ОГЛАВЛЕНИЕ

[ВСТУПЛЕНИЕ 5](#_Toc530597412)

[2. КОДИРОВАНИЕ ИНФОРМАЦИИ 7](#_Toc530597413)

[3. ВИДЫ КОДИРОВАНИЯ ИНФОРМАЦИИ 11](#_Toc530597414)

[4. КОДИРОВАНИЕ ИНФОРМАЦИИ ПО МЕТОДУ ХЭММИНГА 14](#_Toc530597415)

[5. АЛГОРИТМ КОДИРОВАНИЯ МЕТОДА ХЭММИНГА 17](#_Toc530597416)

[6. АЛГОРИТМ ДЕКОДИРОВАНИЯ ХЭММИНГА 20](#_Toc530597417)

[7. ОСОБЕННОСТИ ПРОГРАММНОЙ РЕАЛИЗАЦИИ С ТОЧКИ ЗРЕНИЯ ПРОГРАММИСТА 21](#_Toc530597418)

[8. РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ 22](#_Toc530597419)

[ЗАКЛЮЧЕНИЕ 23](#_Toc530597420)

[СПИСОК ЛИТЕРАТУРЫ 24](#_Toc530597421)

[ПРИЛОЖЕНИЕ А 25](#_Toc530597422)

## Вступление

С появлением технической возможности хранить и пересылать большие объёмы данных, появилась необходимость бороться с возникающимися в этих данных ошибками. Эта практическая потребность дала рождение теории кодирования – науке о надежном хранении и передаче информации. Типичный вопрос, изучаемый этой наукой: как передавать по каналу связи полезную информацию, если один процент пересылаемых битов теряется или искажается?

Методы, развитые в теории кодирования, оказываются эффективны в задачах, не связанных непосредственно с защитой сообщений от шума (например, в теории коммуникационной сложности, в схемах разделения секрета, при дерандомизации вероятностных алгоритмов, и т.д.).

Для написания программы я выбрала язык программирования Visual StudioC#, таккак этот язык программирования, предназначенный для разработки самых разнообразных приложений, предназначенных для выполнения в среде .NET Framework. Библиотека классов .NET Framework предоставляет доступ ко многим службам операционной системы и другим полезным классам, что существенно ускоряет цикл разработки.

Работа состоит из двух частей - теоретической и практической.

В теоретической части приведены определения таких понятий, как кодирование информации, виды кодирования информации, практическое применение кодирования, определены основные задачи теории кодирования, основные аспекты кодирования информации по методу Хэмминга.

В практической части составлена программа для кодирования и декодирования информации.

Актуальность моей работы определяется необходимостью рассматривать вопросы, связанные с кодированием информации, в виду их большой практической значимости. Код Хэмминга используется в некоторых прикладных программах в области хранения данных, особенно в [RAID 2](https://ru.wikipedia.org/wiki/RAID); кроме того, метод Хэмминга давно применяется в памяти типа [ECC](https://ru.wikipedia.org/wiki/ECC) и позволяет «на лету» исправлять однократные и обнаруживать двукратные ошибки.

**Объект исследования:** кодирование и декодирование информации.

**Предмет исследования:** кодирование и декодирование информации методом Хэмминга.

**Постановка задачи:** составить программу для кодирования и декодирования информации методом Хэмминга.

## Кодирование информации

В настоящее время во всех вычислительных машинах информация представляется с помощью электрических сигналов. При этом возможны две формы ее представления – в виде непрерывного сигнала (с помощью сходной величины – аналога) и в виде нескольких сигналов (с помощью набора напряжений, каждое из которых соответствует одной из цифр представляемой величины) [1].

Первая форма представления информации называется аналоговой, или непрерывной. Величины, представленные в такой форме, могут принимать принципиально любые значения в определенном диапазоне. Количество значений, которые может принимать такая величина, бесконечно велико. Отсюда названия – непрерывная величина и непрерывная информация. Слово непрерывность отчетливо выделяет основное свойство таких величин – отсутствие разрывов, промежутков между значениями, которые может принимать данная аналоговая величина. При использовании аналоговой формы для создания вычислительной машины потребуется меньшее число устройств (каждая величина представляется одним, а не несколькими сигналами), но эти устройства будут сложнее (они должны различать значительно большее число состояний сигнала). Непрерывная форма представления используется в аналоговых вычислительных машинах (АВМ). Устройства для обработки непрерывных сигналов обладают более высоким быстродействием, они могут интегрировать сигнал, выполнять любое его функциональное преобразование и т. п. Однако из-за сложности технической реализации устройств выполнения логических операций с непрерывными сигналами, длительного хранения таких сигналов, их точного измерения АВМ не могут эффективно решать задачи, связанные с хранением и обработкой больших объемов информации [1].

Вторая форма представления информации называется дискретной (цифровой). Такие величины, принимающие не все возможные, а лишь вполне определенные значения, называются дискретными (прерывистыми). В отличие от непрерывной величины, количество значений дискретной величины всегда будет конечным. Дискретная форма представления используется в цифровых электронно-вычислительных машинах (ЭВМ), которые легко решают задачи, связанные с хранением, обработкой и передачей больших объемов информации [1].

Для автоматизации работы ЭВМ с информацией, относящейся к различным типам, очень важно унифицировать их форму представления – для этого обычно используется прием кодирования.

Кодирование – это представление сигнала в определенной форме, удобной или пригодной для последующего использования сигнала. Говоря строже, это правило, описывающее отображение одного набора знаков в другой набор знаков. Тогда отображаемый набор знаков называется исходным алфавитом, а набор знаков, который используется для отображения, – кодовым алфавитом, или алфавитом для кодирования. При этом кодированию подлежат как отдельные символы исходного алфавита, так и их комбинации. Аналогично для построения кода используются как отдельные символы кодового алфавита, так и их комбинации [2].

Совокупность символов кодового алфавита, применяемых для кодирования одного символа (или одной комбинации символов) исходного алфавита, называется кодовой комбинацией или кодом символа. При этом кодовая комбинация может содержать один символ кодового алфавита.

Взаимосвязь символов (или комбинаций символов, если кодируются не отдельные символы исходного алфавита) исходного алфавита с их кодовыми комбинациями составляет таблицу соответствия (или таблицу кодов) [2].

В качестве примера можно привести систему записи математических выражений, азбуку Морзе, морскую флажковую азбуку, систему Брайля для слепых и др.

В вычислительной технике также существует своя система кодирования – она называется двоичным кодированием и основана на представлении данных последовательностью всего двух знаков: 0 и 1 (используется двоичная система счисления). Эти знаки называются двоичными цифрами, или битами (binary digital) [2].

К основным задачам теории кодирования информации необходимо отнести следующие:

• разработка принципов наиболее экономичного кодирования информации;

• согласование параметров передаваемой информации с особенностями канала связи;

• разработка приемов, обеспечивающих надежность передачи информации по каналам связи, т. е. отсутствие потерь информации.

Основными атрибутами кодирования являются:

* Код – это набор знаков, упорядоченных в соответствии с определенными правилами того или иного языка, для передачи информации.
* Знак – это метка, предмет, которым обозначается что-нибудь (буква, цифра, отверстие). Знак вместе с его значением называют символом. Существует множество классификаций знаков.
* Язык – это сложная система символов, каждый из которых имеет определенное значение. Языковые символы, будучи общепринятыми и соответственно общепонятными в пределах данного сообщества, в процессе речи комбинируются друг с другом, порождая разнообразные по своему содержанию сообщения.

***Практическое применение кодирования информации:***

* ***Персональные данные.* П**од фотографией в паспорте на просвет просматриваются магнитные метки с записанной информацией, которая считывается только электронным способом и недоступна владельцу документа. Подписываясь под этой графой в паспорте (пока не заполняемой по техническим причинам), человек дает согласие на присвоение ему кода вместо имени, т.е. производится замена имени числом.
* ***Штрих-коды.*** С развитием информационной техники, широким внедрением средств вычислительной техники во многие сферы деятельности все острее встает вопрос быстрого и надежного ввода информации. Ручной ввод кода изделия требуют больших затрат ручного труда, времени, часто приводит к ошибкам. По мнению специалистов, системы штрихового кодирования имеют перспективу и дают возможность решить одну из самых сложных компьютерных проблем - ввод данных. В настоящее время штриховые коды широко используются не только при производстве и в торговле товарами, но и во многих отраслях промышленного производства. Товарный штриховой код присваивается продукции (товару) на этапе запуска его в производство. Штрих-коды получили широкое практическое применение почти во всех сферах деятельности человека.

## Виды Кодирования информации

Большие объемы информации требуют ее постоянной систематизации и кодировки для корректной и точной передачи любого количества и качества полученных данных, необходимых для работы, упрощая их восприятие для одного человека или группы людей. Информационное насыщение и свободная обработка подготовленных блоком требует тщательной кодировки данных, для их сохранения и передачи, поэтому современной наукой разработано несколько эффективных видов и способов кодировки данных [2].

**Кодирование текстовой информации**

Для кодирования текстовых данных используются специально разработанные таблицы кодировки, основанные на сопоставлении каждого символа алфавита с определенным целым числом. Восьми двоичных разрядов достаточно для кодирования 256 различных символов. Этого хватит, чтобы выразить различными комбинациями восьми битов все символы английского и русского языков, как строчные, так и прописные, а также знаки препинания, символы основных арифметических действий и некоторые общепринятые специальные символы. Но не все так просто, и существуют определенные сложности. В первые годы развития вычислительной техники они были связаны с отсутствием необходимых стандартов, а в настоящее время, наоборот, вызваны изобилием одновременно действующих и противоречивых стандартов. Практически для всех распространенных на земном шаре языков созданы свои кодовые таблицы. Для того чтобы весь мир одинаково кодировал текстовые данные, нужны единые таблицы кодирования, что до сих пор пока еще не стало возможным [2].

**Кодирование графической информации**

Кодирование графической информации основано на том, что изображение состоит из мельчайших точек, образующих характерный узор, называемый растром. Каждая точка имеет свои линейные координаты и свойства (яркость), следовательно, их можно выразить с помощью целых чисел – растровое кодирование позволяет использовать двоичный код для представления графической информации. Черно-белые иллюстрации представляются в компьютере в виде комбинаций точек с 256 градациями серого цвета – для кодирования яркости любой точки достаточно восьмиразрядного двоичного числа [2].

Для кодирования цветных графических изображений применяется принцип декомпозиции (разложения) произвольного цвета на основные составляющие. При этом могут использоваться различные методы кодирования цветной графической информации. Например, на практике считается, что любой цвет, видимый человеческим глазом, можно получить путем механического смешивания основных цветов. В качестве таких составляющих используют три основных цвета: красный (Red, R), зеленый (Green, G) и синий (Blue, B). Такая система кодирования называется системой RGB [2].

**Кодирование звуковой информации**

Приемы и методы кодирования звуковой информации пришли в вычислительную технику наиболее поздно и до сих пор далеки от стандартизации. Множество отдельных компаний разработали свои корпоративные стандарты, хотя можно выделить два основных направления.

Метод FM (Frequency Modulation) основан на том, что теоретически любой сложный звук можно разложить на последовательность простейших гармоничных сигналов разной частоты, каждый из которых представляет правильную синусоиду, а следовательно, может быть описан числовыми параметрами, то есть кодом. В природе звуковые сигналы имеют непрерывный спектр, то есть являются аналоговыми. Их разложение в гармонические ряды и представление в виде дискретных цифровых сигналов выполняют специальные устройства – аналогово-цифровые преобразователи (АЦП). Обратное преобразование для воспроизведения звука, закодированного числовым кодом, выполняют цифро-аналоговые преобразователи (ЦАП). При таких преобразованиях часть информации теряется, поэтому качество звукозаписи обычно получается не вполне удовлетворительным и соответствует качеству звучания простейших электромузыкальных инструментов с «окрасом», характерным для электронной музыки [2].

Метод таблично-волнового синтеза (Wave-Table) лучше соответствует современному уровню развития техники. Имеются заранее подготовленные таблицы, в которых хранятся образцы звуков для множества различных музыкальных инструментов. В технике такие образцы называются сэмплами. Числовые коды выражают тип инструмента, номер его модели, высоту тона, продолжительность и интенсивность звука, динамику его изменения. Поскольку в качестве образцов используются «реальные» звуки, то качество звука, полученного в результате синтеза, получается очень высоким и приближается к качеству звучания реальных музыкальных инструментов [2].

## Кодирование информации по методу Хэмминга

В середине 40-х годов Ричард Хемминг работал в знаменитых Лабораториях Белла на счётной машине Bell Model V. Это была электромеханическая машина, использующая релейные блоки, скорость которых была очень низка: один оборот за несколько секунд. Данные вводились в машине с помощью перфокарт, и поэтому в процессе чтения часто происходили ошибки. В рабочие дни использовались специальные коды, чтобы обнаруживать и исправлять найденные ошибки, при этом оператор узнавал об ошибке по свечению лампочек, исправлял и запускал машину. В выходные дни, когда не было операторов, при возникновении ошибки машина автоматически выходила из программы и запускала другую [3].

Р. Хемминг часто работал в выходные дни, и все больше и больше раздражался, потому что часто был должен перегружать свою программу из-за ненадежности перфокарт. На протяжении нескольких лет он проводил много времени над построением эффективных алгоритмов исправления ошибок. В 1950 году он опубликовал способ, который на сегодняшний день мы знаем как код Хемминга [4].

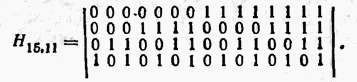
Код Хемминга, как и любой (*n,k*) код, содержит *k* информационных и избыточных символов. Избыточная часть кода строится таким образом, чтобы при декодировании можно было бы установить не только факт наличия ошибок в принятой – комбинации, но и указать номер позиции, в которой произошла ошибка. Это достигается за счет многократной проверки принятой комбинации на четность. Каждой проверкой должны охватываться часть информационных символов и один из избыточных символов. При каждой проверке получают двоичный контрольный символ. Если результат проверки дает четное число, то контрольному символу присваивается значение 0, если нечетное число– 1. В результате всех проверок получается *p*-разрядное двоичное число, указывающее номер искаженного символа. Для исправления ошибки достаточно лишь изменить значение данного символа на обратное [4].

К ним обычно относятся коды с минимальным кодовым расстояниемисправляющие все одиночные ошибки, и коды с расстояниемисправляющие все одиночные и обнаруживающие все двойные ошибки. Длина кода Хэмминга:

(2)

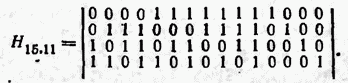
(r – количество проверочных разрядов).

Характерной особенностью проверочной матрицы кода с является то, что ее столбцы представляют собой любые различные ненулевые комбинации длиной *r*.Например, при *r*=4 и*n*=5 для кода (15,11), проверочная матрица может иметь следующий вид (рис.3)



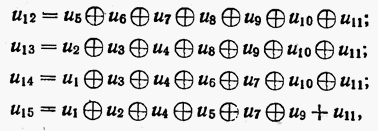
*Рис.3. Проверочная матрица*

Перестановкой столбцов, содержащих одну единицу, данную матрицу можно привести к виду(рис.4)



*Рис. 4.Измененная матрица*

Использование такого кода позволяет исправить любую одиночную ошибку или обнаружить произвольную ошибку кратности два.Если информационные и проверочные разряды кода нумеровать слева направо, то в соответствии с матрицей получаем систему проверочных уравнений, с помощью которых вычисляем проверочные разряды(рис.5):



*Рис.5. Система проверочных уравнений*

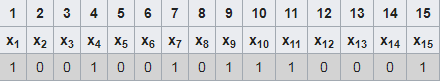
где - -проверочные разряды; - -информационные разряды

Двоичный код Хэмминга с кодовым расстоянием получается путем добавления к коду Хэмминга с одного проверочного разряда, представляющего собой результат суммирования по модулю два всех разрядов кодового слоя. Длина кода при этом разрядов, из которых являются проверочными [4].

Операция кодирования может выполняться в два этапа. На первом этапе определяется кодовая комбинация с использованием матрицы H, соответствующей коду с на втором — добавляется один проверочный разряд, в котором записывается результат суммирования по модулю два всех разрядов кодового слова, полученного на первом этапе. Операция декодирования также состоит из двух этапов. На первом вычисляется синдром, соответствующий коду на втором — проверяется последнее проверочное соотношение [4].

## Алгоритм Кодирования метода Хэмминга

Предположим, что нужно сгенерировать код Хэмминга для некоторого информационного кодового слова. В качестве примера возьмём 15-битовое кодовое слово x1…x15, хотя алгоритм пригоден для кодовых слов любой длины. В приведённой ниже таблице в первой строке даны номера позиций в кодовом слове, во второй — условное обозначение битов, в третьей — значения битов [5].

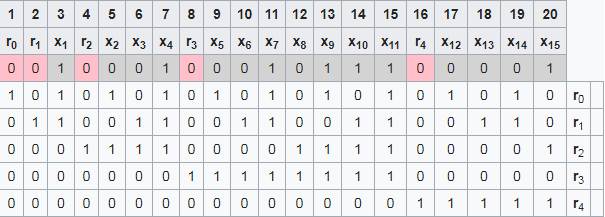


Вставим в информационное слово контрольные биты **r**0…**r**4 таким образом, чтобы номера их позиций представляли собой целые степени двойки: 1, 2, 4, 8, 16. Получим 20-разрядное слово с 15 информационными и 5 контрольными битами. Первоначально контрольные биты устанавливаем равными нулю. На рисунке контрольные биты выделены розовым цветом.



В общем случае количество контрольных бит в кодовом слове равно двоичному логарифму числа, на единицу большего, чем количество бит кодового слова (включая контрольные биты); логарифм округляется в большую сторону. Например, информационное слово длиной 1 бит требует двух контрольных разрядов, 2-, 3- или 4-битовое информационное слово — трёх, 5…11-битовое — четырёх, 12…26-битовое — пяти и т. д [5].

Добавим к таблице 5 строк (по количеству контрольных битов), в которые поместим матрицу преобразования. Каждая строка будет соответствовать одному контрольному биту (нулевой контрольный бит — верхняя строка, четвёртый — нижняя), каждый столбец — одному биту кодируемого слова. В каждом столбце матрицы преобразования поместим двоичный номер этого столбца, причём порядок следования битов будет обратный — младший бит расположим в верхней строке, старший — в нижней. Например, в третьем столбце матрицы будут стоять числа 11000, что соответствует двоичной записи числа три: 00011.



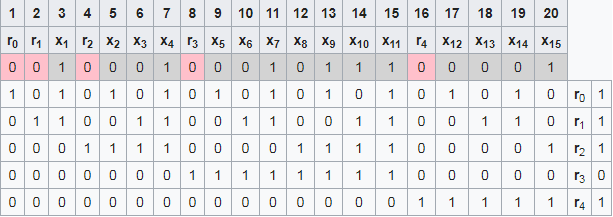
В правой части таблицы мы оставили пустым один столбец, в который поместим результаты вычислений контрольных битов. Вычисление контрольных битов производим следующим образом. Берём одну из строк матрицы преобразования (например, r0) и находим её скалярное произведение с кодовым словом, то есть перемножаем соответствующие биты обеих строк и находим сумму произведений. Если сумма получилась больше единицы, находим остаток от его деления на 2. Иными словами, мы подсчитываем сколько раз в кодовом слове и соответствующей строке матрицы в одинаковых позициях стоят единицы и берём это число по модулю 2.

Если описывать этот процесс в терминах матричной алгебры, то операция представляет собой перемножение матрицы преобразования на матрицу-столбец кодового слова, в результате чего получается матрица-столбец контрольных разрядов, которые нужно взять по модулю 2.

Например, для строки r0:

r0 =(1·0+0·0+1·1+0·0+1·0+0·0+1·1+0·0+1·0+0·0+1·1+0·0+1·1+0·1+1·1+0·0+1·0+0·0+1·0+0·1) mod 2 = 5 mod 2 = 1.

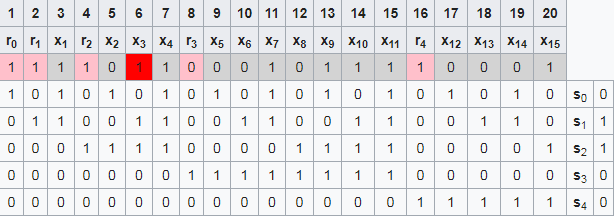
Полученные контрольные биты вставляем в кодовое слово вместо стоявших там ранее нулей. По аналогии находим проверочные биты в остальных строках. Кодирование по Хэммингу завершено. Полученное кодовое слово — 11110010001011110001.



## Алгоритм деКодирования Хэмминга

Алгоритм декодирования по Хэммингу абсолютно идентичен алгоритму кодирования. Матрица преобразования соответствующей размерности умножается на матрицу-столбец кодового слова и каждый элемент полученной матрицы-столбца берётся по модулю 2. Полученная матрица-столбец получила название «матрица синдромов». Легко проверить, что кодовое слово, сформированное в соответствии с алгоритмом, описанным в предыдущем разделе, всегда даёт нулевую матрицу синдромов [5].

Матрица синдромов становится ненулевой, если в результате ошибки (например, при передаче слова по линии связи с шумами) один из битов исходного слова изменил своё значение. Предположим для примера, что в кодовом слове, полученном в предыдущем разделе, шестой бит изменил своё значение с нуля на единицу (на рисунке обозначено красным цветом). Тогда получим следующую матрицу синдромов [5].



Заметим, что при однократной ошибке матрица синдромов всегда представляет собой двоичную запись (младший разряд в верхней строке) номера позиции, в которой произошла ошибка. В приведённом примере матрица синдромов (01100) соответствует двоичному числу 00110 или десятичному 6, откуда следует, что ошибка произошла в шестом бите [3].

## ОСОБЕННОСТИ ПРОГРАММНОЙ РЕАЛИЗАЦИИ С ТОЧКИ ЗРЕНИЯ ПРОГРАММИСТА

## РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

## ЗАКЛЮЧЕНИЕ

## СПИСОК ЛИТЕРАТУРЫ

|  |  |
| --- | --- |
| [1] | В. М., Основы кодирования, Москва: Техносфера, 2004. |
| [2] | «Википедия,» 9 июль 2011. [В Интернете]. Available: https://ru.wikipedia.org/wiki/Код\_Хэмминга#Алгоритм\_кодирования. |
| [3] | Б. Р., Теория и практика кодов, контролирующих ошибки., Мир, 1986. |
| [4] | К. И.В., .Основы теории информации и кодирования, Киев, 1986. |
| [5] | М.-С. Р., Искусство помехоустойчивого кодирования – методы, алгоритмы, применение., Москва, 2005. |

## ПРИЛОЖЕНИЕ А