

**ФЕДЕРАЛЬНОЕ АГЕНТСТВО СВЯЗИ**

**Федеральное государственное образовательное бюджетное  
учреждение высшего профессионального образования  
«САНКТ-ПЕТЕРБУРГСКИЙ  
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТЕЛЕКОММУНИКАЦИЙ  
им. проф. М. А. БОНЧ-БРУЕВИЧА»**

---

**С. С. Владимиров**

**СИСТЕМЫ И УСТРОЙСТВА  
ПЕРЕДАЧИ ДАННЫХ**

**Практикум**

**СПб ГУТ)))**

**Санкт-Петербург  
2014**

УДК 621.391(077)

ББК 32.811.3я73

В 57

Рецензент

заведующий кафедрой ОПДС, профессор,  
доктор технических наук *О. С. Когновицкий*

*Рекомендовано к печати редакционно-издательским советом СПбГУТ*

**Владимиров, С. С.**

В 57 Системы и устройства передачи данных : практикум / С. С. Владимиров ; СПбГУТ. — СПб, 2014 — 36 с.

Учебное пособие призвано ознакомить студентов старших курсов с одномерными алгоритмами сжатия цифрового факсимильного сигнала и методами линейного кодирования. Представленный материал служит справочным и методическим пособием при выполнении курса практических работ по дисциплинам «Системы и устройства передачи данных» и «Системы документальной электросвязи».

Предназначено для студентов, обучающихся по направлениям 210700.62 «Инфокоммуникационные технологии и системы связи» и 230100.62 «Информатика и вычислительная техника».

**УДК 621.391(077)**

**ББК 32.811.3я73**

- © Владимиров С. С., 2014
- © Федеральное государственное образовательное бюджетное учреждение высшего профессионального образования «Санкт-Петербургский государственный университет телекоммуникаций им. проф. М. А. Бонч-Бруевича», 2014

# **1. Исследование эффективности одномерных алгоритмов сжатия цифрового факсимильного сигнала**

## **1.1. Цель работы**

На примере сжатия двух строк двухградационного изображения оценить эффективность следующих одномерных алгоритмов сжатия цифрового факсимильного сигнала:

1. Кодирование длин серий (КДС).
2. Адресно-позиционное кодирование (АПК).
3. Модифицированный код Хаффмана.

## **1.2. Теоретические сведения**

При переходе от аналоговых факсимильных аппаратов к цифровым возникла необходимость использования алгоритмов сжатия для передачи факсимильного изображения. Основной причиной явилось то, что при переходе к цифровому факсимильному сигналу увеличивается полоса частот сигнала, а следовательно, требуется более широкий канал связи для передачи по сравнению с передачей аналогового сигнала.

Основным органом, издающим рекомендации по вопросам телекоммуникаций, является одно из подразделений Международного Союза Электросвязи, МСЭ (International Telecommunications Union, ITU) — ITU-T (International Telecommunication Union — Telecommunication sector. Сектор стандартизации электросвязи МСЭ).

Первые разработанные в ITU-T стандарты факсимильной связи называются T.2 (известен также как Group 1) и T.3 (Group 2). В 1980 они были заменены алгоритмом T.4 (Group 3), а позднее, в 1988, был введен T.6 (Group 4). В настоящее время Group 3 используется в факс-машинах, разработанных для телефонных сетей общего пользования. Эти аппараты работают на скорости 9600 бод. Алгоритм Group 4 разработан для эксплуатации в цифровых сетях, таких как ISDN. Они обычно работают на скорости 64 кбод. Оба метода обеспечивают фактор сжатия 10:1 и выше, сокращая время передачи страницы типичного документа до минуты в первом случае, и до нескольких секунд во втором. Данные алгоритмы предполагают обработку ошибок на более низком уровне. Рекомендации T.4 и T.6 позволяют пересылать не только чёрно-белые изображения, но и изображения в оттенках серого и цветные.

Спецификации T.4 и T.6 описывают различные требования к аппаратуре, включая разрешение сканирования и печати, допуск на размеры и временные ограничения. Процесс кодирования и декодирования может быть реализован как программно, так и на аппаратном уровне.

Документ сканируется построчно, с переводом каждой строки в последовательность черных и белых элементов, называемых пелами или пикселями (от англ. pel/pixel, Picture Element). Термин «пел» на сегодня считается устаревшим. Часто, особенно в названиях единиц измерения, употребляется термин «точка» (англ. dot). Горизонтальное разрешение всегда составляет 8,05 точек на миллиметр (примерно 205 точек на дюйм). Таким образом, строка стандартной длины 8,5 дюймов (формат Letter 8,5 × 11 дюймов — федеральный стандарт США) конвертируется в 1728 пикселей. Стандарт Т.4, однако, предписывает сканирование строки длиной около 8,2 дюймов, что производит 1664 пикселя. В модификации стандарта Т.4, датированной 1992 г., добавляется новый режим кодирования с более высоким разрешением: 16 точек на миллиметр по горизонтали.

Вертикальное разрешение составляет 3,85 линий на миллиметр (стандартная мода) или 7,7 линий на миллиметр (тонкая мода). Во многих факс-машинах имеется также сверхтонкая мода, при которой сканируется 15,4 линий на миллиметр (модификация Т.4 1992 года).

В СССР и Российской Федерации вопросы факсимильной связи регламентируются государственными и отраслевыми стандартами, часть которых приведена ниже.

1. ГОСТ 23151–78. Аппаратура факсимильная. Термины и определения
2. ГОСТ 25872–83. Аппаратура факсимильная со средствами сокращения избыточности для передачи и приема метеорологических карт. Основные параметры
3. ГОСТ 26348–84. Аппаратура факсимильная со средствами сокращения избыточности для передачи и приема документальной информации. Основные параметры
4. ГОСТ 26631–85. Аппаратура факсимильная со средствами сокращения избыточности. Основные параметры
5. ГОСТ 26557–85. Сигналы передачи данных, поступающие в каналы связи. Энергетические параметры
6. ГОСТ 12922–89. Аппаратура факсимильная. Типы
7. ОСТ 45.120–97. Фотокопии электронного изображения. Технические требования
8. ГОСТ 25807–2000. Фотокопии газет факсимильные. Технические требования
9. ГОСТ Р 51826–2001. Системы и аппаратура факсимильной связи. Параметры

Для сжатия факсимильного сигнала используют одномерные и двумерные алгоритмы сжатия. Среди одномерных алгоритмов сжатия факсимильного изображения можно выделить следующие:

1. Кодирование длин серий, КДС (Run-length encoding, RLE).
2. Адресно-позиционное кодирование, АПК.
3. Модифицированный код Хаффмана (Modified Huffman, MH).

Для оценки методов сжатия используется коэффициент сжатия, который определяется по формуле:

$$K = \frac{N}{N_{\text{СЖ}}},$$

где  $N$  — число элементов на изображении,  $N_{\text{СЖ}}$  — число двоичных разрядов для представления сжатого изображения (с учетом служебных кодовых комбинаций).

Еще одним критерием оценки различных методов сжатия является сложность построения их алгоритмов, которая в свою очередь определяет максимальные затраты времени на обработку сигнала процессором и объем вычислительных ресурсов. Благодаря постоянному совершенствованию цифровой элементной базы, относительная стоимость обработки сигналов постоянно снижается по сравнению с общей стоимостью аппаратуры факсимильной связи. Кроме того, тактовые частоты современных процессоров в среднем на пять порядков выше верхней частоты факсимильного сигнала, поэтому время передачи в меньшей степени зависит от сложности алгоритмов программ и, соответственно, от скорости обработки цифрового сигнала изображения. В значительной степени скорость передачи факсимильной информации определяется скоростью развертки и временем прохождения трафика в сети. Таким образом, сложность алгоритмов сжатия и восстановления не является определяющим критерием при обосновании и выборе различных методов.

При рассмотрении алгоритмов сжатия будем учитывать, что исходный двухуровневый факсимильный сигнал от каждой строки развертки изображения разбивается на отдельные элементы — дискретизируется. Элементам белого приписывается значение логического нуля, а элементам черного — единицы. В результате получается массив строк, состоящих из последовательностей нулей (белые участки/серии) и последовательностей единиц (черные участки/серии).

Далее будут подробно рассмотрены одномерные алгоритмы сжатия факсимильного изображения.

### 1.2.1. Кодирование длин серий

В алгоритме кодирования длин серий (КДС) для кодирования длин участков выбираются два числа:

1.  $m_B$  — для кодирования длин белых участков.
2.  $m_C$  — для кодирования длин черных участков.

При этом, с учётом того, что, как правило, в среднем длины черных участков изображения много короче белых (поскольку чаще всего по факсимильной связи передаются документы, содержащие текстовую информацию), выполняется следующее условие:

$$m_B > m_C.$$

Длины белых участков кодируются  $m_B$ -элементными кодовыми комбинациями, а длины черных —  $m_C$ -элементными кодовыми комбинациями.

ГОСТ 25872–83 на факсимильную аппаратуру для передачи и приема метеорологических карт описывает алгоритм построения кодирования факсимильного изображения, который незначительно отличается от алгоритма КДС. Согласно этому алгоритму  $m_B = 6$  и  $m_C = 3$ . Соответственно, белые участки кодируются шестиэлементными кодовыми комбинациями, а черные участки — трехэлементными.

Согласно алгоритму КДС, строка факсимильного изображения всегда начинается с белой серии. Если строка начинается с черного участка, считается, что перед ним стоит белая серия длиной ноль.

Для определения начала строки передается служебная комбинация «код начала строки» (КНС), состоящая из  $m_B + m_C$  нулей.

**Кодирование белых участков.** Одной  $m_B$ -элементной кодовой комбинацией можно закодировать до  $N_B = 2^{m_B} - 1$  белых пикселей. Таким образом, возникает три варианта кодирования белого участка длиной  $L_B$ :

1.  $L_B$  меньше  $N_B$ . В этом случае белый участок кодируется одной  $m_B$ -элементной кодовой комбинацией.
2.  $L_B$  больше и не кратно  $N_B$ . Белый участок кодируется несколькими  $m_B$ -элементными комбинациями, причем все они, кроме последней, состоят из одних единиц.
3.  $L_B$  равно или кратно  $N_B$ . Белый участок кодируется одной или несколькими  $m_B$ -элементными комбинациями, состоящими из одних единиц, и для определения окончания белого участка передается служебная кодовая комбинация из  $m_B$  нулей.

**Кодирование черных участков** производится аналогично. Одной  $m_{\text{ч}}$ -элементной кодовой комбинацией можно закодировать до  $N_{\text{ч}} = 2^{m_{\text{ч}}} - 1$  черных пелов. Таким образом, возникает три варианта кодирования черного участка длиной  $L_{\text{ч}}$ :

1.  $L_{\text{ч}}$  меньше  $N_{\text{ч}}$ . В этом случае черный участок кодируется одной  $m_{\text{ч}}$ -элементной кодовой комбинацией.
2.  $L_{\text{ч}}$  больше и не кратно  $N_{\text{ч}}$ . Черный участок кодируется несколькими  $m_{\text{ч}}$ -элементными комбинациями, причем все они, кроме последней, состоят из одних единиц.
3.  $L_{\text{ч}}$  равно или кратно  $N_{\text{ч}}$ . Черный участок кодируется одной или несколькими  $m_{\text{ч}}$ -элементными комбинациями, состоящими из одних единиц, и для определения окончания черного участка передается служебная кодовая комбинация из  $m_{\text{ч}}$  нулей.

Метод КДС обладает низкой помехоустойчивостью. Неверно принятый код длины серии приводит к так называемому «треку ошибок», поскольку декодер воспроизведет серию не той длины. Таким образом, даже одиночная ошибка за счет помехи в канале приводит к «разрушению» строки.

***Пример сжатия строки изображения алгоритмом КДС***

Для примера взяты две строки двухградационного изображения длиной 1664 пикселя каждая (рис. 1.1).



Рис. 1.1. Строки двухградационного изображения для сжатия по методу КДС

Для кодирования длин белых и черных участков изображения выбраны следующие параметры:

$$m_{\text{б}} = 6; \quad m_{\text{ч}} = 4.$$

Одной  $m_{\text{б}}$ -элементной кодовой комбинацией можно закодировать до  $N_{\text{б}} = 2^{m_{\text{б}}} - 1 = 2^6 - 1 = 63$  белых пикселей.

Одной  $m_{\text{ч}}$ -элементной кодовой комбинацией можно закодировать до  $N_{\text{ч}} = 2^{m_{\text{ч}}} - 1 = 2^4 - 1 = 15$  черных пикселей.

Служебная комбинация КНС будет состоять из  $m_{\text{б}} + m_{\text{ч}} = 6 + 4 = 10$  нулей: 0000000000.

**Кодирование первой строки** (рис. 1.1). Строка начинается с серии белых пикселей длиной 19. Поскольку 19 меньше  $N_B = 63$ , серия кодируется одной кодовой комбинацией длиной  $m_B = 6$ : 010011.

Далее идет серия черных пикселей длиной 26. 26 больше  $N_C = 15$ , но не кратно ему. Разложим 26 на участки:  $26 = 15 + 11$ . Таким образом, серия кодируется одной кодовой комбинацией длиной  $m_C = 4$ , состоящей из единиц, и одной комбинацией, кодирующей остаток 11: 1111.1011. Точка между комбинациями приведена для лучшей читаемости и играет роль разделителя.

Длина следующей серии белых пикселей (126) кратна  $N_B = 63$ . То есть,  $126 = 63 + 63 + 0$ . Таким образом, серия кодируется двумя кодовыми комбинациями длиной  $m_B = 6$ , состоящими из единиц, и комбинацией длиной 6, состоящей из нулей (код нулевого остатка): 111111.111111.000000.

По аналогии кодируется и последняя серия черных пикселей длиной 1493. Ее можно разложить на 99 участков длиной 15 и участок-остаток длиной 8. Таким образом, эта строка закодируется как:  $99 \times 1111.1000$ . Для наглядности вместо выписывания всех 99 комбинаций указан множитель.

Процесс кодирования первой строки можно представить в сокращенном виде (табл. 1.1).

Таблица 1.1

Кодирование первой строки примера по методу КДС

Серия	Длина	Кодовая комбинация
КНС		0000000000
БС	19	010011
ЧС	$26 = 15 + 11$	1111.1011
БС	$126 = 63 + 63 + 0$	111111.111111.000000
ЧС	$1493 = 99 \cdot 15 + 8$	$99 \times 1111.1000$

**Кодирование второй строки** (рис. 1.1) производится аналогично. Единственным отличием является то, что вторая строка начинается с серии черных пикселей, следовательно, согласно алгоритму КДС, при кодировании такой строки считается, что она начинается с белой серии длиной ноль. Процесс кодирования второй строки показан в сокращенном виде в табл. 1.2.

Таблица 1.2

Кодирование второй строки примера по методу КДС

Серия	Длина	Кодовая комбинация
КНС		0000000000
БС	0	000000
ЧС	6	0110
БС	$63 = 63 + 0$	111111.000000
ЧС	$120 = 8 \cdot 15 + 0$	$8 \times 1111.0000$
БС	$1475 = 23 \cdot 63 + 26$	$23 \times 111111.011010$



Полученные таблицы можно «раскрыть», записав закодированные строки в линию:

0000000000.010011.1111.1011.111111.111111.000000.99×1111.1000  
 0000000000.000000.0110.111111.000000.8×1111.0000.23×111111.011010

Длины закодированных строк равны 442 и 212 бит соответственно. Следовательно, суммарная их длина равна 554 бита, в то время как суммарная длина двух незакодированных строк равна  $1664 \cdot 2 = 3328$  пикселей. Таким образом, коэффициент сжатия этих строк алгоритмом КДС равен

$$K_{\text{сж КДС}} = \frac{3328}{554} \approx 6,01.$$

Следовательно, алгоритм КДС обеспечил сжатие двух строк изображения примерно в шесть раз.

### 1.2.2. Адресно-позиционное кодирование

Алгоритм адресно-позиционного кодирования (АПК) основан на том, что строку двухградационного изображения можно представить в виде последовательных переходов от белого цвета к черному и обратно. Учитывая то, что длина строки изображения зафиксирована в стандарте и является постоянной (1728 или 1664 в зависимости от рекомендации), зная позиции этих переходов относительно начала строки и цвет первой серии пикселей, можно однозначно восстановить исходную последовательность белых и черных серий.

Структура закодированной по методу адресно-позиционного кодирования строки показана на рис. 1.2.

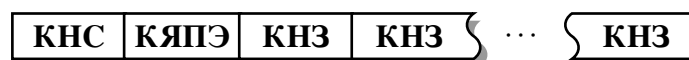


Рис. 1.2. Структура закодированной по методу АПК строки двухградационного изображения

В алгоритме АПК используется равномерное кодирование, при котором все кодовые комбинации (за исключением КЯПЭ) имеют одинаковую длину. При этом длина кодовых комбинаций определяется по формуле

$$k = \log_2 N,$$

где  $N$  — общее число белых и черных пикселей в исходной строке, то есть длина строки.

Код начала строки (КНС) представляет из себя служебную комбинацию, состоящую из  $k$  нулей.

Код яркости первого элемента (КЯПЭ) определяет цвет первой серии пикселей:

- «0» — строка начинается с белой серии;
- «1» — строка начинается с черной серии.

Код нового значения яркости (КНЗ) определяет координату элемента с новым значением яркости (цвета) относительно начала строки изображения. Каждая координата кодируется  $k$ -элементной кодовой комбинацией.

Данный метод не эффективно применять для обработки текстовых факсимильных документов. При передаче контурных карт, чертежей и подобных им изображений избыточность сокращается примерно в 5–6 раз.

Метод АПК менее чувствителен к ошибкам, т. е. обладает большей помехоустойчивостью, чем многие другие методы сжатия. Помехоустойчивость АПК заключается в том, что поражение ошибкой координаты нового значения яркости приводит к искажению небольшой группы элементов строки до следующего нового значения яркости. При кодировании длин серий (КДС) случайная ошибка приводит к сдвигу всех последующих серий до конца строки. Таким образом, подтверждается непреложное правило: помехоустойчивость тем ниже, чем меньше избыточности в передаваемых данных и наоборот; то есть, выигрывая в эффективности сжатия, безусловно теряем в помехозащищенности.

### ***Пример сжатия строки изображения алгоритмом АПК***

Для примера взяты две строки двухградационного изображения длиной 1664 пикселя каждая (рис. 1.3).



*Рис. 1.3. Строки двухградационного изображения для сжатия по методу АПК*

Перед началом кодирования необходимо рассчитать длину кодовых комбинаций, используемых для кодирования координат элемента с новым значением яркости.

$$k = \log_2 N = \log_2 1664 = 11.$$

Таким образом, координату любого элемента строки с новым значением яркости относительно начала строки изображения можно закодировать кодовой комбинацией из 11 бит.

Код начала строки состоит из  $k = 11$  нулей: 00000000000.

**Кодирование первой строки** (рис. 1.3). Первая строка начинается с серии пелов белого цвета, следовательно, код яркости первого элемента (КЯПЭ) будет равен «0».

Координата первого элемента с новым значением яркости будет соответствовать позиции первого элемента черной серии длиной 26 и будет равна длине начальной белой серии плюс единица. Следовательно, его КНЗ будет равен  $19 + 1 = 20$ , а соответствующая кодовая комбинация будет равна: 00000010100.

Следующий элемент с новым значением яркости будет первым элементом белой серии длиной 126, а его КНЗ будет равен  $19 + 26 + 1 = 20 + 26 = 46$ . Кодовая комбинация будет равна: 00000101110.

Последним элементом с новым значением яркости будет первый элемент черной серии длиной 1493. Его код с новым значением яркости равен  $19 + 26 + 126 + 1 = 46 + 126 = 172$ . Кодовая комбинация равна: 00010101100. Далее до конца строки будут располагаться только черные пиксели, и, следовательно, кодирование первой строки на этом завершается. Длина строки фиксирована, поэтому факсимильный аппарат на приемном конце сможет безошибочно восстановить строку.

Кодирование первой строки в сокращенном виде показано в табл. 1.3.

Таблица 1.3

Кодирование первой строки примера (рис. 1.3) по методу АПК

Серия	Позиция КНЗ	Кодовая комбинация
КНС		00000000000
КЯПЭ		0
КНЗ	$20 = 19 + 1$	00000010100
КНЗ	$46 = 20 + 26$	00000101110
КНЗ	$172 = 46 + 26$	00010101100

**Кодирование второй строки** (рис. 1.3) производится таким же образом (табл. 1.4). Единственным отличием является то, что вторая строка начинается с серии черных пикселей, поэтому КЯПЭ для нее будет равен «1».

Таблица 1.4

Кодирование второй строки примера (рис. 1.3) по методу АПК

Серия	Позиция КНЗ	Кодовая комбинация
КНС		00000000000
КЯПЭ		1
КНЗ	$7 = 6 + 1$	00000000111
КНЗ	$70 = 7 + 63$	00001000110
КНЗ	$190 = 70 + 120$	00010111110

Полученные таблицы можно «раскрыть», записав закодированные строки в линию:

```
000000000000.0.00000010100.00000101110.00010101100  
000000000000.1.00000000111.00001000110.00010111110
```

Длина каждой из закодированных строк равна 45 бит. Следовательно, суммарная их длина равна 90 бит, в то время как суммарная длина двух незакодированных строк равна 3328 пикселей. Таким образом, коэффициент сжатия этих строк алгоритмом АПК равен

$$K_{\text{сж АПК}} = \frac{3328}{90} \approx 36,98.$$

Алгоритм АПК обеспечил сжатие двух строк изображения примерно в тридцать семь раз.

### ***1.2.3. Модифицированный код Хаффмана***

Модифицированный код Хаффмана (Modified Huffman — МН) относится к одномерным алгоритмам оптимального неравномерного кодирования. Он учитывает статистику исходного двоичного изображения: наиболее вероятные длины серий кодируются наиболее короткими кодовыми комбинациями, а наименее вероятные — наиболее длинными. При использовании этого кода серии черных и белых элементов кодируются в соответствии с кодовой таблицей (в Рек. Т.4 (11/1988) это таблицы 1 и 2). Размер кодовой комбинации в таблице обратно пропорционален вероятности появления соответствующей длины серии. Согласно Рек. Т.4 МН используется как алгоритм сжатия «по умолчанию» для факсимильной аппаратуры третьей группы.

Характерная особенность кодовых комбинаций кода МН заключается в том, что они не содержат повторяющихся последовательностей единичных и нулевых элементов. Этим обеспечивается однозначное декодирование, и, кроме того, не требуется вводить дополнительные элементы для обозначения начала или конца кодовых комбинаций. Иначе говоря, в любой последовательности кодовых комбинаций всегда однозначно определяются начало и конец каждой из них, т. е. код является самосинхронизирующимся.

Методы сжатия с использованием кодов Хаффмана обладают низкой помехоустойчивостью. Неверно принятая кодовая комбинация приводит к так называемому «треку ошибок», поскольку декодер воспроизведет серию не той длины (и не той градации яркости). Таким образом, даже одиночная ошибка за счет помехи в канале может вызвать поток ошибок при декодировании сигнала.

Общая схема кодирования факсимильного документа кодом МН представлена на рис. 1.4.

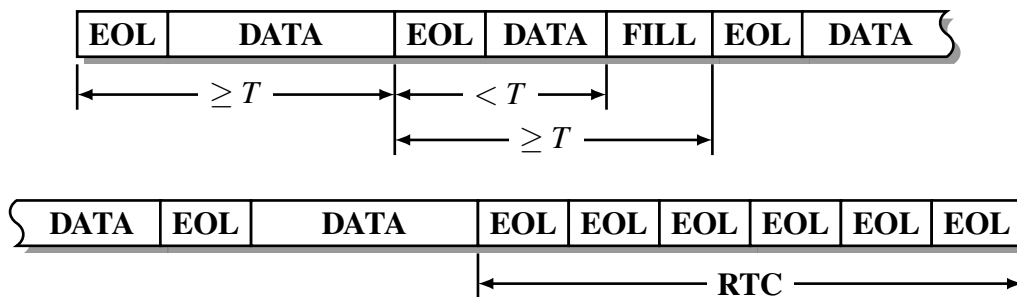


Рис. 1.4. Схема кодирования факсимильного документа кодом МН

**Данные (Data).** Строка Data состоит из серий кодовых слов, которыми последовательно закодированы чередующиеся серии белых и черных пикселей, составляющие строку исходного изображения длиной 1728 пикселей. Для того, чтобы гарантировать удержание синхронизации и цвета на приеме, все строки Data начинаются кодовым словом длины серии белых пикселей. Если фактическая строка развертки начинается с черной серии, передается кодовое слово, соответствующее нулевой длине белой серии. Длина серий черного или белого вплоть до максимальной длины одной строки развертки (1728 элементов изображения) определяется с помощью кодовых слов по табл. 1.5 и табл. 1.6. Существуют кодовые слова двух типов: завершающие и начальные кодовые слова. Каждая длина серии представляется либо одним словом завершающего кода, либо одним начальным кодовым словом, за которым следует завершающее кодовое слово.

Длины серий в диапазоне от 0 до 63 пикселей кодируются соответствующими завершающими кодовыми словами (табл. 1.5). Для длин черных и белых серий существуют разные перечни кодовых слов.

Длины серий в диапазоне от 64 до 1728 пикселей кодируются сперва начальным кодовым словом (табл. 1.6), представляющим длину серии, которая равна или короче требуемой. Затем следует завершающее кодовое слово (табл. 1.5), соответствующее разности между реальной длиной серии и длиной серии, представленной начальным кодом. Необходимо заметить, что в случае, когда реальная длина серии пикселей кратна 64, т. е. соответствует значению из таблицы начальных кодов, всё равно используется завершающее кодовое слово, соответствующее серии длиной ноль.

Следует отметить, что, хотя Рек. Т.4 указывает, что длина строки должна быть равна 1728 или 1664 пикселя, в таблице завершающих кодовых слов (табл. 1.6) присутствуют комбинации для серий пикселей длиной от 1792 до 2560, которые были добавлены с расчетом на возможное развитие стандарта.

**Конец строки (EOL).** Это кодовое слово следует за каждой строкой Data. Это специальное кодовое слово, которое никогда не появляется в действительной строке Data, поэтому возможна ресинхронизация после пакета

ошибок. Кроме того, этот сигнал передается перед первой строкой Data каждой страницы.

Код EOL состоит из одиннадцати нулей и единицы: 000000000001.

Таблица 1.5

Таблица кодов завершения для кода МН (табл. 1 Рек. Т.4)

Длина серии	Белые пелы	Черные пелы	Длина серии	Белые пелы	Черные пелы
0	00110101	0000110111	32	00011011	000001101010
1	000111	010	33	00010010	000001101011
2	0111	11	34	00010011	000011010010
3	1000	10	35	00010100	000011010011
4	1011	011	36	00010101	000011010100
5	1100	0011	37	00010110	000011010101
6	1110	0010	38	00010111	000011010110
7	1111	00011	39	00101000	000011010111
8	10011	000101	40	00101001	000001101100
9	10100	000100	41	00101010	000001101101
10	00111	0000100	42	00101011	000011011010
11	01000	0000101	43	00101100	000011011011
12	001000	0000111	44	00101101	000001010100
13	000011	00000100	45	00000100	000001010101
14	110100	00000111	46	00000101	000001010110
15	110101	000011000	47	00001010	000001010111
16	101010	0000010111	48	00001011	000001100100
17	101011	0000011000	49	01010010	000001100101
18	0100111	0000001000	50	01010011	000001010010
19	0001100	00001100111	51	01010100	000001010011
20	0001000	00001101000	52	01010101	000000100100
21	0010111	00001101100	53	00100100	000000110111
22	0000011	00000110111	54	00100101	000000111000
23	0000100	00000101000	55	01011000	000000100111
24	0101000	00000010111	56	01011001	000000101000
25	0101011	00000011000	57	01011010	000001011000
26	0010011	000011001010	58	01011011	000001011001
27	0100100	000011001011	59	01001010	000000101011
28	0011000	000011001100	60	01001011	000000101100
29	00000010	000011001101	61	00110010	000001011010
30	00000011	000001101000	62	00110011	000001100110
31	00011010	000001101001	63	00110100	000001100111

Таблица 1.6

Таблица начальных кодов для кода МН (табл. 2 Рек. Т.4)

Длина серии	Белые пелы	Черные пелы
64	11011	0000001111
128	10010	000011001000
192	010111	000011001001
256	0110111	000001011011
320	00110110	000000110011
384	00110111	000000110100
448	01100100	000000110101
512	01100101	0000001101100
576	01101000	0000001101101
640	01100111	0000001001010
704	011001100	0000001001011
768	011001101	0000001001100
832	011010010	0000001001101
896	011010011	0000001110010
960	011010100	0000001110011
1024	011010101	0000001110100
1088	011010110	0000001110101
1152	011010111	0000001110110
1216	011011000	0000001110111
1280	011011001	0000001010010
1344	011011010	0000001010011
1408	011011011	0000001010100
1472	010011000	0000001010101
1536	010011001	0000001011010
1600	010011010	0000001011011
1664	011000	0000001100100
1728	010011011	0000001100101
1792	00000001000	00000001000
1856	00000001100	00000001100
1920	00000001101	00000001101
1984	000000010010	000000010010
2048	000000010011	000000010011
2112	000000010100	000000010100
2176	000000010101	000000010101
2240	000000010110	000000010110
2304	000000010111	000000010111
2368	000000011100	000000011100
2432	000000011101	000000011101
2496	000000011110	000000011110
2560	000000011111	000000011111

**Заполнение (FILL).** Существует минимальное время передачи полной кодированной строки развертки ( $T$ ), которое задается при установлении соединения для передачи факсимильного изображения. В том случае, если суммарное время передачи Data и EOL будет меньше  $T$ , между Data и EOL вставляется сигнал FILL.

FILL представляет из себя последовательность нулей переменной длины. При этом должно соблюдаться условие:

$$T_{Data} + T_{FILL} + T_{EOL} \geq T.$$

**Возврат к управлению (RTC).** Сигнал RTC обозначается путем передачи шести последовательных сигналов EOL. После сигнала RTC передатчик будет передавать команды постсообщения в соответствии с цикловым форматом и скоростями передачи сигналов управления, определенных в Рек. Т.30.

### *Пример сжатия строки изображения алгоритмом МН*

Для примера взяты две строки двухградационного изображения длиной 1664 пикселя каждая (рис. 1.5).

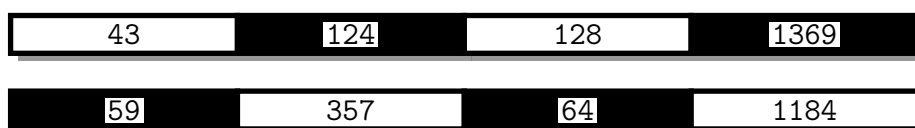


Рис. 1.5. Строки двухградационного изображения для сжатия по методу МН

При кодировании примера не указывается сигнал RTC — считается, что после этих строк будут передаваться и другие. Таким образом, необходимо указать только служебный сигнал EOL в начале передачи и в конце каждой строки. Сигнал FILL также не учитывается.

**Кодирование первой строки** (рис. 1.5). Первая строка начинается с серии пикселей белого цвета длиной 43. Поскольку длина меньше 64, то эта серия кодируется соответствующей кодовой комбинацией из таблицы кодов завершения (табл. 1.5): 00101100.

Следующая серия черных пикселей имеет длину 124. Больше 63, следовательно ее нужно разбить на две части. Ближайший снизу код из таблицы начальных кодов (табл. 1.6) — 64. Остаток равен  $124 - 64 = 60$ . Он кодируется из таблицы кодов завершения. В итоге будет получена кодовая комбинация: 0000001111.000000101100.

Серия белых пикселей длиной 128 точно соответствует значению из таблицы начальных кодов, поэтому она кодируется этой кодовой комбинацией, а в качестве завершения используется кодовая комбинация, соответствующая белой серии нулевой длины: 10010.00110101.



Завершающая черная серия длиной 1369 кодируется начальным кодом 1344 и кодом завершения 25: 0000001010011.00000011000.

Сокращенно кодирование первой строки представлено в табл. 1.7.

Таблица 1.7

Кодирование первой строки примера (рис. 1.5) по методу МН

Серия	Длина	Кодовая комбинация
EOЛ		000000000001
БС	43	00101100
ЧС	124 = 64 + 60	0000001111.000000101100
БС	128 = 128 + 0	10010.00110101
ЧС	1369 = 1344 + 25	0000001010011.00000011000
EOЛ		000000000001

При кодировании второй строки (рис. 1.5) необходимо учесть, что она начинается с серии черных пелов, следовательно в начале необходимо указать кодовое слово, соответствующее нулевой длине белой серии. Код EOЛ ставится только в конце этой строки.

Кодирование второй строки в сокращенном виде показано в табл. 1.8.

Таблица 1.8

Кодирование второй строки примера (рис. 1.5) по методу МН

Серия	Длина	Кодовая комбинация
БС	0	00110101
ЧС	59	000000101011
БС	357 = 320 + 37	00110110.00010110
ЧС	64 = 64 + 0	0000001111.0000110111
БС	1184 = 1152 + 32	011010111.00011011
EOЛ		000000000001

Полученные таблицы можно «раскрыть», записав закодированные строки в линию:

000000000001.00101100.0000001111.000000101100.10010.00110101  
 0000001010011.00000011000.000000000001  
 00110101.000000101011.00110110.00010110.0000001111.0000110111  
 011010111.00011011.000000000001

Длины закодированных строк равны 90 и 85 бит соответственно. Следовательно, суммарная их длина равна 175 бит, в то время как суммарная длина двух незакодированных строк равна 3328 пикселей. Таким образом, коэффициент сжатия этих строк кодом МН равен

$$K_{\text{сж МН}} = \frac{3328}{175} \approx 19,02.$$

Алгоритм МН обеспечил сжатие двух строк изображения примерно в девятнадцать раз.

### **1.3. Рекомендуемая литература**

1. Электронный курс «Системы документальной электросвязи». Размещен на сайте кафедры.
2. Электронный курс «Программное обеспечение систем передачи данных». Размещен на сайте кафедры.
3. Д. Сэломон «Сжатие данных, изображений и звука».
4. Р. Л. Хаммел «Последовательная передача данных».
5. В. П. Шувалов, Н. В. Захарченко и др. «Передача дискретных сообщений. Учебник для вузов».

### **1.4. Порядок выполнения задания**

Задание выполняется каждым учащимся индивидуально на бланке, приведённом на странице [21](#).

В работе требуется закодировать в соответствии с заданным вариантом две строки двухградационного изображения, используя алгоритмы КДС, АПК и МН, а также оценить эффективность каждого из алгоритмов сжатия, посчитав для сжатых строк коэффициент сжатия.

Параметры для алгоритма КДС берутся из табл. [1.9](#) по значению последней цифры номера зачетной книжки, две строки двухградационного изображения выбираются из табл. [1.10](#) (первая строка) и табл. [1.11](#) (вторая строка) по значению предпоследней цифры номера зачетной книжки.

### **1.5. Порядок защиты практической работы**

Защита работы может осуществляться одним из нижеперечисленных способов или их сочетанием на усмотрение преподавателя.

1. Устный ответ по теме работы.
2. Тестирование по теме работы
3. Задача на кодирование строки двухградационного изображения.
4. Задача на декодирование закодированной строки двухградационного изображения.
5. Иные варианты на усмотрение преподавателя.

Таблица 1.9

*Параметры для алгоритма КДС  
№ варианта — последняя цифра номера зачетной книжки*

№	0	1	2	3	4	5	6	7	8	9
<b>m<sub>б</sub></b>	5	5	6	6	7	7	7	8	8	8
<b>m<sub>ч</sub></b>	2	3	2	3	3	4	5	3	4	5

Таблица 1.10

*Первая строка исходного изображения  
№ варианта — предпоследняя цифра номера зачетной книжки*

№ вар-та	Белая серия	Черная серия	Белая серия	Черная серия	Белая серия	Черная серия
0	31	3	63	7	1472	152
1	63	120	1280	3	255	7
2	127	144	1408	15	31	3
3	255	31	31	15	1216	180
4	1344	15	63	31	31	244
5	255	212	1152	15	63	31
6	127	3	255	7	1088	248
7	63	440	960	3	255	7
8	31	128	1408	31	127	3
9	1396	31	31	192	63	15

Таблица 1.11

*Вторая строка исходного изображения  
№ варианта — предпоследняя цифра номера зачетной книжки*

№ вар-та	Черная серия	Белая серия	Черная серия	Белая серия	Черная серия	Белая серия
0	31	255	64	1236	15	127
1	15	31	31	127	128	1396
2	7	63	256	255	31	1116
3	3	127	7	63	128	1400
4	184	1152	3	127	7	255
5	7	31	192	1368	3	127
6	31	1268	320	31	15	63
7	15	127	31	1204	320	31
8	7	255	15	63	448	940
9	3	127	7	255	640	696



## Работа 1

### Исследование эффективности одномерных алгоритмов сжатия цифрового факсимильного сигнала

Группа: \_\_\_\_\_ Студент: \_\_\_\_\_

№ зачетной книжки: \_\_\_\_\_

#### Задание

$$m_{\text{бел}} = \quad m_{\text{чер}} =$$

Строки:



#### Алгоритм КДС

$$m_{\text{бел}} = \quad \rightarrow \text{длина серии} = \quad m_{\text{чер}} = \quad \rightarrow \text{длина серии} =$$

Строка 1			Строка 2		
Серия	Длина	Кодовая комбинация	Серия	Длина	Кодовая комбинация

$$K_{\text{сж КДС}} = \text{---} =$$

### Алгоритм АПК

$$L = \quad \rightarrow k = \log_2 \quad \leq$$

Строка 1			Строка 2		
Серия	КНЗ	Кодовая комбинация	Серия	КНЗ	Кодовая комбинация

$$K_{\text{сж АПК}} = \text{---} =$$

### Алгоритм МН

Строка 1			Строка 2		
Серия	Длина	Кодовая комбинация	Серия	Длина	Кодовая комбинация

$$K_{\text{сж ОНК}} = \text{---} =$$

## 2. Методы линейного кодирования

### 2.1. Цель работы

Изучить принципы цифрового линейного кодирования информации. Рассмотреть на примере и получить навыки в кодировании битовых последовательностей линейными кодами и восстановлении последовательностей из закодированного сигнала.

### 2.2. Теоретические сведения

При передаче цифрового двоичного сигнала практически никогда не используется кодирование бита «1» уровнем напряжения, а бита «0» отсутствием напряжения, поскольку в этом случае невозможно отличить бит «0» от отсутствия сигнала. В связи с этим используется линейное кодирование, которое позволяет принимающей стороне однозначно определить начало, конец и середину каждого принятого бита без помощи внешнего таймера. Линейное кодирование на физическом уровне позволяет приёмнику синхронизироваться с передатчиком.

В литературе часто используют термины «цифровое кодирование», «кодирование цифровых сигналов» и «физическое кодирование» для обозначения линейного кодирования. Также не следует путать системы линейного кодирования цифровых сигналов с помехоустойчивыми линейными кодами.

#### *2.2.1. Виды линейного кодирования*

Можно выделить следующие основные виды систем линейного кодирования в зависимости от количества уровней напряжения, которые используются для формирования сигналов.

1. Бинарное (двухуровневое) кодирование.
2. Тринарное (трёхуровневое) кодирование.
3. Тетрачное (четырёхуровневое) кодирование.
4. Кодирование с большим числом уровней.

Подробнее рассмотрим системы, относящиеся к этим видам.

#### *2.2.2. Бинарное (двухуровневое) кодирование*

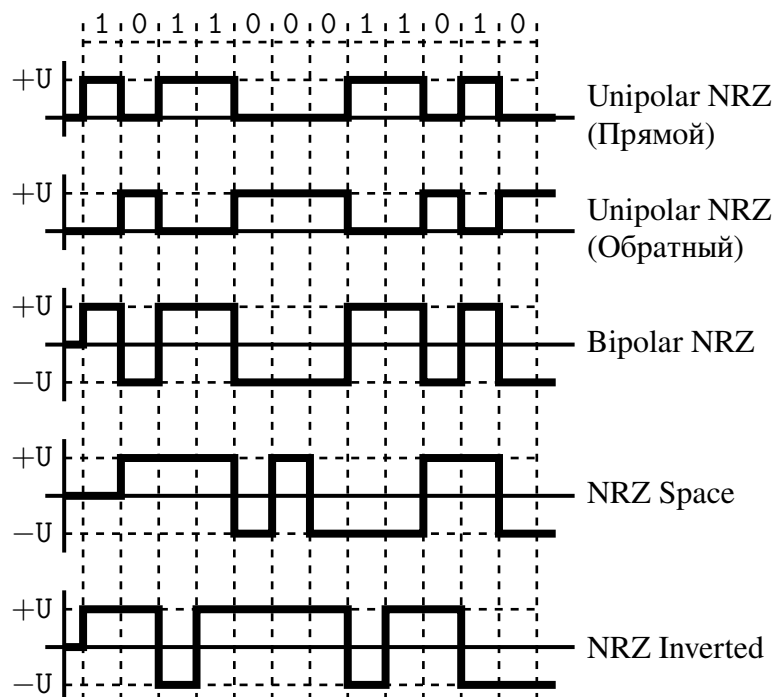
В случае бинарного кодирования используется цифровой сигнал имеющий два возможных уровня напряжения. Как правило, это положительный сигнал  $+U$  и отрицательный сигнал  $-U$ , где  $U$  — уровень напряжения, который должен находиться в пределах определённых границ. Например, в биполярном линейном коде NRZ, используемом в интерфейсе RS-232, уровень

напряжения  $U$  должен находиться в границах  $+5 \dots +12\text{В}$  для передачи «0» и  $-5 \dots -12\text{В}$  для передачи «1».

### **Кодирование без возврата к нулю**

Кодирование без возврата к нулю (Non-Return-to-Zero, NRZ) или потенциальное кодирование является простейшим случаем двухуровневого кодирования. Различают несколько вариантов кодирования NRZ (рис. 2.1).

**Unipolar NRZ Level.** В нём выделяют *Прямой NRZ*, в котором «0» представляется напряжением  $0\text{В}$ , а «1» —  $+U\text{В}$ , и *Обратный NRZ*, где «0» —  $+U\text{В}$ , а «1» —  $0\text{В}$ .



*Рис. 2.1. Кодирование без возврата к нулю (NRZ)*

**Bipolar NRZ Level.** Передаваемый «0» представляется напряжением  $-U\text{В}$ , а «1» —  $+U\text{В}$ .

**NRZ Space.** При передаче «1» уровень сигнала не меняется, «0» кодируется сменой уровня сигнала. Такая схема со сменой уровня сигнала на «нуле» используется, например, в бит-ориентированном протоколе HDLC и протоколе USB. В обоих этих протоколах, чтобы избежать длительных временных интервалов без изменения уровня сигнала, используется вставка «нулевого» бита. В протоколе HDLC «нулевой» бит вставляется после каждых пяти последовательных «1» (за исключением разделителя кадров «01111110»), а в USB — после каждых шести «1».



**NRZ Inverted (NRZI).** Система обратная NRZ Space. «1» кодируется сменой уровня сигнала, при передаче «0» уровень не меняется. Для того, чтобы избавиться от длинных последовательностей «нулей», используется совместно с RLL-кодированием.

### *Манчестерское кодирование*

Такой вид кодирования также называют *фазовым кодированием*. Каждый такт делится на две части. Информация кодируется перепадами потенциала в середине каждого такта (рис. 2.2). «Единица» кодируется перепадом от низкого уровня сигнала к высокому, а «ноль» — обратным перепадом (по стандарту IEEE 802.3, 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.

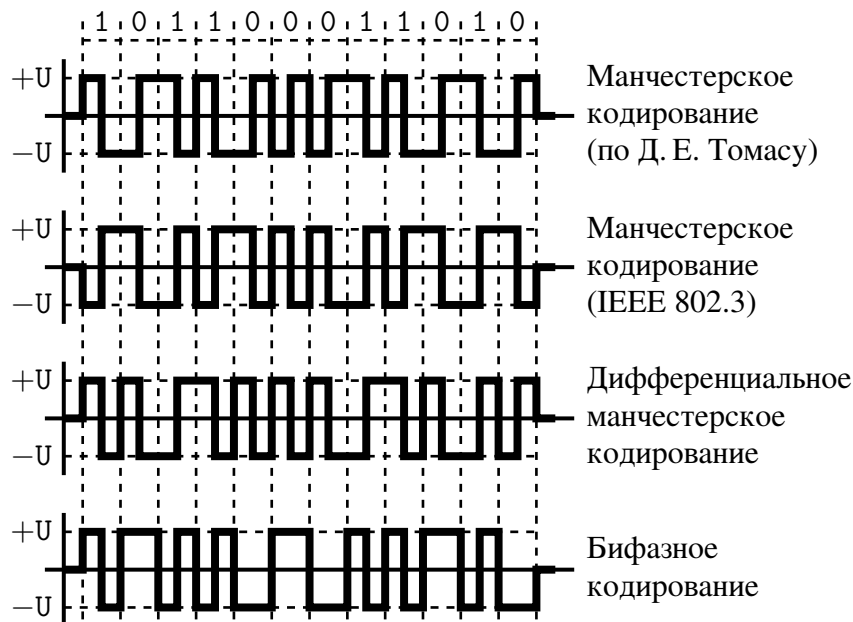


Рис. 2.2. Манчестерское кодирование

Отдельно выделяют *Дифференциальное манчестерское кодирование*. В этом варианте кодирования при передаче «нуля» направление перепада не меняется, а при передаче «единицы» направление перепада меняется на противоположное.

## Бифазное кодирование. *Viphase mark coding (VBC)*

При бифазном кодировании «единица» кодируется перепадом сигнала в середине такта, а «ноль» — тактом без перепада. При этом полярность первого полутакта «единицы» противоположна полярности идущего до неё «нуля». Полярность «нуля» противоположна полярности предыдущего «нуля» или полярности второго полутакта предыдущей «единицы» (рис. 2.2).

Бифазное кодирование используется в интерфейсе S/PDIF, используемом для передачи цифрового аудиосигнала, а также во многих картах с магнитной полосой — в этом случае кодирование чаще называют F2F (frequency/double frequency). Принцип кодирования описан в стандарте ISO/IEC 7811. Также этот способ кодирования использовался в флоппи-дисках одиночной плотности.

### 2.2.3. Тринарное (трёхуровневое) кодирование

В тринарном кодировании используется цифровой сигнал, имеющий три возможных уровня напряжения: положительный сигнал  $+U$ , отрицательный сигнал  $-U$ , где  $U$  — уровень напряжения, который должен находиться в пределах определённых границ, и «нулевой» сигнал.

#### Код с возвратом к нулю

В коде *Return to Zero (RZ)* 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 отрицательный импульс, логической единице — положительный (рис. 2.3). Таким образом, код является квазитроичным. Здесь на 1 бит приходится 2 изменения уровня напряжения, поэтому для скорости в 10 Мбит/сек требуется пропускная способность в 10 МГц.

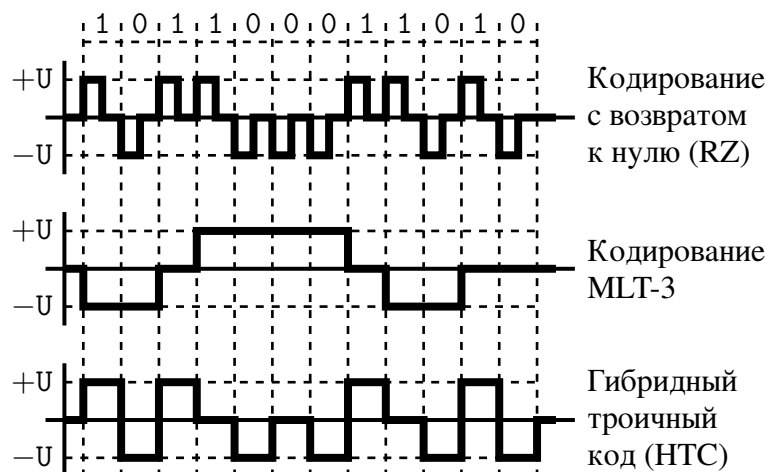


Рис. 2.3. Кодирование с возвратом к нулю. Код MLT-3. Гибридный троичный код

### *Multi Level Transmission 3 (MLT-3)*

Код MLT-3 схож с кодом NRZ-I, но в отличие от него имеет три уровня сигнала. В MLT-3 циклично перебираются уровни напряжений  $-U$ ,  $0$ ,  $+U$ . Смена уровня соответствует передаче сигнала 1 бит, при передаче 0 бит уровень не изменяется (рис. 2.3). Как и в случае NRZ-I, код MLT-3 имеет кодировую эффективность, равную 1 бит/бод, при этом для возврата в предыдущее состояние требуется четыре перехода (бода):  $-U \rightarrow 0$ ,  $0 \rightarrow +U$ ,  $+U \rightarrow 0$ ,  $0 \rightarrow -U$ . В связи с этим реальная частота уменьшается до четверти бода. Передача таким сигналом больше подходит для медных линий. Впервые код MLT-3 был предложен компанией Crescendo Communications для использования в технологии CDDI (FDDI по медному кабелю).

#### *Гибридный троичный код. Hybrid ternary code*

Гибридный троичный код объединяет принципы кодирования без возврата к нулю (NRZ Level) и кодирования с возвратом к нулю (RZ).

В гибридном троичном коде есть три уровня представления сигнала: положительный ( $+U$ ), нулевой ( $0$ ) и отрицательный ( $-U$ ). Эти три уровня представлены тремя состояниями. Код может находиться в любом из этих трёх состояний. Переход к следующему состоянию зависит от значения входного бита и текущего состояния на выходе кодера (рис. 2.3). Саму процедуру кодирования можно свести к таблице 2.1.

Таблица 2.1

*Принцип кодирования гибридным троичным кодом*

Входной бит	Предыдущий уровень на выходе	Уровень на выходе кодера
0	$+U$	$-U$
	$0$	
	$-U$	
1	$+U$	$0$
	$0$	
	$-U$	

Словами процедуру кодирования можно описать тремя постулатами.

1. Если на входе «единица», то кодер подает на выход  $+U$ ; если «ноль», то  $-U$ .
2. Если при этом сигнал на выходе будет повторять предыдущее состояние, то на выход вместо этого подается  $0$ .
3. Изначально, до прихода на вход первого бита, кодер находится в нулевом состоянии.

## Биполярный код АМІ

Код АМІ (Alternate Mark Inversion) использует следующие представления битов:

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

Пример показан на рис. 2.4.

Код АМІ 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.

Кроме того, код АМІ послужил основой для целого ряда систем линейного кодирования, получивших общее название «модифицированные АМІ-коды» (Modified АМІ code). Эти коды в основном применяются в технологии PDH американского стандарта Tх и европейского стандарта Eх.

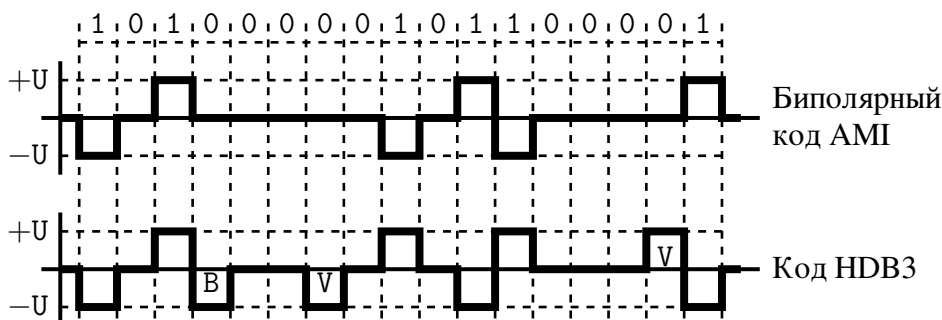


Рис. 2.4. Биполярный код АМІ. Код HDB3

## Код HDB3

Код HDB3 (high density bipolar of order 3) используется на всех уровнях европейского стандарта PDH, начиная с E1.

Основной принцип HDB3, как и прочих «модифицированных АМІ-кодов», состоит в замене последовательностей нулей на служебные комбинации, чтобы исправить указанный выше недостаток кода АМІ, связанный с потерей синхронизации.

В коде HDB3 осуществляется замена последовательностей из четырёх идущих подряд нулей. Используется следующее правило: если перед блоком из четырёх нулей исходный код содержал нечётное число единиц, то комбинация 0000 заменяется на служебный код 000V, а если чётное — то B00V.

Отсчёт количества единиц производится с начала передачи исходного кода и начинается заново после каждого сигнала V (рис. 2.4).

V — сигнал запрещённой полярности, то есть сигнал той же полярности, что и предшествующая ему единица.

B — сигнал правильной полярности, то есть сигнал полярности, обратной предшествующей ему единицы.

### ***Код B8ZS***

Код B8ZS (bipolar with eight-zero substitution) используется в американском стандарте PDH на уровне T1 (1.544 Mbit/s). В нём каждая серия из восьми последовательно идущих нулей заменяется на специальную комбинацию 000VB0VB (рис. 2.5).

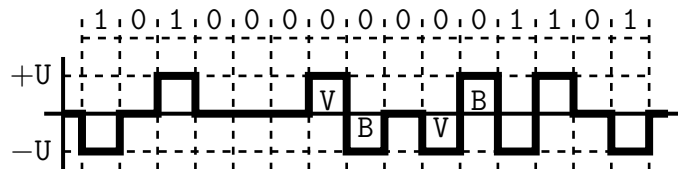


Рис. 2.5. Код B8ZS

### ***Код B6ZS***

Код B6ZS (bipolar with six-zero substitution) используется в американском стандарте PDH на уровне T2 (6.312 Mbit/s). В нём каждая серия из шести последовательно идущих нулей заменяется на специальную комбинацию 00VB0VB (рис. 2.6).

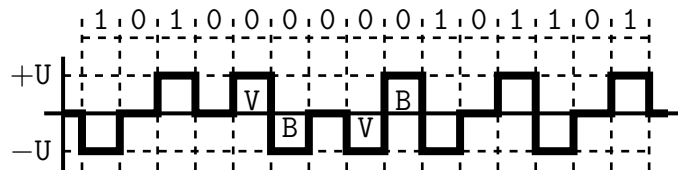


Рис. 2.6. Код B6ZS

### ***Код B3ZS***

Код B3ZS (bipolar with three-zero substitution) используется в американском стандарте PDH на уровне T3 (44.736 Mbit/s). Подобно коду HDB3, если перед блоком из трёх нулей исходный код содержал нечётное число единиц, то комбинация 000 заменяется на служебный код 00V, а если чётное — то B0V. Отсчёт количества единиц производится с начала передачи исходного кода и начинается заново после каждого сигнала V (рис. 2.7).

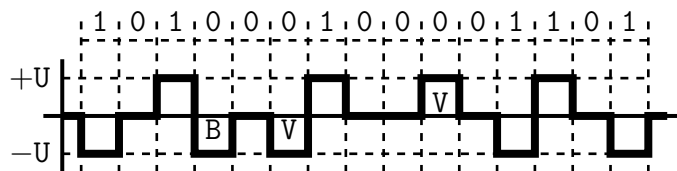


Рис. 2.7. Код B3ZS

### 2.2.4. Тетрарное (четырёхуровневое) кодирование

В случае тетрарного кодирования используются четыре уровня сигнала. Среди подобных алгоритмов линейного кодирования можно выделить алгоритм 2B1Q.

#### Код 2B1Q

Линейное кодирование 2B1Q (2 Binary 1 Quandary) было разработано для использования в качестве протокола физического уровня в точке сопряжения U BRI-интерфейса сетей ISDN. Алгоритм 2B1Q представляет собой один из вариантов реализации амплитудно-импульсной модуляции с четырьмя уровнями выходного напряжения без возвращения к нулевому уровню. Код 2B1Q передает за один временной интервал сразу два бита данных. Каждой возможной паре в соответствие ставится свой уровень из четырех возможных уровней потенциала в соответствии с таблицей 2.2

Таблица 2.2

Принцип кодирования кодом 2B1Q

Биты данных	Уровень потенциала на выходе
00	-2.5 В
01	-0.833 В
11	+0.833 В
10	+2.5 В

Пример кодирования кодом 2B1Q показан на рис. 2.8.

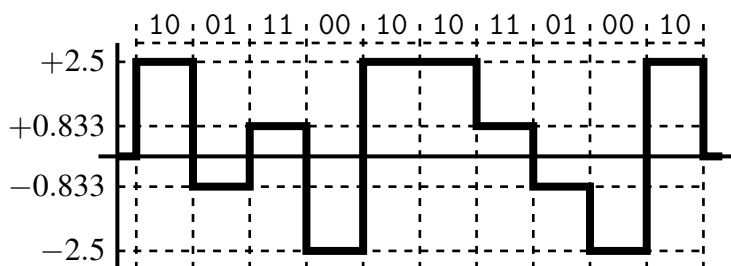


Рис. 2.8. Код 2B1Q

Информационная скорость кода 2B1Q в два раза превышает символьную. Следовательно, модуляционная схема 2B1Q обеспечивает постоянную величину спектральной эффективности модулированного сигнала 2 бита/Гц. Реализация этого метода требует более мощного передатчика и более сложного приемника, который должен различать четыре уровня.

Помимо сетей ISDN алгоритм линейного кодирования 2B1Q нашел достаточно широкое применение в устройствах xDSL с симметричными скоростями передачи данных — HDSL и SDSL.

### 2.3. Рекомендуемая литература

1. В. Г. Олифер, Н. А. Олифер. «Компьютерные сети».
2. В. В. Крухмалев, В. Н. Гордиенко. «Цифровые системы передачи».
3. С. В. Кунегин. «Системы передачи информации. Курс лекций».
4. Л. С. Левин, М. А. Плоткин. «Цифровые системы передачи информации».

### 2.4. Порядок выполнения задания

Задание выполняется каждым учащимся индивидуально на бланке, приведённом на странице 33.

В работе требуется закодировать двоичную информационную последовательность различными алгоритмами линейного кодирования в соответствии с вариантом.

**Исходная информационная последовательность** формируется, исходя из последних пяти цифр номера зачётной книжки (или студенческого) учащегося. Каждая цифра номера представляется двоичной последовательностью длиной 4 бита, согласно табл. 2.3. Перед полученной последовательностью ставится 10, а в конце — 01. В итоге получается информационная последовательность длиной 24 бита.

Таблица 2.3

*Соответствие цифры номера зачетной книжки и двоичной последовательности*

Цифра	Код	Цифра	Код	Цифра	Код	Цифра	Код	Цифра	Код
1	0001	3	0011	5	0101	7	0111	9	1001
2	0010	4	0100	6	0110	8	1000	0	1010

**Пример формирования последовательности:** Пусть номер зачетной книжки 650327. Для формирования строки берем последние пять цифр: 50327. В результате кодирования по табл. 2.3 и добавления начальной и конечной комбинаций будет получена информационная последовательность

10 0101 1010 0011 0010 0111 01

**Алгоритмы линейного кодирования выбираются** из табл. 2.4 в зависимости от четности номера зачетной книжки.

Таблица 2.4

Алгоритмы линейного кодирования в зависимости от номера зачетной книжки

Номер	Алгоритмы
Четный	Unipolar NRZ (прямой) / NRZ Space / Манчестерский код (по Д. Е. Томасу) / Диффер. манч. код. / С возвратом к нулю / Гибр. троичный код / HDB-3 / B6ZS / 2B1Q
Нечетный	Unipolar NRZ (обратный) / NRZ Inverted / Манчестерский код (IEEE 802.3) / Бифазное код. / MLT-3 / AMI / B8ZS / B3ZS / 2B1Q

## 2.5. Порядок защиты практической работы

Защита работы может осуществляться одним из нижеперечисленных способов или их сочетанием на усмотрение преподавателя.

1. Устный ответ по теме работы.
2. Тестирование по теме работы
3. Задача на кодирование/декодирование информационной последовательности.
4. Иные варианты на усмотрение преподавателя.



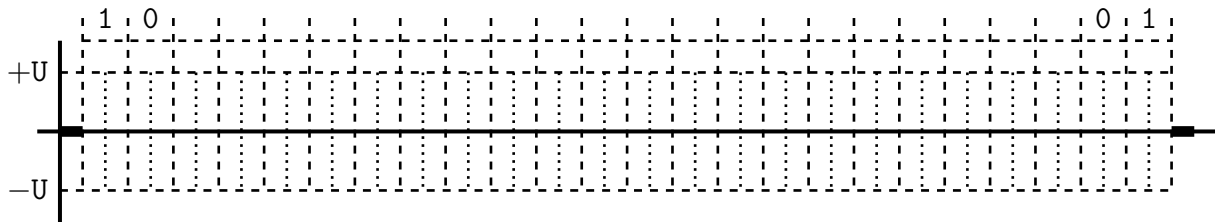
## Работа 2

### Методы линейного кодирования

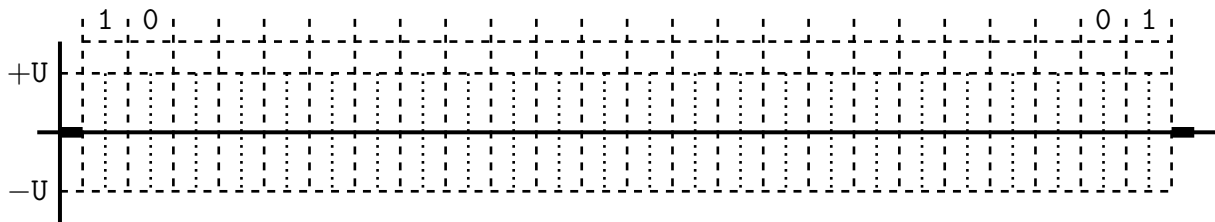
Группа: \_\_\_\_\_ Студент: \_\_\_\_\_

№ зачетной книжки: \_\_\_\_\_

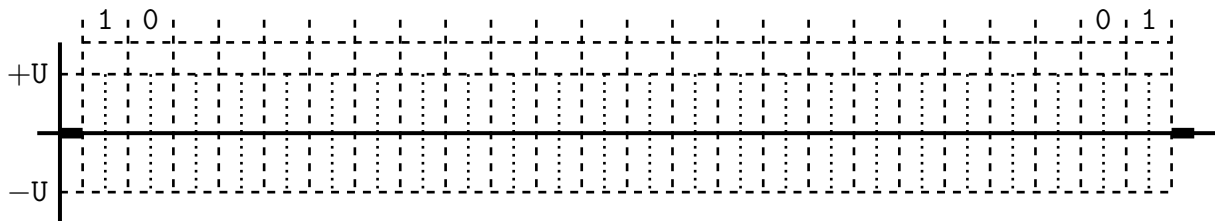
Алгоритм кодирования: \_\_\_\_\_



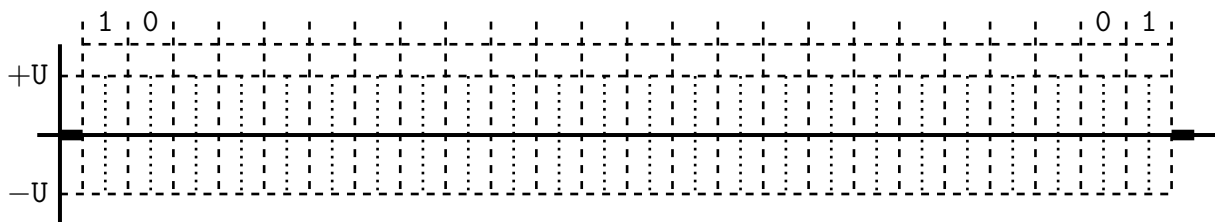
Алгоритм кодирования: \_\_\_\_\_



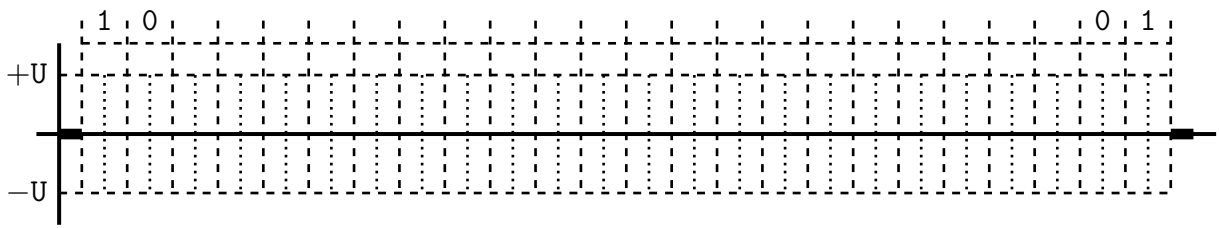
Алгоритм кодирования: \_\_\_\_\_



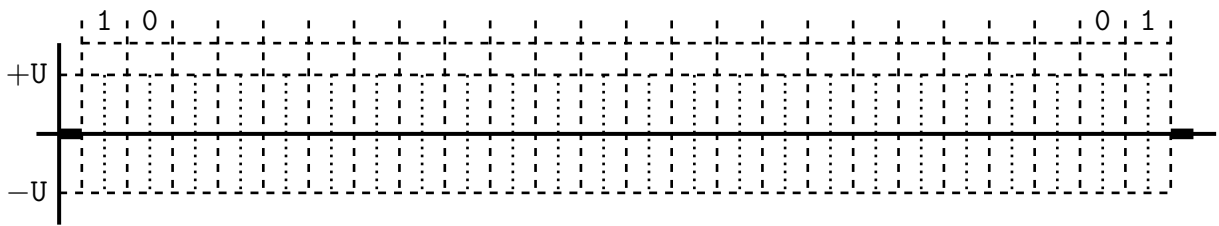
Алгоритм кодирования: \_\_\_\_\_



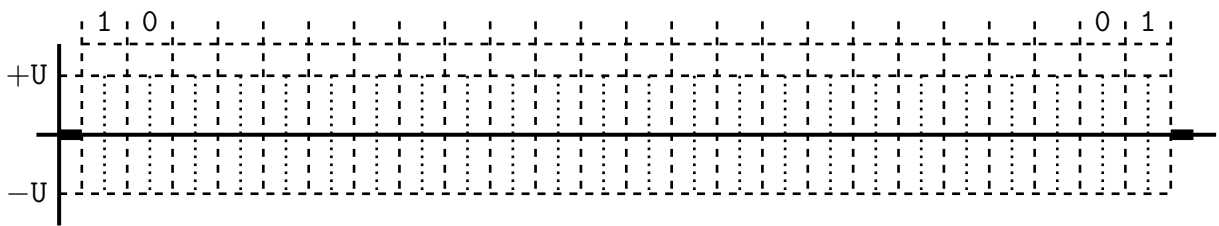
Алгоритм кодирования: \_\_\_\_\_



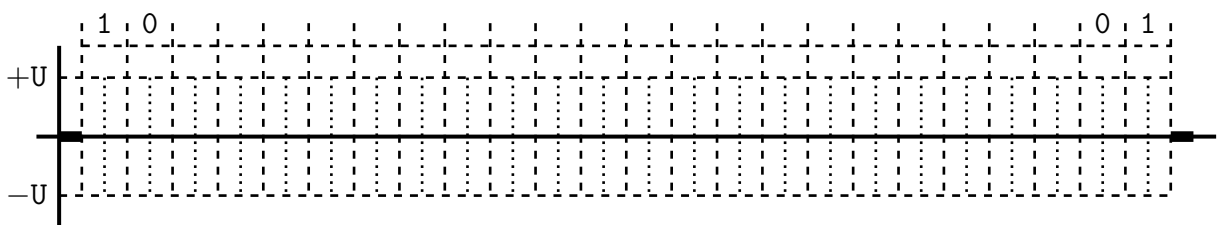
Алгоритм кодирования: \_\_\_\_\_



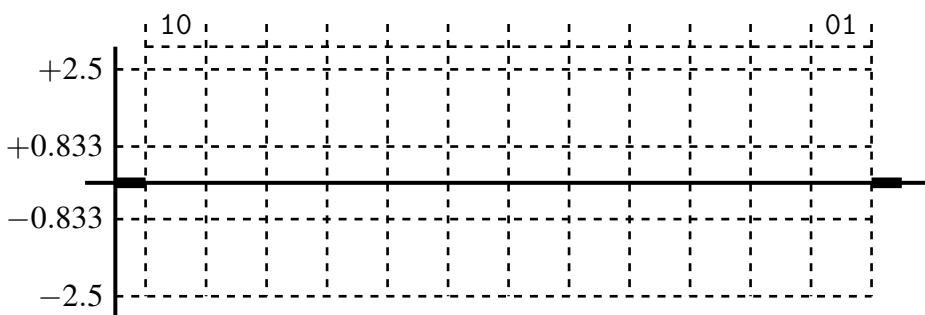
Алгоритм кодирования: \_\_\_\_\_



Алгоритм кодирования: \_\_\_\_\_



Алгоритм кодирования: 2B1Q



# Содержание

<b>1</b>	<b>Исследование эффективности одномерных алгоритмов сжатия цифрового факсимильного сигнала</b>	<b>3</b>
1.1	Цель работы . . . . .	3
1.2	Теоретические сведения . . . . .	3
1.3	Рекомендуемая литература . . . . .	18
1.4	Порядок выполнения задания . . . . .	18
1.5	Порядок защиты практической работы . . . . .	18
	<b>Бланк к работе 1</b>	<b>21</b>
<b>2</b>	<b>Методы линейного кодирования</b>	<b>23</b>
2.1	Цель работы . . . . .	23
2.2	Теоретические сведения . . . . .	23
2.3	Рекомендуемая литература . . . . .	31
2.4	Порядок выполнения задания . . . . .	31
2.5	Порядок защиты практической работы . . . . .	32
	<b>Бланк к работе 2</b>	<b>33</b>

**Владимиров Сергей Сергеевич**

**СИСТЕМЫ И УСТРОЙСТВА ПЕРЕДАЧИ ДАННЫХ**

**Практикум**

Издано в авторской редакции

План изданий 2014–2015 гг., доп. п. 10

Подписано к печати 30.06.2014  
Объем 2,25 усл.-печ. л. Тираж 15 экз. Заказ 487

Редакционно-издательский центр СПбГУТ  
191186 СПб., наб. р. Мойки, 61  
Отпечатано в СПбГУТ