды с повторением

Один заданный информационный символ повторяется n раз. Это (n, 1)-код. Для него минимальное расстояние равно n, и при предположении, что большинство принятых битов совпадает с переданным информационным битом, может быть исправлено (n — 1)/2 ошибок.

0 ↔ 00000

1 ↔ 11111

Расстояние Хэмминга. Кодовое расстояние

На множестве двоичных слов длины ***m*** расстоянием ***d(a,b)*** – ***расстоянием Хэмминга*** - между двумя словами **a** и **b** называют число несовпадающих позиций этих слов.

Например: расстояние между словами **a=0110100 и b=0010101** равно **2**.

$$\begin{array}{r} 0110100 \\ \oplus\ 0010101 \\ \hline 0100001 \end{array}$$

Определение: Минимальное расстояние, взятое по всем парам кодовых разрешенных комбинаций кода, называют (***минимальным)*** ***кодовым расстоянием*** (d_{0min}).

Пример: Рассмотрим код, заданный таблицей

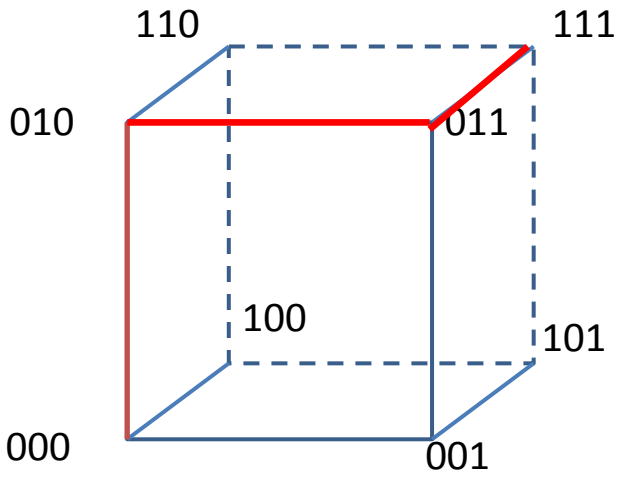
00	↔	10101	d(10101, 10010) = 3
01	↔	10010	d(10101, 01110) = 4
10	↔	01110	d(10101, 11111) = 2
11	↔	11111	d(10010, 01110) = 3
			d(10010, 11111) = 3
			d(01110, 11111) = 2

$d_{0min} = 2$ – *кодовое расстояние для данного кода*

Геометрическая интерпретация кодового расстояния и расстояния Хэмминга

При взаимно независимых ошибках наиболее вероятен переход в кодовую комбинацию, отличающуюся от данной в наименьшем числе символов.

Рассмотрим код, исправляющий ошибку. Идея построения такого кода наглядно иллюстрируется геометрической моделью трехзначного двоичного кода на все сочетания, которая представляет собой куб.



Для каждой вершины куба имеются три вершины, которые отстоят от нее на один шаг (на расстоянии одного ребра куба), еще три вершины, которые отстоят на два шага, и одна вершина — на три шага.

Расстояние между ближайшими кодовыми комбинациями *называется* ***кодовым расстоянием.***

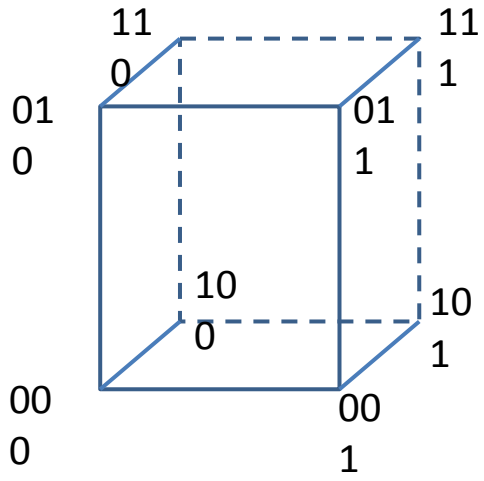
Замечание. Кодовое расстояние — параметр, характеризующий помехоустойчивость кода и заложенную в нем избыточность.

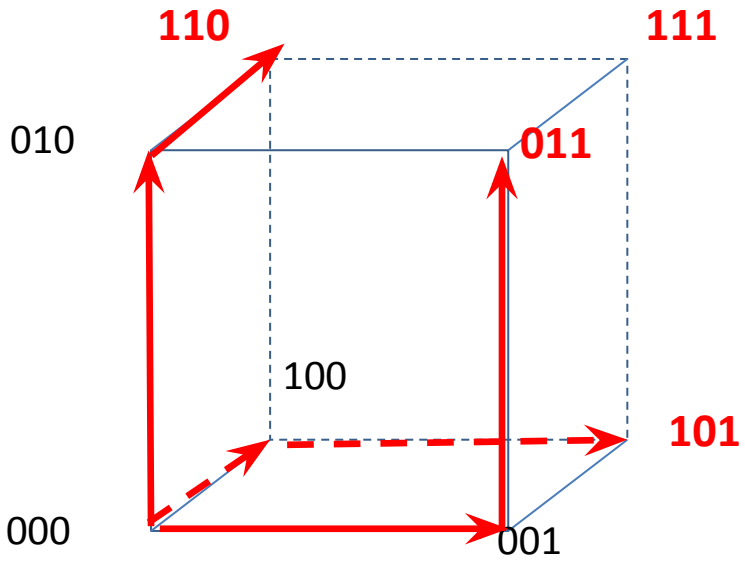
Кодовое расстояние (d)

Если кодовое расстояние *d* = 1 (избыточность в коде отсутствует), то не могут быть обнаружены даже единичные искажения, так как искаженная комбинация будет совпадать с одной из разрешенных.

Если кодовое расстояние *d = 2,* то такой код позволяет обнаруживать одиночные ошибки, так как уже есть возможность сделать так, чтобы искаженная комбинация не входила в число разрешенных.

По рисунку легко определить кодовые комбинации, обнаруживающие ошибку в комбинации **000**. Они должны отличаться друг от друга в двух символах, т. е. отстоять от точки **0** на два шага.





Как видно из рисунка ими являются 110, 011, 101. Для исправления одиночной ошибки расстояние от точки **0** следует увеличить еще на один шаг. Такая комбинация будет только одна — 111.

Для трехмерного куба корректирующие комбинации расположены на противоположных вершинах куба. Это пары 000—111, 010—101, 001—110, 011—100 (Коды-спутники).

Исправление ошибки в кодах-спутниках

Идея исправления ошибки в кодах-спутниках весьма проста. Главное, чтобы при искажении любой комбинации не могла быть образована соседняя рабочая комбинация. Процесс исправления ошибки заключается в том, что искаженная комбинация отождествляется с ближайшей разрешенной комбинацией.

Например, если передавать буквы алфавита, которым соответствуют следующие комбинации двоичного кода: *А* — 00000, *Б* — 00111 и *В* — 11100, то при искажении любого одного знака легко определить, какая комбинация была передана, так как каждая из них отличается друг от друга не меньше чем в трех символах (кодовое расстояние $d \geq 3$).

Декодирование после приема производится таким образом, что принятая кодовая комбинация отождествляется с той разрешенной, которая находится от нее на **наименьшем** кодовом расстоянии.

Такое декодирование называется декодированием ***по методу максимального правдоподобия***

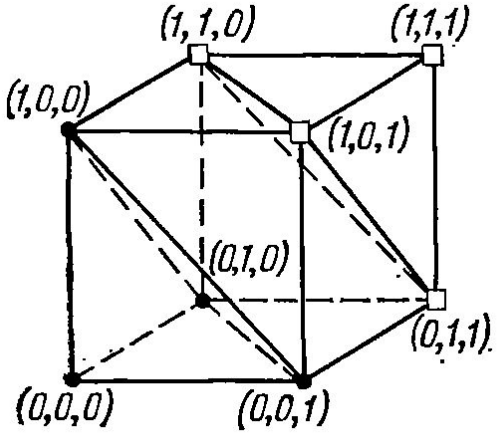
В общем случае при необходимости обнаруживать ошибки кратности до r включительно минимальное хэммингово расстояние между разрешенными кодовыми комбинациями должно быть по крайней мере на единицу больше r.

$$d_{0\ min} \geq r+1$$

В общем случае для обеспечения возможности исправления всех ошибок кратности до s включительно при декодировании по методу максимального правдоподобия, каждая из ошибок должна приводить к запрещенной комбинации, относящейся к подмножеству исходной разрешенной кодовой комбинации

Пример. Рассмотрим (2,3)–код с проверкой на четность.

Множество кодовых слов есть 000, 101, 011, 110. Минимальное расстояние между кодовыми словами равно 2. Этот код способен обнаруживать однократную ошибку.



Общее выражение для определения кодового расстояния в случае одновременного обнаружения и исправления ошибок

$$d = r + s + 1$$

, где

r — число обнаруживаемых ошибок;

s — число исправляемых ошибок;

d — минимальное количество элементов, в которых одна кодовая комбинация отличается от другой.

Если требуется определить кодовое расстояние исходя только из количества исправляемых ошибок, то применяют формулу

$$d = 2s + 1$$

Примеры кодов

Код Хэмминга

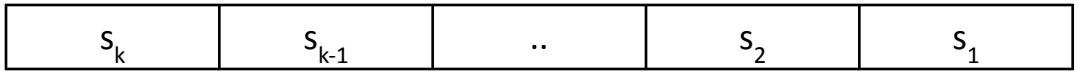
$d_{0min} = 3$

Для $\forall m \in (2^m-1, 2^{n-1}-m)$ код Хэмминга

Это систематический код, с **m** информационными и **k = (n-m)** проверочными битам. Код Хэмминга является кодом с проверкой на четность, с той лишь разницей, что эта проверка производится **k** раз.

При каждой проверке охватывается часть информационных символов и один избыточный, при этом получается один контрольный символ.

Если результат проверки дает четное число, то контрольному символу присваивается значение '**0**', если нечетное – '**1**'.



k - разрядное двоичное число

$$2^{n-k} \leq \frac{2^n}{1+n}$$

Когда при передаче кодового слова возникает одиночная ошибка, окажутся невыполненными те проверочные соотношения, в которые входит значение ошибочного разряда.

Пусть передается слово $\bar{u} = (u_1, u_2, \ldots, u_m),\ u_i \in \Sigma = \{0,1\}$

Порядок проведения кодирования и проверок

$s1 = u_1 \oplus u_3 \oplus u_5 \oplus u_7 \oplus u_9 \oplus u_{11} \oplus \ldots$

$s2 = u_2 \oplus u_3 \oplus u_6 \oplus u_7 \oplus u_{10} \oplus u_{11} \oplus \ldots$

$s3 = u_4 \oplus u_5 \oplus u_6 \oplus u_7 \oplus u_{12} \oplus u_{13} \oplus \ldots$

$s4 = u_8 \oplus u_9 \oplus u_{10} \oplus u_{11} \oplus u_{12} \oplus u_{13} \oplus \ldots$

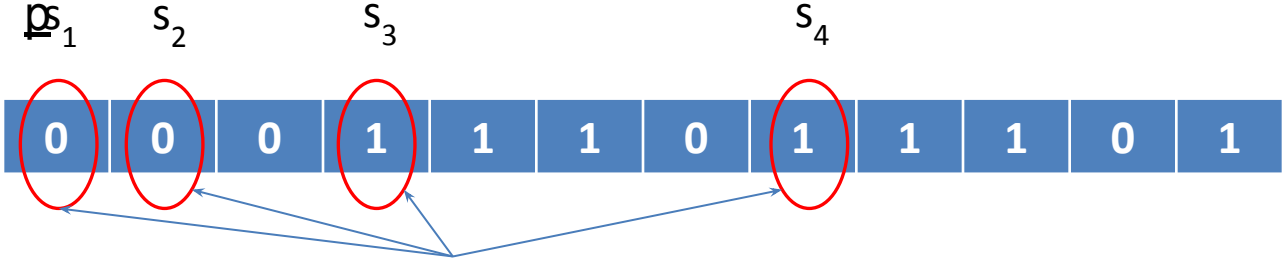
Позиции контрольных битов

№ проверок	Контролируемые биты											
1	**1**	3	5	7	9	11	13	15	17	19	21	…
2	**2**	3	6	7	10	11	14	15	18	19	22	…
3	**4**	5	6	7	12	13	14	15	20	21	22	…
4	**8**	9	10	11	12	13	14	15	24	25	26	…
5	**16**	17	18	19	20	21	22	23	24	25	26	…
6	**32**	33	34	35	36	37	38	39	40	41	42	…

Целесообразно выбирать такое размещение проверочных символов в кодовой комбинации, при которой каждый из них включается в минимальное число проверяемых групп (лучше в одну).

Приме

ps



Проверочные

Алгоритм декодирования кода Хэмминга:

1. Провести проверку всех битов чётности
2. Если все биты чётности верны, то перейти к п 5.
3. Вычислить сумму номеров всех неправильных битов чётности
4. Инвертировать содержимое бита, номер которого равен сумме, найденной в п.3
5. Исключить биты чётности , передать правильный информационный код

Избыточность кодов Хемминга для различных длин передаваемых последовательностей

Число информационных битов	Число контрольных битов	Избыточность, L
8	4	1,50
16	5	1,31
32	6	1,06

При построении кодов перед разработчиками стоит задача определения числа добавочных, корректирующих символов k, или исходя из числа информационных разрядов m, либо из общей длины кода n.

Для обнаружения и исправления **одиночной** ошибки соотношение между числом информационных разрядов $n_и$ и числом корректирующих разрядов $n_к$ должно удовлетворять следующим условиям:

$$2^k \geq n + 1$$

$$2^m \leq \frac{2^n}{n+1}$$

При этом подразумевается, что общая длина кодовой комбинации n=m+k

Для практических расчетов при определении числа контрольных разрядов кодов с минимальным кодовым расстоянием $d_{0min} = 3$ удобно пользоваться выражениями:

1. Если известна длина полной кодовой комбинации *n*

$$n_{k_{1(2)}} = [\log_2(n+1)]$$

2. Если при расчетах удобнее исходить из заданного числа информационных символов m

$$n_{k_{1(2)}} = [\log_2\{(m+1) + [\log_2(m+1)]\}]$$

Для кодов, обнаруживающих все **трехкратные ошибки** ($d_{0min} = 4$)

$$k_{1(3)} \geq 1 + \log_2(n+1)$$

$$k_{1(3)} \geq 1 + \log_2[(m+1) + \log_2(m+1)]$$

Для кодов длиной в *n* символов, исправляющих **одну или две ошибки** ($d_{0min} = 5$),

$$k_2 \geq \log_2(C_n^2 + C_n^1 + 1)$$

Для практических расчетов можно пользоваться выражением

$$k_2 = [\log_2 \frac{n^2 + n + 1}{2}]$$

Для кодов, исправляющих **три ошибки** ($d_{0min} = 7$),

$$k_2 = [\log_2 \frac{n^3 + n^2 + n + 1}{6}]$$

Для кодов, исправляющих **S ошибок** ($d_{0min} = 2S + 1$),

$$\log_2\left(C_n^s + C^{s-1}_{n} + \cdots + 1\right) < k_s < \log_2\left(C^{2s-1}_{n-1} + C^{2s-2}_{n-1} + \cdots + 1\right)$$

Выражение слева известно как нижняя граница Хэмминга, а выражение справа как верхняя граница ***Варшамова — Гильберта***.

В настоящее время разработаны десятки кодов, которые теоретически могут обнаруживать произвольное количество ошибок.