



# Проект по теме кодирование



Выполнил:

Студент группы 1-15 ПКС  
Черномазов К. П.

Принял:

Осмоловская Н.С.



# Содержание

- Введение
- Представление о кодировании информации
- Особенности кодирования информации в персональном компьютере
- Двоичные совокупности
- Дискретизация
- Единицы измерения
- Тест по теме кодирование
- Кроссворд

# Введение

**Кодирование информации** — процесс преобразования сигнала из формы, удобной для непосредственного использования информации, в форму, удобную для передачи, хранения или автоматической переработки.

**Сетевое кодирование** — раздел теории информации, изучающий вопрос оптимизации передачи данных по сети с использованием техник изменения пакетов данных на промежуточных узлах.

**Энтропийное кодирование** — кодирование последовательности значений с возможностью однозначного восстановления с целью уменьшения объёма данных (длины последовательности) с помощью усреднения вероятностей появления элементов в закодированной последовательности.

**Дельта-кодирование** — способ представления данных в виде разницы (дельты) между последовательными данными вместо самих данных.

**Кодирование (программирование)** — процесс написания программного кода, скриптов, с целью реализации определённого алгоритма на определённом языке программирования.

**Кодирование (медицина)** — научнообразные методы внушения в наркологии.

# ПРЕДСТАВЛЕНИЕ О КОДИРОВАНИИ ИНФОРМАЦИИ

Когда человек или какой-нибудь другой живой организм или какое-то устройство участвуют в информационном процессе, то все они представляют информацию в той или иной форме.

Независимо от формы представления и способа передачи информации, она всегда передается с помощью какого-либо языка.

**Язык** – это знаковая форма представления информации.

Таким образом, языки можно рассматривать, как системы знаков и правил их использования для представления информации. Есть разговорные языки, возникшие и развивающиеся вместе с народом-носителем этого языка. Их называют *естественными*. А есть искусственные (*формальные*) языки, специально созданные для представления какого-либо особого вида информации в определенной области человеческой деятельности.

Представление информации с помощью какого-либо языка часто называют *кодированием*. Когда мы представляем информацию в разных формах или преобразуем ее из одной формы в другую, мы информацию *кодируем*.

**Код** – это система условных знаков для представления информации.

**Кодирование** – это операция преобразования символов или группы символов одного кода в символы или группы символов другого кода.

Основу любого языка составляет **алфавит, синтаксис и грамматика.**

**Алфавит** – набор однозначно определенных знаков (символов), из которых формируется сообщение. При разговоре код передается звуками, при письме – буквами. Одну и ту же информацию можно представить с помощью различных кодов. Например посредством русских букв, или специальных значков.

**Синтаксис** – совокупность правил, согласно которым образуются предложения языка.

**Грамматика** – совокупность правил правописания.

# ЯЗЫКИ

**ЕСТЕСТВЕННЫЕ**  
(НОСЯТ  
национальный  
характер)

Пример: русский,  
английский,  
немецкий...

Алфавит: кириллица,  
латиница, иероглифы...

Синтаксис и грамматика: большое  
количество правил, из которых  
существуют исключения

**ФОРМАЛЬНЫЕ**  
(НОСЯТ  
интернациональный  
характер)

Пример: язык  
математики, химии,  
программирования...

Алфавит жестко зафиксирован:  
цифры, ноты, дорожные знаки,  
точки и тире...

Синтаксис и грамматика:  
наличие строгих правил

$$A = \pi r^2 \sum_{\substack{0 \leq i \leq m \\ 0 < j < n}} P(i, j)$$

$$(x + a)^n = \sum_{k=0}^n \binom{n}{k} x^k a^{n-k}$$

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \quad A = \pi r^2 \sum_{\substack{0 \leq i \leq m \\ 0 < j < n}} P(i, j)$$

$$f(x) = a_0 + \sum_{n=1}^{\infty} (a_n \cos \frac{n\pi x}{L} + b_n \sin \frac{n\pi x}{L})$$

$$\sin \alpha \pm \sin \beta = 2 \sin \frac{1}{2}(\alpha \pm \beta) \cos \frac{1}{2}(\alpha \mp \beta)$$

$$A(1+x)^n = 1 + \frac{nx}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots, \quad \max_{0 \leq x \leq 1} x e^{-x^2} < \infty$$
$$\lim_{n \rightarrow \infty} \left(1 + \frac{1}{n}\right)^n = e$$
$$\frac{dy}{dx} = 2 \sin \frac{1}{2}(\alpha \pm \beta) \cos \frac{1}{2}(\alpha \mp \beta)$$
$$a^2 + b^2 = c^2$$
$$e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots$$

**ОДНУ И ТУ ЖЕ ИНФОРМАЦИЮ МОЖНО КОДИРОВАТЬ РАЗНЫМИ СПОСОБАМИ. ЭТО ЗАВИСИТ ОТ РЯДА ПРИЧИН, КАКОВ НОСИТЕЛЬ ИНФОРМАЦИИ, КАКОВА ОКРУЖАЮЩАЯ ОБСТАНОВКА, КТО ИЛИ ЧТО ЯВЛЯЕТСЯ ИСТОЧНИКОМ ИНФОРМАЦИИ И ПОЛУЧАТЕЛЕМ, ОТ ЗАСЕКРЕЧЕННОСТИ И Т.Д.**

**ОБЫЧНО ПРИ КОДИРОВАНИИ ИНФОРМАЦИИ ИСПОЛЬЗУЕТСЯ СРЕДСТВО КОДИРОВАНИЯ - НЕКОТОРАЯ ТАБЛИЦА, КОТОРАЯ УСТАНАВЛИВАЕТ СООТВЕТСТВИЕ МЕЖДУ ЗНАКАМИ РАЗНЫХ КОДОВ**

**ПРИМЕРЫ:  
- АЗБУКА МОРЗЕ  
- ФЛАЖКОВАЯ АЗБУКА**

**В ПРОЦЕССЕ ОБМЕНА ИНФОРМАЦИЕЙ КРОМЕ КОДИРОВАНИЯ ИНФОРМАЦИИ ПРОИСХОДИТ И ЕЕ ДЕКОДИРОВАНИЕ**

### Азбука Морзе

А	· —	Р	· · · —
Б	— · · ·	С	· · · ·
В	— · —	Т	— · ·
Г	· — ·	У	· · —
Д	— · ·	Ф	· · · —
Е	· · · ·	Х	· — · —
Ж	· · · —	Ц	· · · ·
З	— · · ·	Ч	— · — ·
И	· · ·	Ш	— · — ·
Й	· · —	Щ	— · — ·
К	— · —	Ъ	— · — ·
Л	· — · ·	Ы	— · — ·
М	— — ·	Ь	· · · ·
Н	— · —	Э	· · · ·
О	— — —	Ю	· · · —
П	— · —	Я	· · · —

### Флажковая азбука

А	Б	В	Г	Д
Е,Э	Ж	З	И,И	К
Л	М	Н	О	П
Р	С	Т	У	Ф
Х	Ц	Ч	Ш	Щ
Ъ,Ь	Ы	Ю	Я	

Источник информации

Кодирующее устройство

Передача информации

Декодирующее устройство

Получатель информации

Пример как создаётся новая мелодия

Обрез мелодии

Запись мелодии нотами

Передача музыканту носителя с нотами

Перевод исполнителем нот в звуки

Прослушивание мелодии

Огромное количество способов кодирования информации привело к попыткам создать универсальный язык или азбуку для кодирования. Эта проблема была успешно реализована лишь в отдельных областях техники, науки и культуры. Хотя человек издавна стремился упростить работу с информацией, по настоящему универсальный инструмент для этого появился не так давно, и инструмент этот – компьютер. С технической точки зрения наиболее рациональным оказалось использование в нем языка двоичных кодов.

Для обработки компьютером любая информация представляется в виде последовательности, состоящей из двух знаков 0 и 1. Цифры 0 и 1 называются двоичными (bit сокр. от англ. binary digit - двоичная цифра). Они являются символами, из которых состоит язык, понимаемый и используемый компьютером. Информация, с которой работает компьютер, «кодируется» с помощью этого языка. Таким образом, любая информация в компьютере представляется с помощью двоичных цифр. Наименьшим количеством информации является одно из двух возможных значений - 0 или 1. Такое количество информации называется бит.



# ОСОБЕННОСТИ КОДИРОВАНИЯ ИНФОРМАЦИИ В ПЕРСОНАЛЬНОМ КОМПЬЮТЕРЕ

Числовая информация внутри персонального компьютера (ПК) кодируется в двоичной или в двоично-десятичной системах счисления. При вводе и выводе любой информации в ПК используются специальные коды представления информации – коды ASCII (American Standard Code for Information Interchange), эти же коды применяются для кодирования буквенной и символьной информации и внутри ПК.

Для удобства работы введены следующие термины для обозначения совокупностей двоичных разрядов (табл. 3). Эти термины обычно используются в качестве единиц измерения объемов информации, хранимой или обрабатываемой в компьютере.

# Двоичные совокупности

Количество двоичных разрядов в группе	Наименование единицы измерения
	Бит
	Байт
	Параграф
$8 \cdot 1024^1$	Кбайт (килобайт)
$8 \cdot 1024^2$	Мбайт (мегабайт)
$8 \cdot 1024^3$	Гбайт (гигабайт)
$8 \cdot 1024^4$	Тбайт (терабайт)
$8 \cdot 1024^5$	Пбайт (пентабайт)

Последовательность нескольких бит или байт часто называют *полем данных*. Биты в числе (в слове, в поле и т. п.) нумеруются справа налево, начиная с 0-го разряда. В ПК могут обрабатываться поля *постоянной* и *переменной* длины.

Поля постоянной длины:

- q слово - 2 байт;
- q двойное слово – 4 байт;
- q полуслово – 1 байт;
- q расширенное слово – 8 байт.

Числа с фиксированной запятой чаще всего имеют формат слова и полуслова; числа плавающей запятой – формат двойного и расширенного слова. Поля переменной длины могут иметь любой размер от 0 до 255 байт, но обязательно равный целому числу байт.

Пример. Структурно запись двоичного числа  $-11000001_2$  равного десятичному  $-193_{10}$ , в разрядной сетке ПК представлена на рис. 1 и 2.

Разряд																				
Число																				
	Знак числа	Абсолютная величина числа																		

**Рис. 1. Число с фиксированной запятой формата слова со знаком**

Разряд																					...		
Число																						...	
	Знак числа	Порядок	Мантисса																				

**Рис. 2. Число с плавающей запятой формата двойного слова**

Двоично-кодированные десятичные числа могут быть представлены в ПК полями переменной длины в так называемых упакованном (рис. 3) и распакованном форматах. В упакованном формате для каждой десятичной цифры отводится по четыре двоичных разряда (полбайта), при этом знак числа кодируется в крайнем правом полубайте числа (1100 – знак «+» и 1101 – знак «-»).



Рис. 3. Структура поля упакованного формата

Упакованный формат используется обычно в ПК при выполнении операций сложения и вычитания двоично-десятичных чисел.

В распакованном формате (рис. 4) для каждой десятичной цифры отводится по целому байту, при этом старшие полубайты (зона) каждого байта (кроме самого младшего) в ПК заполняются кодом 0011 (в соответствии с ASCII-кодом), а в младших (левых) полубайтах обычным образом кодируются десятичные цифры. Старший полубайт (зона) самого младшего (правого) байта используется для кодирования знака числа



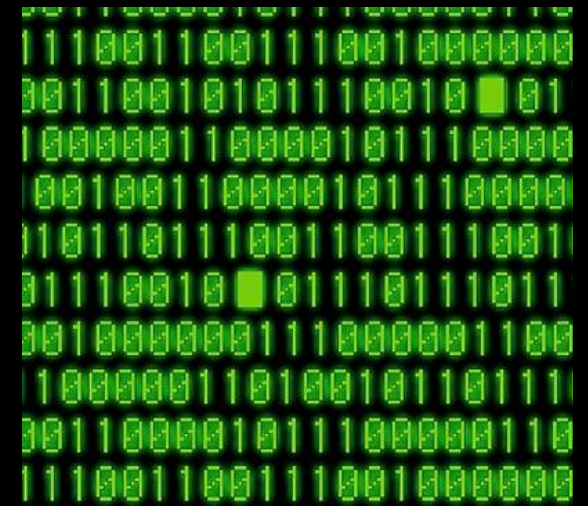
Рис. 4. Структура поля распакованного формата

Распакованный формат используется в ПК при вводе-выводе информации, а также при выполнении операций умножения и деления двоично-десятичных чисел.

Например, число  $-193_{(10)} = -000110010011_{(2-10)}$  в ПК будет представлено:  
q в упакованном формате: 0001 1001 0011 1101;  
q в распакованном формате: 0011 0001 0011 1001 1101 0011.

Код ASCII (American Standard Code for Information Interchange – американский стандартный код для обмена информацией) имеет основной стандарт и расширение (рис. 5). Основной стандарт для кодирования символов использует шестнадцатеричные коды 00-7F, расширение стандарта – коды 80-FF.

Основной стандарт является международным и используется для кодирования управляющих символов и букв латинского алфавита; в расширении стандарта кодируются символы псевдографики и буквы национального алфавита (естественно, в разных странах разные). Пользоваться таблицей достаточно просто. Следует приписать шестнадцатеричную цифру номера строки справа к шестнадцатеричной цифре номера столбца. Так получится шестнадцатеричный код символа.



										А	В	С	Д	Е	Ф
		▶		@	Р	·	Р	А	Р	а	▒	Л	Ш	Р	≡
☺	◀	!		А	Q	a	q	Б	С	б	▒	┌	┐	С	+
	↕	"		В	R	b	r	В	Т	в	▒	t	┐	т	≥
	!!	#		С	S	c	s	Г	У	г			┐	у	≤
	π	\$		D	T	d	t	Д	Ф	д		-	t	ф	
		%		E	U	e	u	Е	Х	е	≡	+	F	Х	
	■	&		F	V	f	v	Ж	Ц	ж	≡	≡	≡	ц	÷
'	↕	'		G	W	G	w	З	Ч	з	≡	≡	≡	ч	≈
▣	↑	(		Н	X	Н	x	И	Ш	и	≡	≡	≡	ш	○
	↓	)		I	Y	i	y	Й	Щ	й	≡	≡	≡	щ	'
<b>А</b>		→	*	:	J	Z	j	К	Ъ	к	≡	±	≡	ъ	·
<b>В</b>	♂	←	+	;	K	[	k	{	Л	Ы	л	≡	▀	ы	√
<b>С</b>	♀	┌	,	<	L	\			М	Ь	м	≡	▀	ь	π
<b>Д</b>	♪	↔	-	=	M	]	m	}	Н	Э	н		≡	э	²
<b>Е</b>		▲	/	>	N	^	n	~	О	Ю	о	≡	▀	ю	▪
<b>Ф</b>	*	▼	/	?	O	-	o	Δ	П	Я	п	≡	▀	я	

Рис. 5. Таблица кодов ASCII

**Примечание.** Любой символ, представленный в этой таблице, при работе в DOS может быть введен в ПК с клавиатуры набором его десятичного кода (соответствующего шестнадцатеричному ASCII-коду) на малой цифровой клавиатуре при нажатой клавише Alt.

# Дискретизация

**Видеосигнал** - представление непрерывного аналогового сигнала последовательностью его значений (отсчетов). Эти отсчеты берутся в моменты времени, отделенные друг от друга интервалом, который называется интервалом дискретизации. Величину, обратную интервалу между отсчетами, называют частотой дискретизации. На рис. 1 показаны исходный аналоговый сигнал и его дискретизированная версия. Картинки, приведенные под временными диаграммами, получены в предположении, что сигналы являются телевизионными видеосигналами одной строки, одинаковыми для всего телевизионного растра.

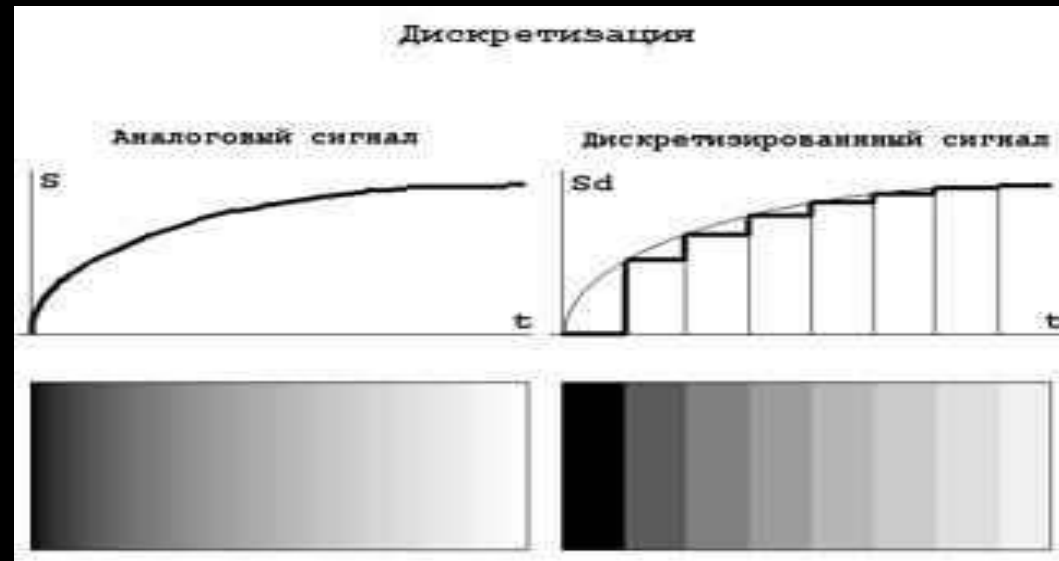


Рис.6 Аналого-цифровое преобразование. Дискретизация.

**Аналоговый сигнал** — сигнал данных, у которого каждый из представляющих параметров описывается функцией времени и непрерывным множеством возможных значений.

Различают два пространства сигналов — пространство  $L$  (непрерывные сигналы), и пространство  $l$  ( $L$  малое) — пространство последовательностей. Пространство  $l$  ( $L$  малое) есть пространство коэффициентов Фурье (счётного набора чисел, определяющих непрерывную функцию на конечном интервале области определения), пространство  $L$  — есть пространство непрерывных по области определения (аналоговых) сигналов. При некоторых условиях, пространство  $L$  однозначно отображается в пространство  $l$  (например, первые две теоремы дискретизации Котельникова)

Аналоговые сигналы описываются непрерывными функциями времени, поэтому аналоговый сигнал иногда называют **континуальным сигналом**. Аналоговым сигналам противопоставляются *дискретные* (квантованные, цифровые).



Если потребовать, чтобы в процессе дискретизации не возникало искажений ТВ сигнала (рис. 9) с граничной частотой, например, 6 МГц, то частота дискретизации должна быть не меньше 12 МГц. Однако, чем ближе частота дискретизации к удвоенной граничной частоте сигнала, тем труднее создать фильтр нижних частот, который используется при восстановлении, а также при предварительной фильтрации исходного аналогового сигнала.

Это объясняется тем, что при приближении частоты дискретизации к удвоенной граничной частоте дискретизируемого сигнала предъявляются все более жесткие требования к форме частотных характеристик восстанавливающих фильтров - она все точнее должна соответствовать прямоугольной характеристике.

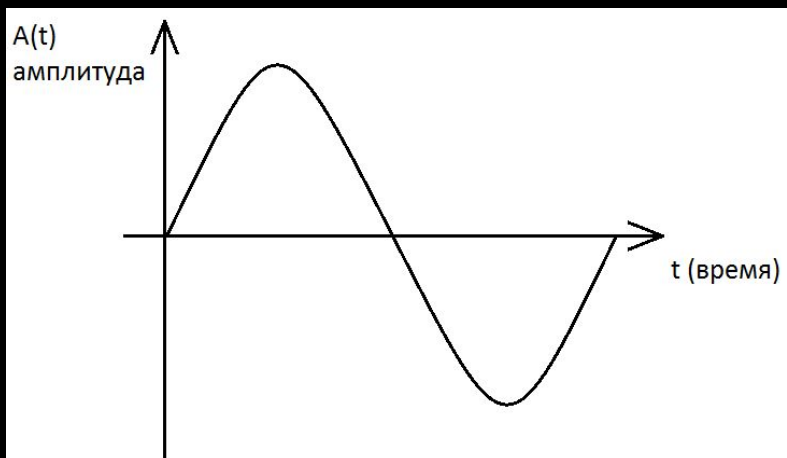
Следует подчеркнуть, что фильтр с прямоугольной характеристикой не может быть реализован физически. Такой фильтр, как показывает теория, должен вносить бесконечно большую задержку в пропускаемый сигнал. Поэтому на практике всегда существует некоторый интервал между удвоенной граничной частотой исходного сигнала и частотой дискретизации.



Рис. 9

**Звуковой сигнал** - компьютер является мощнейшим устройством для обработки различных типов информации, в том числе и звуковой. Но аналоговый звук непригоден для обработки на компьютере, его необходимо **преобразовать в цифровой**. Для этого используются специальные устройства - **аналого-цифровые преобразователи или АЦП**. В компьютере роль АЦП выполняет **звуковая карта**. Каким же образом АЦП преобразует сигнал из аналогового в цифровой вид? Давайте разберемся.

Пусть у нас есть источник звука с частотой **440Гц**, пусть это будет гитара. Сначала звук нужно превратить в электрический сигнал. Для этого используем микрофон. На выходе микрофона мы получим электрический сигнал с частотой **440Гц**. Графически он выглядит таким образом:



Следующая задача - преобразовать этот сигнал в цифровой вид, то есть в последовательность цифр. Для этого используется **временная дискретизация** - аналоговый звуковой сигнал разбивается на отдельные маленькие временные участки и для каждого такого участка устанавливается определенная величина интенсивности звука, которая зависит от амплитуды.

Другими словами через какие-то промежутки времени мы измеряем уровень аналогового сигнала. Количество таких измерений за одну секунду называется частотой дискретизации. **Частота дискретизации измеряется в Герцах**. Соответственно, если мы будем измерять наш сигнал 100 раз в секунду, то частота дискретизации будет равна 100Гц.

Вот примеры некоторых используемых частот дискретизации звука:

*8 000 Гц - телефон, достаточно для речи;*

*11 025 Гц;*

*16 000 Гц;*

*22 050 Гц - радио;*

*32 000 Гц;*

*44 100 Гц - используется в Audio CD;*

*48 000 Гц - DVD, DAT;*

*96 000 Гц - DVD-Audio (MLP 5.1);*

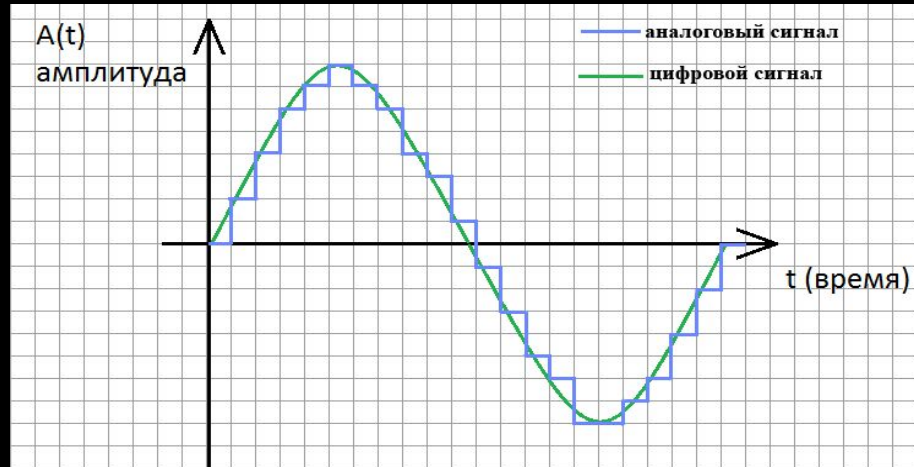
*192 000 Гц - DVD-Audio (MLP 2.0);*

*2 822 400 Гц - SACD, процесс однобитной дельта-сигма модуляции, известный как DSD - Direct Stream Digital, совместно разработан компаниями Sony и Philips;*

*5,644,800 Гц - DSD с удвоенной частотой дискретизации, однобитный Direct Stream Digital с частотой дискретизации вдвое больше, чем у SACD. Используется в некоторых профессиональных устройствах записи DSD.*

Современные звуковые карты способны оцифровывать звук с частотой дискретизации 96КГц и даже 192 кГц.

В итоге наш аналоговый сигнал превратится в цифровой, а график станет уже не гладким, а ступенчатым, дискретным:



**Глубина кодирования звука** - это количество возможных уровней сигнала. Другими словами глубина кодирования это точность измерения сигнала. **Глубина кодирования измеряется в битах.** Например, если количество возможных уровней сигнала равно **255**, то глубина кодирования такого звука **8 бит**. **16-битный** звук уже позволяет работать с **65536** уровнями сигнала.

Современные звуковые карты обеспечивают **глубину кодирования в 16 и даже 24 бита**, а это возможность кодирования **65536 и 16 777 216 различных уровней громкости соответственно.** Зная **глубину кодирования**, можно легко **узнать количество уровней сигнала цифрового звука.** Для этого используем формулу

## Единицы измерения

**Байт (англ. byte)** - единица измерения количества информации, в стандартном виде [источник не указан 42 дня] байт считается равным восьми битам (в этом случае может принимать 256 (28) различных значений) .

Байт в современных x86 совместимых компьютерах - это минимально адресуемая последовательность фиксированного числа битов. При хранении данных в памяти существует также бит чтения-записи, а для цифровых микросхем - бит синхронизации, т. е. в реализации для x86 байт состоит обычно из 10 битов (из них 8 бит данных и 2 системных бита поддержки). Для того, чтобы подчеркнуть, что имеется в виду восьми битный байт, в описании сетевых протоколов используется термин **«октет» (лат. octet)**.

Иногда байтом называют последовательность битов, которые составляют подполе машинного слова, используемое для кодирования одного текстового символа (хотя правильней это называть символом, а не байтом) . На некоторых компьютерах возможна адресация слов разной длины. Это предусмотрено инструкциями извлечения полей ассемблеров LDB и DPB на PDP-10 и в языке Common Lisp.

Ряд ЭВМ 1950-х и 1960-х годов использовали 6-битовые символы в 48-битовых или 60-битовых машинных словах. В некоторых моделях ЭВМ производства Burroughs Computer Corporation (ныне Unisys) размер символа был равен 9 битам. Во многих современных цифровых сигнальных процессорах используется машинное слово длиной 16 бит и больше.

Название **Байт** было впервые использовано в 1956 году В. Бухгольцем при проектировании первого суперкомпьютера IBM 7030 для пучка одновременно передаваемых в устройствах ввода-вывода битов (шести штук) , позже в рамках того же проекта расширили байт до восьми (23) бит.

Байтовая адресация памяти была впервые применена в системе IBM System/360. В более ранних компьютерах адресовать можно было только целиком машинное слово, состоявшее из нескольких символов, что затрудняло обработку нечисловых данных.

Кратные приставки для образования производных единиц для байта применяются не как обычно: во-первых, уменьшительные приставки не используются совсем, а единицы измерения информации меньше чем байт называются специальными словами (нибл и бит) ; во-вторых, увеличительные приставки означают за каждую тысячу  $1024=2^{10}$  (килобайт равен 1024 байтам, мегабайт равен 1024 килобайтам, или 1 048 576 байтам; и т. д. с гига-, тера- и петабайтами (больше пока не употребляются)) . Разница возрастает с ростом веса приставки. Более правильно использовать двоичные приставки, но на практике они пока не применяются, возможно, из-за неблагозвучности — кибибайт, меbibайт, йобибайт и т. п.

Иногда десятичные приставки используются и в прямом смысле, например, при указании ёмкости жёстких дисков: у них гигабайт может обозначать миллион кибибайтов, т. е. 1 024 000 000 байтов, а то и просто миллиард байтов, а не 1 073 741 824 байта, как, например, в модулях памяти; а так же при указании пропускной способности каналов передачи данных (сетей).

1 б (байт) = 8 бит (8 двоичных разрядов)

1 Кб (Килобайт) =  $2^{10}$  б = 1024 б.

1 Мб (Мегабайт) =  $2^{20}$  б = 1024 Кб.

1 Гб (Гигабайт) =  $2^{30}$  б = 1024 Мб.

1 Тб (Терабайт) =  $2^{40}$  б = 1024 Гб.

1 Пб (Петабайт) =  $2^{50}$  б = 1024 Тб.

# **Тест по теме кодирование**



Вопрос №1. Набор однозначных символов это...

Вопрос №2. Роль АЦП в компьютере выполняет ... карта.

Вопрос№3.

Представление непрерывного аналогового сигнала последовательностью его значений это...

Вопрос№4. Система условных знаков для представления информации это?

Вопрос№5. Первая кодировка текста в которой был русский алфавит?

Вопрос№6. Наука о методах обеспечения конфиденциальности?

Вопрос №7. Периодического процесса, равна количеству повторений или возникновения событий?

Вопрос№8. Инструмент для открывания замка?



Вопрос №9. Непростая задача, для решения которой, как правило, требуется сообразительность?

Вопрос№10. Процесс раскрытия секретной информации?

# Результаты

**Всего заданий выполнено:**

**Выполнено верно:**

**Процент выполнения:**

**Оценка:**

# Кроссворд



1. Наука о математических методах обеспечения конфиденциальности
2. Совокупность знаков (символов) и система определенных правил
3. Свойство, противопоставляемое непрерывности, прерывность.
4. Информация в криптографии, используемая алгоритмом для преобразования сообщения при шифровании или расшифровке.
5. Физическая величина, характеристика периодического процесса, равна количеству повторений или возникновения событий (процессов) в единицу времени.
6. Роль АЦП в компьютере выполняет ... карта.
7. Процесс раскрытия секретной информации?
8. Совокупность операций для определения отношения одной величины к другой однородной величине, принятой за единицу, хранящаяся в техническом средстве.
9. Числительный знак, выражающий число это, 1
10. Сторона языка программирования, которая описывает структуру программ как наборов символов