**Донской государственный технический университет**

# **Кафедра радиоэлектроники**

**Практическое занятие № 4**

**Исследование методов преобразования сигналов в системах**

**документальной связи**

**Практическое занятие № 4**

**Тема: Исследование методов преобразования сигналов в системах**

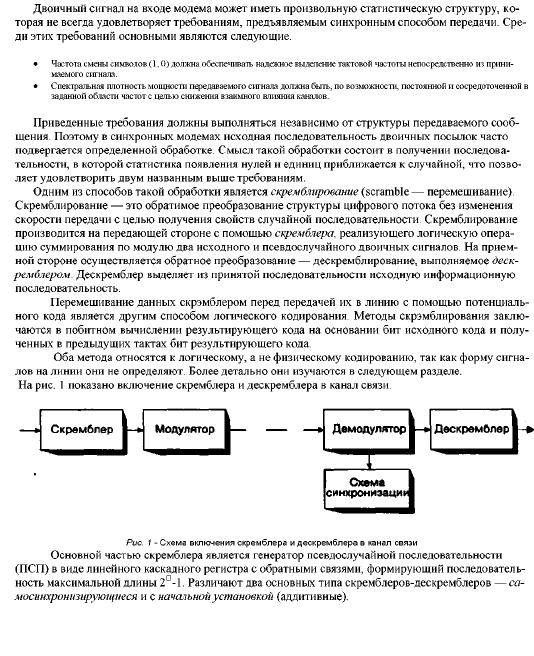
**документальной связи**

**1. Теоретическая часть**

1.1 Метод скремблирования

1.2 Методы потенциального кодирования

**1.1 Метод скремблирования**

****

Применительно к системам документальной электросвязи скремблирование повышает надёжность синхронизации устройств, подключенных к линии связи (обеспечивает надёжное выделение тактовой частоты непосредственно из принимаемого сигнала), и уменьшает уровень помех, излучаемых на соседние линии многожильного кабеля. Другая область применения скремблеров — защита передаваемой информации от несанкционированного доступа.

Для алгоритмов скремблирования исключительно важны скорость работы и случайный характер последовательности, чтобы его нельзя было восстановить в случае перехвата противником. Процесс скремблирования может включать в себя добавление определённых компонент к исходному сигналу либо изменение важных частей сигнала для того, чтобы усложнить восстановление вида исходного сигнала либо для придания сигналу определённых статистических свойств.

Обычно скремблирование осуществляется на последнем этапе цифровой обработки непосредственно перед модуляцией.

## Типы скремблеров

* Самосинхронизирующиеся скремблеры (СС)
* Аддитивные скремблеры (с установкой)

### 1.2 Самосинхронизирующиеся скремблеры

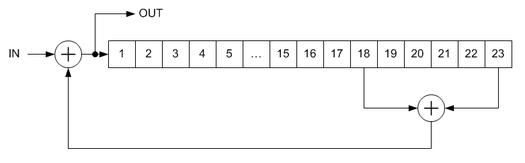
[](http://commons.wikimedia.org/wiki/File:Scrambler_randomizer_multiplicative_scrambler.png?uselang=ru)

Рисунок 1

Основной частью самосинхронизирующегося скремблера является [генератор псевдослучайной последовательности (ПСП)](http://ru.wikipedia.org/wiki/%D0%93%D0%B5%D0%BD%D0%B5%D1%80%D0%B0%D1%82%D0%BE%D1%80_%D0%BF%D1%81%D0%B5%D0%B2%D0%B4%D0%BE%D1%81%D0%BB%D1%83%D1%87%D0%B0%D0%B9%D0%BD%D1%8B%D1%85_%D1%87%D0%B8%D1%81%D0%B5%D0%BB) в виде [линейного n-каскадного регистра с обратными связями](http://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%B3%D0%B8%D1%81%D1%82%D1%80_%D1%81%D0%B4%D0%B2%D0%B8%D0%B3%D0%B0_%D1%81_%D0%BB%D0%B8%D0%BD%D0%B5%D0%B9%D0%BD%D0%BE%D0%B9_%D0%BE%D0%B1%D1%80%D0%B0%D1%82%D0%BD%D0%BE%D0%B9_%D1%81%D0%B2%D1%8F%D0%B7%D1%8C%D1%8E), формирующий последовательность максимальной длины 2^n - 1.

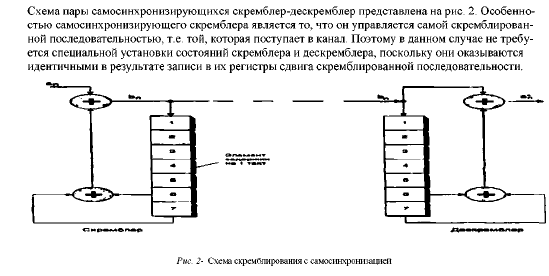
Особенностью ***самосинхронизирующегося скремблера* (*СС скремблера*)** является то, что он управляется скремблированной последовательностью, т.е. той, которая передаётся в канал. Поэтому, при данном виде скремблирования не требуется специальной установки состояний скремблера и дескремблера: скремблированная последовательность записывается в регистры сдвига скремблера и дескремблера, устанавливая их в идентичное состояние. При потере синхронизма между скремблером и дескремблером время восстановления синхронизма не превышает числа тактов, равного числу ячеек регистра скремблера.

На приёмной стороне выделение исходной последовательности происходит путём [**сложения по модулю 2**](http://ru.wikipedia.org/wiki/%D0%A1%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D0%B5_%D0%BF%D0%BE_%D0%BC%D0%BE%D0%B4%D1%83%D0%BB%D1%8E_2)принятой скремблированной последовательности с последовательностью на выходе сдвигового регистра. Например, для схемы, приведённой на рисунке 1, входная последовательность INс помощью скремблера в соответствии с соотношением OUT = IN \oplus (R_{18} \oplus R_{23}) преобразуется в посылаемую двоичную последовательность OUT. В приёмнике из этой последовательности таким же регистром сдвига, как на приёме, формируется последовательность

IN' = OUT \oplus (R_{18} \oplus R_{23}) = IN \oplus (R_{18} \oplus R_{23}) \oplus (R_{18} \oplus R_{23}) = IN.

Как следует из принципа действия схемы, при одной ошибке в последовательности INошибочными получаются также последующие восемнадцатый и двадцать третий символы (в данном примере). В общем случае влияние ошибочно принятого бита будет сказываться *a* раз, где *a* - число обратных связей в регистре сдвига. Таким образом, **СС скремблер-дескремблер обладает свойством размножения ошибок.** Данный недостаток СС скремблера-дескремблера ограничивает число обратных связей в регистре сдвига; практически это число не превышает *a* = 2.

**Второй недостаток СС скремблера связан с возможностью появления на его выходе при определенных условиях так называемых «критических ситуаций»**, когда выходная последовательность приобретает периодический характер с периодом, меньшим длины ПСП. Чтобы предотвратить это, в скремблере и дескремблере предусматриваются специальные дополнительные схемы контроля, которые выявляют наличие периодичности элементов на входе и нарушают её.



При потере синхронизма между скремблером и дескремблером время его восстановления не превышает числа тактов, равного числу ячеек регистра скремблера. На приёмной стороне выделение информационной последовательности происходит сложением по модулю два принятой скремблированной последовательности с псевдослучайной последовательностью регистра (ПСП). Например, для схемы, изображённой на рисунке 2, входная последовательность *an*с помощью скремблера в соответствии с выражением

*bn=an+(bn-6+ bn-7)*

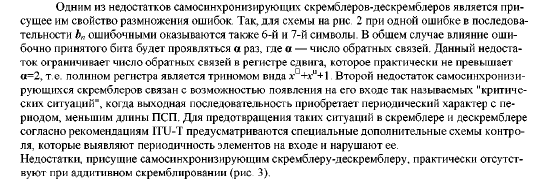
преобразуется в двоичную последовательность *bn*, посылаемую в канал. В приёмнике из этой последовательности таким же регистром сдвига, как и на передаче, формируется последовательность

*an\* =bn+(bn-6+ bn-7),*

которая идентична последовательности *an*. Это легко проверяется при преобразовании первого выражения к виду

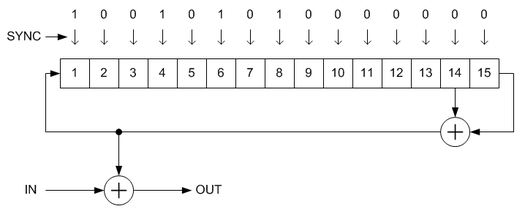
*an=bn+(bn-6+ bn-7)*

и сравнении полученного выражения с предыдущим.



### 1.3. Аддитивные скремблеры

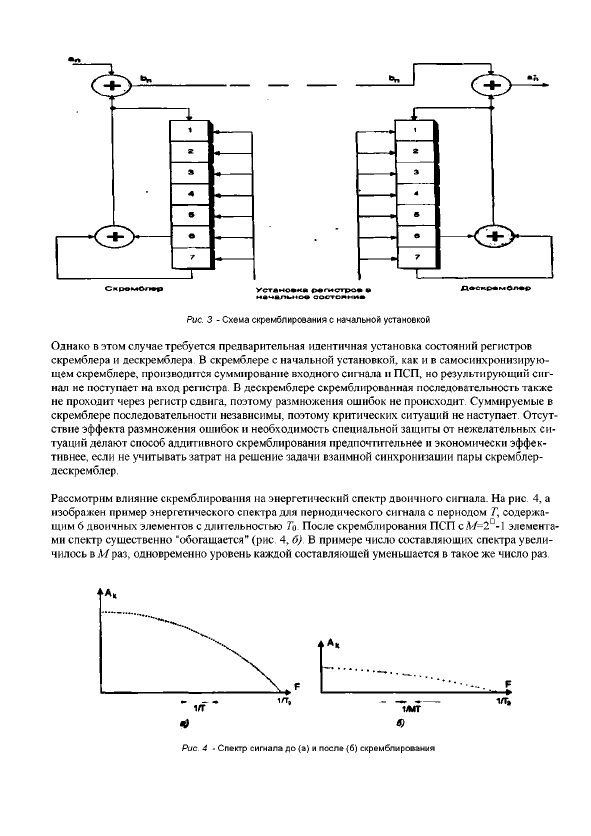
При ***аддитивном скремблировании*** требуется предварительная идентичная установка состояний регистров скремблера и дескремблера.

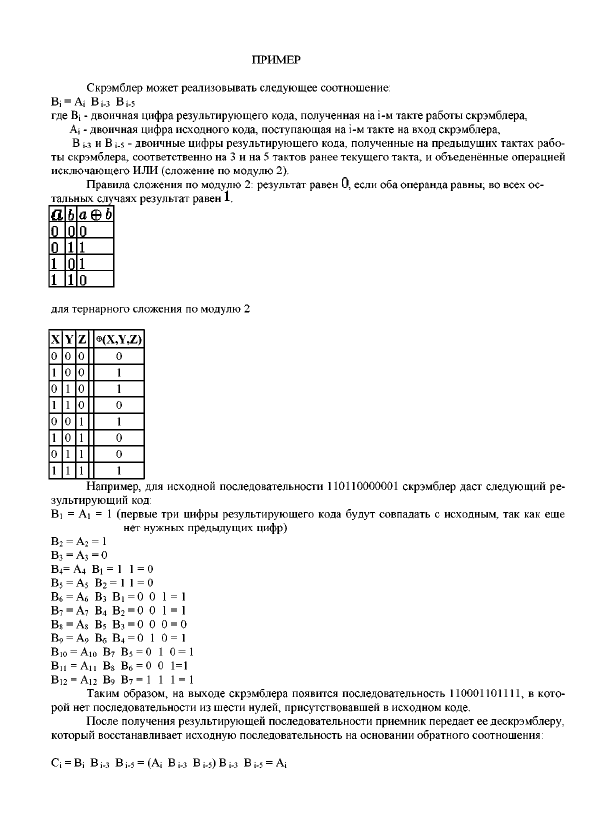
[](http://commons.wikimedia.org/wiki/File:Scrambler_randomizer_additive.png?uselang=ru)

**Скремблер аддитивный**

В ***скремблере с установкой (АД-скремблере)*,** также, как и в СС скремблере, производится суммирование входного сигнала и ПСП, но результирующий сигнал не поступает на вход регистра. В дескремблере скремблированный сигнал также не проходит через регистр сдвига, поэтому размножения ошибок не происходит.

Суммируемые в скремблере последовательности независимы, поэтому их период всегда равен наименьшему общему кратному величин периодов входной последовательности и ПСП и критическое состояние отсутствует. Отсутствие эффекта размножения ошибок и необходимости в специальной логике защиты от нежелательных ситуаций делают способ аддитивного скремблирования предпочтительнее, если не учитывать затрат на решение задачи синхронизации скремблера и дескремблера. В качестве сигнала установки в цифровых системах передачи данных используют сигнал цикловой синхронизации.





Существую и более простые методы борьбы с последовательностями единиц или нулей, также относящиеся к классу скремблирования.

**2. Методы потенциального кодирования**

**2.1 Форматы кодов**

Каждый бит кодового слова передаётся или записывается с помощью дискретных сигналов, например, импульсов. Способ представления исходного кода определёнными сигналами определяется форматом кода. Известно большое количество форматов, каждый из которых имеет свои достоинства и недостатки и предназначен для использования в определённой аппаратуре.

* **Формат БВН (без возвращения к нулю)** естественным образом соответствует режиму работы логических схем. Единичный бит передаётся 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, так как в его спектре не содержится тактовая частота. Соответствующий коду формата БВН сигнал содержит низкочастотные компоненты (при передаче длинных серий нулей или единиц перепады не возникают).
* **Формат БВН-1 (без возвращения к нулю с перепадом при передаче 1)** является разновидностью формата БВН. В отличие от последнего в БВН-1 уровень не передает данные, так как и положительные и отрицательные перепады соответствуют единичным битам. Перепады сигнала формируются при передаче 1. При передаче 0 уровень не меняется. Для декодирования требуются тактовые импульсы.
* **Формат БВН −0 (без возвращения к нулю с перепадом при передаче 0)** является дополнительным к БВН-1 (перепады соответствуют нулевым битам исходного кода). В многодорожечных системах записи цифровых сигналов вместе с кодом в формате БВН надо записывать тактовые импульсы. Возможным вариантом является запись двух дополнительных сигналов, соответствующих кодам в форматах БВН-1 и БВН-0. В одном из двух сигналов перепады происходят в каждом такте, что позволяет получить импульсы тактовой частоты.
* **Формат ВН (с возвращением к нулю)** требует передачи импульса, занимающего только часть тактового интервала (например, половину), при одиночном бите. При нулевом бите импульс не формируется.
* **Формат ВН-П (с активной паузой)** означает передачу импульса положительной полярности при единичном бите и отрицательной — при нулевом бите. Сигнал этого формата имеет в спектре компоненты тактовой частоты. Он применяется в ряде случаев для передачи данных по линиям связи.
* **Формат ДФ-0 (двухфазный со скачком фазы при передаче 0)** соответствует способу представления, при котором перепады формируются в начале каждого такта. При единичных битах сигнал в этом формате меняется с тактовой частотой, то есть в середине каждого такта происходит перепад уровня. При передаче нулевого бита перепад в середине такта не формируется, то есть имеет место скачок фазы. Код в данном формате обладает возможностью самосинхронизации и не требует передачи тактовых сигналов.

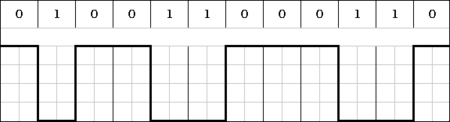
Направление перепада при передаче сигнала единицы не имеет значения. Поэтому изменение полярности кодированного сигнала не влияет на результат декодирования. Он может передаваться по симметричным линиям без постоянной составляющей. Это также упрощает его магнитную запись. Этот формат известен также под названием «Манчестер 1». Он используется в адресно-временном коде SMPTE, широко применяющемся для синхронизации носителей звуковой и видеоинформации.

**Бинарное кодирование**

**Без возврата к нулю**

Потенциальное кодирование, также называется кодированием без возвращения к нулю ([**NRZ**](http://en.wikipedia.org/wiki/Non-return-to-zero)). При передаче нуля он передает потенциал, который был установлен на предыдущем такте (то есть не меняет его), а при передаче единицы потенциал инвертируется на противоположный. Этот код называется потенциальным кодом с инверсией при единице (**NRZI**).

[**NRZ**](http://en.wikipedia.org/wiki/Non-return-to-zero)

[](http://commons.wikimedia.org/wiki/File:NRZ-code.png?uselang=ru)

**Потенциальный код NRZ (перевёрнутый)**

Для передачи единиц и нулей используются два устойчиво различаемых потенциала:

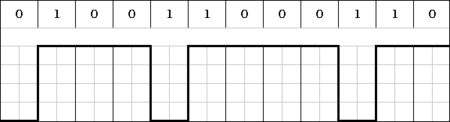
[**NRZ (прямой)**](http://en.wikipedia.org/wiki/Non-return-to-zero)**:**

* биты 0 представляются нулевым напряжением 0 (В);
* биты 1 представляются значением U (В).

**NRZ (перевёрнутый):**

* биты 0 представляются значением U (В);
* биты 1 представляются нулевым напряжением 0 (В).

[**NRZI**](http://en.wikipedia.org/wiki/Non-return-to-zero#Non-Return-to-Zero_Inverted_.28NRZI.29)

[](http://commons.wikimedia.org/wiki/File:NRZI-code.png?uselang=ru)

**Потенциальный код NRZI**

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

**Достоинства метода NRZ:**

— Простота реализации.

— Метод обладает хорошей распознаваемостью ошибок (благодаря наличию двух резко отличающихся потенциалов).

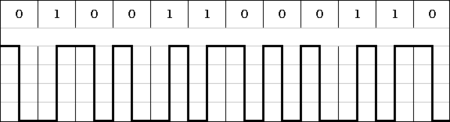
— Основная гармоника f0 имеет достаточно низкую частоту (равную N/2 Гц, где N — битовая скорость передачи дискретных данных [бит/с]), что приводит к узкому спектру.

**Недостатки метода NRZ:**

— Метод не обладает свойством самосинхронизации. Даже при наличии высокоточного тактового генератора приёмник может ошибиться с выбором момента съёма данных, так как частоты двух генераторов никогда не бывают полностью идентичными. Поэтому при высоких скоростях обмена данными и длинных последовательностях единиц или нулей небольшое рассогласование тактовых частот может привести к ошибке в целый такт и, соответственно, считыванию некорректного значения бита.

— Вторым серьёзным недостатком метода, является наличие низкочастотной составляющей, которая приближается к постоянному сигналу при передаче длинных последовательностей единиц и нулей. Из-за этого многие линии связи, не обеспечивающие прямого гальванического соединения между приёмником и источником, этот вид кодирования не поддерживают. Поэтому в сетях код NRZ в основном используется в виде различных его модификаций, в которых устранены как плохая самосинхронизация кода, так и проблемы постоянной составляющей.

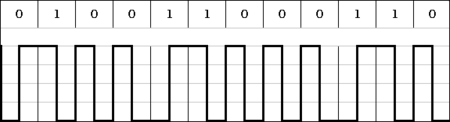
[**Манчестерское кодирование**](http://en.wikipedia.org/wiki/Manchester_code)

[](http://commons.wikimedia.org/wiki/File:Manchester-code.png?uselang=ru)

**Манчестерское кодирование**

При манчестерском кодировании 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 (по стандарту IEEE 802.3, хотя по Д.Е. Томасу кодирование происходит наоборот). В начале каждого такта может происходить служебный перепад сигнала, если нужно представить несколько единиц или нулей подряд. Так как сигнал изменяется, по крайней мере, один раз за такт передачи одного бита данных, то манчестерский код обладает хорошими самосинхронизирующими свойствами. У манчестерского кода нет постоянной составляющей (меняется каждый такт), а основная гармоника в худшем случае (при передаче последовательности единиц или нулей) имеет частоту N Гц, а в лучшем случае (при передаче чередующихся единиц и нулей) — N/2 Гц, как и у NRZ. В среднем ширина спектра при манчестерском кодировании в два раза шире, чем при NRZ кодировании.

[**Дифференциальное манчестерское кодирование**](http://en.wikipedia.org/wiki/Differential_Manchester_encoding)

[](http://commons.wikimedia.org/wiki/File:Differential-manchester-code.png?uselang=ru)

**Дифференциальное манчестерское кодирование**

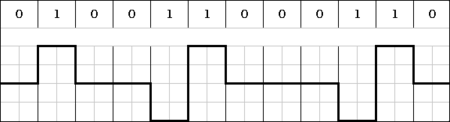
При дифференциальном манчестерском кодировании в течение битового интервала (времени передачи одного бита) уровень сигнала может меняться дважды. Обязательно происходит изменение уровня в середине интервала, этот перепад используется для синхронизации. Получается, что при передаче нуля в начале битового интервала происходит перепад уровней, а при передаче единицы такой перепад отсутствует.

**Тринарное кодирование**

[**RZ**](http://en.wikipedia.org/wiki/Return-to-zero) **(c возвратом к нулю)**

То есть каждый бит передается 3-мя уровнями напряжения. Поэтому требует в 2 раза больше скорости по сравнению с обычной скоростью. Это квазитроичный код, то есть изменение сигнала происходит между 3-мя уровнями.

[**Биполярный код AMI**](http://en.wikipedia.org/wiki/Bipolar_encoding)

[](http://commons.wikimedia.org/wiki/File:AMI-code.png?uselang=ru)

**Биполярный код AMI**

AMI-код использует следующие представления битов:

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

AMI-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. Используется в телефонии уровня передачи данных, когда используются потоки мультиплексирования.

[**HDB3**](http://en.wikipedia.org/wiki/Modified_AMI_code#HDB3_.28European_E-carrier.29)

Код HDB3 исправляет любые 4 подряд идущие нули в исходные последовательности. Правило формирования кода следующее: каждые 4 нуля заменяются 4 символами в которых имеется хотя бы один сигнал V. Для подавления постоянной составляющей полярность сигнала V чередуется при последовательных заменах. Для замены используются два способа: 1)если перед заменой исходный код содержал нечётное число единиц то используется последовательность 000V, если чётное то 100V

V-сигнал единицы запрещённого для данного сигнала полярности

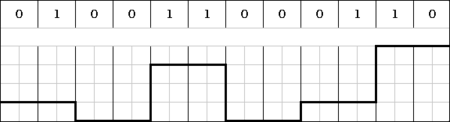
Тоже что и AMI, только кодирование последовательностей из четырех нулей заменяется на код -V, 0, 0, -V или +V, 0, 0, +V — в зависимости от предыдущей фазы сигнала.

[**MLT-3**](http://en.wikipedia.org/wiki/MLT-3_encoding)

**MLT-3** Multi Level Transmission — 3 (многоуровневая передача) — метод кодирования, использующий три уровня сигнала. Метод основывается на циклическом переключении уровней -U, 0, +U. Единице соответствует переход с одного уровня сигнала на следующий. Так же как и в методе **NRZI** при передаче «нуля» сигнал не меняется. В случае наиболее частого переключения уровней (длинная последовательность единиц) для завершения цикла необходимо четыре перехода. Это позволяет вчетверо снизить частоту несущей относительно тактовой частоты, что делает **MLT-3** удобным методом при использовании медных проводов в качестве среды передачи. Метод разработан Cisco Systems для использования в сетях [**FDDI**](http://ru.wikipedia.org/wiki/FDDI) на основе медных проводов, известных как **CDDI.** Также используется в [**Fast Ethernet**](http://ru.wikipedia.org/wiki/Ethernet)[**100BASE-TX**](http://ru.wikipedia.org/wiki/100BASE-TX).

**Тетрарное кодирование**

[**Потенциальный код 2B1Q**](http://en.wikipedia.org/wiki/2B1Q)

[](http://commons.wikimedia.org/wiki/File:2B1Q-code.png?uselang=ru)

**Потенциальный код 2B1Q**

Код 2B1Q передает пару бит за один битовый интервал. Каждой возможной паре в соответствие ставится свой уровень из четырех возможных уровней потенциала. Паре

00 соответствует потенциал −2.5 В,

01 соответствует −0.833 В,

11 — +0.833 В,

10 — +2.5 В.

**Достоинство метода 2B1Q:** 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 метода 2B1Q:** Реализация этого метода требует более мощного передатчика и более сложного приемника, который должен различать четыре уровня.

**2.2 Иерархия кодирования**

Система кодирования сигналов имеет многоуровневую иерархию. Самым нижним уровнем в иерархии кодирования является физическое кодирование, которое определяет число дискретных уровней сигнала (амплитуды напряжения, амплитуды тока, амплитуды света).

Физическое кодирование рассматривает кодирование только на самом низшем уровне иерархии кодирования - на физическом уровне и не рассматривает более высокие уровни в иерархии кодирования, к которым относятся логические кодирования различных уровней.

С точки зрения физического кодирования цифровой сигнал может иметь два, три, четыре, пять и т. д. уровней амплитуды напряжения, амплитуды тока, амплитуды света. В совокупности физическое кодирование и логическое кодирование образуют **систему кодирования** самого низшего уровня.

Ни в одной из версий технологии [**Ethernet**](http://ru.wikipedia.org/wiki/Ethernet) не применяется прямое двоичное кодирование бита 0 напряжением 0 вольт и бита 1 — напряжением +5 вольт, так как такой способ приводит к неоднозначности. Если одна станция посылает битовую строку 00010000, то другая станция может интерпретировать её либо как 10000, либо как 01000, так как она не может отличить «отсутствие сигнала» от бита 0. Поэтому принимающей машине необходим способ однозначного определения начала, конца и середины каждого [**бита**](http://ru.wikipedia.org/wiki/%D0%91%D0%B8%D1%82) без помощи внешнего таймера. Кодирование сигнала на физическом уровне позволяет приёмнику синхронизироваться с передатчиком по смене напряжения в середине периода битов.

**Физическое кодирование**

Самым нижним уровнем в иерархии кодирования является физическое кодирование, которое определяет число дискретных уровней сигнала (амплитуды напряжения, амплитуды тока, амплитуды света). Физическое кодирование рассматривает кодирование только на самом низшем уровне иерархии кодирования - на физическом уровне и не рассматривает более высокие уровни в иерархии кодирования, к которым относятся логические кодирования различных уровней.

С точки зрения физического кодирования цифровой сигнал может иметь два, три, четыре, пять и т. д. уровней амплитуды напряжения, амплитуды тока, амплитуды света. Естественным становится вопрос:"Какое число уровней сигнала имеет наибольшую плотность кодирования?" Ответ на этот вопрос даёт "Теорема о наиболее экономичной системе счисления (системе кодирования)", описанная в книжке С. В. Фомина "Системы счисления", в § 14. Об одном замечательном свойстве троичной системы.

Вторым уровнем в иерархии кодирования является самый нижний уровень логического кодирования с разными назначениями.

В совокупности физическое кодирование и логическое кодирование образуют **систему кодирования** самого низшего уровня.

**Системы кодирования**

**Системы кодирования данных**

Наиболее часто используемые системы кодирования:

* NRZ (Non Return Zero) — без возврата к нулю
* Манчестер II
* RZ (Return to Zero) — с возвратом к нулю

**Системы с двухуровневым кодированием**

[**NRZ**](http://en.wikipedia.org/wiki/Non-return-to-zero)

Простейший код, обычный цифровой (дискретный) сигнал (может быть преобразован на обратную полярность или изменены уровни соответствующие нулю и единице).

Достоинства — простая реализация; не нужно кодировать и декодировать на концах. Высокая скорость передачи при заданной полосе пропускания (для обеспечения пропускной способности в 10Мбит/сек полоса пропускания составит 5 МГц, так как одно колебание равно 2 битам). Для синхронизации передачи байта используется старт-стоповый бит. Недостатки - Наличие постоянной составляющей, из за чего невозможно обеспечить [**гальваническую развязку**](http://ru.wikipedia.org/wiki/%D0%93%D0%B0%D0%BB%D1%8C%D0%B2%D0%B0%D0%BD%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B0%D1%8F_%D1%80%D0%B0%D0%B7%D0%B2%D1%8F%D0%B7%D0%BA%D0%B0) с помощью трансформатора. Высокие требования к синхронизации частот на приёмном и передающем конце - за время передачи одного слова (байта) приемник не должен сбиться больше, чем на бит (например для слова длиной в байт с битом старта и стопа, т.е. всего 10 бит канальной информации, рассинхронизация частот приёмника и передатчика не может превышать 10% в обе стороны, для слова в 16 бит, т.е. 18 бит канальной информации, рассинхронизация не должна превышать 5,5%, а в физических реализациях и того меньше).

[**Манчестерский код, код Манчестер-2**](http://en.wikipedia.org/wiki/Manchester_code)

Является [**самосинхронизирующимся**](http://ru.wikipedia.org/w/index.php?title=%D0%A1%D0%B0%D0%BC%D0%BE%D1%81%D0%B8%D0%BD%D1%85%D1%80%D0%BE%D0%BD%D0%B8%D0%B7%D0%B8%D1%80%D1%83%D1%8E%D1%89%D0%B8%D0%B5%D1%81%D1%8F_%D0%BA%D0%BE%D0%B4%D1%8B&action=edit&redlink=1)**,** то есть не требует специальной кодировки синхроимпульса, который бы занимал полосу данных и поэтому является самым плотным кодом на единицу частоты. Логическому нулю соответствует положительный переход в центре бита, то есть первая половина битового интервала — низкий уровень, вторая половина — высокий. Логической единице соответствует отрицательный переход в центре бита. (Возможна и обратная реализация, обычно в приёмнике это проверяется вначале передачи. "Обратная реализация может возникнуть например в трансформаторе, если перепутать направление обмоток").

Обязательное наличие перехода в центре бита позволяет легко выделить синхросигнал. Допустимое расхождение частот передачи - до 25 % (это означает, что код Манчестер-2 - самый устойчивый к рассинхронизации, он самосинхронизуется в каждом бите передаваемой информации).

Плотность кода 1 бит/герц. В спектре сигнала, закодированного Манчестером-2, присутствует 2 частоты - частота передачи и половинная частота передачи (она образуется когда рядом стоят 0 и 1 или 1 и 0. При передаче гипотетической последовательности одних 0 или 1 в спектре будет присутствовать только частота передачи). Важным достоинством манчестерского кода является возможность обеспечить гальваническую развязку с помощью трансформатора, так как у него отсутствует постоянная составляющая. Вторым важным преимуществом является отсутствие необходимости в синхронизующих битах (как в NRZ-коде) и, вследствие этого, данные могут передаваться подряд сколь угодно долго, из-за чего плотность данных в общем потоке кода приближается к 100% (например для кода NRZ 1-8-0 она равна 80%).

[**NRZI**](http://en.wikipedia.org/wiki/Non-return-to-zero#Non-Return-to-Zero_Inverted_.28NRZI.29)

NRZI — Non Return to Zero Invertive (инверсное кодирование без возврата к нулю). Этот метод является модифицированным методом Non Return to Zero (NRZ), где для представления 1 и 0 используются потенциалы двух уровней. В коде NRZ I также используется 2 потенциала, но его текущее значение зависит от предыдущего. Если текущее значение бита “1”, то полученный потенциал должен быть инверсией от предыдущего, если значение бита “0” — такой же.

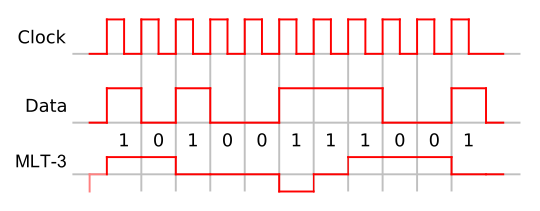
Поскольку код незащищен от долгих последовательностей “нулей” или “единиц”, то это может привести к проблемам синхронизации. Поэтому перед передачей, заданную последовательность битов рекомендуется предварительно закодировать кодом предусматривающим скремблирование (скремблер предназначен для придания свойств случайности передаваемой последовательности данных с целью облегчения выделения тактовой частоты приемником). Используется в [**Fast Ethernet**](http://ru.wikipedia.org/wiki/Fast_Ethernet)100Base-FX и 100Base-T4.

**Системы с трёхуровневым кодированием**

[**RZ**](http://en.wikipedia.org/wiki/Return-to-zero)

Это трехуровневый код. После значащего уровня сигнала в первой половине передаваемого бита информации следует возврат к нулевому уровню. Переход к нему происходит в середине бита. Логическому нулю при этом соответствует положительный импульс, логической единице — отрицательный. Здесь на 1 бит приходится 2 изменения уровня напряжения, поэтому для скорости в 10 Мбит/сек требуется пропускная способность в 10 МГц.

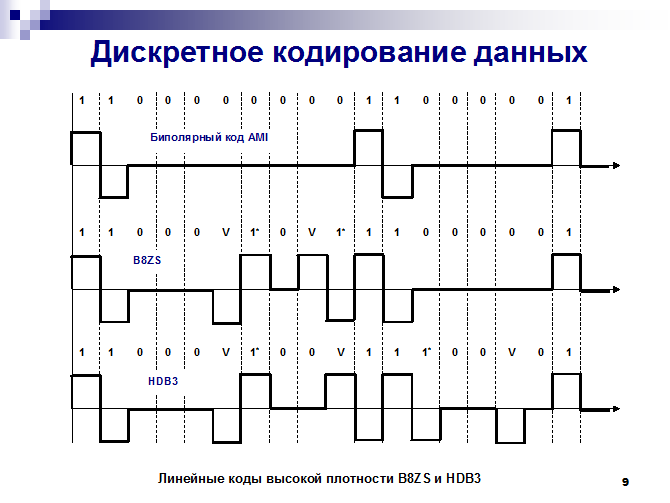
[**MLT-3**](http://en.wikipedia.org/wiki/MLT-3_encoding)

[](http://commons.wikimedia.org/wiki/File:MLT3encoding.svg?uselang=ru)

Пример кодирования MLT-3.

MLT-3 Multi Level Transmission — 3 (многоуровневая передача) — немного схож с кодом NRZI, но в отличие от последнего имеет три уровня сигнала. Единице соответствует переход с одного уровня сигнала на другой, причем изменение уровня сигнала происходит последовательно с учетом предыдущего перехода. При передаче “нуля” сигнал не меняется.

Этот код, так же как и NRZI нуждается в предварительном кодировании. Используется в [**Fast Ethernet 100Base-TX**](http://ru.wikipedia.org/wiki/Fast_Ethernet#100BASE-T)**.**



**Для улучшения кода Bipolar AMI используются два метода, основанные на искусственном искажении последовательности нулей запрещёнными символами.**

**B8ZS** (Bipolar with 8 Zeros Substitution) и метод **HDB3** (High-Density Bipolar 3-Zeros) для корректировки кода AMI.

**Исходный код состоит из двух длинных последовательностей нулей: в первом случае — из 8, а во втором — из 5.**

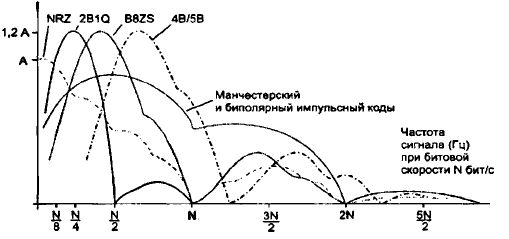
**Код B8ZS исправляет только последовательности, состоящие из 8 нулей.** Для этого он после первых трех нулей вместо оставшихся пяти нулей вставляет пять цифр: V - l\* - 0 –V - l\*. V здесь обозначает сигнал единицы, запрещенной для данного такта полярности, то есть сигнал, не изменяющий полярность предыдущей единицы, 1\* — сигнал единицы корректной полярности, а знак звездочки отмечает тот факт, что в исходном коде в этом такте была не единица, а ноль. В результате на 8 тактах приемник наблюдает 2 искажения — очень маловероятно, что это случилось из-за шума на линии или других сбоев передачи. Поэтому приемник считает такие нарушения кодировкой 8 последовательных нулей и после приёма заменяет их на исходные 8 нулей.

Код B8ZS построен так, что его постоянная составляющая равна нулю при любых последовательностях двоичных цифр.

**Код HDB3 исправляет любые четыре подряд идущих нуля в исходной последовательности.**

Правила формирования кода HDB3 более сложные, чем кода B8ZS. Каждые четыре нуля заменяются четырьмя сигналами, в которых имеется один сигнал V. Для подавления постоянной составляющей полярность сигнала V чередуется при последовательных заменах. Кроме того, для замены используются два образца четырёхтактовых кодов. Если перед заменой исходный код содержал нечётное число единиц, то используется последовательность 000V, а если число единиц было чётным — последовательность l\*00V.

Улучшенные потенциальные коды обладают достаточно узкой полосой пропускания для любых последовательностей единиц и нулей, которые встречаются в передаваемых данных. Коды, полученные из потенциального путём логического кодирования, обладают более узким спектром, чем манчестерский, даже при повышенной тактовой частоте. Этим объясняется применение потенциальных избыточных и скрэмблированных кодов в современных технологиях, подобных FDDI, Fast Ethernet, [Gigabit Ethernet](http://click02.begun.ru/click.jsp?url=8erDkbK*v75PiiYYwqQC3SP9lJZVxG86bTBcyZLtEKozKOf5d4Cx5vo2rE33lHF2aWrsHbR2VfNZ0NVjhdIAePKCYerjbstmjVssx71AYJ2BEHUUDKsr9cER5VE3HSVVeo0*YAW9N2Ud*okQL4mhjQUN3yWzdIM5P-DFuWv-rjOpAkNzs0FBudMj6cbxhY-t0fprb0hqAkL54AzgwsJVmk1P2fQvcMe2Nmk-k*6ZRzJ1kuEURvXbRckawHMttm8QAEvHKhXugDOtk21VpvZwfmCYc4z2yGBUXW2puYQBBwjMXMdWvUrLU52QQp4R3Tz0wNiT1OzuOdYovoOpK5a20N-r7o6Xo6mBbfjFkQEfGenT6TmraAf*agrfWqi4nTDkksAIfYBNRSvsrtq70tABpNLUw-IDWIJI746vOLyxGDc1uc6aXUmEcP-9V05mdvoIEy9Aefg-cVs*pAoOvArb0JgqaEMLVqy5kSg7pFLjxhpoOd6nBFA6zKZZn17eC9lbM2knqphuwDIWI6LNib8kTlAVCQxPc4Sb3ViNKCuiHnQKPbvvR3rk8NSvK7O2ldv4xiSo-w&eurl%5B%5D=8erDkTc2NzZhJi5Wdx9r8eVWj7s9FpZgFJc8vtZaUBA3f32r), ISDN и т. п. вместо манчестерского и биполярного импульсного кодирования.



Спектры потенциальных импульсных кодов.

**2. Практическая часть**

**Задания первого уровня сложности**

**Задание 1.**

Скремблер реализует соотношение **Bi=Ai Bi-3Bi-5.**

Код исходной последовательности **001010010011.**

**Найти:**

Код на выходе скремблера.

**Записать:**

Последовательность формирования кода на выходе дескремблера.

**Задание 2.**

Скремблер реализует соотношение **Bi=Ai Bi-2Bi-4.**

Код исходной последовательности **101110110011.**

**Найти:**

Код на выходе скремблера.

**Записать:**

Последовательность формирования кода на выходе дескремблера.

**Задание 3.**

Скремблер реализует соотношение **Bi=Ai Bi-1Bi-2.**

Код исходной последовательности **101000110011.**

**Найти:**

Код на выходе скремблера.

**Записать:**

Последовательность формирования кода на выходе дескремблера.

**Задание 4.**

Нарисовать диаграмму напряжений для двоичного кода 1101011 в форматах БВН (NRZ), БВН-1 (NRZI), БВН-0 (NRZ0), ВН, ВН-П и ДФ-0.

**Задание 5.**

Ответить на контрольные вопросы.

**Задания второго уровня сложности**

**Задание 1.**

Выполнить задания первого уровня сложности.

**Задание 2.**

Нарисовать диаграмму напряжений для двоичного кода 1101011 при манчестерском кодировании.

**Задание 3.**

Нарисовать диаграмму напряжений для двоичного кода 1101011 при тринарном кодировании.

**Задание 4.**

Нарисовать диаграмму напряжений для двоичного кода 1101011 при кодировании биполярным кодом AMI

**Задание 5.**

Нарисовать диаграмму напряжений для двоичного кода 1101011 при кодировании кодом B8ZS.

**Задание 6.**

Нарисовать диаграмму напряжений для двоичного кода 1101011 при кодировании кодом HDB3.

**Задание 7.**

Нарисовать диаграмму напряжений для двоичного кода 1101011 при кодировании кодом 2B1Q.

**Задание 8.**

Нарисовать диаграмму напряжений для двоичного кода 1101011 при кодировании кодом MLT-3

**Контрольные вопросы**

1. Требования к двоичным сигналам на входе модема.

2. Сущность скремблирования.

3. Сущность скремблирования сложением сигнала с ПСП.

4. Сущность скремблирования с помощью потенциального кода.

5. Сущность самосинхронизирующегося и аддитивного скремблеров.

6. Задачи, решаемые скремблированием в СДЭ.

7. Сущность свойства «размножения ошибок» в скремблере с самосинхронизацией.

8. Сущность «критических ситуаций» и борьба с ними в скремблере с самосинхронизацией.

9. Достоинства аддитивного скремблера.

10. Как влияет скремблирование на спектр дискретного сигнала?

11. Сущность физического и логического кодирования.