

The background of the slide features a complex, abstract design. It includes several network cables with RJ45 connectors, some of which are plugged into circular ports. Overlaid on these are concentric circles and various data-like patterns, including the word 'WWW' repeated in a circular fashion. The overall aesthetic is technical and digital, rendered in a light gray tone against a white background.

# **Кодирование сигнала**

Автор: Г.И. Осипова  
ст. группы ВТм-112

# Кодирование сигналов

---

При выборе способа кодирования нужно одновременно стремиться к достижению нескольких целей:

- минимизировать ширину спектра сигнала, полученного в результате кодирования;
- обеспечивать синхронизацию между передатчиком и приемником;
- обеспечивать устойчивость к шумам;
- обнаруживать и по возможности исправлять битовые ошибки;
- минимизировать мощность передатчика.

# Кодирование сигналов

---

Более **узкий спектр сигнала** позволяет на одной и той же линии (с одной и той же полосой пропускания) добиваться более высокой скорости передачи данных.

**Спектр сигнала** в общем случае зависит как от способа кодирования, так и от тактовой частоты передатчика.

# Кодирование сигналов

**Синхронизация передатчика и приемника** нужна для того, чтобы приемник точно знал, в какой момент времени считывать новую порцию информации с линии связи.

При передаче дискретной информации время всегда разбивается на такты, и приемник старается считать новый сигнал в середине каждого такта, то есть синхронизировать свои действия с передатчиком.



Рис. 1 Синхронизация приемника и передатчика на небольших расстояниях

# Кодирование сигналов

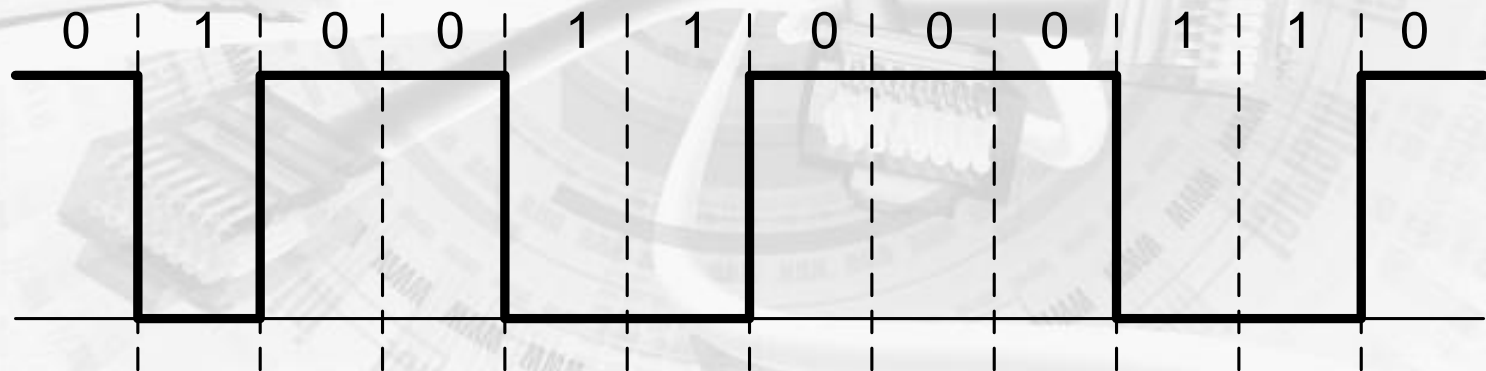
---

В сетях применяются так называемые **самосинхронизирующиеся коды**, сигналы которых несут для приемника указания о том, в какой момент времени начать распознавание очередного бита. Любой резкий перепад сигнала — фронт — может служить указанием на необходимость синхронизации приемника с передатчиком.

# Потенциальный код NRZ (Non Return to Zero)

---

- Единичный бит передается в пределах такта, уровень не меняется. Положительный перепад означает переход из 0 к 1 в исходном коде, отрицательный — от 1 к 0.
- Отсутствие перепадов показывает, что значения предыдущего и последующего битов равны.
- Для декодирования кодов в формате БВН необходимы тактовые импульсы



# Потенциальный код NRZ (Non Return to Zero)

---

## Достоинства метода NRZ:

- простота реализации;
- метод обладает хорошей распознаваемостью ошибок (благодаря наличию двух резко отличающихся потенциалов);
- основная гармоника  $f_0$  имеет достаточно низкую частоту (равную  $N/2$  Гц), что приводит к узкому спектру.

## Недостатки метода NRZ:

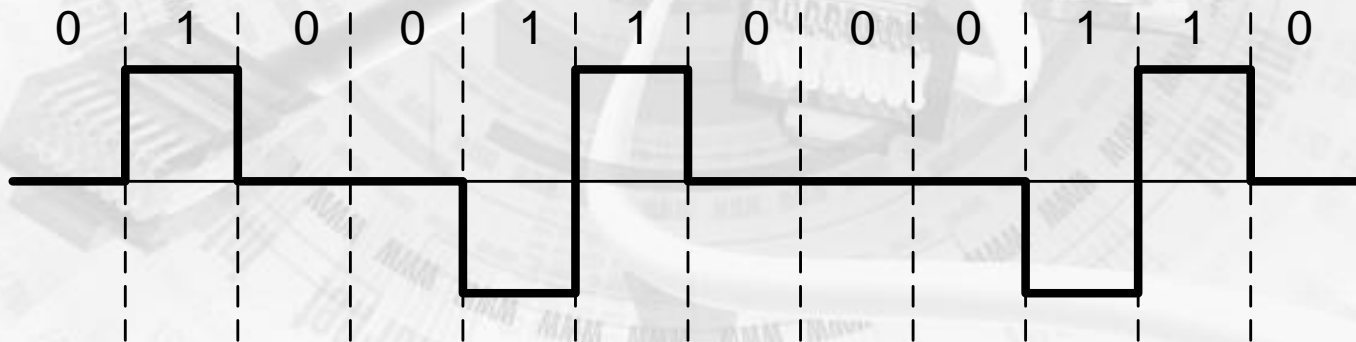
- метод не обладает свойством самосинхронизации;
- наличие низкочастотной составляющей, которая приближается к постоянному сигналу при передаче длинных последовательностей единиц или нулей.

# Биполярное кодирование АМІ

Это метод **тринарного кодирования**, то есть каждый бит передается 3-мя уровнями напряжения.

АМІ-код использует следующие представления битов:

- биты 0 представляются нулевым напряжением (0 В);
- биты 1 представляются поочерёдно значениями  $-U$  или  $+U$  (В).





# Биполярное кодирование АМІ

---

**Достоинства:** АМІ-код обладает хорошими синхронизирующими свойствами при передаче серий единиц и сравнительно прост в реализации.

**Недостатком** кода является ограничение на плотность нулей в потоке данных, поскольку длинные последовательности нулей ведут к потере синхронизации.

# Сравнение AMI и NRZ

---

«+»

- При передаче длинных последовательностей единиц код AMI частично решает проблемы наличия постоянной составляющей и отсутствия самосинхронизации, присущие коду NRZ.
- Для различных комбинаций битов на линии использование кода AMI приводит к более узкому спектру сигнала, чем для кода NRZ, а значит, и к более высокой пропускной способности линии.
- Код AMI предоставляет некоторые возможности по распознаванию ошибочных сигналов (нарушение строгой очередности в полярности сигналов говорит о ложном импульсе или исчезновении с линии корректного импульса).

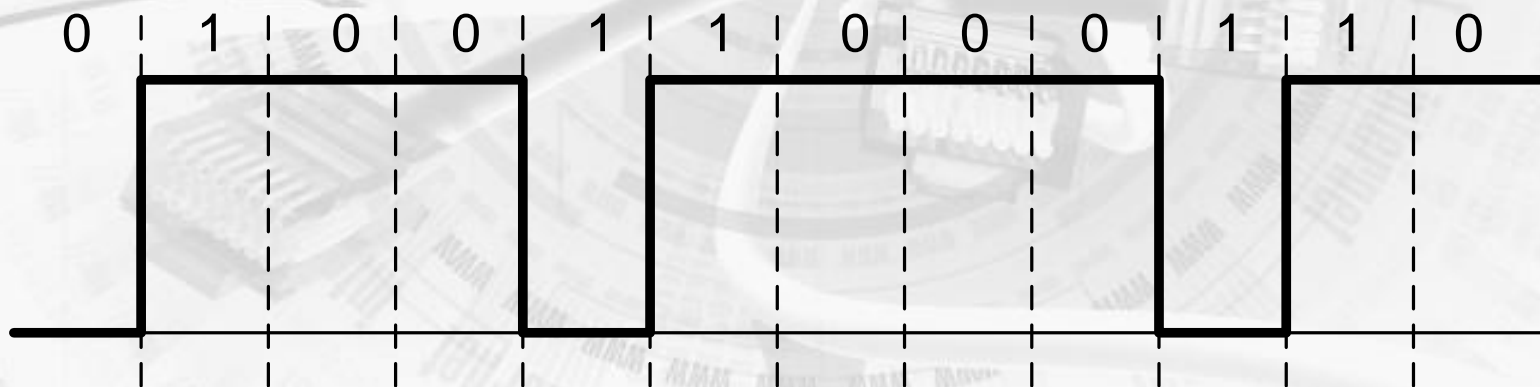
«-»

- Длинные последовательности нулей для кода AMI столь же опасны, как и для кода NRZ — сигнал вырождается в постоянный потенциал нулевой амплитуды.
- В коде AMI используются не два, а три уровня сигнала на линии.
- Дополнительный уровень требует увеличение мощности передатчика примерно на 3 дБ для обеспечения той же достоверности приема битов на линии.

# Потенциальный код NRZI (Non Return to Zero with ones Inverted)

---

При передаче последовательности единиц, сигнал не возвращается к нулю в течение такта. То есть смена сигнала происходит при передаче единицы, а передача нуля не приводит к изменению напряжения.



# Потенциальный код NRZI (Non Return to Zero with ones Inverted)

---

## Достоинства:

- NRZI требует меньше изменений сигнала при передаче произвольной двоичной информации, чем манчестерский код, за счет чего спектр его сигналов уже.

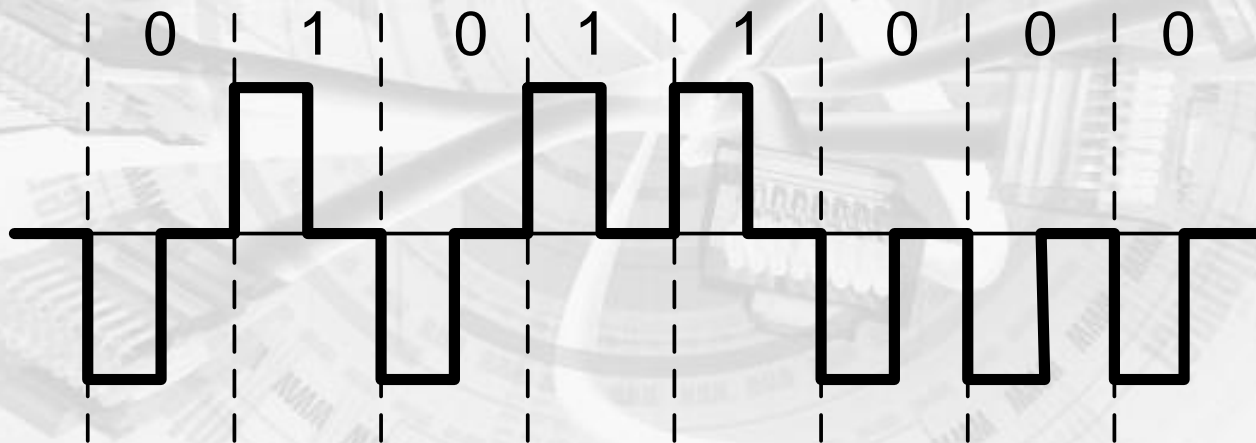
## Недостатки:

- код NRZI обладает плохой самосинхронизацией, так как при передаче длинных последовательностей нулей сигнал вообще не меняется и, значит, у приемника исчезает возможность синхронизации с передатчиком на значительное время, что может приводить к ошибкам распознавания данных.

# Биполярный импульсный код

---

Единица представляется импульсом одной полярности, а ноль — другой. Каждый импульс длится половину такта.



# Биполярный импульсный код

---

## Достоинства:

- подобный код обладает отличными самосинхронизирующими свойствами;
- спектр у данного кода шире, чем у потенциальных кодов. Так, при передаче всех нулей или единиц частота основной гармоники кода равна 1МГц, что в два раза выше основной гармоники кода NRZ и в четыре раза выше основной гармоники кода AMI при передаче чередующихся единиц и нулей.

## Недостатки:

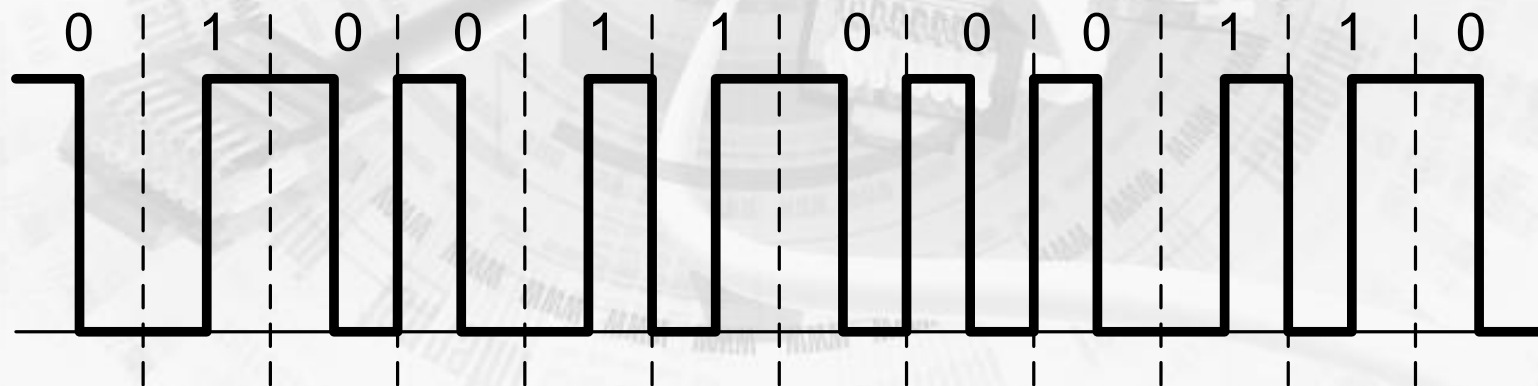
- может присутствовать постоянная составляющая, например, при передаче длинной последовательности единиц или нулей.
- из-за слишком широкого спектра биполярный импульсный код используется редко.

# Манчестерский код

---

Каждый такт делится на две части. Информация кодируется перепадами потенциала в середине каждого такта.

Единица кодируется перепадом от низкого уровня сигнала к высокому, а ноль — обратным перепадом.



# Манчестерский код

---

## Достоинства:

- манчестерский код обладает хорошими самосинхронизирующими свойствами;
- у манчестерского кода нет постоянной составляющей (меняется каждый такт);
- основная гармоника в худшем случае (при передаче последовательности 1 или 0) имеет частоту  $N$  Гц, а в лучшем случае (при передаче чередующихся 1 и 0) —  $N/2$  Гц, как и у NRZ;
- в среднем ширина спектра при манчестерском кодировании в два раза шире чем при NRZ кодировании.

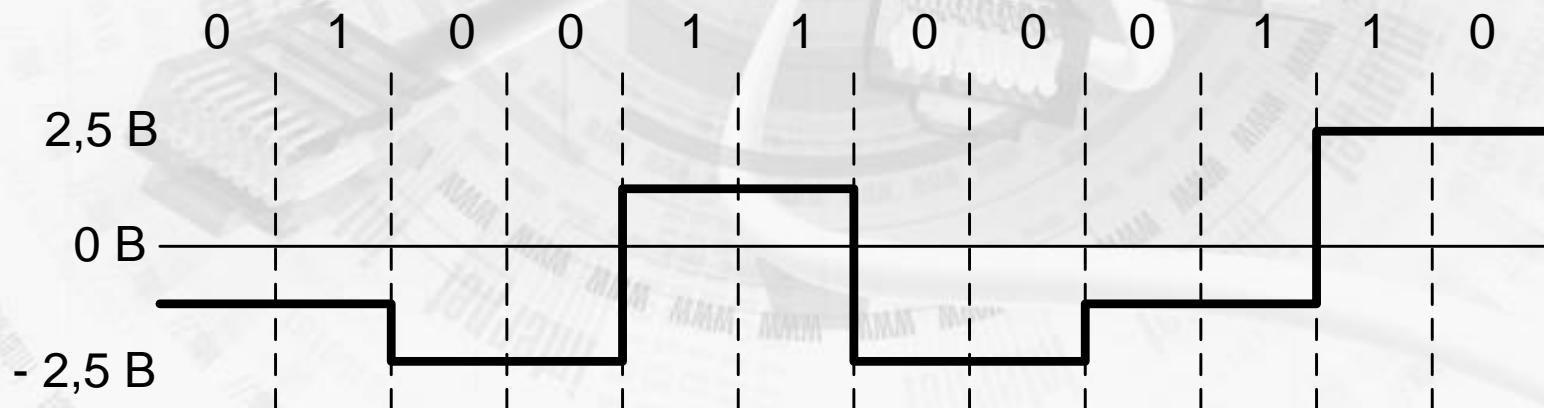
Применяется в технологиях Ethernet и Token Ring.



# Потенциальный код 2B1Q

Название кода 2B1Q отражает его суть — каждые два бита (2B) передаются за один такт (1) сигналом, имеющим четыре состояния (Q — Quadra).

- Паре битов 00 соответствует потенциал -2,5 В
- паре 01 — потенциал -0,833 В
- паре 11 — потенциал +0,833 В
- паре 10 — потенциал +2,5 В.



# Потенциальный код 2B1Q

---

**Достоинство** : сигнальная скорость у этого метода в два раза ниже, чем у кодов NRZ и AMI, а спектр сигнала в два раза уже. Следовательно с помощью 2B1Q-кода можно по одной и той же линии передавать данные в два раза быстрее.

**Недостаток**: реализация этого метода требует более мощного передатчика и более сложного приемника, который должен различать четыре уровня.

# Избыточный код 4В/5В

---

Избыточные коды основаны на разбиении исходной последовательности битов на порции, которые часто называют символами. Затем каждый исходный символ заменяется новым с большим количеством битов, чем исходный.

В избыточном коде 4В/5В исходные символы длиной 4 бит заменяются символами длиной 5 бит. Так как результирующие символы содержат избыточные биты, то общее количество битовых комбинаций в них больше, чем в исходных (см. табл.1).

# Избыточный код 4В/5В

---

Таблица 1 - соответствие исходных и результирующих кодов 4В/5В

Исходный код	Результирующий код	Исходный код	Результирующий код
0000	11110	1000	10010
0001	01001	1001	10011
0010	10100	1010	10110
0011	10101	1011	10111
0100	01010	1100	11010
0101	01011	1101	11011
0110	01110	1110	11100
0111	01111	1111	11101

# Скремблирование

---

**Скремблирование** заключается в побитном вычислении результирующего кода на основании битов исходного кода и полученных в предыдущих тактах битов результирующего кода.

Например, скрэмблер может реализовывать следующее соотношение:

$$V_i = A_i \oplus V_{i-3} \oplus V_{i-5}$$

где  $V_i$  — двоичная цифра результирующего кода, полученная на  $i$ -м такте работы скрэмблера,  $A_i$  — двоичная цифра исходного кода, поступающая на  $i$ -м такте на вход скрэмблера,  $V_{i-3}$  и  $V_{i-5}$  — двоичные цифры результирующего кода, полученные на предыдущих тактах работы скрэмблера (соответственно на 3 и на 5 тактов ранее текущего такта) и объединенные операцией исключающего ИЛИ (сложение по модулю 2).

# Скремблирование

---

Таким образом, на выходе скремблера появится код 110001101111, в котором нет последовательности из шести нулей, присутствовавшей в исходном коде.

$$B_1 = A_1 = 1$$

$$B_2 = A_2 = 1$$

$$B_3 = A_3 = 0$$

$$B_4 = A_4 \oplus B_1 = 1 \oplus 1 = 0$$

$$B_5 = A_5 \oplus B_2 = 1 \oplus 1 = 0$$

$$B_6 = A_6 \oplus B_3 \oplus B_1 = 0 \oplus 0 \oplus 1 = 1$$

$$B_7 = A_7 \oplus B_4 \oplus B_2 = 0 \oplus 0 \oplus 1 = 1$$

$$B_8 = A_8 \oplus B_5 \oplus B_3 = 0 \oplus 0 \oplus 0 = 0$$

$$B_9 = A_9 \oplus B_6 \oplus B_4 = 0 \oplus 1 \oplus 0 = 1$$

$$B_{10} = A_{10} \oplus B_7 \oplus B_5 = 0 \oplus 1 \oplus 0 = 1$$

$$B_{11} = A_{11} \oplus B_8 \oplus B_6 = 0 \oplus 0 \oplus 1 = 1$$

$$B_{12} = A_{12} \oplus B_9 \oplus B_7 = 1 \oplus 1 \oplus 1 = 1$$

# Скремблирование

---

После получения результирующей последовательности приемник передает ее дескремблеру, который восстанавливает исходную последовательность на основании обратного соотношения:

$$C_i = V_i \quad V_{i-3} \quad V_{i-5} = (A_i \quad V_{i-3} \quad V_{i-5}) \quad V_{i-3} \quad V_{i-5} = A_i$$

# Методы обнаружения ошибок

---

Методы обнаружения ошибок основаны на передаче в составе блока данных избыточной служебной информации, по которой можно судить с некоторой степенью вероятностно достоверности принятых данных.

Избыточную служебную информацию принято называть **контрольной суммой**, или **контрольной последовательностью кадра** (Frame Check Sequence, FCS).

Контрольная сумма вычисляется как функция от основной информации. Принимающая сторона повторно вычисляет контрольную сумму кадра по известному алгоритму и в случае ее совпадения с контрольной суммой, вычисленной передающей стороной, делает вывод о том, что данные были переданы через сеть корректно.



# Методы обнаружения ошибок

---

Наиболее распространенные алгоритмы вычисления контрольной суммы:

- Контроль по паритету
- Вертикальный и горизонтальный контроль по паритету
- Циклический избыточный контроль (Cyclic Redundancy Check, CRC)

# Методы обнаружения ошибок

---

**Контроль по паритету** - заключается в суммировании по модулю 2 всех битов контролируемой информации.

Нетрудно заметить, что для информации, состоящей из нечетного числа единиц, контрольная сумма всегда равна 1, а при четном числе единиц — 0.

Например, для данных 100101011 результатом контрольного суммирования будет значение 1.

Результат суммирования также представляет собой один дополнительный бит данных, который пересылается вместе с контролируемой информацией. При искажении в процессе пересылки любого одного бита исходных данных (или контрольного разряда) результат суммирования будет отличаться от принятого контрольного разряда, что говорит об ошибке. Однако, двойная ошибка, например 110101010, будет неверно принята за корректные данные.

# Методы обнаружения ошибок

---

**Достоинство** - представляет собой наиболее простой метод контроля данных

## **Недостатки:**

- Это наименее мощный алгоритм контроля, так как с его помощью можно обнаруживать только одиночные ошибки в проверяемых данных. Поэтому контроль по паритету применяется к небольшим порциям данных, как правило, к каждому байту.
- Метод редко используется в компьютерных сетях из-за значительной избыточности и невысоких диагностических возможностей.

# Методы обнаружения ошибок

---

**Вертикальный и горизонтальный контроль по паритету** представляет собой модификацию предыдущего метода.

Исходные данные рассматриваются в виде матрицы, строки которой составляют байты данных. Контрольный разряд подсчитывается отдельно для каждой строки и для каждого столбца матрицы.

**Достоинство** – данный метод позволяет обнаруживать большую часть двойных ошибок.

**Недостаток** – этот метод обладает еще больше избыточностью.

# Методы обнаружения ошибок

---

**Циклический избыточный контроль (Cyclic Redundancy Check, CRC)** является в настоящее время наиболее популярным методом контроля.

Метод основан на представлении исходных данных в виде одного многоразрядного двоичного числа.

Контрольной информацией считается остаток от деления этого числа на известный делитель  $R$ . Обычно в качестве делителя выбирается семнадцати- или тридцатитрехразрядное число.

При получении кадра данных снова вычисляется остаток от деления на тот же делитель  $R$ , но при этом к данным кадра добавляется и содержащаяся в нем контрольная сумма.

Если остаток от деления на  $R$  равен нулю, то делается вывод об отсутствии ошибок в полученном кадре, в противном случае кадр считается искаженным.

# Методы обнаружения ошибок

---

## Достоинства:

- диагностические возможности данного метода гораздо выше, чем у методов контроля по паритету;
- метод CRC позволяет обнаруживать все одиночные ошибки, двойные ошибки и ошибки в нечетном числе битов;
- метод обладает также невысокой степенью избыточности.

**Недостатки:** высокая вычислительная сложность.

# Методы коррекции ошибок

---

Техника кодирования, которая позволяет приемнику не только понять, что присланные данные содержат ошибки, но и исправить их, называется **прямой коррекцией ошибок** - (Forward Error Correction, FEC).

Коды, которые обеспечивают прямую коррекцию ошибок, требуют введения большей избыточности в передаваемые данные, чем коды, только обнаруживающие ошибки.



# Методы коррекции ошибок

---

- Для того чтобы оценить количество дополнительных битов, требуемых для исправления ошибок, нужно знать так называемое **расстояние Хемминга** между разрешенными комбинациями кода. **Расстоянием Хемминга** называется минимальное число битовых разрядов, в которых отличается любая пара разрешенных кодов. Для схем контроля по паритету расстояние Хемминга равно 2.
- Так как коды с контролем по паритету имеют расстояние Хемминга, равное 2, то они могут только обнаруживать однократные ошибки и не могут исправлять ошибки.
- Можно доказать, что если мы сконструировали избыточный код с расстоянием Хемминга, равным  $n$ , то такой код будет в состоянии распознавать  $(n-1)$ -кратные ошибки и исправлять  $(n-1)/2$ -кратные ошибки.



# Методы коррекции ошибок

---

**Коды Хемминга** эффективно обнаруживают и исправляют изолированные ошибки, то есть отдельные искаженные биты, которые разделены большим количеством корректных битов.

Однако при появлении длинной последовательности искаженных битов (пульсации ошибок) коды Хемминга не работают.

# Методы коррекции ошибок

---

Пульсации ошибок характерны для беспроводных каналов, в которых применяют **сверточные коды**. Поскольку для распознавания наиболее вероятного корректного кода в этом методе задействуется решетчатая диаграмма, то такие коды еще называют **решетчатыми**. Эти коды используются не только в беспроводных каналах, но и в модемах.