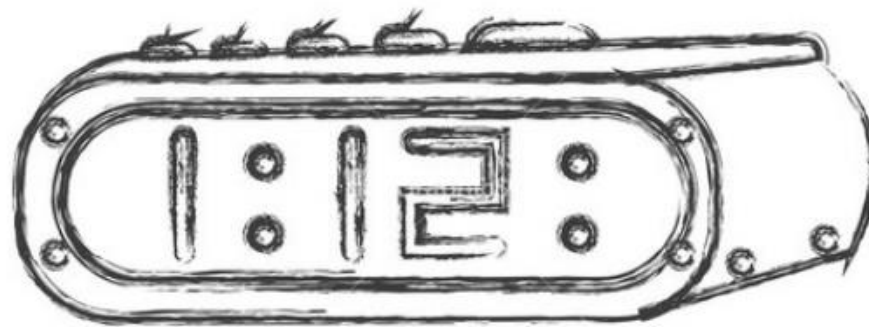


VS.



# **DIGITALNO VS ANALOGNO**



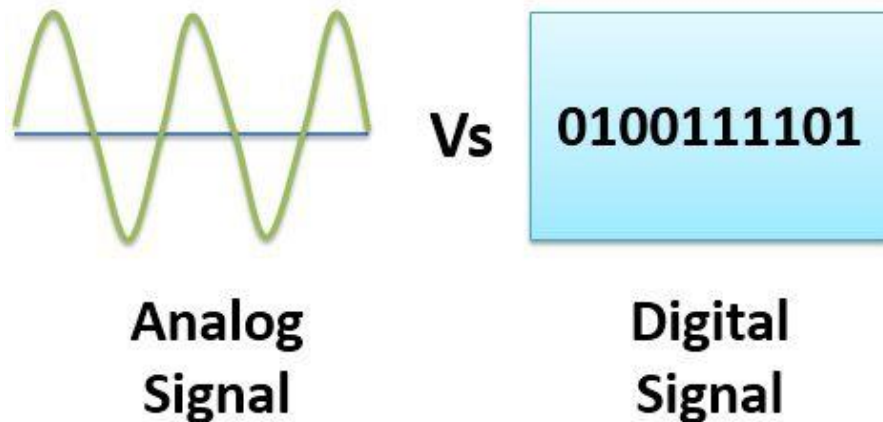
23:59:59



Tlak, temperatura, jačina zvuka, nijanse boja primjeri su **analognih** veličina – riječ je o neprekidnim (**kontinuiranim**) veličinama.

# ANALOGNO VS DIGITALNO

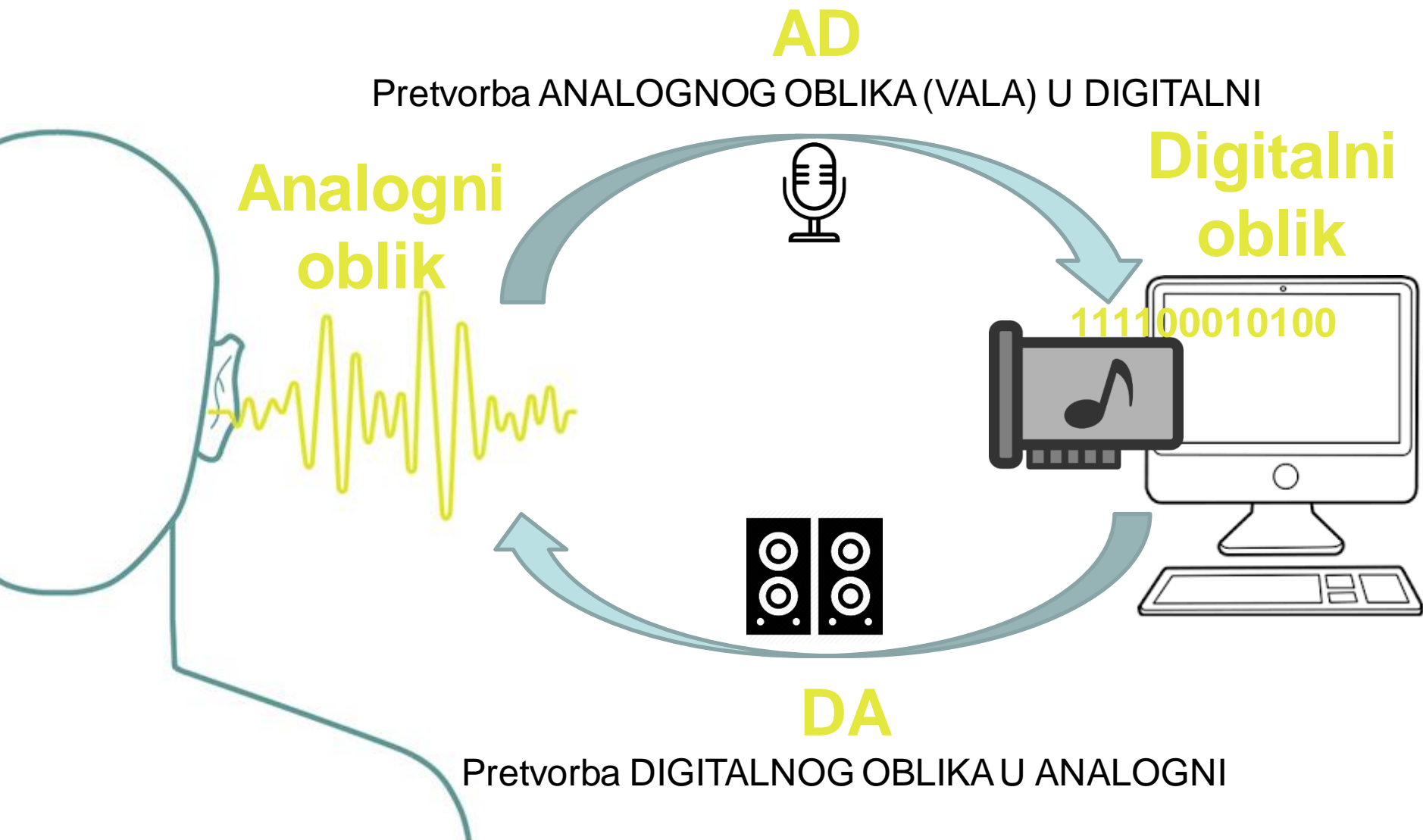
- Za razliku od digitalnoga sustava, u kojem ne postoje međustanja, u analognome sustavu može se postići bilo koja vrijednost između krajnjih vrijednosti.



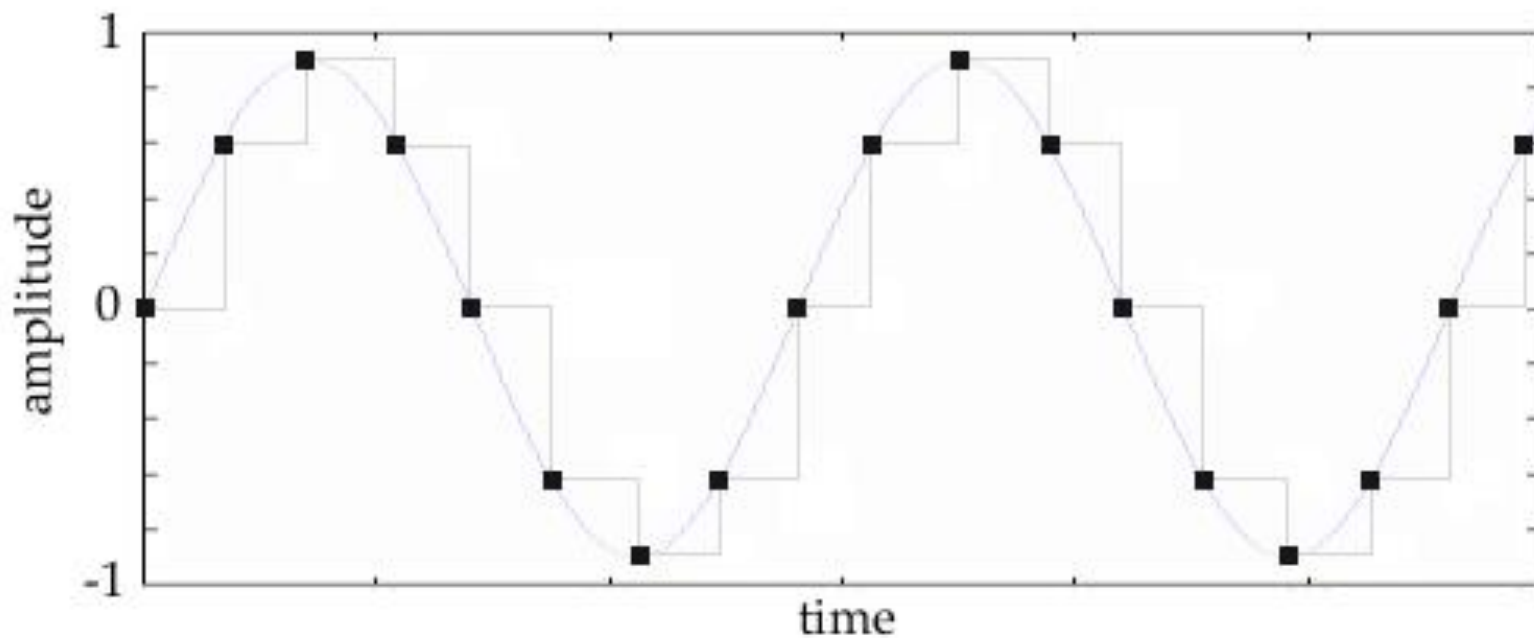


**ZVUK NA  
RAČUNALU?**

# ZVUK NA RAČUNALU?

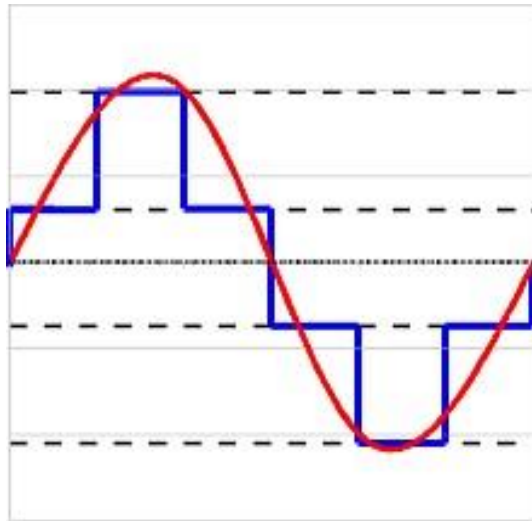


# AD pretvorba zvuka

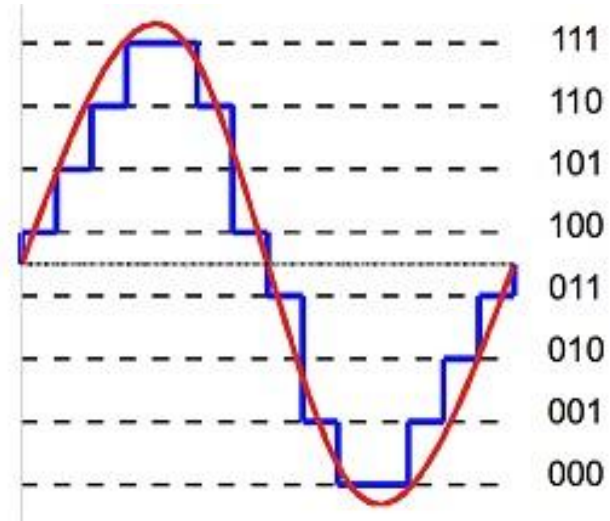


1. uzimanje uzorka zvuka
2. pridruživanje uzorku niza bitova

# Veći broj bitova - veća kvaliteta i preciznost



11  
10  
01  
00



, ali i veća datoteka 😊



# ANALOGNI SIGNAL

- ima puno međustanja, bezbroj stanja.
- analognu informaciju je teško kopirati
- ne postoje dvije jednake kopije analogne informacije
- prilikom svakog kopiranja nastaju neke izmjene
- podložan „šumu”



# DIGITALNI SIGNAL

- ima samo dva osnovna stanja
- dig. informaciju je lako ju je kopirati i prenositi na velike udaljenosti bez gubitka kvalitete
- kopije se tvore bez gubitka kvalitete



# Prednost digitalnog zapisa

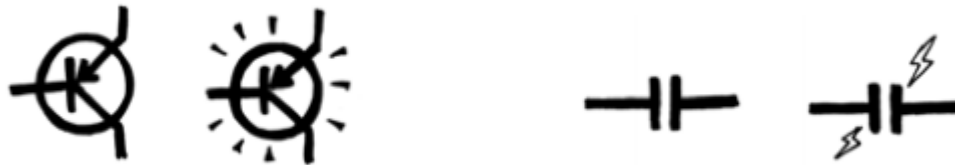
- može **se prenositi na velike udaljenosti i kopirati bez gubitka kvalitete**
- lako kopirati informaciju koja se sastoji od dva stanja, za razliku od umnožavanja analogne informacije temeljene na bezbroj stanja

# ZAPIS PODATAKA NA RAČUNALU



# BIT (engl. binary digit)

- binarna znamenka
- Predstavlja jedno od dva moguća stanja: ima struje-napona (1), nema struje –napona (0).
- razlog tome je što digitalna elektronika počiva na **tranzistoru**, koji ima dva stabilna stanja: ili vodi struju (1) ili ne vodi struju (0).



# Drugi načini postizanja 0 i 1 u računalu

- Kod magnetskih medija npr. **tvrdog diska različiti smjer magnetskog polja** na dijelu diska označava 0 ili 1.



# Drugi način postizanja dva stanja u računalu

- Kod **optičkih uređaja** i medija (CD, DVD) 0 nam može značiti da se laserska zraka nije, a 1 da se jest reflektirala od medija u glavu za čitanje.



# Bit (b), bajt (B), riječ

- u računalu i digitalnim uređajima uvijek imamo neki način da zabilježimo jedno od ta dva stanja.
- Jedan bit sam za sebe ne može predstaviti puno podataka, pa se bitovi grupiraju u bajtove  $1\text{ B} = 8\text{ b}$

# STANJA NIZA BITOVA

Broj bitova:

1

2

3

4

5

6

7

8

0	00	000	0000
1	01	001	0001
	10	010	0010
	11	011	0011
		100	0100
		101	1000
		110	1001
		111	1010
			1100
			1110
			1111

Broj mogućih stanja:

2

4

8

16

32

64

128

256



# BAJT

- BIT – najmanja jedinica za količinu podataka, može imati samo jednu od dvije različite vrijednosti (0 ili 1)
- Niz od 8 bitova zovemo BAJT.
- PRIMJERI BAJTOVA: 00111001, 00110011, 00000000, 11111111
- Bajt je osnovna mjerna jedinica za kapacitet (veličinu) memorije računala.

# Zašto baš 8?

Jer s 8 bitova se može zapisati 256 znakova, a to je u većoj mjeri dovoljno za kodiranje znakova koje čovjek koristi.

# STANJA NIZA BITOVA

Broj bitova:

1

2

3

4

5

6

7

8

0	00	000	0000
1	01	001	0001
	10	010	0010
	11	011	0011
		100	0100
		101	1000
		110	1001
		111	1010
			1100
			1110
			1111

*Zašto baš  
8?*

1 bajt = 1 znak

Broj mogućih stanja:

2

4

8

16

32

64

128

256

# Bajt i njegovih 256 stanja

- Jedan bajt (8 bitova) omogućuje **256** različitih zapisa bitova.
- 8 bitova je dogovorom uzeto da tvori jedan bajt jer je 256 kombinacija dovoljno za predočavanje **(svih)** znakova na tipkovnici: **brojeva** (0-9), **slova** (A-Ž, a-ž) i **ostalih znakova** (\* + - . ! “ # \$ % itd)
- **1 bajt = 1 znak**

*Baš svih?*

# KOD

- Kad pritisnemo neku tipku na tipkovnici računalo odmah prevede taj znak u **dogovoreni** niz 0 i 1 - **kod**.
- Svaki znak na tipkovnici ima pridružen binarni broj - KOD.
- Najrašireniji kod kojim se služe računala je **ASCII** (čitaj: aski) i sastoji se od **8 bitova za svaki znak**.
- 256 znakova – nije dovoljno da se kodiraju svi znakovi (svih zemalja)

# KODIRANJE

**Vježba** – Ako u programu Notepad (Blok za pisanje) napišemo svoje ime i prezime, koliko memorije će zauzimati ta datoteka?

binarni kod	ZNAK		binarni kod	ZNAK
0100 0000	@		0110 0000	`
0100 0001	A		0110 0001	a
0100 0010	B		0110 0010	b
0100 0011	C		0110 0011	c
0100 0100	D		0110 0100	d
0100 0101	E		0110 0101	e
0100 0110	F		0110 0110	f
0100 0111	G		0110 0111	g
0100 1000	H		0110 1000	h
0100 1001	I		0110 1001	i
0100 1010	J		0110 1010	j
0100 1011	K		0110 1011	k
0100 1100	L		0110 1100	l
0100 1101	M		0110 1101	m
0100 1110	N		0110 1110	n
0100 1111	O		0110 1111	o
0101 0000	P		0111 0000	p
0101 0001	Q		0111 0001	q
0101 0010	R		0111 0010	r
0101 0011	S		0111 0011	s
0101 0100	T		0111 0100	t
0101 0101	U		0111 0101	u
0101 0110	V		0111 0110	v
0101 0111	W		0111 0111	w
0101 1000	X		0111 1000	x


# Nova kodiranja

- UTF-8
- trenutno je najpopularniji način kodiranja
- koristi 1-6 bajtova za kodiranje jednog znaka
- Prvih 128 znakova je kompatibilno s ASCII kôdom

```
<html>
<head>
  <title>3.sat - stilovi</title>
  <meta charset="utf-8">
</head>
<body>
  <h1>EKO VODIČ</h1>
  <h2>Koliko je vremena potrebno da se razgradi bačeno smeće?</h2>
  <img src = "3sat.png"><p>
  Staklo - 100godina<br>
  Plastična vrećica - od 100 do 1000 godina <br>
  Limenke - od 10 do 100 godina <br>
  Stiropor - nikada <br>
</p>
<p>
  <strong>Recikliranje</strong> je izdvajanje materijala iz otpada
```

# BAJT (B)

- Kilobajt            1 KB= 1024 B
- Megabajt          1 MB= 1024 KB
- Gigabajt           1 GB= 1024 MB
- Terabajt           1 TB= 1024 GB

- 640 KB = 640\*1024 B
- 1KB = 1024\*8 b
- 1kb = 1000 b 

Zašto 1KB  $\neq$  1000 B  
nego 1KB = 1024B ?

Zato što se veće mjerne  
jedinice tvore umnožkom  
s brojem 2, a ne s 10.