

# **Кодирование символьной, графической, звуковой информации**

# Кодирование символьной информации

Каждому символу можно сопоставить целое положительное число и назвать его **кодом символа**.

Соответствие символа числовому значению кода формирует **таблицу кодирования**.

Как закодировать все используемые символы?



таблица кодирования **ASCII**

(American Standard Code for Information Interchange )

Базовая

Расширенная

значения кодов  
от 0 до 127

значения кодов  
от 128 до 255

Для представления одного символа выделяется 1 байт (8 бит).

Набор из 8 разрядов может принять  $2^8 = 256$  различных значений.

ASCII Code Chart																
0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI	
DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US	
!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/		
0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?	
@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	
P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	-	
~	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	
p	q	r	s	t	u	v	w	x	y	z	{		}	-	DEL	

Первые 32 кода (от 0 до 31) базовой таблицы выделены для управления работой технических устройств.

В 60-х годах XX века в *американском национальном институте стандартизации (ANSI)* была разработана таблица кодирования символов, которая впоследствии была использована во всех операционных системах.

В системе **ASCII** закреплены две таблицы кодирования –  
**базовая 0 до 127**  
**и расширенная 128 до 255.**

Первые 32 кода (от 0 до 31) базовой таблицы выделены производителям аппаратных средств (в первую очередь компьютеров и печатающих устройств). Это управляющие коды, которым не соответствуют никакие символы, ими можно управлять работой технических устройств.

С 32 по 127 - основная часть таблицы, входят десятичные цифры, буквы латинского алфавита (заглавные и строчные), знаки препинания (точка, запятая, скобки и др.), а также пробел и различные служебные символы (табуляция, перевод строки и др.).

# Таблица символов ASCII

Dec	Hex	Char	Cmd	Dec	Hex	Char	Cmd	Dec	Hex	Char	Cmd
0	00		NUL	32	20		(sp)	64	40	@	
1	01	␀	SOH	33	21	!		65	41	A	
2	02	␁	STX	34	22	"		66	42	B	
3	03	␂	ETX	35	23	#		67	43	C	
4	04	␃	EOT	36	24	\$		68	44	D	
5	05	␄	ENQ	37	25	%		69	45	E	
6	06	␅	ACK	38	26	&		70	46	F	
7	07	•	BEL	39	27	'		71	47	G	
8	08	␆	BS	40	28	(		72	48	H	
9	09	◦	TAB	41	29	)		73	49	I	
10	0A	␇	LF	42	2A	*		74	4A	J	
11	0B	♂	VT	43	2B	+		75	4B	K	
12	0C	♀	FF	44	2C	,		76	4C	L	
13	0D	␈	CR	45	2D	-		77	4D	M	
14	0E	♪	SO	46	2E	.		78	4E	N	
15	0F	␉	SI	47	2F	/		79	4F	O	
16	10	▶	DLE	48	30	0		80	50	P	
17	11	◀	DC1	49	31	1		81	51	Q	
18	12	↕	DC2	50	32	2		82	52	R	
19	13	!!	DC3	51	33	3		83	53	S	
20	14	¶	DC4	52	34	4		84	54	T	
21	15	§	NAK	53	35	5		85	55	U	
22	16	—	SYN	54	36	6		86	56	V	
23	17	↕	ETB	55	37	7		87	57	W	
24	18	↑	CAN	56	38	8		88	58	X	
25	19	↓	EM	57	39	9		89	59	Y	
26	1A	→	SUB	58	3A	:		90	5A	Z	
27	1B	←	ESC	59	3B	;		91	5B	[	
28	1C	∟	FS	60	3C	<		92	5C	\	
29	1D	↔	GS	61	3D	=		93	5D	]	
30	1E	▲	RS	62	3E	>		94	5E	^	
31	1F	▼	US	63	3F	?		95	5F	_	

Продолжение таблицы символов ASCII имеет множество различных реализаций, которые определяются региональными (языковыми) особенностями.

Dec	Hex	Char									
128	80	А	160	A0	а	192	C0	҂	224	E0	р
129	81	Б	161	A1	б	193	C1	҃	225	E1	с
130	82	В	162	A2	в	194	C2	҄	226	E2	т
131	83	Г	163	A3	г	195	C3	҅	227	E3	у
132	84	Д	164	A4	д	196	C4	҆	228	E4	ф
133	85	Е	165	A5	е	197	C5	҇	229	E5	х
134	86	Ж	166	A6	ж	198	C6	҈	230	E6	ц
135	87	З	167	A7	з	199	C7	҉	231	E7	ч
136	88	И	168	A8	и	200	C8	Ҋ	232	E8	ш
137	89	Й	169	A9	й	201	C9	ҋ	233	E9	щ
138	8A	К	170	AA	к	202	CA	Ҍ	234	EA	ъ
139	8B	Л	171	AB	л	203	CB	ҍ	235	EB	ы
140	8C	М	172	AC	м	204	CC	Ҏ	236	EC	ь
141	8D	Н	173	AD	н	205	CD	ҏ	237	ED	э
142	8E	О	174	AE	о	206	CE	Ґ	238	EE	ю
143	8F	П	175	AF	п	207	CF	ґ	239	EF	я
144	90	Р	176	B0	Ғ	208	D0	ғ	240	F0	Ё
145	91	С	177	B1	Ҕ	209	D1	Ҕ	241	F1	ë
146	92	Т	178	B2	ҕ	210	D2	Җ	242	F2	€
147	93	У	179	B3	җ	211	D3	҈	243	F3	€
148	94	Ф	180	B4	ҙ	212	D4	҉	244	F4	Ї
149	95	Х	181	B5	Ҋ	213	D5	Ҋ	245	F5	ї
150	96	Ц	182	B6	ҋ	214	D6	ҋ	246	F6	Ӯ
151	97	Ч	183	B7	Ҍ	215	D7	Ҍ	247	F7	ӹ
152	98	Ш	184	B8	ҍ	216	D8	ҍ	248	F8	Ӱ
153	99	Щ	185	B9	Ҏ	217	D9	Ҏ	249	F9	.
154	9A	҂	186	BA	ҏ	218	DA	ҏ	250	FA	.
155	9B	҃	187	BB	Ґ	219	DB	Ґ	251	FB	√
156	9C	҄	188	BC	҈	220	DC	҈	252	FC	№
157	9D	҅	189	BD	҉	221	DD	҉	253	FD	□
158	9E	҆	190	BE	Ҋ	222	DE	Ҋ	254	FE	■
159	9F	҇	191	BF	ҋ	223	DF	ҋ	255	FF	

# Кодирование символьной информации



## Универсальная система кодирования Unicode

- набор графических символов (символы, имеющие видимое изображение) и способ их кодирования для компьютерной обработки текстовых данных.

Графические символы включают в себя *следующие группы*:

- ❖ буквы, содержащиеся хотя бы в одном из обслуживаемых алфавитов;
- ❖ цифры;
- ❖ знаки пунктуации;
- ❖ специальные знаки (математические, технические и пр.);
- ❖ разделители.

Используя Unicode можно закодировать до  $2^{31}$  (2 147 483 648) кодовых позиций.

В Unicode первые 128 кодов совпадают с таблицей ASCII.

Чаще используется сокращенная 16-битовая версия ( $2^{16} = 65\ 536$  значений), где кодируются все современные алфавиты.

**Юникод** (*Unicode*) — стандарт кодирования символов, включающий в себя знаки почти всех письменных языков мира. Стандарт предложен в 1991 году.

Позволяет закодировать большое число символов из разных систем письменности:

в документах, закодированных по стандарту Юникод, могут соседствовать китайские иероглифы, математические символы, буквы греческого алфавита, латиницы и кириллицы, символы музыкальной нотной нотации, при этом становится ненужным переключение кодовых страниц.

Однако информационная емкость 31-битового Unicode все равно остается слишком большой.

Поэтому чаще используется сокращенная 16-битовая версия  
( $2^{16} = 65\,536$  значений), где кодируются все современные алфавиты.

Присваивается каждому символу код в виде неотрицательного целого числа, записываемого в шестнадцатеричной форме с префиксом U+.

Область с кодами от U+0000 до U+007F содержит символы набора ASCII, и коды этих символов совпадают с их кодами в ASCII.

Под символы кириллицы выделены области знаков с кодами от U+0400 до U+052F, от U+2DE0 до U+2DFF, от U+A640 до U+A69F.

## Основная многоязычная плоскость.

В стандарте [Юникод](#) **плоскость** — это непрерывный диапазон из 65 536 ( $2^{16}$ ) кодовых позиций.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	Латинская письменность
10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	Нелатинские европейские письменности
20	21	22	23	24	25	26	27	28	29	2A	2B	2C	2D	2E	2F	Письменности Африки
30	31	32	33	34	35	36	37	38	39	3A	3B	3C	3D	3E	3F	Письменности Среднего Востока и Юго-Западной Азии
40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	Письменности Южной и Центральной Азии
50	51	52	53	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F	Письменности Юго-Восточной Азии
60	61	62	63	64	65	66	67	68	69	6A	6B	6C	6D	6E	6F	Письменности Восточной Азии
70	71	72	73	74	75	76	77	78	79	7A	7B	7C	7D	7E	7F	Идеограммы ККЯ
80	81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E	8F	Письменности Индонезии и Океании
90	91	92	93	94	95	96	97	98	99	9A	9B	9C	9D	9E	9F	Письменности Америки
A0	A1	A2	A3	A4	A5	A6	A7	A8	A9	AA	AB	AC	AD	AE	AF	Системы нотописи
B0	B1	B2	B3	B4	B5	B6	B7	B8	B9	BA	BB	BC	BD	BE	BF	Знаки
C0	C1	C2	C3	C4	C5	C6	C7	C8	C9	CA	CB	CC	CD	CE	CF	Область для частного использования
D0	D1	D2	D3	D4	D5	D6	D7	D8	D9	DA	DB	DC	DD	DE	DF	Суррогатные пары UTF-16
E0	E1	E2	E3	E4	E5	E6	E7	E8	E9	EA	EB	EC	ED	EE	EF	Свободные кодовые позиции
F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	FA	FB	FC	FD	FE	FF	

По состоянию на версию Юникода 17.0

Основная многоязычная плоскость. Каждая пронумерованная ячейка содержит 256 кодовых позиций.

# Кодирование графической информации

**Компьютерная графика** (computer graphics, CG) —

это область информационных технологий, которая охватывает создание, обработку и манипулирование визуальной информацией с помощью специализированного программного обеспечения.

К компьютерной графике относится любое изображение, созданное или обработанное на компьютере.

Создано разнообразное аппаратное и программное обеспечение для получения изображений самого различного вида и назначения - от простых чертежей до реалистических образов естественных объектов.



Растровое

Векторное

Изображение

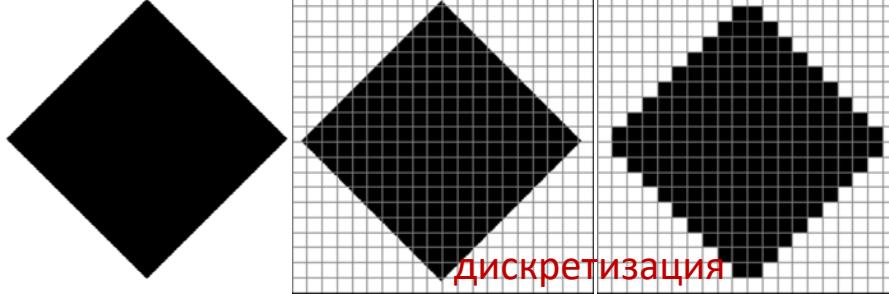
# Кодирование графической информации

## Растровое кодирование

### Растровое

представляет собой сетку пикселей разных цветов.

### Получение растрового изображения

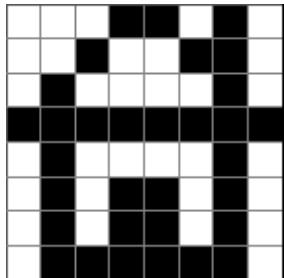


- на изображение ромба наложена сетка, которая разбивает его на квадраты. Такая сетка называется *растром*.
- для каждого квадрата определим цвет (черный или белый).
- **Дискретизация** – разбили единый объект на отдельные элементы.

**Пиксель** (англ. pixel = picture element, элемент рисунка) – это наименьший элемент рисунка, для которого можно задать свой цвет.

### Построение двоичного кода:

- ✓ заменяем белые пиксели «0», а черные – «1»;
- ✓ выписываем строки полученной таблицы одну за другой.



0	0	0	1	1	0	1	0
0	0	1	0	0	1	1	0
0	1	0	0	0	0	1	0
1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	0
0	1	0	1	1	0	1	0
0	1	0	1	1	0	1	0
0	1	1	1	1	1	1	0

**Разрешение** – это количество пикселей, приходящихся на дюйм размера изображения.

**Разрешение измеряется в пикселях на дюйм:**

254 ppi – на дюйм (25,4 мм) приходится 254 пикселя, т.е. пиксель имеет размер 0,1×0,1 мм.

Часть исходной информации при кодировании была потеряна. Чтобы уменьшить потери информации, нужно уменьшать размер пикселя.

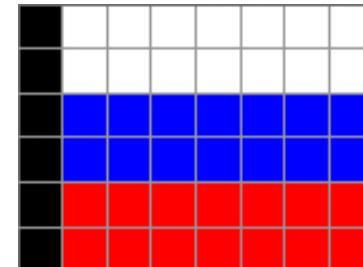


Чем больше разрешение, тем точнее кодируется рисунок (меньше информации теряется), однако одновременно растет и объем файла.

## Изображение

Для кодирования одного из четырех вариантов нужно 2 бита, поэтому код каждого цвета (и код каждого пикселя) будет состоять из 2-х бит.

Пусть 00 обозначает черный цвет, 01 – красный, 10 – синий и 11 – белый.

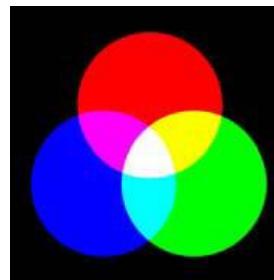
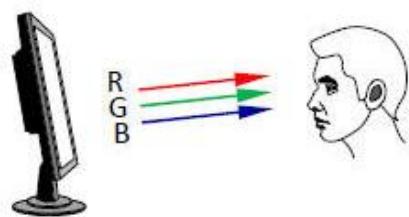


00	11	11	11	11	11	11	11	11
00	11	11	11	11	11	11	11	11
00	10	10	10	10	10	10	10	10
00	10	10	10	10	10	10	10	10
00	01	01	01	01	01	01	01	01
00	01	01	01	01	01	01	01	01

**Проблема:** как определить, какой цвет соответствует тому или другому коду?

Любой цвет приближенно раскладывается на три составляющих – красную, зеленую и синюю.

## Модель RGB



**Код цвета** – это тройка чисел (R,G,B), яркости отдельных каналов.

В модели **RGB** яркость каждой составляющей кодируется целым числом от 0 до 255. Всего есть по 256 вариантов яркости каждого из трех цветов.

(0,0,0)

(255,255,255)

(255,150,150)

(100,0,0)

Меняя силу этих составляющих, можно составить любые цвета.

**Глубина цвета** – количество бит, необходимое для кодирования цвета пикселя.

всего **256** вариантов яркости каждого из трех цветов

можно закодировать  **$256^3 = 16\ 777\ 216$**  оттенков

каждая составляющая цвета занимает в памяти **8 бит (1 байт)**

вся информация о каком-то цвете – **24 бита (3 байта)**

## Модель цвета RGB - *red* (красный), *green* (зеленый) и *blue*(синий).

Любой цвет (в том числе и «белый») приближенно раскладывается на три составляющих – красную, зеленую и синюю. Меняя силу этих составляющих, можно составить любые цвета.

В модели **RGB** яркость каждой составляющей (или каждого канала) кодируется целым числом от 0 до 255, каждая из трех составляющих занимает в памяти 8 бит или 1 байт,  $2^8 = 256$ , получается по 256 вариантов яркости каждого из трех цветов.

При этом код цвета – это тройка чисел (R,G,B), яркости отдельных каналов, вся информация о каком-то цвете – 24 бита (или 3 байта)

Это позволяет закодировать  $256^3 = 16\ 777\ 216$  оттенков, что более чем достаточно для человека.

### Примеры модели RGB

b=3			
Цвет	R	G	B
Черный	0	0	0
Синий	0	0	1
Зеленый	0	1	0
Голубой	0	1	1
Красный	1	0	0
Пурпурный	1	0	1
Желтый	1	1	0
Белый	1	1	1

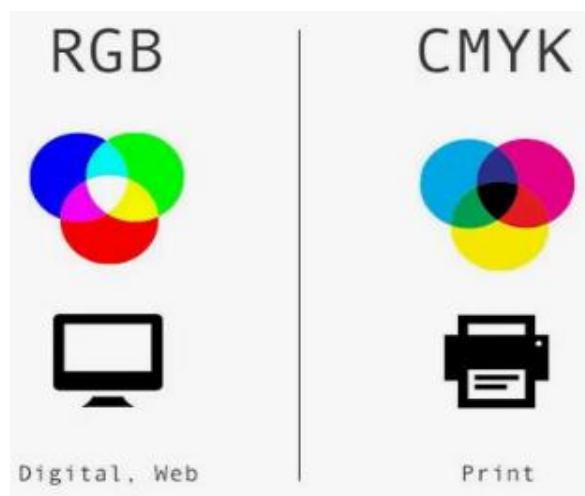
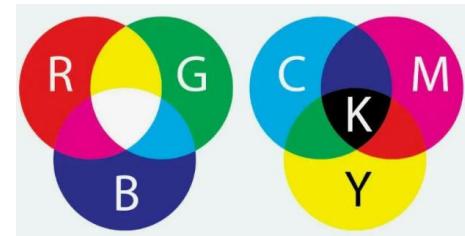
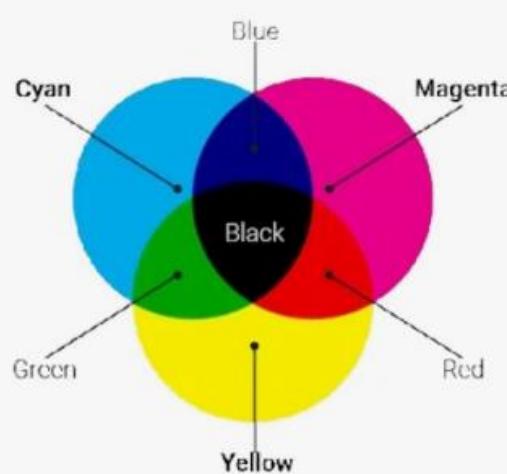
b=24 (True Color)			
Серый (100, 100, 100)			
Черный (0, 0, 0)			
Синий (0, 0, 255)			
Зеленый (0, 255, 0)			
Голубой (0, 255, 255)			
Красный (255, 0, 0)			
Пурпурный (255, 0, 255)			
Желтый (255, 255, 0)			
Белый (255, 255, 255)			

Если все составляющие имеют равную яркость, получаются оттенки серого цвета, от черного до белого.

Чтобы сделать светло-красный (розовый) цвет, нужно в красном цвете (255,0,0) одинаково увеличить яркость зеленого и синего каналов, например, цвет (255, 150, 150) – это розовый. Равномерное уменьшение яркости всех каналов делает темный цвет, например, цвет с кодом (100,0,0) – тёмно-красный.

## CMYK — ЦВЕТОВАЯ МОДЕЛЬ, КОТОРУЮ ИСПОЛЬЗУЮТ ПРИ СОЗДАНИИ ИЗОБРАЖЕНИЙ ДЛЯ ПЕЧАТИ.

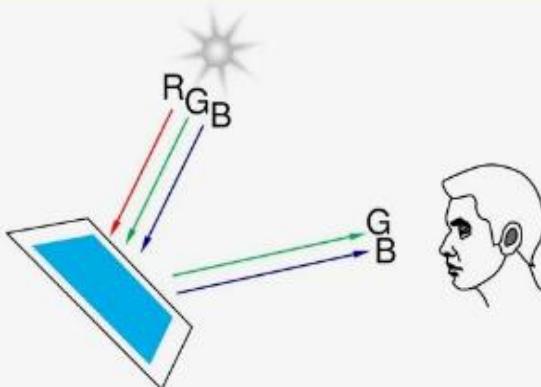
Аббревиатура CMYK расшифровывается как:  
C – циан (Cyan) – ярко-голубой;  
M – маджента (Magenta) – ярко-розовая;  
Y – желтый (Yellow) – ярко-желтая;  
K – черный (Black) – черная краска.



**CMYK** ПРЕДПОЧТИТЕЛЬНЕЕ для печатных материалов, так как он обеспечивает большую согласованность и точность цветов. Цвета в CMYK могут быть более точно воспроизведены на физических носителях, таких как бумага. CMYK идеален для брошюр, визиток и плакатов.

**RGB** лучше подходит для цифровых проектов, отображаемых на экранах. RGB способен воспроизводить более яркие и насыщенные цвета по сравнению с CMYK. Эта цветовая палитра применяется при создании дизайна сайта, социальных сетей, видеороликов или презентаций.

# Цветовая модель CMYK (при печати)



C M Y

0 0 0

255 255 0

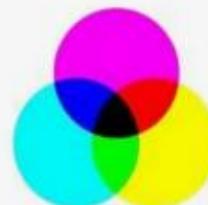
255 0 255

0 255 255

255 255 255



C = Cyan  
M = Magenta  
Y = Yellow



Модель CMY

Модель CMYK: + Key color



- меньший расход краски и лучшее качество для чёрного и серого цветов

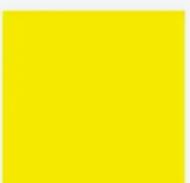
CMYK COLOR CODES



CYAN



MAGENTA



YELLOW



KEY BLACK



TURQUOISE

C	100%
M	0%
Y	0%
K	0%

C	0%
M	100%
Y	0%
K	0%

C	0%
M	0%
Y	100%
K	0%

C	0%
M	0%
Y	0%
K	100%

ЦВЕТОВОЕ ПРОСТРАНСТВО.

**RGB:** КАЖДЫЙ ЦВЕТ ОПРЕДЕЛЯЕТСЯ ТРЕМЯ ЗНАЧЕНИЯМИ (от 0 до 255).

**CMYK:** КАЖДЫЙ ЦВЕТ ОПРЕДЕЛЯЕТСЯ ЧЕТЫРЬМА ЗНАЧЕНИЯМИ (в процентах от 0% до 100%).

**RGB** ПОЗВОЛЯЕТ ДОБИТЬСЯ ЯРКИХ И НАСЫЩЕННЫХ ЦВЕТОВ, КОТОРЫЕ ХОРОШО СМОТРЯТСЯ НА ЭКРАНАХ.

**CMYK** — ТОЧНОГО ВОСПРОИЗВЕДЕНИЯ ЦВЕТОВ НА ПЕЧАТНЫХ МАТЕРИАЛАХ.

**CMYK**, суть модели: ЦВЕТА ФОРМИРУЮТСЯ ПУТЁМ СМЕШИВАНИЯ И ПОГЛОЩЕНИЯ ОДНИМ ЦВЕТОМ ДРУГОГО, А ИНТЕНСИВНОСТЬ ЗАДАЁТСЯ ПРОЦЕНТАМИ от 0% до 100%.

**CMYK** - БЕЛОГО ЦВЕТА В ЭТОЙ СХЕМЕ НЕ ПРЕДУСМОТРЕНО. БЕЛЫЙ ЦВЕТ ПРИ ПОЛНОЦВЕТНОЙ ОФСЕТНОЙ ПЕЧАТИ ПОЯВЛЯЕТСЯ ТОЛЬКО ТАМ, ГДЕ ВСЕ ЧЕТЫРЕ КРАСКИ ОТСУТСТВУЮТ, И ЕГО ОТТЕНОК ЗАВИСИТ ОТ МАТЕРИАЛА, ТО ЕСТЬ БУМАГИ.

Т.е. изображения в растровом формате строится на основе разбиения изображения на прямоугольную матрицу элементов.

Существует подход хранения пикселей - хранить числа, соответствующие пикселям, одно за другим.

Разрешение

### **DPI и PPI: ключевые определения и отличия**

**DPI** (Dots Per Inch) и **PPI** (Pixels Per Inch) — параметры, определяющие разрешение и качество изображения, однако относятся они к принципиально разным сферам применения.

**DPI (точек на дюйм)** — физический параметр, показывающий, сколько отдельных чернильных точек способно нанести печатающее устройство на отрезке в один дюйм (2,54 см). Это характеристика принтера или иного печатного оборудования. При печати документа принтер формирует изображение из микроскопических капель чернил или тонера — точек.

**PPI (пикселей на дюйм)** — цифровой параметр, указывающий на плотность пикселей (элементарных точек цифрового изображения) на дюйм экрана или в файле изображения. Каждый пиксель имеет один определенный цвет, а их совокупность формирует видимое нами цифровое изображение.

Ключевое различие состоит в том, что DPI — это физическая характеристика вывода изображения на материальный носитель, в то время как PPI — параметр самого цифрового файла или характеристика дисплея.

Одно и то же изображение с определенным значением PPI может быть напечатано с различными настройками DPI.

**Глубина цвета** – количество бит, необходимое для кодирования цвета пикселя.

Количество цветов (например, отображаемых на экране монитора) может быть вычислено по формуле:

$N=2^I$  , где  $N$  – количество цветов,  $I$  – глубина цвета или битовая глубина.

$$8 : 2^8 = 256$$

$$16 : 2^{16} = 65\,536$$

$$24 : 2^{24} = 16\,777\,216$$

$$32 : 2^{32} = 429\,496\,7296$$

Например:

**16** - битное кодирование (англ. *HighColor* – «высокий» цвет),  
когда на красную и синюю составляющую отводится по 5 бит,  
а на зеленую, к которой человеческий глаз более чувствителен – 6 бит.

**24** - битное кодирование цвета часто называют режимом **истинного цвета** (англ. *TrueColor* – истинный цвет).

Растровое

Векторное

Изображение

# Кодирование графической информации

Векторное

основано на использовании совокупности графических примитивов, каждый из которых описывается математической формулой.

Векторное изображение



Простые графические примитивы

Отрезки

Параметры отрезков, параметры линий, тип заливки

Математические формулы

Кодирование буквенной и цифровой информации

Изображение может быть полностью разложено на простейшие геометрические фигуры, в этом случае при кодировании нет потери информации.

Информация о векторном изображении кодируется как обычная буквенно-цифровая и обрабатывается специальными программами.

Изображение закодировано в виде набора простейших геометрических фигур, параметры которых (размеры, координаты вершин, углы наклона, цвет контура и заливки) хранятся в виде чисел.

Векторный способ кодирования рисунки обладает значительными **преимуществами** в сравнении с растровым тогда, когда изображение может быть полностью разложено на простейшие геометрические фигуры (например, чертеж, схема, карта, диаграмма).

В этом случае при кодировании нет потери информации.

Объем файлов напрямую зависит от сложности рисунка – чем меньше элементов, тем меньше места занимает файл.

Как правило, векторные рисунки значительно меньше по объему, чем растровые.

## **ФОРМАТЫ ГРАФИЧЕСКИХ ИЗОБРАЖЕНИЙ**

### **Растровая графика**

Основные форматы: JPEG, PNG, BMP, GIF

### **Векторная графика**

Основные форматы: SVG, EPS, AI, CDR

# Кодирование звуковой информации

**Звук** — физическое явление, представляющее собой распространение в виде упругих волн механических колебаний в твёрдой, жидкой или газообразной среде.

Звуковые волны в воздухе



Звук  
непрерывен

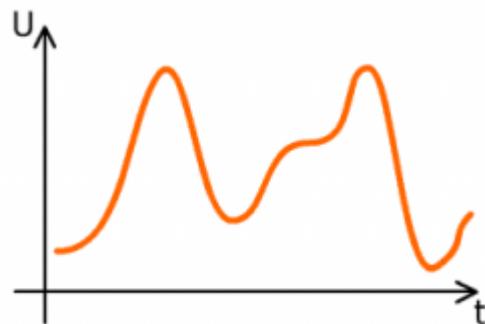
Как хранить звук на компьютере?

Информация  
дискретна

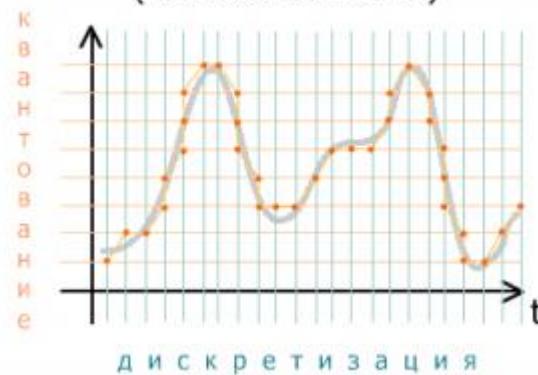
**Аналоговый сигнал** — сигнал данных, у которого каждый из представляющих параметров описывается функцией времени и непрерывным множеством возможных значений.

**Цифровой сигнал** — сигнал данных, у которого каждый из представляющих параметров описывается функцией дискретного времени и конечным множеством возможных значений.

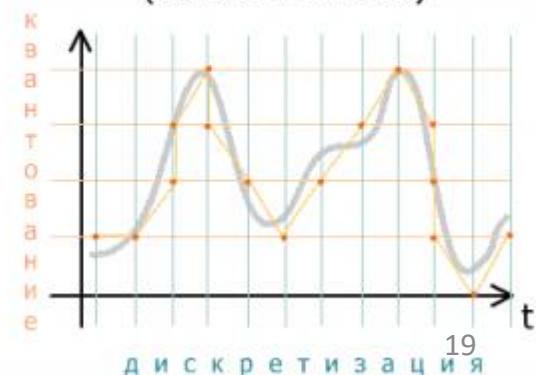
Пример аналогового сигнала



Кодирование сигнала в  
цифровой вид  
(высокое качество)



Кодирование сигнала в  
цифровой вид  
(низкое качество)



# Кодирование звуковой информации

- Звук можно описать в виде совокупности синусоидальных волн определенных частоты и амплитуды. Частота волны определяет высоту звукового тона, амплитуда – громкость звука. Частота измеряется в герцах.
- Диапазон слышимости для человека составляет от 20 до 17000 Гц.
- Физически звук представляет собой волновые колебания давления в той или иной среде. Каковы бы ни были физические характеристики колебаний, в данном случае важно то, что звук представляет собой нечто неделимое на части (непрерывное), пробегающее в пространстве и времени.
- Чтобы записать звук на какой-нибудь носитель можно соотнести его уровень (силу) с какой-нибудь измеряемой характеристикой этого носителя.
- Например, степень намагниченности магнитной ленты в различных ее местах зависит от особенностей звука, который на нее записывался.
- Намагниченность может непрерывно изменяться на протяжении ленты, подобно тому, как параметры звука могут меняться в воздухе. Т.е. магнитная лента прекрасно справляется с задачей хранения звука.
- Хранит его в **аналоговой форме**, когда значения изменяются непрерывно (плавно), что близко к естественному звуку.

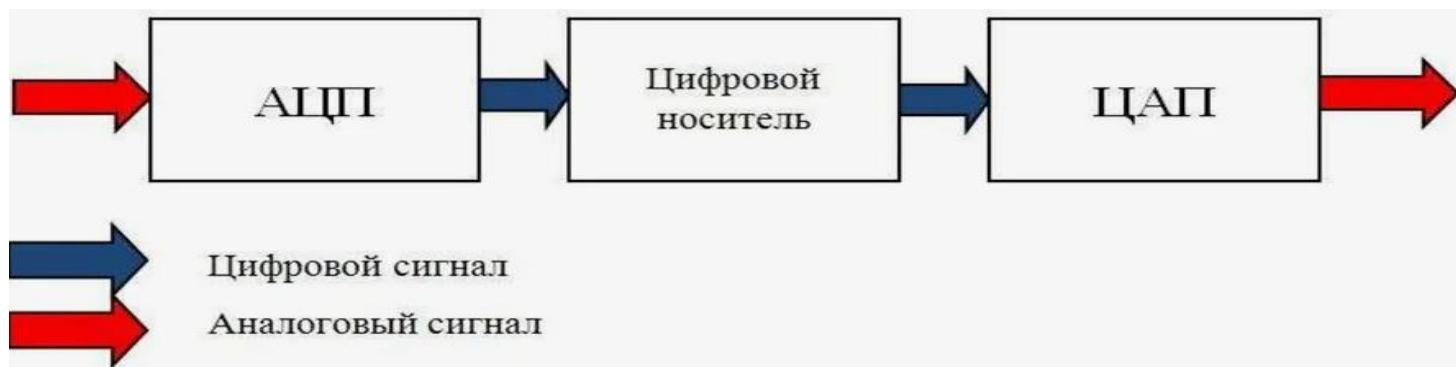
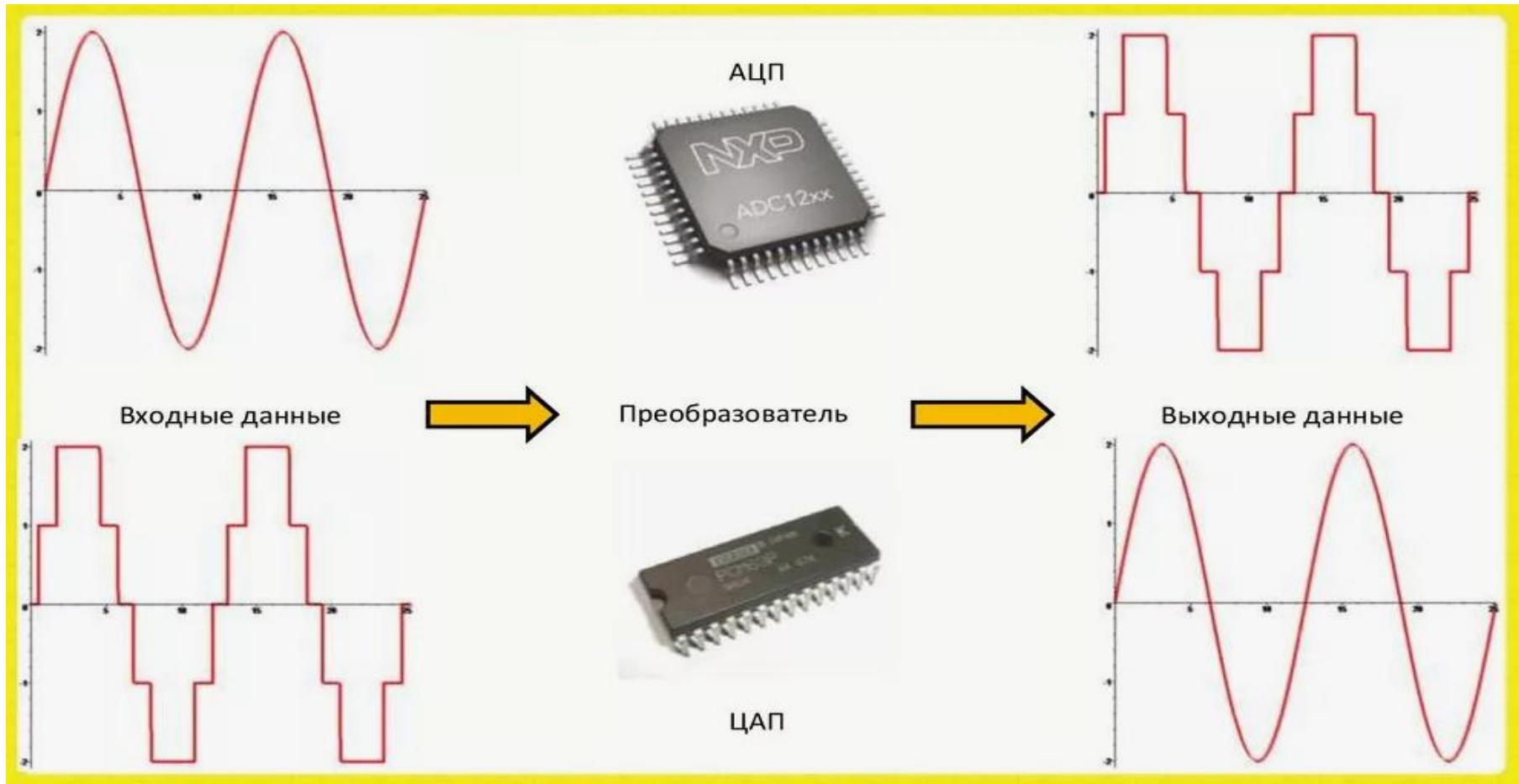
## **Как хранить звук на компьютере?**

Здесь любая информация представлена в **цифровой форме**. Данные должны быть представлены числами, а, следовательно, информация в компьютере дискретна (разделена).

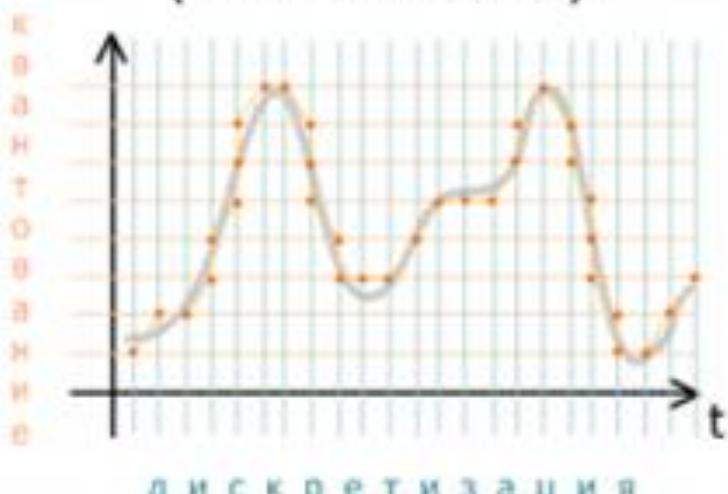
Для того, чтобы записать звук на цифровой носитель информации (например, жесткий диск), его подвергают так называемой оцифровке, механизм которой заключается в измерении параметров звука через определенные промежутки времени (очень малые).

**АЦП** (ADC — «Analog-to-Digital Converter) **аналого-цифровой преобразователь**, преобразует аналоговые сигналы в дискретные (цифровые), поступающие на цифровые устройства для дальнейшей обработки или хранения.

**ЦАП** (DAC — «Digital-to-Analog Converter») **цифро-аналоговый преобразователь**, преобразует выходные цифровые сигналы цифровых устройств в аналоговые.



## Кодирование сигнала в цифровой вид (высокое качество)



## Дискретизация и квантование

При преобразовании звуковой информации в цифровую форму ее подвергают **дискретизации и квантованию**.

**Дискретизация** заключается в замерах величины аналогового сигнала огромное множество раз в секунду.

Полученной величине аналогового сигнала сопоставляется определенное значение из заранее выделенного диапазона: 256 (8 бит) или 65536 (16 бит).

Приведение в соответствие уровня сигнала определенной величине диапазона и есть **квантование**.

**Процесс дискретизации по времени** — процесс получения значений сигнала, который преобразуется, с определенным временным шагом — шагом дискретизации.

- Количество замеров величины сигнала, осуществляемых в одну секунду, называют частотой дискретизации или частотой выборки, или частотой семплирования (от англ. «sampling» — «выборка»).
- Чем меньше шаг дискретизации, тем выше частота дискретизации (то есть, тем чаще регистрируются значения амплитуды), тем более точное представление о сигнале нами будет получено.

Как бы часто мы не проводили измерения, все равно часть информации будет теряться.

**Чем чаще мы проводим замеры, тем точнее будет соответствовать цифровой звук своему аналоговому оригиналу.**

**Чем больше бит отведено под кодирование уровня сигнала (квантование), тем точнее соответствие.**

С другой стороны, звук хорошего качества будет содержать больше данных и, следовательно, больше занимать места на цифровом носителе информации.

### **Пример**

- Для записи качественной музыки аналоговый звуковой сигнал измеряют более 44 000 раз в секунду и квантуют 2 байтами (16 бит дает диапазон из 65536 значений).
- Т.е. за одну секунду записывается 88 000 байт информации. Это равно  $(88\ 000 / 1024)$  примерно 86 Кбайт.
- Минута обойдется уже в 5168 Кбайт  $(86 * 60)$ , что немного больше 5 Мб.

В настоящее время наиболее известный стандарт системы сжатия видеоизображений, формат цифрового видеофайла – **MPEG** (англ. Moving Picture Experts Group — «Экспертная группа по движущимся изображениям»).