

**Predmet: ELEKTRONIKA U MEHATRONICI**  
**Predmetni nastavnik: Dr Nándor Burány**

1. Semestar specijalističkih studija iz  
Meatronike

**2. GLAVA**  
**KOLA DIGITALNE ELEKTRONIKE**

## Teme

- Kombinaциона kola
- Sekvencijalna kola
- Mešovita kola
- Kola sa softverskim programiranjem
- Kola sa hardverskim programiranjem
- Fizičke osobine digitalnih kola
- Digitalne komunikacije

## Vrste digitalnih mreža

- Skup digitalnih kola za rešavanje **složenog zadatka** se zove **digitalna mreža**.
- Obično se razlikuju dve kategorije mreža, čisto **kombinacione** mreže i **sekvencijalne** mreže.
- Kombinacije ovih mreža možemo zvati **mešovitim** mrežama.

## Osobine kombinacionih mreža

- Kod ovih digitalnih kola **logičko stanje izlaza ( $Q_i$ ) zavisi samo od trenutnih vrednosti signala na ulazu ( $X_i$ )**, prema odgovarajućoj logičkoj funkciji.
- **Izlazni signali ne zavise od redosleda** promena ulaznih signala, od **smera** promene i od logičkih vrednosti u prethodnim intervalima.



- Sekvencijalna kola (kasnije će se proučavati) ispoljavaju drugačije ponašanje.

## Uloga kombinacionih mreža

- Proizvode se **SSI** (small scale of integration) i **MSI** (medium scale of integration) kombinacione mreže skoro u svakoj familiji (TTL, CMOS, BiCMOS - različite tehnologije izrade), od različitih kola. Sa ovim kolima se i danas mogu izgraditi složeni digitalni sistemi ali je taj pristup zastareo.
- **Unutrašnja struktura VLSI** (very large scale of integration) **kola** (mikrokontroleri, PLD) takođe **sadrži ove kombinacione elemente**. Pri njihovom opisivanju koriste se osnovni pojmovi koji se ovde obrađuju.

## Tipične kombinacione mreže

Postoje sledeće **vrste** kombinacionih kola:

1. Kola za sprezanje
2. Logičke kapije
3. Dekoderi
4. Koderi
5. Pretvarači koda
6. Multipleksori
7. Demultipleksori

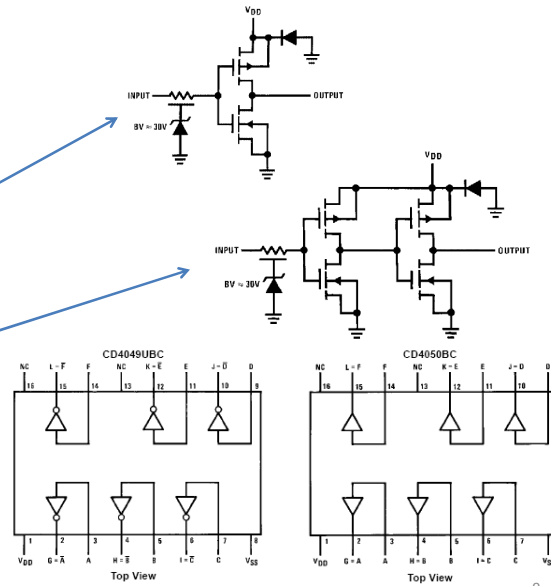
## Kola za sprezanje

Ova kola obavljaju sledeće **zadatke**:

1. **Prilagođenje impedanse** (pojačanje struje)
2. **Prilagođenje logičkog nivoa** (pojačanje ili smanjenje napona)
3. **Invertovanje**
4. **Upotreba zajedničkih vodova.**

## Prilagođenje impedanse

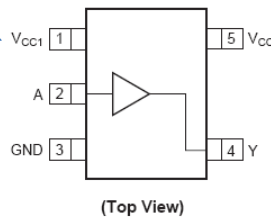
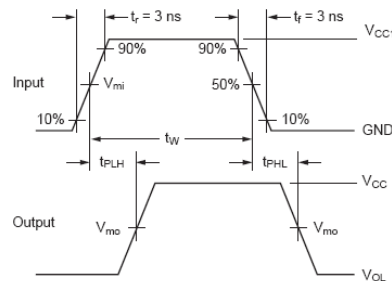
- **Velika ulazna i mala izlazna otpornost (impedansa)**
- Mala ulazna struja, velika opteretljivost izlaza
- **Unutrašnja struktura** invertujućeg kola za sprezanje (logički invertor u CMOS izvedbi)
- **Neinvertujuće kolo** za sprezanje (bafer) (CMOS izvedba)
- **Primer:** **CD4049/4050** -šest invertujućih / neinvertujućih kola za sprezanje u kućištu DIL16 (raspored izvoda)



9

## Prilagođenje nivoa

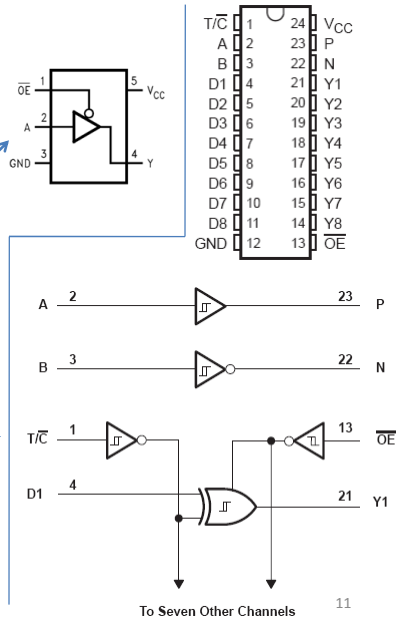
- Prenose se digitalni signali između **dva sistema sa različitim naponima napajanja** ( $V_{CC1}$ ,  $V_{CC}$ ) .
- Usaglase se logički nivoi na pojedinim stranama sa odgovarajućim naponom napajanja.
- **Primer:** Fairchild FXLP34
- $1V < V_{CC}, V_{CC1} < 3,6V$
- $0 < V_{IL} < 0,35V_{CC1}$
- $0,65V_{CC1} < V_{IH} < V_{CC1}$
- $V_{OL} \approx 0V$
- $V_{OH} \approx V_{CC}$



10

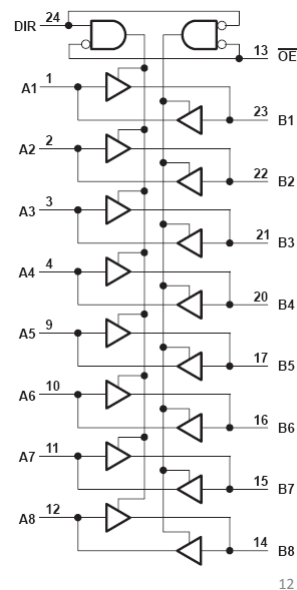
## Kola za sprezanje sa tri stanja

- Signal sa ulaza (A) dolazi na izlaz (Y) samo ako je  $\overline{OE}=0$ .
- Ako je  $\overline{OE}=1$ , izlaz je u trećem stanju (stanje velike impedanse)
- Primer 1.: **NC7SZ125** Tiny Logic UHS Buffer with 3-STATE Output
- Kontrola (OE) može biti aktivna ili za niski ili za visoki logički nivo.
- Po potrebi može da se koristi invertujuće ili neinvertujuće kolo za sprezanje.
- Može se ugraditi histerzis u prenosnu karakteristiku.
- Primer 2.: **SN74LV8151** 10-bit universal Schmitt-trigger buffer with 3-state output
- U zavisnosti od kontrolnog signala (T/C) signali se invertuju ili ne invertuju.



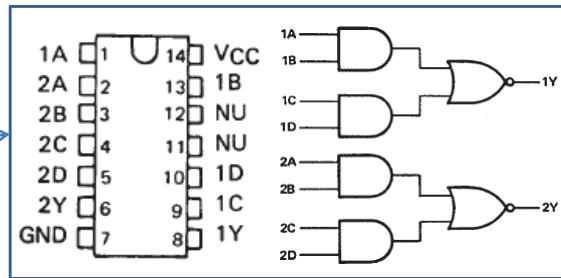
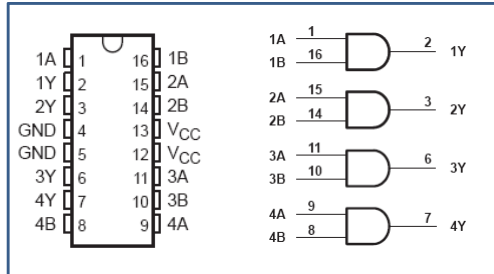
## Dvosmerna kola za sprezanje

- Dve signalne linije se povežu pomoću dva kola za sprezanje.
- U zavisnosti od logičkog nivoa signala DIR, **signali se prenose u jednom ili drugom smeru** ( $A \rightarrow B$  ili  $B \rightarrow A$ ).
- Ako je  $\overline{OE} = 1$ , sa obe strane se dobija velika impedansa (i A i B).
- Primer: **74AC11245** Octal bus transceiver with three state outputs (Texas Instruments)



## Logičke kapije

- Realizuju **proste logičke funkcije**: I, ILI, NI, NILI, isključivo ILI...
- Primer 1.: **74AC11008**, četiri I kola u jednom kućištu.
- Postoje i integrisana kola sa po jednom kapijom u kućištu (**Little Logic-Texas Instruments, Tiny Logic-Fairchild**).
- Mogu se naći i integrisana kola sa kombinacijom različitih vrsta logičkih kapija.
- Primer 2.: **SN74LS51** AND-OR-INVERT gates.



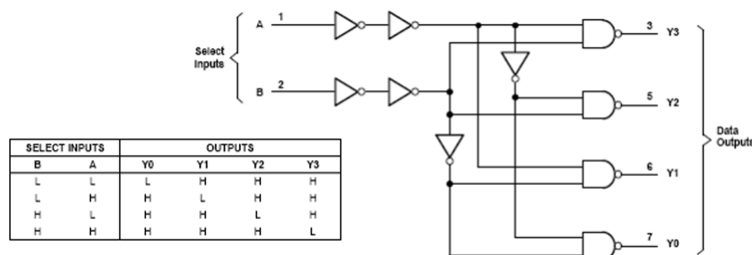
13

## Dekoderi

- Više ulaza ( $n$ ) i više izlaza ( $\leq 2^n$ ).
- Ulazi su **binarno kodirani brojevi**.
- Pri svakom kodu (binarnoj kombinaciji) na ulazu **aktivira se druga linija na izlazu**.

$n$  DEKODER  $2^n$

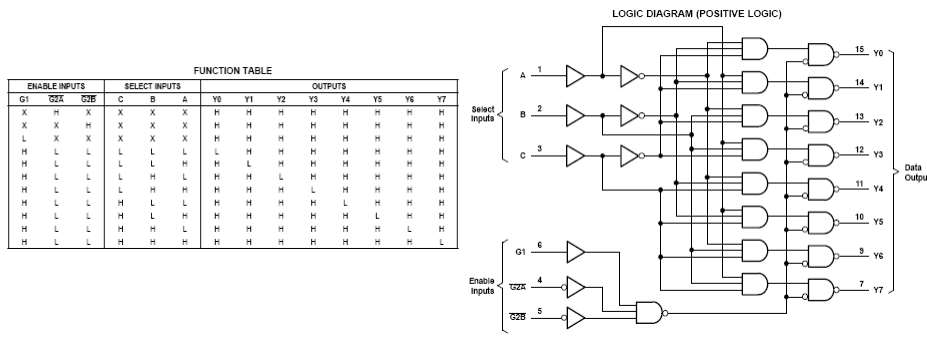
- Primer: **SN74LVC1G139**, običan (potpuni) dekodler 2/4:



14

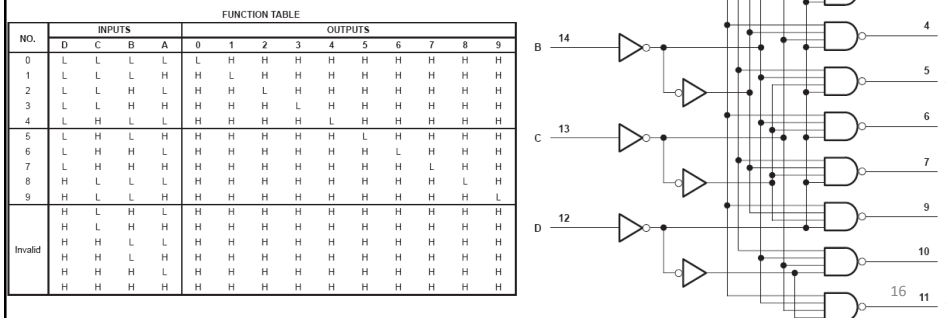
## Potpuni dekodler

- Na svaku varijaciju ulaza aktivira se **jedan ali samo jedan izlaz**.
- Primer: **SN54LVC138A**, potpuni dekodler 3/8.
- Upotrebom ulaza za dozvolu može se proširiti kapacitet ili se može ostvariti demultipleksor.



## Nepotpuni dekodler

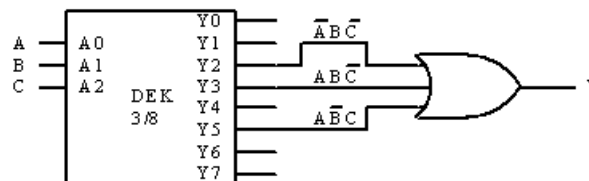
- U slučaju da **nema potrebe za svih mogućih  $2^n$  izlaza...**
- Postoji mogućnost za minimizaciju.
- Najčešći slučaj: nepotpuni dekodler 4/10.
- Primer: **SN74HC42**  
(bez minimizacije)





## Ostvarivanje logičkih funkcija primenom dekodera

- Svaki izlaz dekodera odgovara **jednom logičkom proizvodu** ulaznih promenljivih.
- Sabiranjem odgovarajućih logičkih proizvoda pomoću jednog ILI kola može se realizovati proizvoljna logička funkcija.
- Primer:  $Y = \bar{A}\bar{B}\bar{C} + A\bar{B}\bar{C} + A\bar{B}C$



17

## Koder

- Računari i drugi digitalni uređaji obrađuju **binarno kodirane informacije**.
- Kôd se sastoji od izvesnog broja ( $n$ ) logičkih signala (bit).
- Sa  $n$  bita može se kodirati maksimalno  $2^n$  ulaznih signala.



18

## Potpuni koder

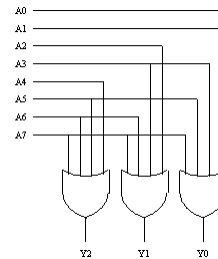
- $2^n$  ulaza, n izlaza
- **Problematična je primena** jer se dobije pogrešan kod kada se istovremeno aktivira više od jednog ulaza.
- Ne proizvodi se zasebno takva komponenta!
- Primer: potpuni koder 8/3.

A7	A6	A5	A4	A3	A2	A1	A0	Y2	Y1	Y0
0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	1	0	0	0	1	0
0	0	0	0	1	0	0	0	0	1	1
0	0	0	1	0	0	0	0	1	0	0
0	0	1	0	0	0	0	0	1	0	1
0	1	0	0	0	0	0	0	1	1	0
1	0	0	0	0	0	0	0	1	1	1

$$Y_0 = A_1 + A_3 + A_5 + A_7$$

$$Y_1 = A_2 + A_3 + A_6 + A_7$$

$$Y_2 = A_4 + A_5 + A_6 + A_7$$



19

## Nepotpuni koder

- $<2^n$  ulaza, n izlaza
- I ovo kolo je **problematično** jer se dobije pogrešan kod kada se istovremeno aktivira više od jednog ulaza.
- Ne proizvodi se ovakvo kolo zasebno!
- Primer: koder 10/4.

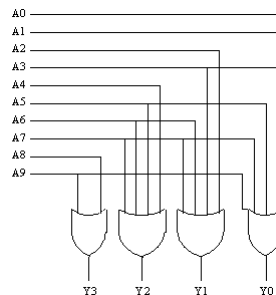
Ai	Y3	Y2	Y1	Y0
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1

$$Y_0 = A_1 + A_3 + A_5 + A_7 + A_9$$

$$Y_1 = A_2 + A_3 + A_6 + A_7$$

$$Y_2 = A_4 + A_5 + A_6 + A_7$$

$$Y_3 = A_8 + A_9$$



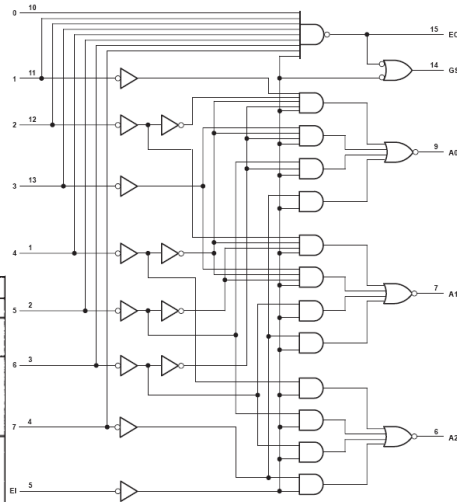
20

## Potpuni prioritetni koder

- Nema problema i ako se **istovremeno aktivira više od jednog ulaza**, uzima se u obzir ulaz sa najvećim prioritonom (sa najvećim rednim brojem).
- Primer: **SN74HC148**, prioritetni koder 8/3.

Input									Output				
EI	D7	D6	D5	D4	D3	D2	D1	D0	GS	O2	Q1	Q0	EO
0	X	X	X	X	X	X	X	X	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	1
1	1	X	X	X	X	X	X	X	1	1	1	1	0
1	0	1	X	X	X	X	X	X	1	1	1	0	0
1	0	0	1	X	X	X	X	X	1	1	0	1	0
1	0	0	0	1	X	X	X	X	1	1	0	0	0
1	0	0	0	0	1	X	X	X	1	0	1	1	0
1	0	0	0	0	0	1	X	X	1	0	1	0	0
1	0	0	0	0	0	0	1	X	1	0	0	1	0
1	0	0	0	0	0	0	0	1	1	0	0	0	0

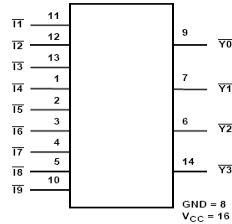
X = Don't Care                      Logic 1 ≡ High                      Logic 0 ≡ Low



21

## Nepotpuni prioritetni koder

- I kod prioriternog kodera može da se namesti broj ulaza manji od  $2^n$ .
- Primer: **SN74HC147**, prioritetni koder 10/4.



INPUTS										OUTPUTS			
I1	I2	I3	I4	I5	I6	I7	I8	I9		Y3	Y2	Y1	Y0
H	H	H	H	H	H	H	H	H		H	H	H	H
X	X	X	X	X	X	X	X	X		L	L	H	H
X	X	X	X	X	X	X	L	H		L	H	H	H
X	X	X	X	X	X	L	H	H		H	L	L	L
X	X	X	X	L	H	H	H	H		H	L	H	L
X	X	X	L	H	H	H	H	H		H	L	H	H
X	X	L	H	H	H	H	H	H		H	H	L	L
X	L	H	H	H	H	H	H	H		H	H	H	L
L	H	H	H	H	H	H	H	H		H	H	H	L

H = High Logic Level, L = Low Logic Level, X = Don't Care

22

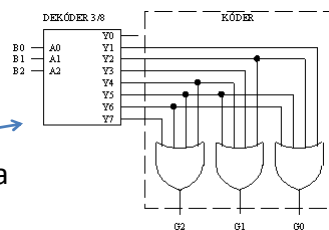
## Pretvarač koda

- Pretvara kod **iz jednog kodnog sistema u drugi.**
- **Standardno rešenje:** kaskadna veza dekodera i koda.
- Redovno postoji **prostije rešenje:**
  1. hardver dobijen minimizacijom logičkih funkcija
  2. softverska metoda, iščitavanje iz tabele.

23

## Pretvarač prirodnog binarnog koda u Gray-ov kod

- **Kaskadnom vezom dekodera sa koderom** dobija se sledeće rešenje:



- Logičke jednačine na bazi tabele su:

$$G_2 = B_2\bar{B}_1\bar{B}_0 + B_2\bar{B}_1B_0 + B_2B_1\bar{B}_0 + B_2B_1B_0$$

$$G_1 = \bar{B}_2B_1\bar{B}_0 + \bar{B}_2B_1B_0 + B_2\bar{B}_1\bar{B}_0 + B_2\bar{B}_1B_0$$

$$G_0 = \bar{B}_2\bar{B}_1B_0 + \bar{B}_2B_1\bar{B}_0 + B_2\bar{B}_1B_0 + B_2B_1\bar{B}_0$$

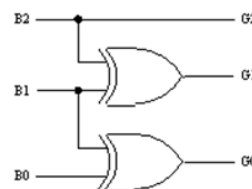
- **Minimizacijom** logičkih funkcija dobijaju se prostiji izrazi:

$$G_2 = B_2$$

$$G_1 = B_2 \oplus B_1$$

$$G_0 = B_1 \oplus B_0$$

- Prostija mreža konstruisana na bazi minimiziranih logičkih funkcija:

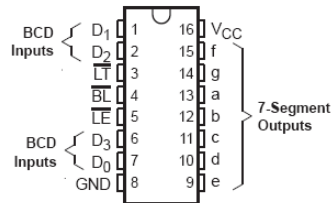


Prirodni binarni kod B2B1B0	Gray-ov kod G2G1G0
000	000
001	001
010	011
011	010
100	110
101	111
110	101
111	100

24

## Pretvarač koda BCD/7 segmenata

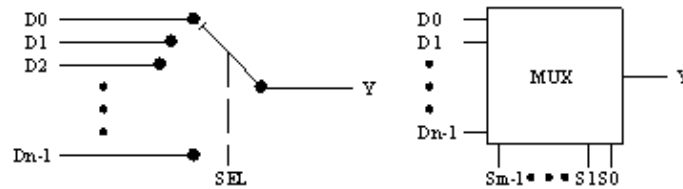
- Sedmo segmentni indikator služi za **indikaciju cifara** decimalnog brojnog sistema.
- Primer: **SN74HCT4511** BCD-to-7 segment latch/decoder/driver.
- Tabela (postoje i kontrolni ulazi).
- Može se minimizirati.
- Raspored nožica.



INPUTS						OUTPUTS								
LE	BL	LT	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>	a	b	c	d	e	f	g	DISPLAY
X	X	L	X	X	X	X	H	H	H	H	H	H	H	8
X	L	H	X	X	X	X	L	L	L	L	L	L	L	Blank
L	H	H	L	L	L	L	H	H	H	H	H	L	L	0
L	H	H	L	L	L	H	L	H	H	L	L	L	L	1
L	H	H	L	L	H	L	H	H	L	H	H	L	L	2
L	H	H	L	L	H	H	H	H	H	H	L	L	H	3
L	H	H	L	H	L	L	L	H	H	L	L	H	H	4
L	H	H	L	H	L	H	H	L	H	H	L	H	H	5
L	H	H	L	H	H	L	L	L	H	H	H	H	H	6
L	H	H	L	H	H	H	H	H	H	L	L	L	L	7
L	H	H	H	L	L	L	H	H	H	H	H	H	H	8
L	H	H	H	L	L	H	H	H	L	L	L	L	H	9
L	H	H	H	L	H	L	L	L	L	L	L	L	L	Blank
L	H	H	H	L	H	H	L	L	L	L	L	L	L	Blank
L	H	H	H	H	L	L	L	L	L	L	L	L	L	Blank
L	H	H	H	H	L	H	L	L	L	L	L	L	L	Blank
L	H	H	H	H	H	H	L	L	L	L	L	L	L	Blank
L	H	H	H	H	H	H	L	L	L	L	L	L	L	Blank
H	H	H	X	X	X	X	t	t	t	t	t	t	t	t

## Multipleksor

- Prosleđivanje digitalnih signala (signalni ulazi,  $D_0, D_1, \dots, D_{n-1}$ ) sa više ulaznih linija na jednu izlaznu liniju (Y).
- Radi kao jedan jednopolni više položajni prekidač.
- Podrazumeva se da istovremeno može da prenosi samo jedan signal - vrši **vremensko multipleksiranje**.
- Izbor signala koji se prenosi na izlaz u datom momentu se vrši pomoću selekcionih ulaza ( $S_0, S_1, S_{m-1}$ ).
- Redovno važi  $n=2^m$ .



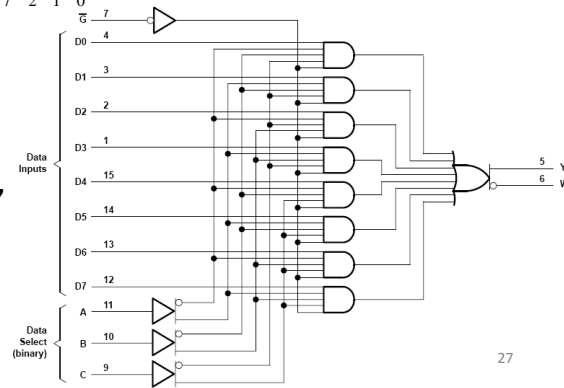
### Konstrukcija digitalnog multipleksora

- Kombinačna tabela (CD74AC151):
- Oblik logičke funkcije za prost multipleksor 8/1:

INPUTS				OUTPUTS	
SELECT			STROBE $\bar{G}$	Y	W
C	B	A		L	H
X	X	X	H	L	H
L	L	L	L	D0	$\overline{D0}$
L	L	H	L	D1	$\overline{D1}$
L	H	L	L	D2	$\overline{D2}$
L	H	H	L	D3	$\overline{D3}$
H	L	L	L	D4	$\overline{D4}$
H	L	H	L	D5	$\overline{D5}$
H	H	L	L	D6	$\overline{D6}$
H	H	H	L	D7	$\overline{D7}$

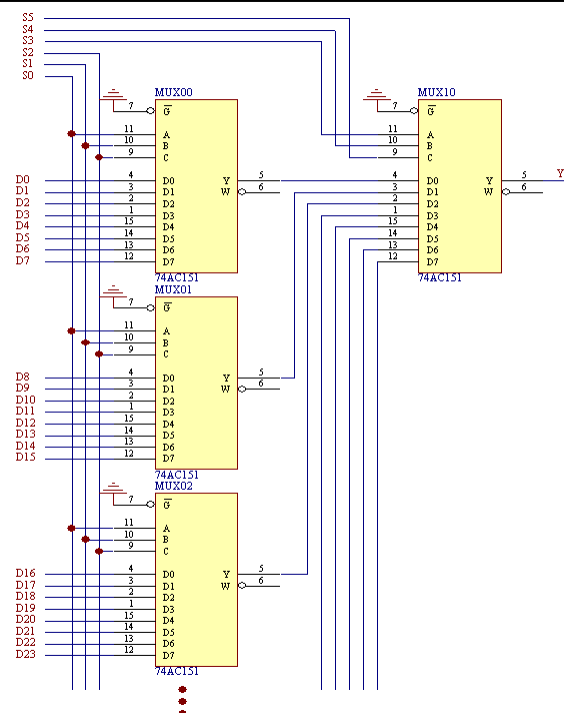
$$Y = D_0\bar{S}_2\bar{S}_1\bar{S}_0 + D_1\bar{S}_2\bar{S}_1S_0 + \dots + D_7S_2S_1S_0$$

- U tabeli se navodi i jedan kontrolni ulaz (STROBE), koji, bez obzira na ostale ulaze, dovodi nulu na izlaz.
- Formira se i invertovani izlaz.



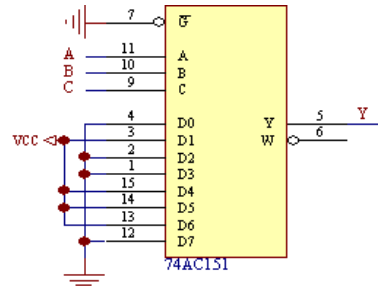
### Proširivanje multipleksora

- Ne proizvode se multipleksori sa više od 16 ulaza.
- Multipleksiranje većeg broja signala se može obaviti sprežanjem potrebnog broja multipleksora.
- Primer: multipleksor sa  $8 \times 8 = 64$  ulaznih kanala.



## Ostvarivanje logičkih funkcija primenom multipleksora

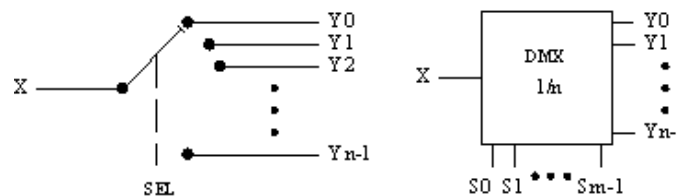
- **Logičke promenljive povezujemo na selekzione ulaze** multipleksora.
- **Na ulaze za podatke** se povezuju one logičke vrednosti koje važe pri datim varijacijama logičkih promenljivih.
- Primer:  $Y = \overline{C}BA + C\overline{B} + CBA$
- Funkcija se mora svesti na normalnu formu:  $Y = \overline{C}BA + C\overline{B}A + C\overline{B}\overline{A} + CBA$
- Postoji efikasnija metoda (mogu se ostvariti i funkcije četiri promenljive sa istim multipleksorom!



29

## Demultipleksor

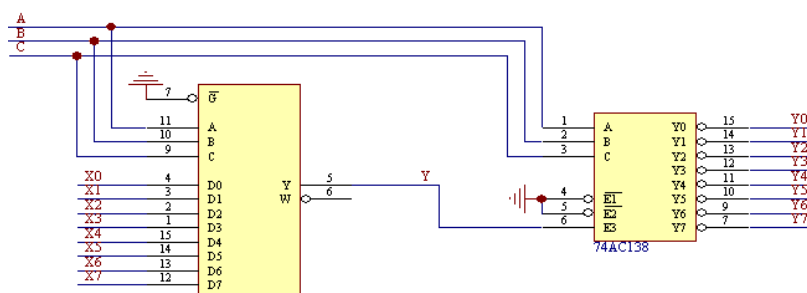
- Prosleđuje **jedan ulazni signal** ( $X$ ) na **više izlaznih linija** ( $Y_0, Y_1, \dots, Y_{n-1}$ ).
- Radi kao jednopolni, više položajni prekidač.
- Istovremeno se signal može prosleđivati samo prema jednom od izlaza - smatra se da su na ulazu **vremenski multipleksirani signali**.
- **Izbor signala** koji će se u datom momentu preneti na izlaz određen je selekcionim ulazima ( $S_0, S_1, S_{m-1}$ ).
- Redovno je  $n=2^m$ .
- U katalozima redovno se ista komponenta nudi i kao dekodeer i kao demultipleksor.



30

## Prenos više signala kroz zajednički kanal – vremenski multipleks

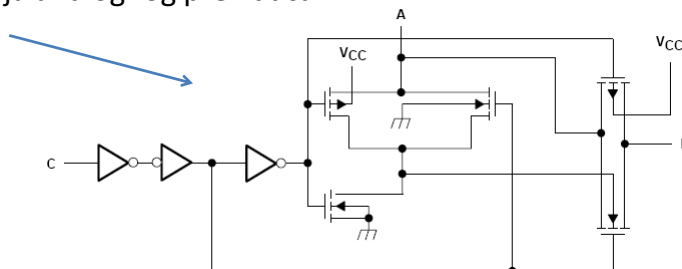
- Kaskadnom vezom multipleksora i demultipleksora može se preneti **više signala kroz zajednički kanal**, po sistemu vremenskog multipleksa.
- Pored signala koji se prenose treba povezati i **selektivne signale** da bi navedeni multipleksor i demultipleksor radili sinhrono.
- U datom primeru, umesto osam signalnih linija se koristi samo četiri. U opštem slučaju potrebno je  $m+1(+1)$  linija, gde je  $n=2^m$  broj signala koje treba preneti.



31

## Analogni multipleksor/demultipleksor – analogni prekidač

- Može da **prenese analogne signale** (naravno, i digitalne).
- Isti element **prenosi signal u oba smera**.
- Sadrži analogne prekidače i dekodeer.
- Konstrukcija analognog prekidača:



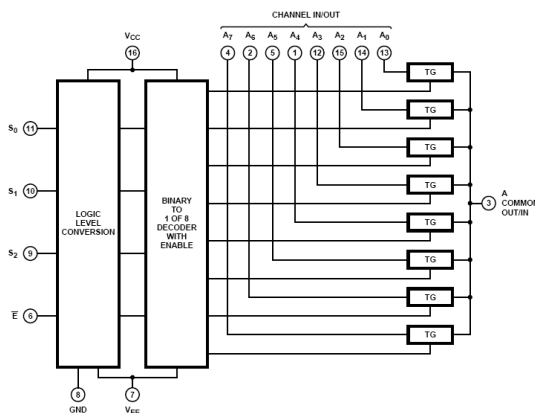
- Povezuje (električno) tačke A i B (mala otpornost).
- C - kontrolni ulaz, C=1 - prekidač provodi.

32



## Konstrukcija analognog multipleksora/demultipleksora

- Primer: **CD74HC4051**, osmo kanalni analogni multipleksor-demultipleksor
- Analognim prekidačima upravlja dekodler 3/8
- Selekcioni ulazi (S0, S1, S2): 0V i 5V.
- Opseg analognog napona: -5V...+5V. ( $V_{CC}=5V$ ,  $V_{EE}=-5V$ ).



33

## Sekvencijalne mreže

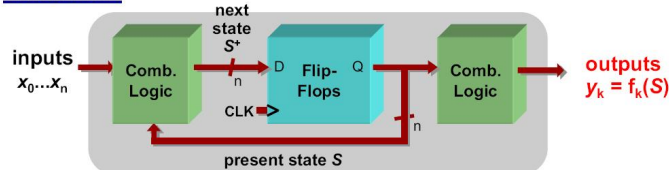
- U pitanju su digitalna **kola sa memorijom** (sposobnost čuvanja informacija)
- Izlazi kola u datom momentu **zavise od trenutnih ulaza, ali zavise i od događaja u prethodnom intervalu** (ulazi u prethodnom intervalu određuju skladištenu informaciju).
- **Logički automati**: tako se zovu pojedine sekvencijalne mreže jer se primenjuju za automatsko upravljanje.
- Engleski naziv: (finite) **state machine**.
- **Memorija** u sekvencijalnim kolima je važna ali je redovno **malog kapaciteta** - svega nekoliko bita, pošto se sa n bita može kodirati  $2^n$  (puno) stanja, prema tome može se realizovati mreža sa složenim ponašanjem.

34

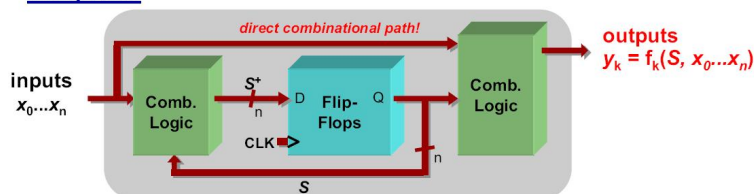
## Struktura sekvencijalnih mreža

Pri konstrukciji sekvencijalnih mreža primenjuju se dve strukture:

### Moore FSM:



### Mealy FSM:



Moore-ova mreža je redovno prostija, dok je Mealy-jeva mreža obično brža.

35

## Takt signal u sekvencijalnim mrežama

- **Moment promene stanja** se redovno sinhroniše sa takt signalom (clock).
- Sinhronizacioni signal je povezan na memorijske elemente.
- Sinhronizacija **nije obavezna ali** dobar deo savremenih digitalnih uređaja koristi sinhronizovane mreže.
- Zahvaljujući sinhronizaciji **eliminisu se mnogi hazardi**.
- Rad sinhronizovanih mreža je mnogo lakše pratiti/sagledati.
- Mealy-jev automat je više podložan hazardu jer su ulazi direktno povezani na ulaze izlaznog kombinacionog kola.

36

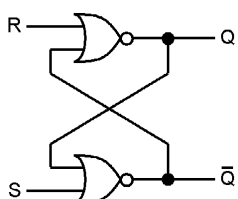
## Elementarne memorije

- Potrebne su za **konstrukciju sekvencijalnih mreža**.
- Oni sami su elementarna sekvencijalna kola bez sinhronizacije.
- Informacija se čuva na bazi **pozitivne povratne sprege** - čuvanje traje dok ima napajanja.
- **Upis nove informacije** se može vršiti pri odgovarajućem **nivou** ili odgovarajućoj **ivici** sinhronizacionog signala.

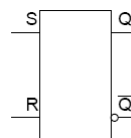
37

## Latch-evi - SR latch

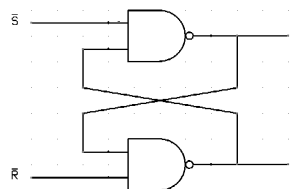
- Elementarne memorije koje **reaguju na logički nivo**.
- SR latch **sa NILI kolima**.



S	R	Q	$\bar{Q}$
0	0	latch	latch
0	1	0	1
1	0	1	0
1	1	0	0

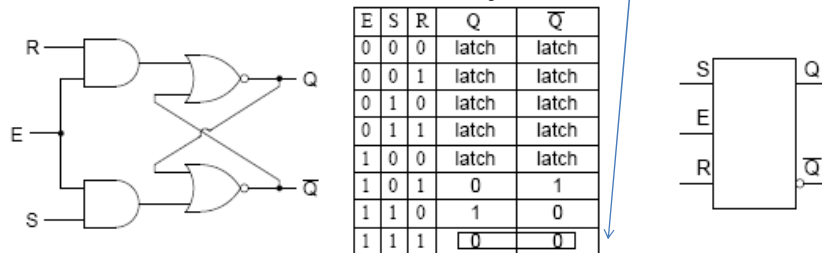


- Nakon prestanka delovanja dve jedinice na ulazu ne može se znati koje stanje će se formirati na izlazu - zato je zabranjeno primeniti takvo upravljanje.
- Slično rešenje **sa NI kolima**, u ovom slučaju upravljanje se vrši sa logičkom nulom.
- U ovom slučaju do neodređenog ponašanja dolazi ako na ulaz dovedemo istovremeno dve nule.



## Sinhronizacija SR latch-a

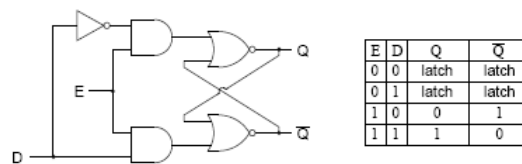
- Signal **E (enable - dozvola)** određuje kada će doći do promene stanja u latch-u.
- Pripreme se logički nivoi na SR ulazima, zatim se dovede signal za sinhronizaciju (E).
- Ako se pri E=1 promene SR ulazi, izlazi će reagovati na te promene (**transparentni latch**).
- Istovremena pojava dve logičke jedinice na SR ulazima dovodi do **nedefinisane situacije**.



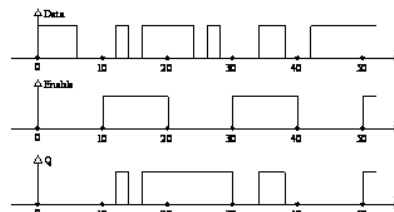
39

## D latch

- **Nedefinisana situacija** kod SR latch-a se može razrešiti modifikacijom koja je prikazana na slici.
- U stvarnosti novo kolo ima samo **jedan ulaz** za podatke (D).



- I D latch ima **transparentno** ponašanje.



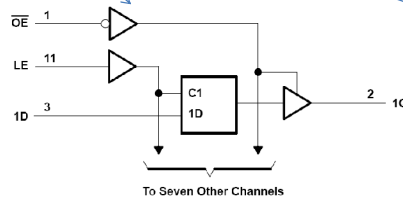
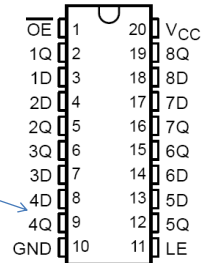
40

## Primer D latch-a

- **SN74AHCT373** - OCTAL TRANSPARENT D-TYPE LATCHES WITH 3-STATE OUTPUTS

- Tablica prelaza, raspored izvoda i logički dijagram.

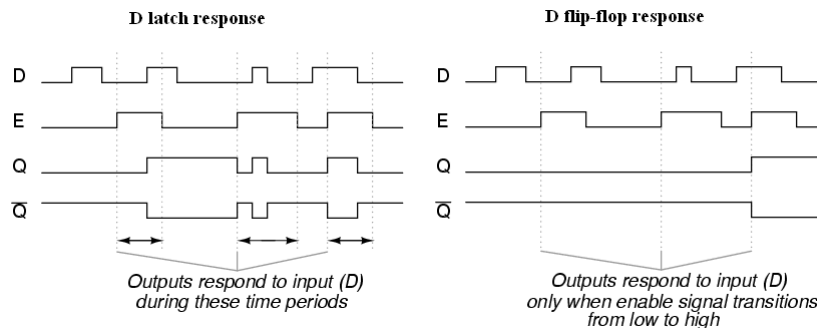
INPUTS			OUTPUT
$\overline{OE}$	LE	D	Q
L	H	H	H
L	H	L	L
L	L	X	$Q_0$
H	X	X	Z



41

## Flip-flop-ovi

- I flip-flop-ovi su **elementarne memorije**.
- Umesto upravljanja nivoom **upravlja se ivicom**.
- Promena stanja (upis podatka) se vrši pri uzlaznoj (pozitivnoj) ili silaznoj (negativnoj) ivici .
- **Razlika** između D latch-a i D flip-flop-a:

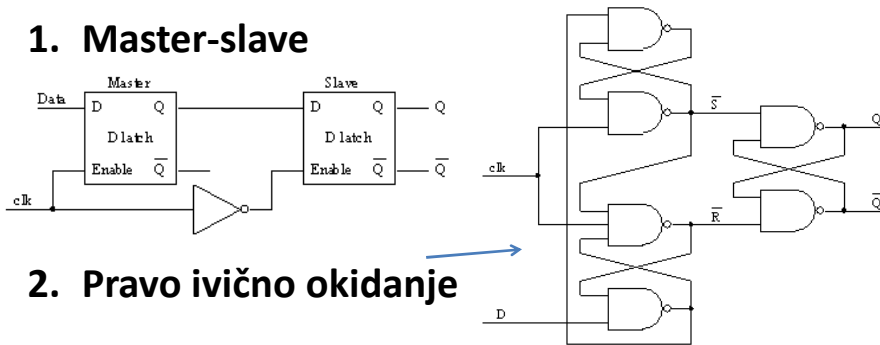


42

## Realizacija okidanja na ivicu

- Moguće realizacije D flip-flop-a:

### 1. Master-slave



### 2. Pravo ivično okidanje

- Tablica prelaza je ista u oba slučaja

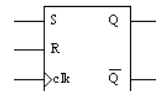
D	Q <sub>n</sub>	Q <sub>n+1</sub>
0	0	0
0	1	0
1	0	1
1	1	1

43

## Drugi tipovi flip-flop-ova

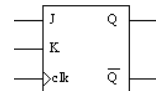
- Razlike u tablicama prelaza.

- SR flip-flop



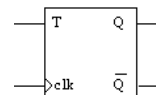
S	R	Q <sub>n</sub>	Q <sub>n+1</sub>
0	0	Q	Q
0	1	Q	0
1	0	Q	1
1	1	Q	mem def.

- JK flip-flop



J	K	Q <sub>n</sub>	Q <sub>n+1</sub>
0	0	Q	Q
0	1	Q	0
1	0	Q	1
1	1	Q	$\bar{Q}$

- T flip-flop



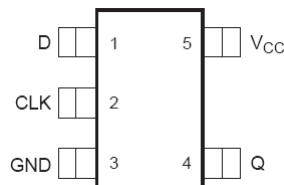
T	Q <sub>n</sub>	Q <sub>n+1</sub>
0	Q	Q
1	Q	$\bar{Q}$

**Bilo koja sekvencijalna mreža se može realizovati sa bilo kojim tipom flip-flop-a.** Danas se uglavnom primenjuju D flip-flop-ovi, pogotovu u VLSI tehnici. Razlog za šarenilo u ranijim godinama je što se neki tipovi sekvencijalnih mreža lakše realizuju sa određenim tipom flip-flop-a.

44

## Primer flip-flop-a

- **SN74AUP1G79** LOW-POWER SINGLE POSITIVE-EDGE-TRIGGERED D-TYPE FLIP-FLOP

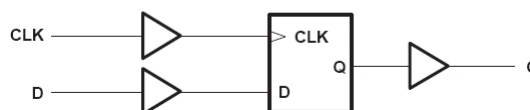


- Kod neki integrisanih kola izlazi su sa tri stanja ili invertovani.
- Za realizaciju trećeg stanja na potreban je odgovarajući kontrolni ulaz.

FUNCTION TABLE

INPUTS		OUTPUT Q
CLK	D	
↑	H	H
↑	L	L
L or H	X	Q <sub>0</sub>

LOGIC DIAGRAM (POSITIVE LOGIC)



45

## Opisivanje i konstruisanje logičkih automata

Obično se misli na neki zadatak iz automatizacije ali istim postupcima se konstruišu i bilo koja druga sekvencijalna kola.

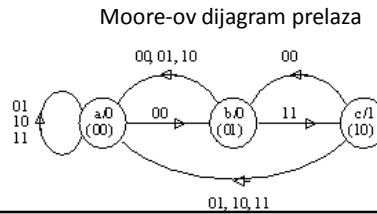
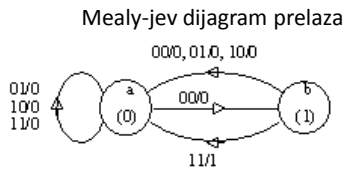
Postupak:

1. Polazi se od **usmenog opisa**.
2. Konstruiše se **dijagram prelaza** ili njemu ekvivalentna **tabela prelaza**.
3. Iz tabele se dobijaju **logičke jednačine** na bazi kojih se konstruišu ulazna i izlazna kombinaciona mreža.
4. Kombinaciona kola se **povezuju sa potrebnim brojem i tipom flip-flop-ova**.
5. Automat je **spreman za rad!**

46

## Dijagram prelaza

- Precizno rešenje za opis sekvencijalne mreže (logičkog automata) .
- **Zadatak:**  
*Posmatraju se ulazne linije A i B. Ako se prvo pojave istovremeno nule na tim linijama, zatim istovremeno jedinice, izlaz treba dići na visoki logički nivo u trajanju od jednog takt intervala (ili do sledeće uzlazne ivice takta). U svim drugim situacijama izlaz treba da bude na niskom logičkom nivou.*
- **Označavanje kod Mealy-jevog automata:** u krugu je oznaka stanja, u zagradi je kôd stanja, pored strelice je ulazna kombinacija/izlazna kombinacija.
- **Označavanje kod Moore-ovog automata:** u krugu je oznaka stanja/izlaz, u zagradi je kôd stanja, pored strelice je ulazna kombinacija.



47

## Tabela prelaza

- Tabela prelaza sadrži istu informaciju kao **dijagram prelaza** (dijagram stanja) ali je pogodnija za pisanje logičkih jednačina kola.
- Treba da se navedu u tabeli **sva stanja automata i sve varijacije ulaznih promenljivih.**
- Za svaku situaciju treba navesti **naredno stanje i vrednosti izlaza.**

Tabela prelaza za Mealy-jev automat

Trenutno stanje	Sledeće stanje				Izlaz Y			
	BA= 00	01	10	11	BA= 00	01	10	11
a (0)	b	a	a	a	0	0	0	0
b (1)	a	a	a	a	0	0	0	1

Tabela prelaza za Moore-ov automat

Trenutno stanje	Sledeće stanje				Izlaz Y
	BA= 00	01	10	11	
a (00)	b	a	a	a	0
b (01)	a	a	a	c	0
c (10)	b	a	a	a	1

48



## Kodiranje stanja

- Sa  $n$  komada flip-flop-ova može se kodirati  $\leq 2^n$  stanja.
- **Ranije** (pri tradicionalnom projektovanju) cilj je je bio **minimalan broj flip-flopova**.
- **Danas** (projektovanje pomoću PLD-a) naglasak nije na minimizaciji broja flip-flopova, **često se koristi princip jedno stanje - jedan flip-flop**.
- Nije sve jedno **kako usvajamo kodove** za pojedina stanja (izbor utiče na nivo složenosti kombinacionih kola) ali, na žalost, ne postoji sistematski postupak za optimizaciju.
- Možemo koristiti **flip-flop-ove bilo kog tipa** (D, SR, JK, T), na žalost, ne može se znati unapred koje rešenje će biti prostije.

49

## Jednačine za upravljanje flip-flop-ovima

- Iz tablice prelaza flip-flop-a određenog tipa može se znati **šta treba dovesti na ulaze** da bi izlaz reagovao na željeni način.
- Zadatak projektanta je da konstruiše takvo kombinaciono kolo koje će na potreban način upravljati flip-flop-ovima.
- U nekim slučajevima upravljanje znači dovođenje **konkretnih logičkih nivoa** (kod D flip-flop-a je uvek tako), ali ima slučajeva kad pojedini ulazi flip-flopa mogu biti **proizvoljni** (na pr. SR flip-flop će dati logičku nulu ako je i dosad bio u tom stanju bilo da ga resetujemo bilo da ne, znači R ima proizvoljnu vrednost, važno je samo da ne bude  $S=1$ ).

50

## Jednačina za upravljanje flip-flop-ovima u primeru

- Za ranije uvedeni logički automat, u slučaju ostvarivanja po **Mealy**-ju, važi sedeća **tabela i jednačina**:

$Q_n$	BA	$Q_{n+1}=D$	Y
0	00	1	0
	01	0	0
	10	0	0
	11	0	0
1	00	0	0
	01	0	0
	10	0	0
	11	0	1

$$D = \overline{Q}BA$$

- Tabela prelaza i jednačina** pri ostvarivanju logičkog automata po **Moore**-u:

$Q_1 Q_0$	BA	$Q_1 Q_{n+1}=D1, Q_0 Q_{n+1}=D0$	Y
00	00	01	0
	01	00	0
	10	00	0
	11	00	0
01	00	00	0
	01	00	0
	10	00	0
	11	10	0
10	00	01	1
	01	00	1
	10	00	1
	11	00	1
11	00	00	0
	01	00	0
	10	00	0
	11	00	0

$$D1 = \overline{Q}1Q0BA$$

$$D0 = \overline{Q}1\overline{Q}0BA + Q1\overline{Q}0BA = \overline{Q}0BA$$

51

## Formiranje izlaza

- Kod **Mealy**-jevog automata **izlazna kombinaciona mreža** formira izlazne logičke nivoe na bazi trenutnih ulaza i trenutnog stanja. U konkretnom slučaju dobija se:

$$Y = QBA$$

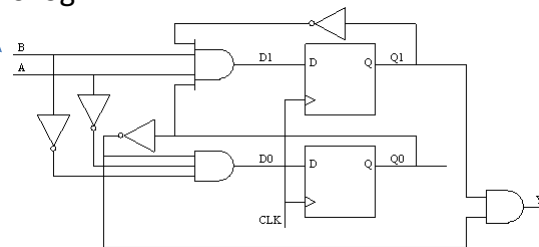
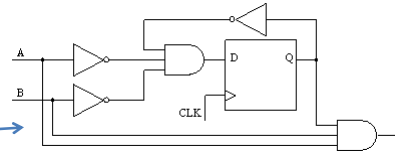
- Kod **Moore**-ovog automata izlazi zavise samo od trenutnog stanja, to stanje treba dovesti na ulaze **izlaznog kombinacionog kola**. U konkretnom slučaju dobija se:

$$Y = Q1\overline{Q}0$$

52

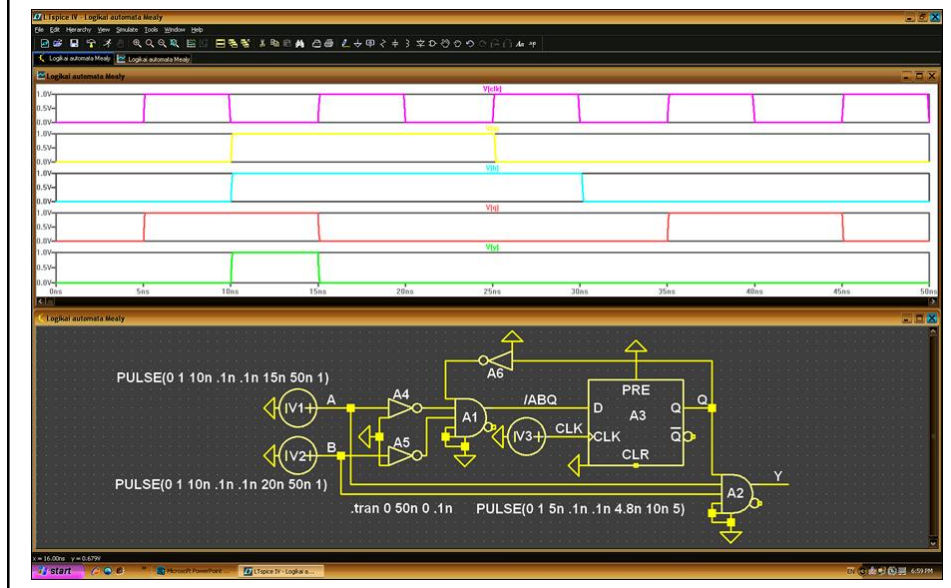
## Konstrukcija kompletnog automata

- Ostvarena sekvencijalna mreža (automat) sadrži ulaznu kombinacionu mrežu, potreban broj i tip flip-flop-ova i izlaznu kombinacionu mrežu.
- Logička šema **Mealy**-jevog automata.
- Logička šema **Moore**-ovog automata.

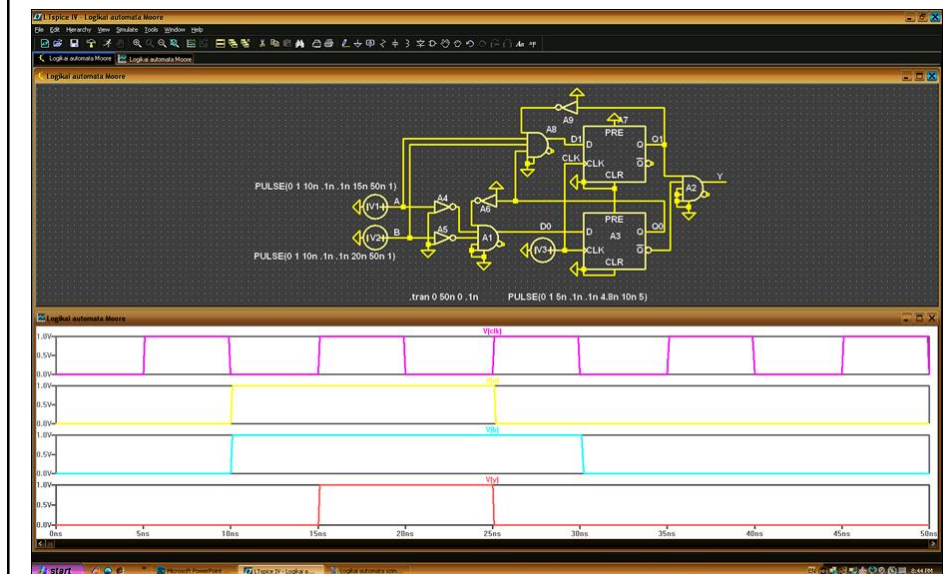


53

## Simulacija rada ostvarenog automata Mealy-jevog tipa



## Simulacija rada ostvarenog automata Moore-ovog tipa

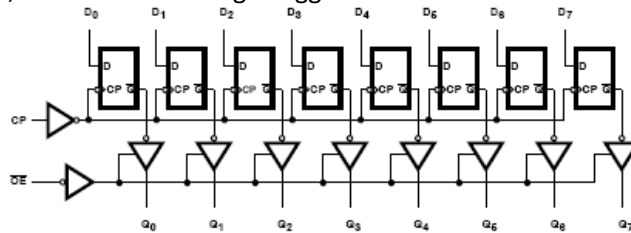
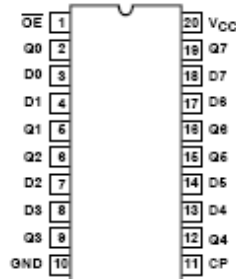


## Registri

- Služe za **čuvanje** (skladištenje) **male količine informacije** (nekoliko bita).
- Struktura: **latch-evi ili flip-flop-ovi** poređani u jednom kućištu, sa zajedničkim upravljačkim vodovima.
- **Tipovi:**
  1. Obični (stacionarni) registri
  2. Pomerački (shift) registri
  3. Kružni registri (kružni brojači)

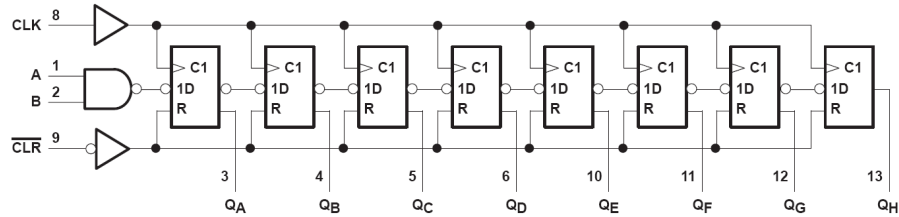
## Obični (stacionarni) registri

- Redovno **zajednički takt** (CLK) ili zajednički signal dozvole (LE).
- **Paralelni upis i čitanje** (svi bitovi istovremeno).
- Izlazi mogu biti sa tri stanja ili invertovani (kod nekih tipova).
- Broj bitova kod MSI kola je od 2 do 32. Sprezanjem više integrisanih kola se može dalje proširivati.
- Primer: **CD54HC374** High-Speed CMOS Logic Octal D-Type Flip-Flop, 3-State Positive-Edge Triggered

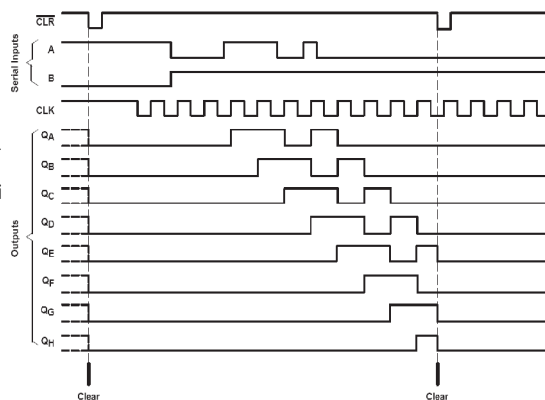
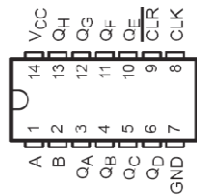


57

## Pomerački (SHIFT) REGISTRARI

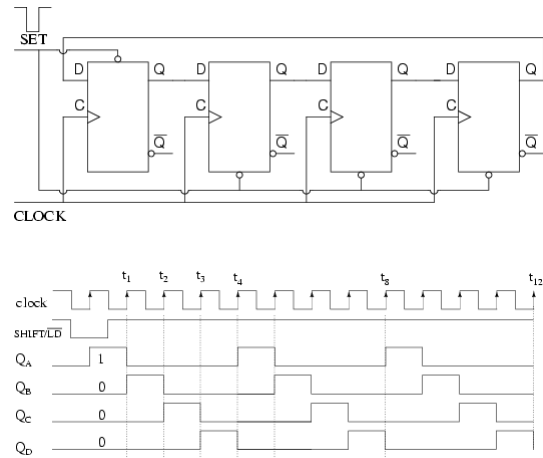


- Sadržaj elementarnih memorija (flip-flop-ova) se **prepisuje iz jednog u drugi**.
- Obično imaju **jedan ulaz i jedan izlaz** ali se može obezbediti i paralelni upis i čitanje.
- Primer: **SN74HC164** 8-bit parallel-out serial shift register: zajednički signal za brisanje, čitanje serijski ili paralelno.



## Kružni registri (kružni brojači)

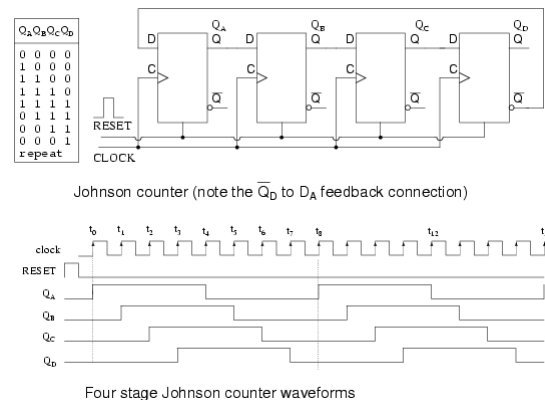
- **Povratna sprega** sa izlaza na ulaz.
- Sadržaj pravi jedan **krug u toku n ciklusa** takta.
- Nekako treba "**pokrenuti**": signal SET upisuje jedinicu u jedan flip-flop i nule u ostale (dozvoljene su i druge kombinacije)



59

## Johnson-ov brojač

- Invertovani izlaz poslednjeg flip-flop-a je **vraćen na ulaz** prvog.
- Sadržaj čini jedan krug u toku **2n ciklusa** (n je broj flip-flop-ova).
- Nije potrebno početno podešavanje.

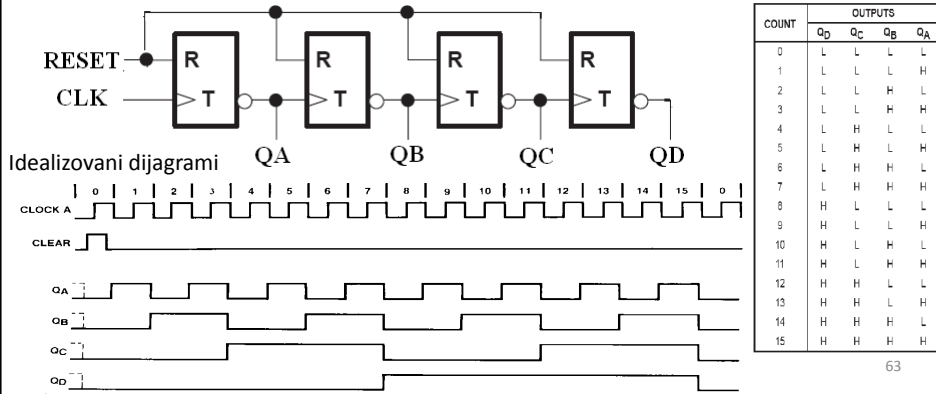


60



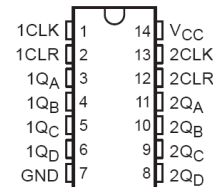
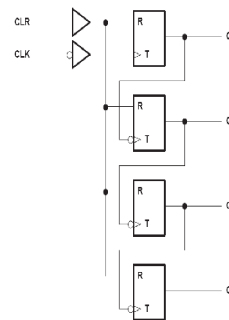
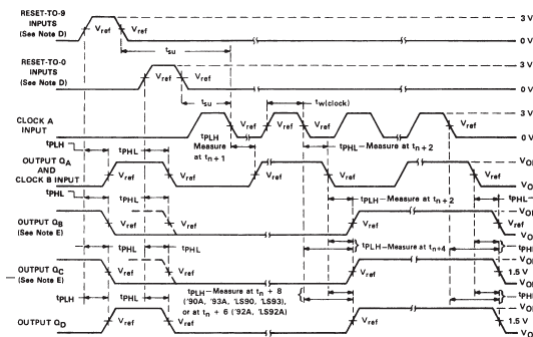
## Asinhroni (redni) brojači

- Obično se pravi od **T flip-flop-ova**.
- Prosta kaskadna veza, **nema ni ulazne ni izlazne kombinacione mreže**.
- **Takt signal (clock)** se dovodi samo **na prvi flip-flop**, ostali dobijaju pobudu jedan od drugog.
- **Promene stanja** pojedinih flip-flop-ova **kasne** u odnosu na takt – zato, u kratkim intervalima posle ivice takta izlazi nisu važeći.



## Primer za asinhroni brojač

- Primer: **SN74HC393** Dual four-bit asynchronous binary counters.
- Modul:  $2^4=16$ .
- Na dijagramu se mogu analizirati **kašnjenja**.





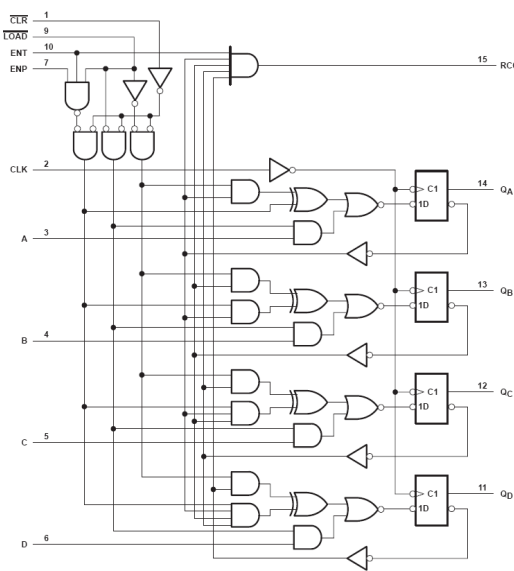
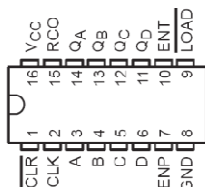
## Sinhroni (paralelni) brojači

- Pojedini flip-flop-ovi dobijaju **isti takt signal**, zato se istovremeno upisuje novi sadržaj (dobija se pravilno ponašanje i pri višim frekvencijama takta).
- **Potrebna je ulazna kombinaciona mreža** za pripremanje novog sadržaja (stanja).
- Konstrukcija ulazne kombinacione mreže se vrši pomoću metode koja je prikazana za **sintezu logičkih automata**.
- Izlazna kombinaciona mreža redovno nije potrebna.
- Modul  $\leq 2^n$ .

65

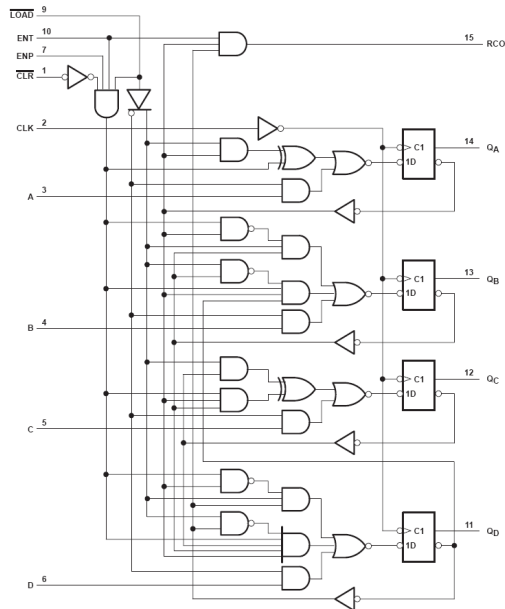
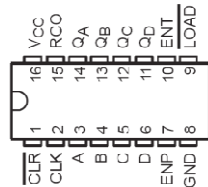
## Primer sinhronog binarnog brojača

- **SN74ALS161B** - SYNCHRONOUS 4-BIT BINARY COUNTER
- Mogućnost paralelnog upisa (A,B,C,D, LOAD)
- Istovremeno brisanje svih flip-flop-ova (CLR).
- Signali ENT, ENP i RCO su potrebni za kaskadno vezivanje više ovakvih brojača (radi povećanja modula).



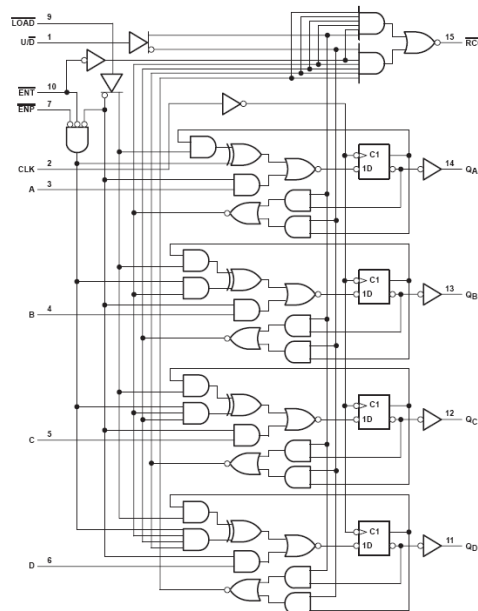
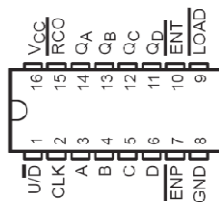
## Primer sinhronog decimalnog brojača

- **SN74ALS162B** - SYNCHRONOUS 4-BIT DECIMAL COUNTER
- Mogućnost paralelnog upisa (A,B,C,D, LOAD).
- Istovremeno brisanje svih flip-flop-ova (CLR).
- Signali ENT, ENP i RCO su potrebni za kaskadno vezivanje više ovakvih brojača (radi povećanja modula).



## Primer sinhronog brojača napred/nazad

- **SN74ALS169B** - SYNCHRONOUS 4-BIT UP/DOWN BINARY COUNTER
- U/D - određuje smer brojanja
- Mogućnost paralelnog upisa (A,B,C,D, LOAD)
- Signali ENT, ENP i RCO su potrebni za kaskadno vezivanje više ovakvih brojača (radi povećanja modula).



## Mešovite mreže

Dve mogućnosti:

1. Digitalna kola koja sadrže i kombinacije i sekvencijalne elemente. Težište može biti i na jednoj i na drugoj vrsti elemenata
2. Kombinacija digitalnih i analognih kola u jednom kućištu.

Podela:

1. Memorijska kola
2. Aritmetičke jedinice
3. D/A pretvarači
4. A/D pretvarači

69

## Memorije

- Mogu trajno ili privremeno da skladište veću količinu podataka.
- Za postizanje velikog kapaciteta potreban je veliki skladišni **prostor** i dobra **organizacija**.
- Postoji i drugi načini pamćenja podataka (optički, magnetni...). Mi se ovde bavimo samo sa poluprovodničkim rešenjima.
- Blok šema memorijskog kola:

A - adresne linije

C - upravljački signali

D - magistrala podataka



70

## Podela i karakteristike memorija

### Principi po kojima delimo memorije:

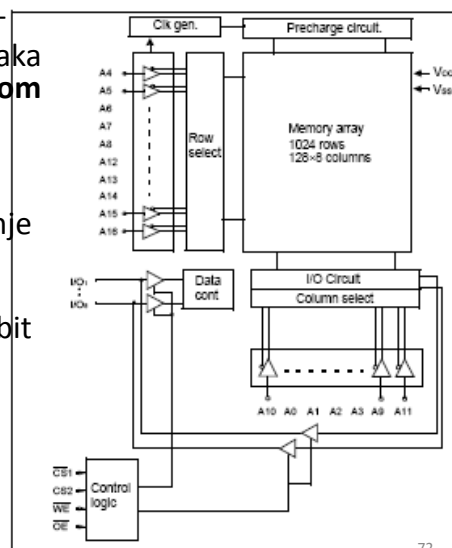
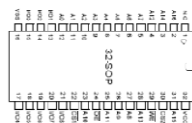
- **Brzi** ili **spori** pristup (može biti velika razlika između brzine i učestalosti upisa i čitanja kod iste memorije),
- **Statički** (koristi flip-flop-ove) ili **dinamički** (koristi parazitne kapacitivnosti) način čuvanja podataka,
- Pristup podacima po nekom **redu** ili u **proizvoljnom redosledu**,
- **Jednobitni** ili **višebitni** podaci,
- Tehnologija proizvodnje: **CMOS** ili **bipolarna**.

71

## RAM

### RAM (random access memory) -

- Upis i čitanje pojedinih podataka se može obaviti po **proizvoljnom redosledu**.
- Čuvanje podataka može biti **statičko** (SRAM) ili **dinamičko** (DRAM). Dinamičko skladištenje zahteva manje elemenata po jednoj ćeliji.
- Primer: **K6T1008V2C** 128Kx8 bit Low Power and Low Voltage CMOS Static RAM

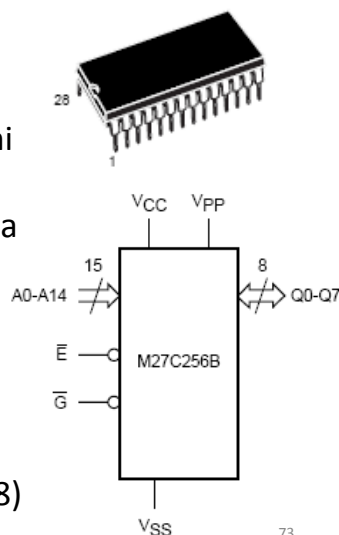


72

## ROM

### ROM (read only memory)

- jednom upisani **sadržaj** se nikad više **ne može promeniti**,
- podaci se čitaju se brzo i neograničeni broj puta,
- po strukturi u pitanju je kombinaciona mreža (konvertor koda),
- OTP ROM - programira korisnik,
- mask programmable ROM - programira proizvođač
- nisu pogodni za razvoj proizvoda
- Primer: **M27C256B** 256 Kbit (32Kb x 8) UV EPROM and OTP EPROM

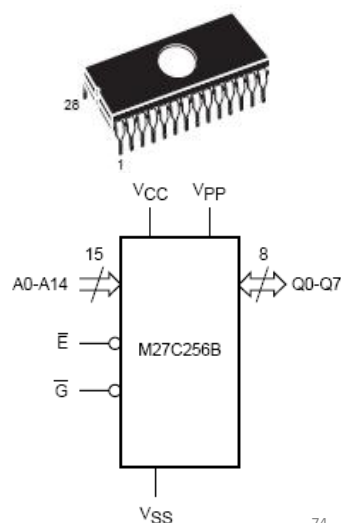


73

## EPROM

### EPROM - electrically programmable ROM

- **programira se** (upis podataka) **električnim signalima** - na gejt MOSFET-a se dovodi opterećenje koje ostaje zarobljeno,
- **brisanje ultraljubičastim zračenjem** - relativno spor proces,
- postoji stakleni prozor na gornjoj strani kućišta,
- Primer: **M27C256B** 256 Kbit (32Kb x 8) UV EPROM and OTP EPROM.

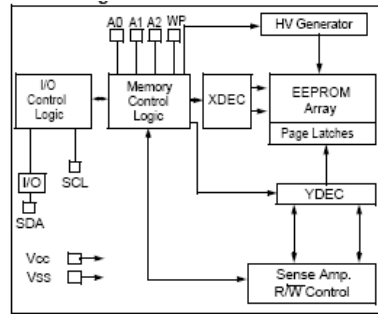
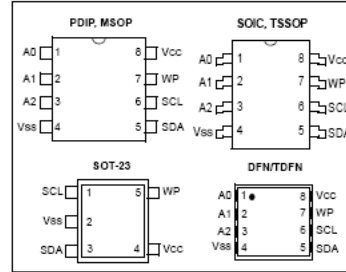


74

# EEPROM

## EEPROM - electrically erasable PROM

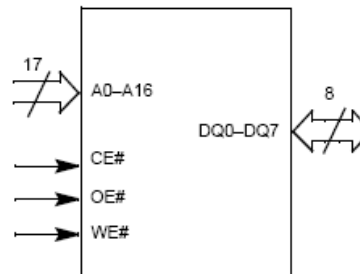
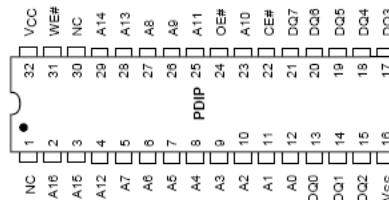
- **brisanje** sadržaja i ponovni **upis električnim signalima** - nema prozora,
- čitanje se može obaviti brzo, upis je nešto sporiji,
- često se pravi sa serijskim upisom/čitanjem (može da se smesti memorija velikog kapaciteta u kućište sa malim brojem nožica)
- Primer: **24AA32A/24LC32A** 32K I2C™ Serial EEPROM



# Flash EEPROM

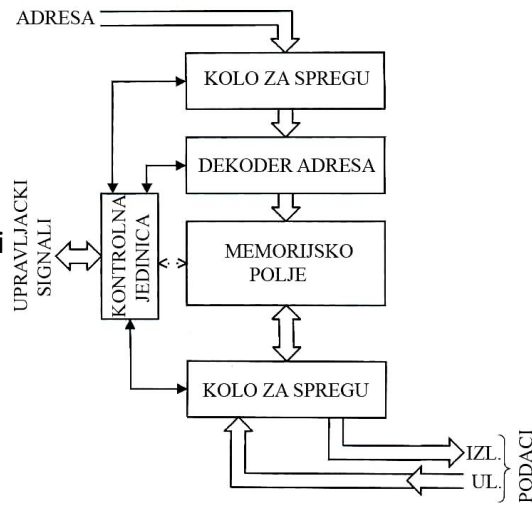
## flash EEPROM (flash memorija)

- relativno brzo brisanje i ponovni upis električnim signalima,
- čuva podatke i nakon isključenja napajanja,
- broj ponovnih upisa je ograničen (na pr. 100,000),
- brisanje sadržaja redovno se može raditi po sektorima,
- **Am29F010** 1 Megabit (128 K x 8-bit) CMOS 5.0 Volt-only, Uniform Sector Flash Memory



## Unutrašnja struktura memorijskih kola

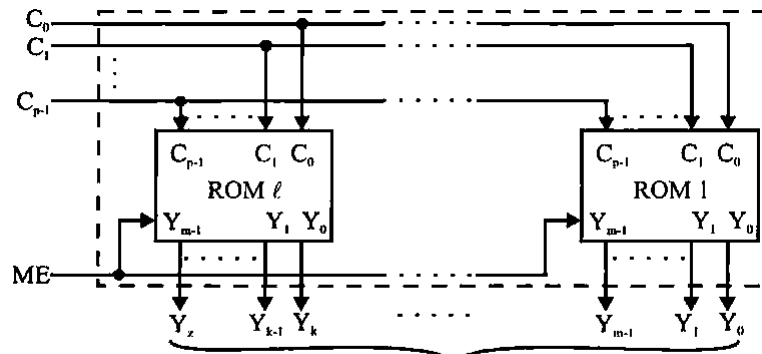
- Zbog velike količine podataka potrebno je efikasno organizovanje.
- **Centralni deo:** polje memorijskih ćelija.
- Uloga **dekodera** je da izabere aktuelnu ćeliju.
- Radi uprošćenja dekodera, dekodera se deli na dve manje jedinice (kolone i vrste).
- Podaci ulaze i izlaze na istim linijama, zahvaljujući dvosmernim kolima za sprezanje
- Upravljački signali: WE, OE, CS



77

## Proširivanje kapaciteta - povećanje dužine reči

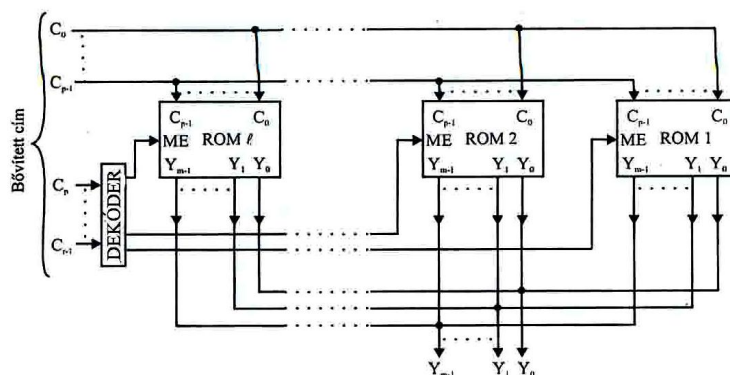
- Koje su nam mogućnosti ako nije dovoljan kapacitet memorije koji se pravi u jednom kućištu?
- **Povećanje dužine reči:** zajedničke adresne linije, linije podataka se stave paralelno.



78

## Proširivanje kapaciteta - povećanje broja reči

- Koje su nam mogućnosti ako nije dovoljan kapacitet memorije koji se pravi u jednom kućištu?
- **Povećanje broja reči** koji se može adresirati. Izbor pojedinih kućišta sa spoljnim dekomerom. Ulazi i izlazi se spajaju na istu magistralu.



79

## Aritmetičke jedinice

Ove funkcije se najviše sreću kao unutrašnji moduli mikrokontrolera ali se nađe i poneko MSI kolo.

Postoje kola za sledeće funkcije:

- sabiranje
- množenje
- komparacija brojeva
- provera parnosti broja jedinica u broju

80



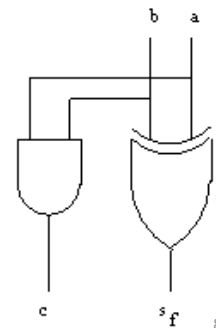
## Kola za sabiranje - polusabirač

- Osnovno kolo je polusabirač: sabere dva bita, odredi sumu i prenos na veću pozicionu vrednost.
- Nije pogodan za kaskadno vezivanje (što bi bilo neophodno za sabiranje više bitnih brojeva)
- Jednačine kola su:

$$s_f = \bar{a}b + a\bar{b} = a \oplus b$$

$$c = ab$$

a	b	c	s <sub>f</sub>
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

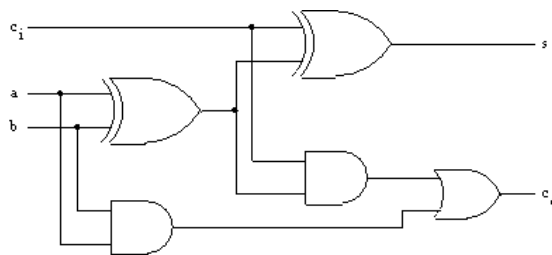


81

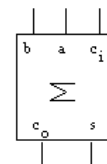
## Kola za sabiranje – potpuni sabirač

- Kod potpunog sabirača postoji mogućnost kaskadnog vezivanja pošto je predviđeno prihvatanje ulaznog prenosa sa niže pozicione vrednosti.

- Jednačine kola su:  $s = a \oplus b \oplus c_i$   
 $c_o = (a \oplus b)c_i + ab$



(a)

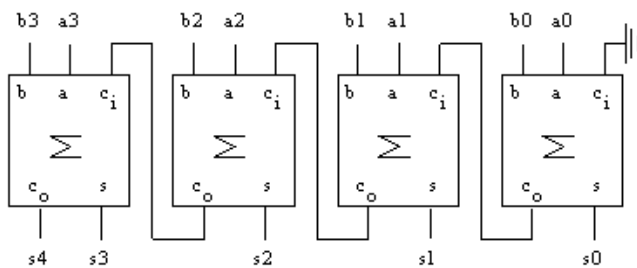


(b)

82

## Sabiranje višebitnih brojeva

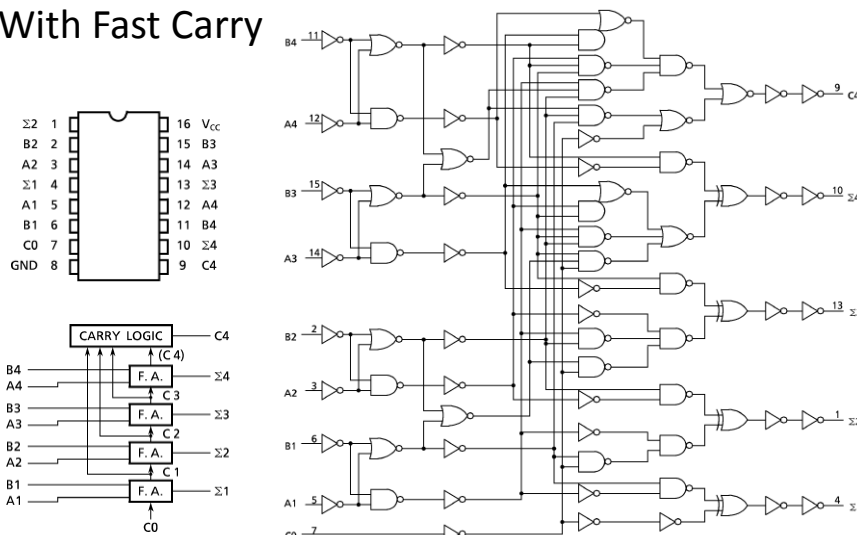
- Sabiranje višebitnih brojeva.
- Kaskadno vezivanje jednobitnih potpunih sabirača.
- Određivanjem prenosa paralelnim postupkom može se ubrzati rad kola (potreba posebna mreža).



83

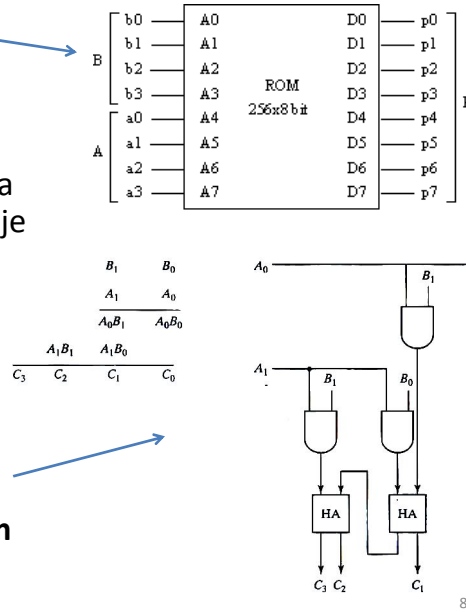
## Primer MSI sabirača

- Primer: **CD54/74ACT283** 4-Bit Binary Full Adder With Fast Carry



## Kola za množenje

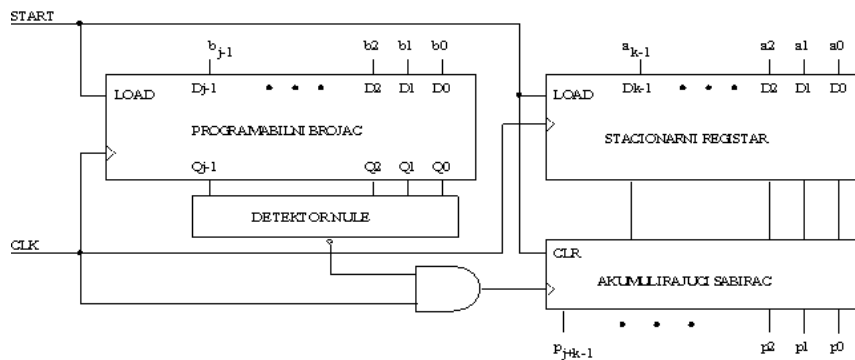
- Primenom **čisto kombinacione mreže** - na osnovu tabele - za svaku ulaznu varijaciju se odredi izlaz. Sa porastom broja bita složenost mreže za množenje nesrazmerno raste (jedna prihvatljiva metoda je korišćenje **ROM-a**).
- Takođe primenom čisto kombinacione mreže: **množenje bit po bit sa 1 kolima i sabiranje međurezultata sa potpunim sabiračima**.



## Množenje svedeno na višestruko sabiranje

Sekvencijalna mreža:

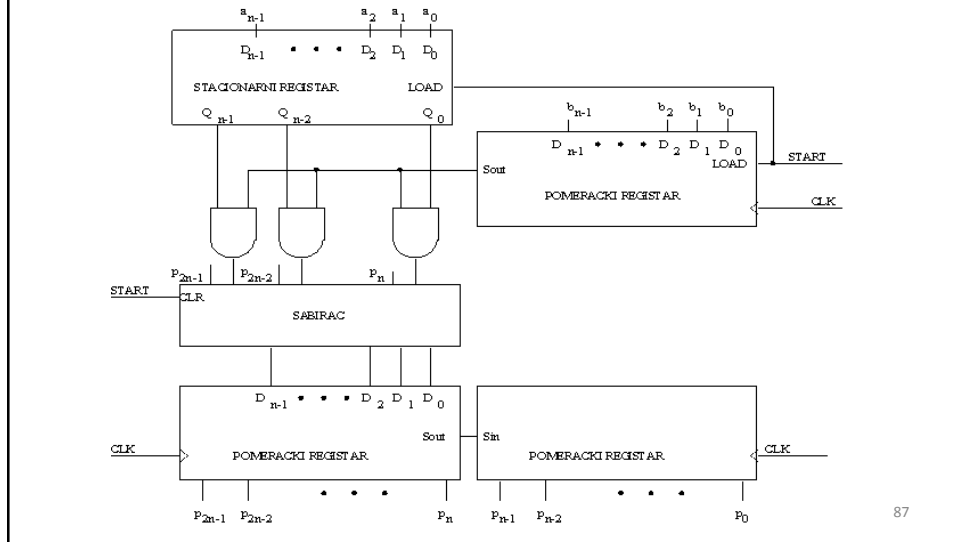
Sabira se množilac onoliko puta kolika je brojna vrednost množitelja.



# Množenje bit po bit

Sekvencijalna mreža:

Množenje bit po bit i sabiranje međurezultata.



87

## Aritmetički (digitalni) komparator

- Upoređenje binarnih brojeva po veličini.
- Mogući rezultati: manje, veće, jednako.
- Za slučaj jednobitnih brojeva važe izrazi:

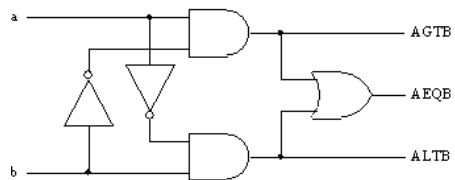
$$AGTB = a\bar{b}$$

$$AEQB = ab + \bar{a}\bar{b} = \bar{a} \oplus \bar{b}$$

$$ALTB = \bar{a}b$$

- Kombinaциона tabela i logička šema:

A	B	AGTB	AEQB	ALTB
0	0	0	1	0
0	1	0	0	1
1	0	1	0	0
1	1	0	1	0

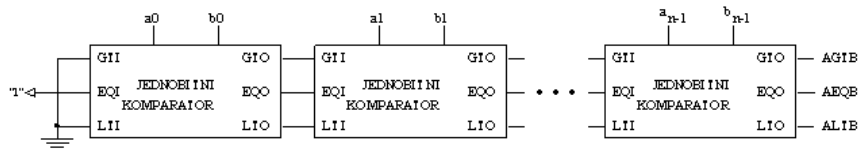


88

## Kaskadna veza jednobitnih komparatora

Upoređenje više bitnih brojeva

1. Kaskadna veza jednobitnih komparatora:

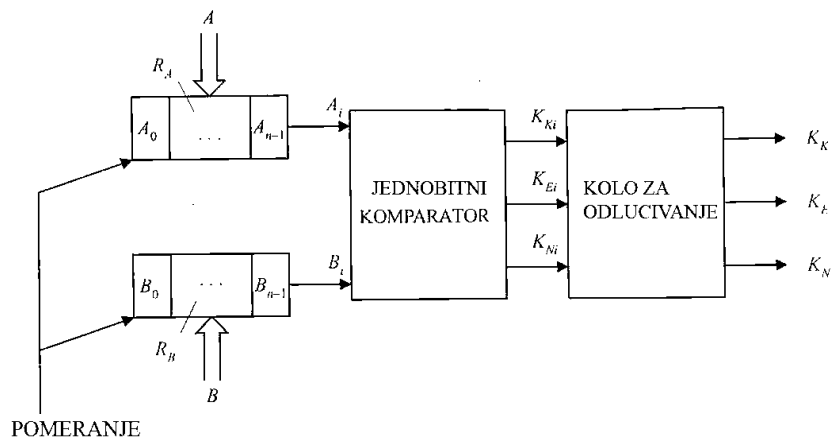


- Za kaskadnu vezu potrebno je obezbediti odgovarajuće ulaze i izlaze.
- Prosto ali relativno sporo rešenje zbog serijskog prenosa.

89

## Sekvencijalni komparator

Upoređenje višebitnih brojeva

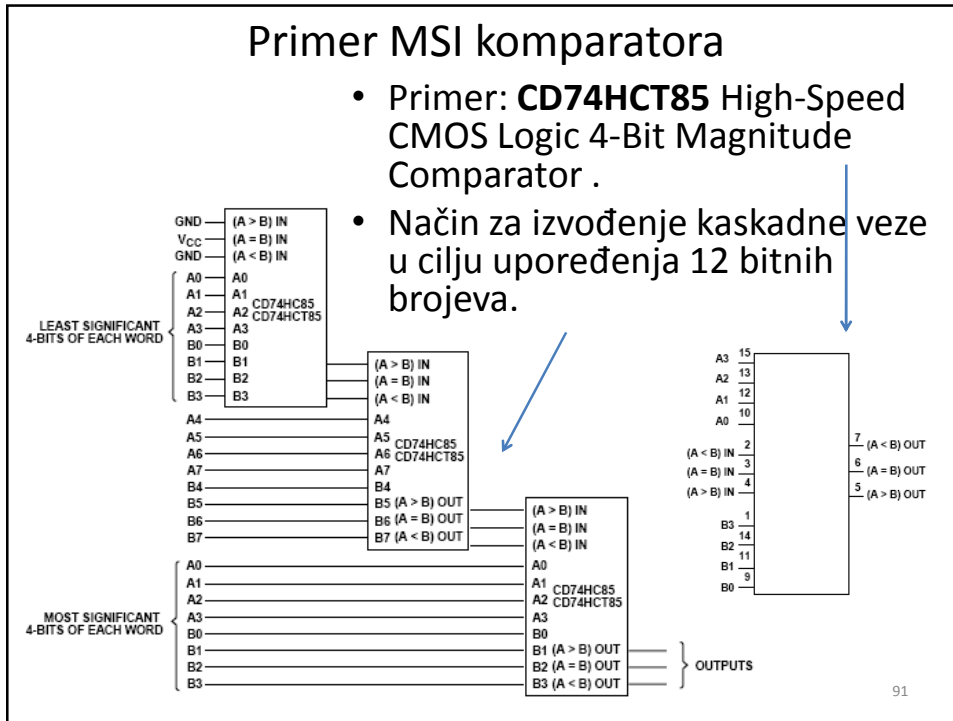


- Razumno je upoređivanje početi od bitova veće težine.

90

### Primer MSI komparatora

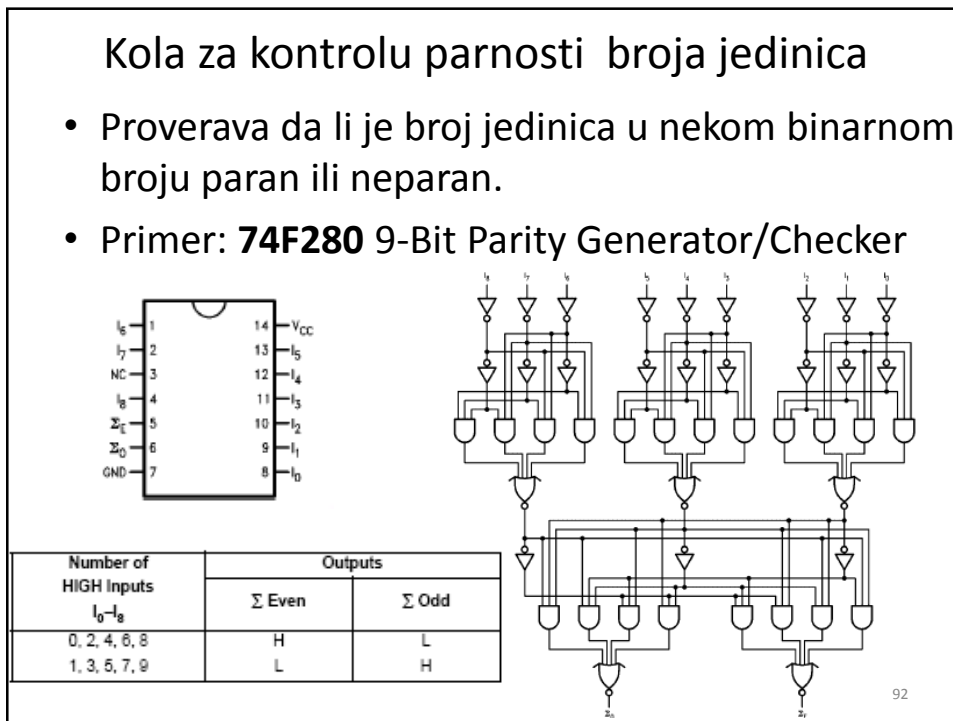
- Primer: **CD74HC85** High-Speed CMOS Logic 4-Bit Magnitude Comparator .
- Način za izvođenje kaskadne veze u cilju upoređenja 12 bitnih brojeva.



91

### Kola za kontrolu parnosti broja jedinica

- Proverava da li je broj jedinica u nekom binarnom broju paran ili neparan.
- Primer: **74F280** 9-Bit Parity Generator/Checker



92

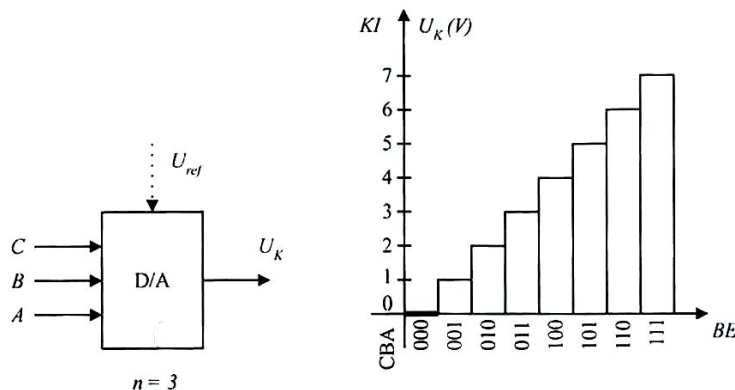
## Digitalno/analogni pretvarači

- Od brojeva formiraju analogne signale (napon određene vrednosti).
- Dobijena vrednost napona je redovno srazmerna sa vrednošću broja.
- Dobijene vrednosti napona formiraju diskretnu skalu.

93

## Princip rada D/A pretvarača

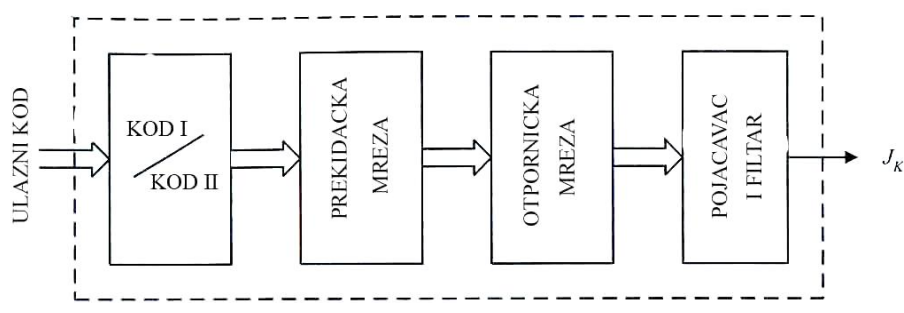
- U slučaju broja (kôd) sa  $n$  bita postoje  $2^n$  mogućih vrednosti izlaznog napona.
- Potreban je jedan referentni napon ( $V_{REF}$ ), vrednosti na skali su srazmerni sa tim naponom.



94

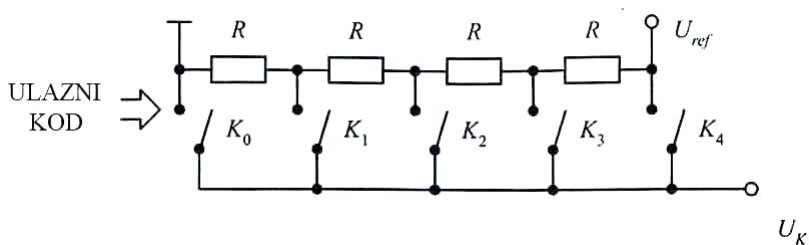
## Konstrukcija D/A pretvarača

- Glavni deo pretvarača je jedna otpornička mreža.
- Odnos razdele napona variramo uključivanjem/isključivanjem analognih prekidača.
- Upravljanje analognim prekidačima vršimo u skladu sa ulaznim brojem (kôdom). U nekim slučajevima potrebno je pretvaranje kôda.
- Na izlazu je redovno potrebna neka analogna obrada (pojačanje, filtracija).



## D/A pretvarač direktnog tipa

- Redno se vežu otpornici iste otpornosti.
- Na izvodima razdelnika se formiraju svi naponi prema predviđenoj skali.
- Za upravljanje analognim prekidačima redovno je potreban dekoder.



96

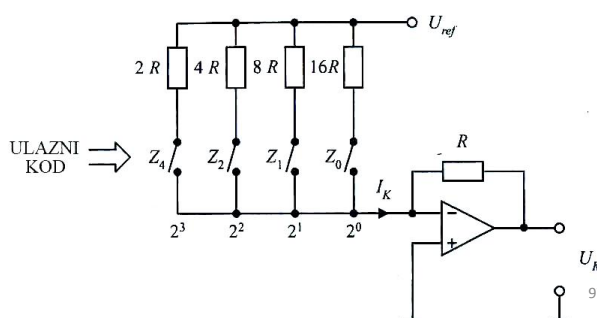


## D/A pretvarač sa težinskom otpornom mrežom

- Vrednosti otpornosti i struje kroz njih su u odnosima  $1:2:4: \dots 2^n$ .
- Formula za računanje izlaznog napona:

$$V_o = -R_f V_{REF} \frac{1}{R} (2^0 Q_0 + 2^1 Q_1 + 2^2 Q_2 + \dots 2^{n-1} Q_{n-1})$$

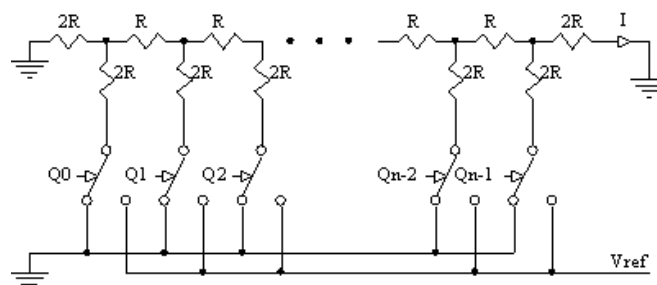
- Nije pogodno za realizaciju u integrisanoj izvedbi.



## D/A pretvarač sa R-2R lestvičastom mrežom

- U integrisanoj izvedbi se redovno proizvode ove vrste pretvarača - potrebno je reprodukovati samo dve vrednosti otpornosti.
- Formula za računanje izlazne struje (izlazni napon je srazmeran sa strujom):

$$I = \frac{V_{REF}}{6R} \cdot \frac{1}{2^{n-1}} (2^{n-1} Q_{n-1} + 2^{n-2} Q_{n-2} + \dots + 2^1 Q_1 + 2^0 Q_0)$$

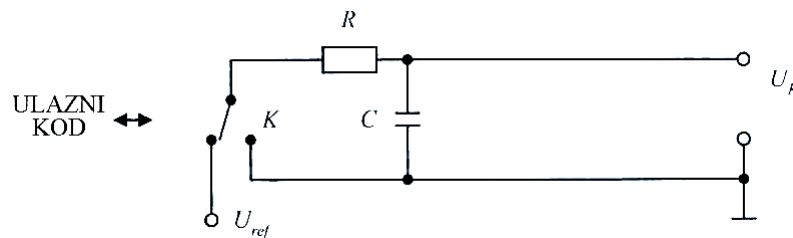


## D/a pretvarač sa impulsno-širinskom modulacijom

- Reč je o pomoćnom rešenju, realizuje se sa jako malim brojem komponenti.
- Rešivo je i sa mikrokontrolerom bez analognog izlaza.
- Formula za izlazni napon je:

$$V_O = D \cdot V_{REF} \quad D = f(Q_0, Q_1 \dots Q_{n-1})$$

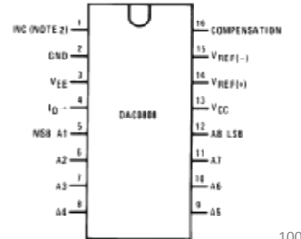
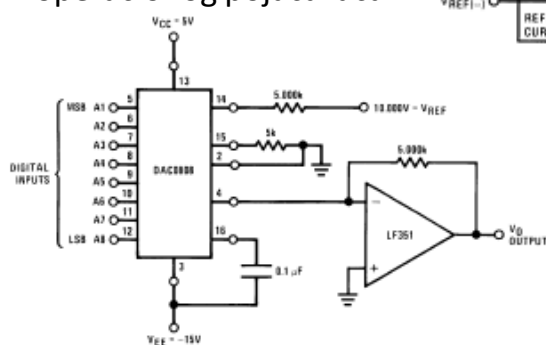
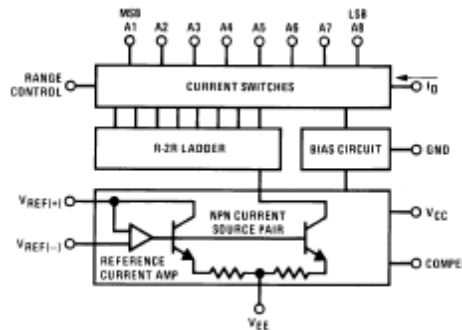
- Zbog kašnjenja filtra sporo radi.



## Primer integrisanog D/A pretvarača

Primer: **DAC0808** 8-Bit D/A Converter.

- Sa R-2R lestvičastom mrežom
- Daje izlazni napon koji se pretvara u struju pomoću operacionog pojačavača.



100

## Karakteristike D/A pretvarača

### Rezolucija

- izražava se brojem ulaznih bitova
- to ujedno određuje i tačnost jer treba garantovati monotonost pretvaračke karakteristike.

### Brzina

- pretvarači sa otporničkim mrežama su relativno brzi, vreme uspostavljanja je obično manje od  $\mu s$  (kašnjenje analognih prekidača i operacionih pojačavača)
- pretvarači sa impulsno-širinskom modulacijom su spori: vreme uspostavljanja je mnogostruka vrednost periode impulsa na izlazu pretvarača.

101

## A/D pretvarači

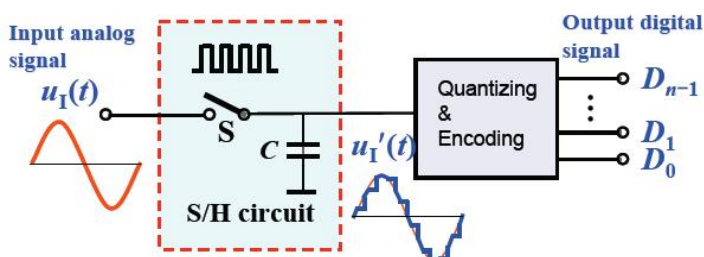
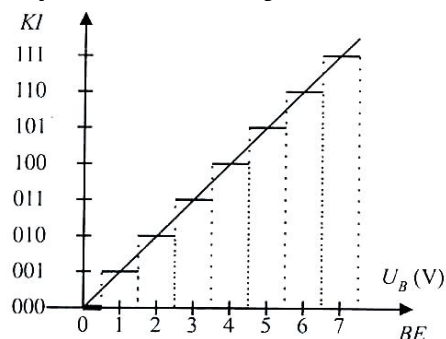
1. Pretvaranje analognog signala (napon) u digitalni signal (broj)
2. Razlozi za pretvaranje:
  - digitalno skladištenje
  - digitalna obrada signala
  - digitalni prenos signala
  - digitalno prikazivanje

102

## Principi A/D pretvaranja

**Zadaci** koje treba rešiti:

1. odabiranje (diskretizacija po vremenu),
2. diskretizacija po amplitudi (upoređenje sa odgovarajućom skalom),
3. kodiranje (svakoj diskretnoj vrednosti se dodeljuje kodna reč).



103

## Konstrukcija A/D pretvarača

U praksi se spominju četiri rešenja:

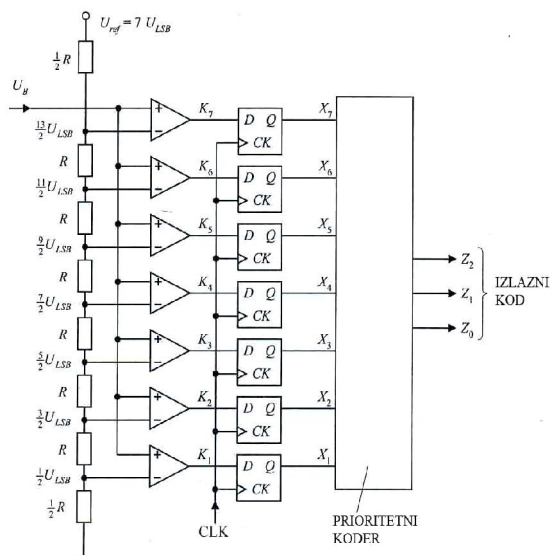
1. **Direktni** (flash tip) pretvarača
2. **Postepeno približavanje** (sukcesivna aproksimacija - pretvaranje bit po bit)
3. **Brojačko** (integratorsko) rešenje
4. **Sigma-delta** A/D pretvarač

Postoje ozbiljne razlike u ceni, brzini i rezoluciji.

104

## Direktni (flash) A/D pretvarač

- Za  $n$ -bitni pretvarač  $2^n - 1$  komparatora vrše diskretizaciju po amplitudi.
- Kodiranje prioriternim koderom.
- Sinhronizacija taktom obezbeđuje da iščitavamo samo važeće kôdove.
- Složeni, skupi hardver.
- Velika brzina.

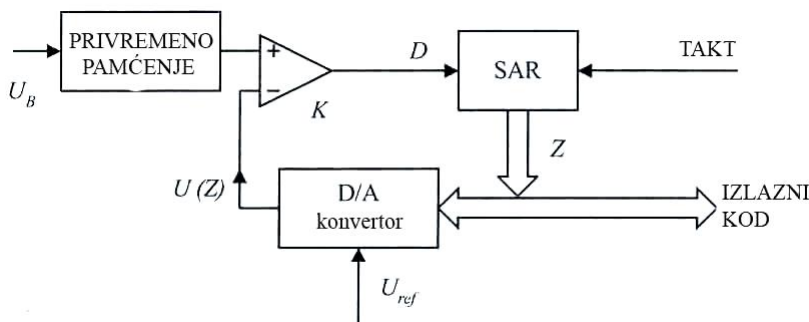


105

## A/D pretvarač sa sukcesivnom aproksimacijom

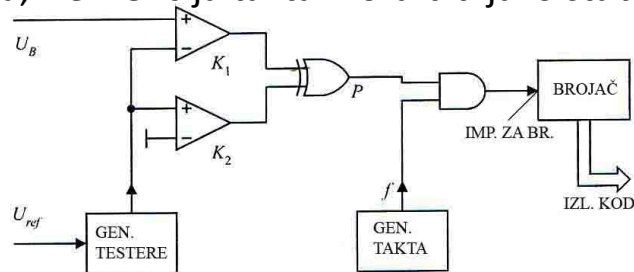
Sadržaj SAR-a (registar) postavljamo na logičku jedinicu bit po bit (počev od bita najveće težine).

- D/A pretvarač proizvodi odgovarajuću analognu vrednost.
- Komparator donosi odluku da li je bilo potrebno postaviti dotični bit na jedinicu.



## A/D konvertor brojačkog tipa – jednostruka integracija

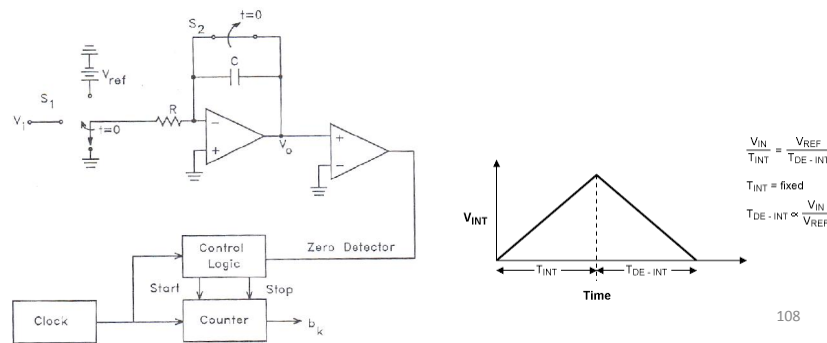
- Analogni napon se upoređuje sa signalom iz generatora testerastog napona - time se napon pretvara u vreme.
- U slučaju većeg napona brojač broji srazmerno duže.
- Slabe tačke: ne sme da se menja nagib testerastog signala, frekvencija takta mora biti jako stabilna.



107

## A/D konvertor brojačkog tipa – dvostruka integracija

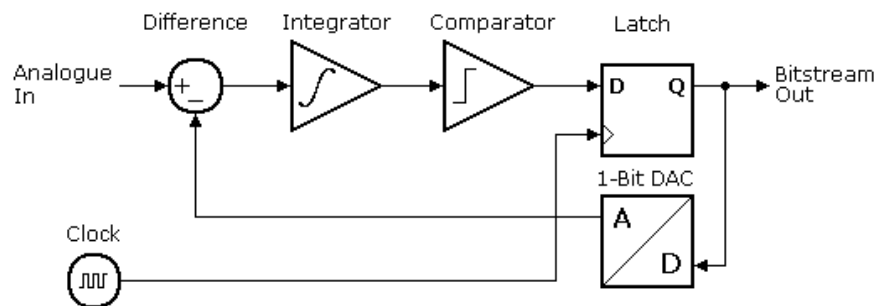
- Samo  $V_{REF}$  treba da je tačne vrednosti.
- Frekvencija takta i elementi integratora treba samo da su stabilne vrednosti na kratkoj stazi (za vreme jednog pretvaranja).



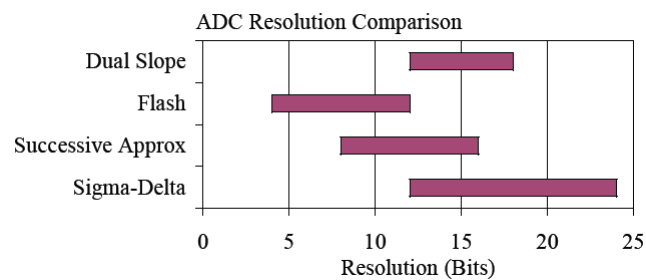
108

## Sigma-delta A/D pretvarača

- Integrator integriše razliku analognog signala i izlaza D/A pretvarača.
- Komparator određuje predznak integrala.
- Izlaz komparatora se upisuje u flip-flop.
- Broj impulsa na izlazu flip-flop-a dobijen u jedinici vremena je srazmeran sa analognim signalom.



## Upoređenje A/D pretvarača

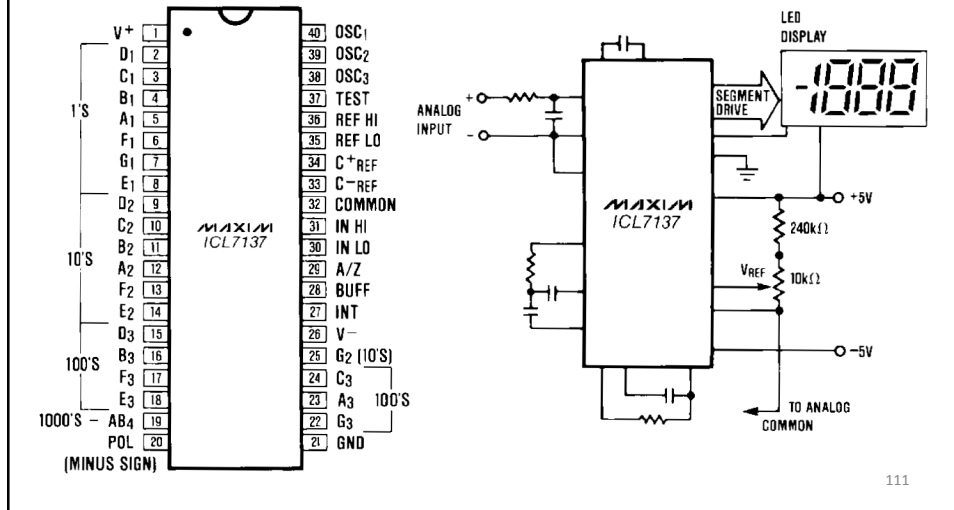


Type	Speed (relative)	Cost (relative)
Dual Slope	Slow	Med
Flash	Very Fast	High
Successive Approx	Medium – Fast	Low
Sigma-Delta	Slow	Low

110

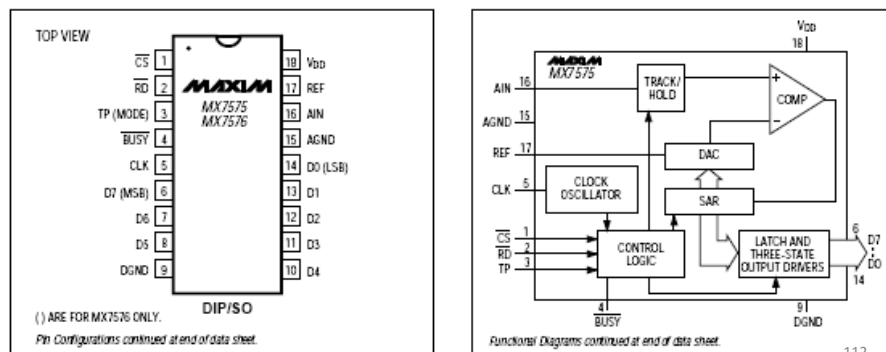
## Primeri integrisanih A/D pretvarača

- **ICL7137** 3 ½ cifarski (decimalne cifre) A/D pretvarač (sa dvostrukom integracijom) (digitalni voltmetar)



## Primeri integrisanih A/D pretvarača

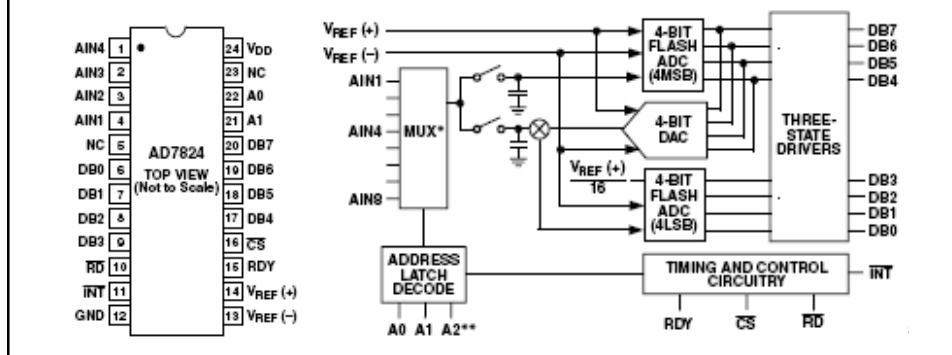
- **MX7575/MX7576** CMOS,  $\mu$ P-Compatible, 5 $\mu$ s/10 $\mu$ s, 8-Bit ADCs (sa sukcesivnom aproksimacijom)
- Napajanje 5V, ugrađeno je kolo za odabiranje, granična frekvencija analognog signala je 50 kHz.





## Primeri integriranih A/D pretvarača

- **AD7824/AD78** Speed 4- and 8-Channel 8-Bit ADCs **28** LC<sup>2</sup>MOS High
- Poseban četvorobitni pretvarač direktnog (flash) tipa za gornjih četiri bita i drugi pretvarač za preostala četiri bita. Vreme pretvaranja 2,5 $\mu$ s.



## VLSI kola sa softverskim programiranjem

Algoritamsko izvršavanje raznih zadataka:

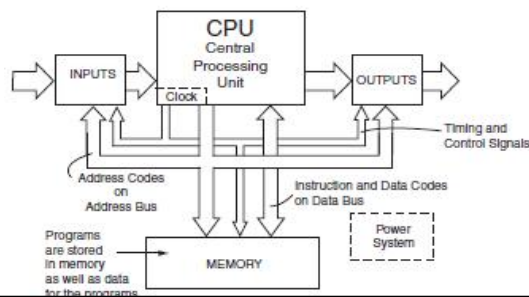
- obrada signala
- kontrola/regulacija
- signalizacija/indikacija
- komunikacija...

Tipovi:

- mikroprocesori
- mikrokontroleri
- signal procesori

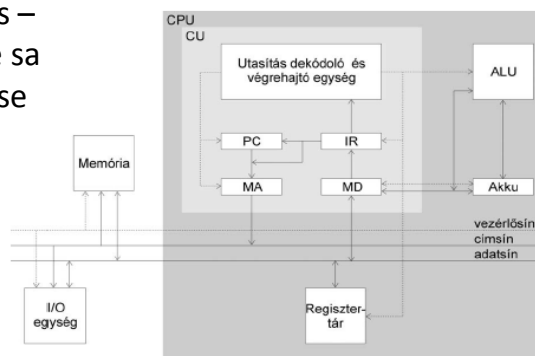
## Softverom vođeni sistem za obradu podataka – digitalni rađunar

- Softverom vođeni sistem za obradu podataka prima signale, memoriše ih, obrađuje ih, i rezultate prosleđuje prema izlazu.
- CPU (central processing unit – centralna procesna jedinica) vrši aritmetičke i logičke operacije, donosi odluke, usmerava protok podataka itd.



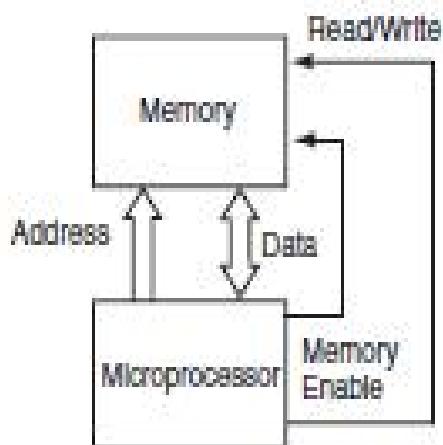
## CPU

- ALU – aritmetička / logička jedinica, izvršava operacije
- CU – control unit - kontroliše rad ALU (dekodira instrukcije i izdaje kontrolne signale za izvršavanje operacija)
- PC – program counter – čuva adresu sledeće operacije
- IR – instruction register – prima kod operacije (instrukcije) i podatke iz memorije
- MA – memory address – čuva adrese memorije sa koje se čita ili na koju se piše.
- MD – memory data - čuva podatke koji se upisuju u memoriju ili koji se primaju iz memorije.



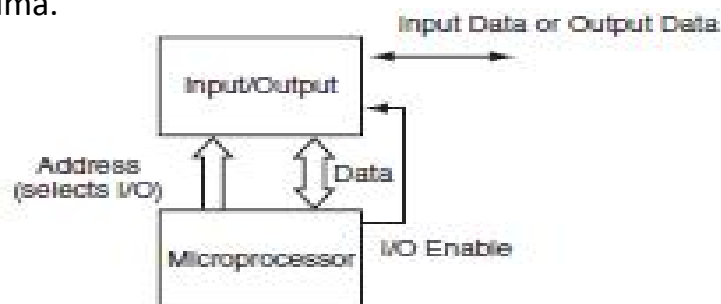
## Operativna memorija

- Služi za čuvanje programa i podataka.
- Neumann-ova arhitektura: i program i podaci su u istoj memoriji.
- Harvard arhitektura – posebna memorijska jedinica za program i podatke.
- Način spajanja CPU i memorije je prikazan na slici.
- Broj adresnih linija zavisi od kapaciteta memorije.



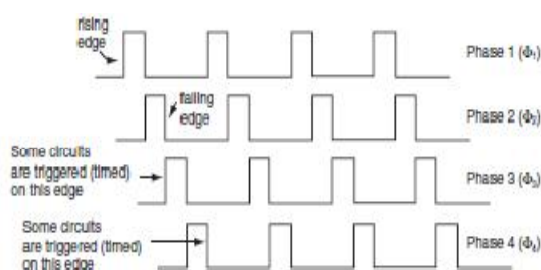
## Ulazno/izlazna jedinica

- Preko ulazno/izlazne jedinice takođe se obavlja prenos podataka.
- Prenos zahteva adresiranje.
- Potrebni su kontrolni signali za usmeravanje podataka.
- Ponekad je potrebno prilagođenje nivoa.
- Ako periferija radi asinhrono, potrebno je ugraditi FIFO (first in, first out memory).
- Prekidi su specijalni ulazi koji informišu CPU o nekim događajima.



## Uloga takta

- CPU je sinhroni logički automat.
- Signal za sinhronizaciju se zove takt ili clock (CLK).
- Frekvencija takta je obično od 1 MHz do 100 MHz.
- Neki procesori zahtevaju ili unutar generišu višefazni takt. U svakoj fazi se obavljaju određene pod-operacije.



## Izvršavanje operacija

Proces izvršavanja jedne operacije se sastoji od sledećih koraka:

- Sadržaj PC registra se upiše u MA registar i na adresnim linijama memorije se pojave adrese.
- Podatak (u ovom slučaju je to kod instrukcije) koji se pojavi na linijama podataka se upiše u MD registar.
- Sadržaj PC registra se poveća za jedan.
- Sličnim postupkom se učitava podatak iz memorije.
- Dekoder instrukcije protumači učitani kod instrukcije i izvrši operaciju.
- Ponovo se poveća sadržaj PC registra da bi ukazivao na sledeću instrukciju.

## Tipovi operacija

- binarne aritmetičke operacije,
- logičke operacije,
- registarske operacije
- operacije vršene nad pojedinim bitovima,
- operacije za prenos podataka,
- operacije skoka

## Načini adresiranja

- Direktno registarsko adresiranje – kod instrukcije ukazuje na registar koji sadrži operand.
- Direktno memorijsko adresiranje – adresa memorijske lokacije je sadržan u kodu instrukcije.
- Indirektno adresiranje – u kodu instrukcije je adresa memorijske lokacije na kojoj se nalazi adresa podatka.
- Indirektno registarsko adresiranje – u kodu instrukcije se ukazuje na registar u kome se nalazi adresa operanda.
- adresiranje sa relativnim baznim registrom – deo koda instrukcije ukazuje na jedan registar ali sadrži i jedan ofset koji treba dodati na sadržaj registra da bi dobili adresu podatka.
- neposredno adresiranje – podataka se nalazi u kodu instrukcije ili odmah iza memorijske lokacije odakle se učitala instrukcija.

## Prekidi i izuzeci

- Pri dobijanju prekida iz okoline treba izvršiti neku rutinu.
- Rutina se redovno ne izvršava odmah, CPU treba da prihvati prekid.
- U slučaju više prekida treba uspostaviti prioritet.
- Pre izvršavanja prekida treba memorisati sadržaj PC registra i nekih drugih registara koji bi se mogli promeniti pri izvršavanju prekidne rutine.
- Nakon završetka rutine treba vratiti prvobitni sadržaj registara.
- Deo memorije koji čuva sadržaj registara se zove stack.
- Ponekad i unutrašnji događaji prouzrokuju prekide – oni se zovu izuzeci.

## Mašinski jezik

- Kodovi instrukcija koji se upisuju u memoriju su binarni nizovi. Procesor radi isključivo na bazi te binarne informacije.
- Program napisan u binarnoj formi je mašinski program.
- Za programera programiranje na mašinskom jeziku je zamorno i nepregledno

## Asembler

- Ako se binarni kodovi zamene sa mnemonicima (kratke reči), govorimo o programiranju na asemblerskom jeziku.
- I registri i podaci imaju svoje mnemonike.
- Asemblerski kod treba prevesti na mašinski jezik da bi se mogao izvršiti programiranje. To prevođenje se sastoji od prostih zamena mnemonika sa binarnim kombinacijama.
- Deo nekog asemblerskog programa se vidi dole.

```

                .ORIG      $100

BEGIN          CLRA
INC_LOOP      INCA
              CMPA      #$1E      ; compare ACCA = $1E
              BNE       INC_LOOP   ; if not equal, go back
              LDAA      #'Z'      ; else, load ASCII 'Z'
              BSR       SEND_CHAR  ; send ACCA to serial port
              BRA       BEGIN     ; start over again

```

## Viši programski jezici

- Radi pojednostavljenja programiranja često se pišu programi na nekom višem jeziku – oni su bliži ljudskom jeziku i načinu razmišljanja.
- Takve programe isto treba prevesti na mašinski jezik.
- Programi za prevođenje se zovu kompajler.

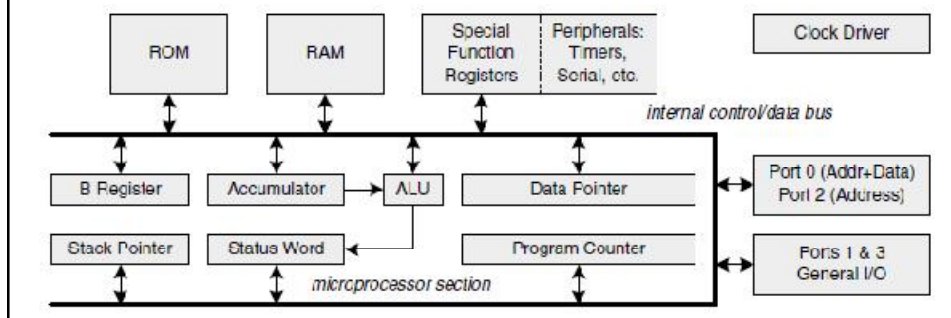
## Familije mikroprocesora

- Intel 8080
- Motorola 6800
- Intel 8086
- Motorola 68000
- Savremene familije: ARM procesori

## Mikrokontroleri

Radi se o VLSI kolima koji sadrže:

- mikroprocesor
- razne memorijske module
- razne ulazno/izlazne jedinice



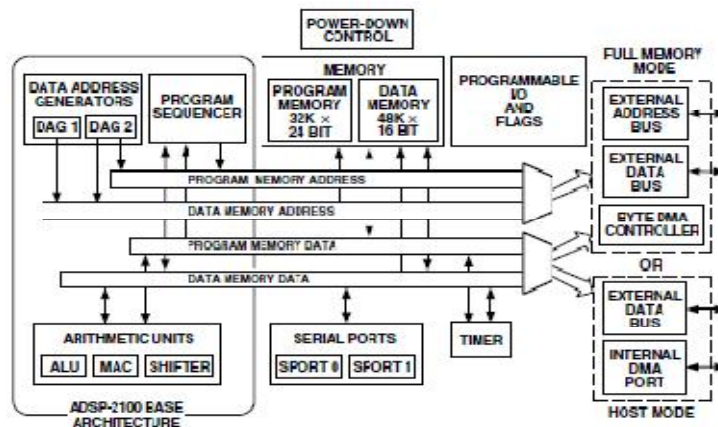


## Familije mikrokontrolera

- Intel 8051
- Microchip PIC
- MSP430

## DSP

- DSP – digital signal processor
- Pogodni za poslove sa puno složenih proračuna
- Mnoge obrade analognih signala se danas vrši u digitalnoj formi.



## Familije DSP

- Analog Devices – ADSP 21xx
- Texas Instruments – TMS320
- Microchip - dsPIC

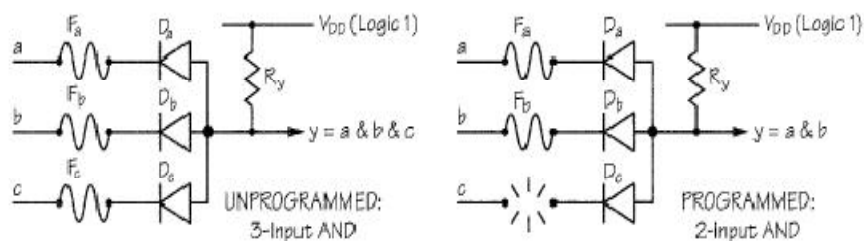
## VLSI kola sa hardverskim programiranjem

Tipovi:

- prost PLD
- CPLD
- FPGA

## Principi hardverskog programiranja

- Kola sa hardverskim programiranjem su poluproizvodi koji se finaliziraju prilikom ugradnje u neki krajnji proizvod.
- Fabrika poluprovodnika realizuje veliki broj malih digitalnih podsklopova u okviru jednog integrisanog kola.
- Unutrašnja struktura i međusobne veze podsklopova se mogu naknadno programirati kod korisnika.
- Programiranje se sastoji od uspostavljanja ili prekidanja određenih veza.



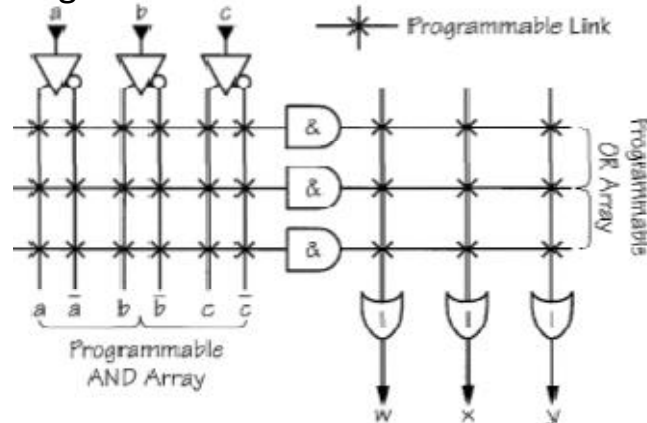
## Prost PLD

- **PLA** – programmable logic
- **PAL** – programmable array logic.
- **ROM** – read only memory.

Smatraju se zastarelim komponentama.

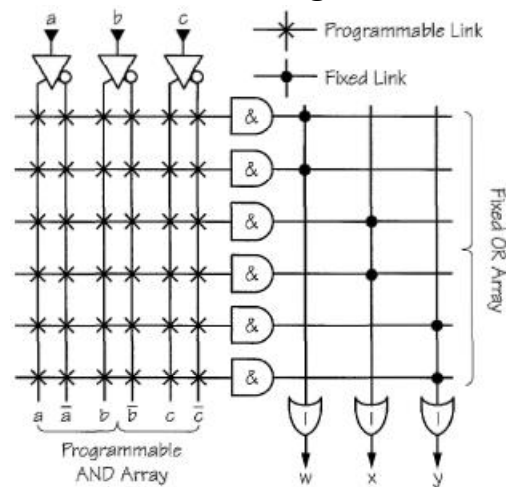
## PLA

- Programmable logic array – sadrži veliki broj I i ILI kola, omogućava realizaciju dvostepene kombinacione mreže gde se može birati koji ulazni signali se dovodi na koje I kolo i na koje ulaze ILI kola se povezuju izlazi određenog I kola.



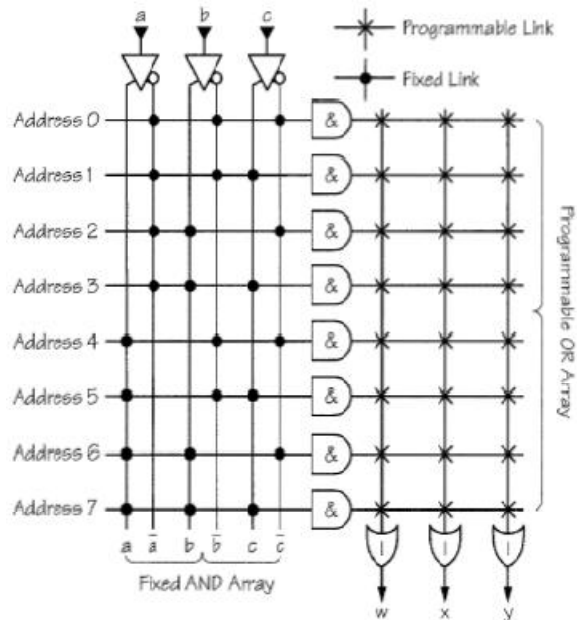
## PAL

- Programmable array logic – isto se može koristiti za realizaciju dvostepene I/ILI mreže, ILI kola su fiksno povezana, mogu se birati samo signali na ulazu I kola.
- Slična komponenta se zove **GAL** – moguće je višestruko programiranje zahvaljujući EEPROM tehnologiji za formiranje veza.



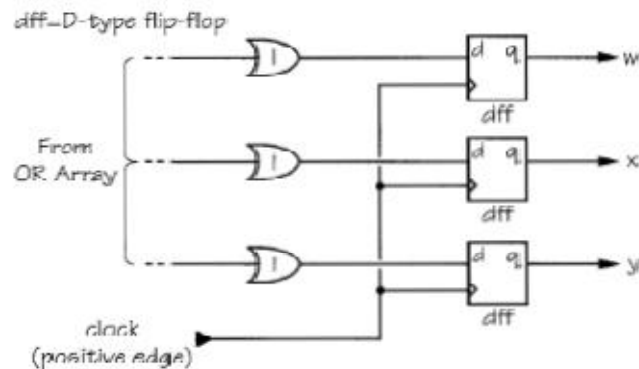
## ROM

- **ROM** – read only memory – obrađena ranije kod memorijskih kola – takođe se može koristiti za realizaciju velikih kombinacionih mreža, programabilna je samo ILI mreža.



## Prosti PLD-ovi sa registarskim izlazom

- Na izlaz kombinacijskih kola se postavlja po jedan flip-flop.
- Takva struktura omogućava realizaciju sekvencijalnih kola.

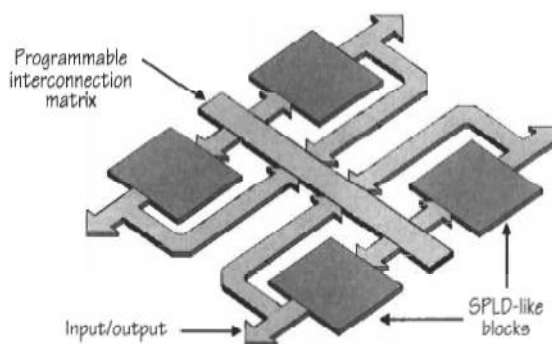


## Složeni PLD

- CPLD
- FPGA

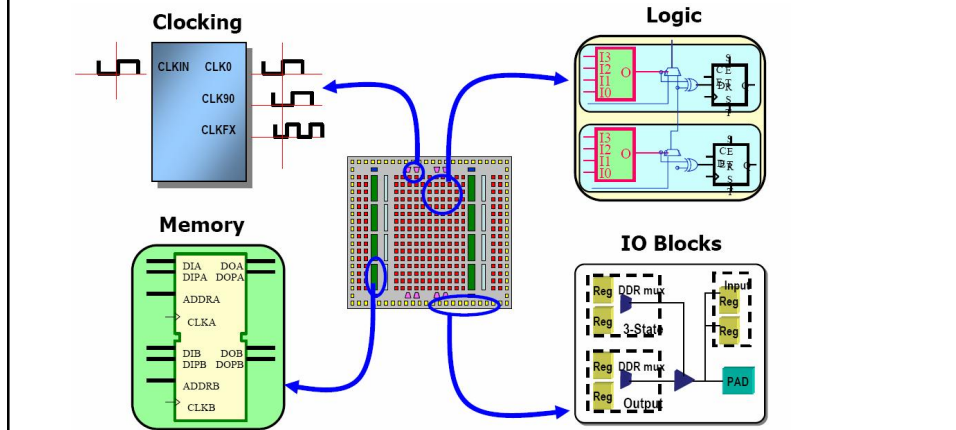
## CPLD

- Complex PLD
- Naslednici GAL kola – programiranje se vrši EEPROM tehnologijom.
- Mnogo složenija struktura, sadrži mali broj velikih kombinacionih blokova i mali broj flip-floпова.
- Sve manje se primenjuju.



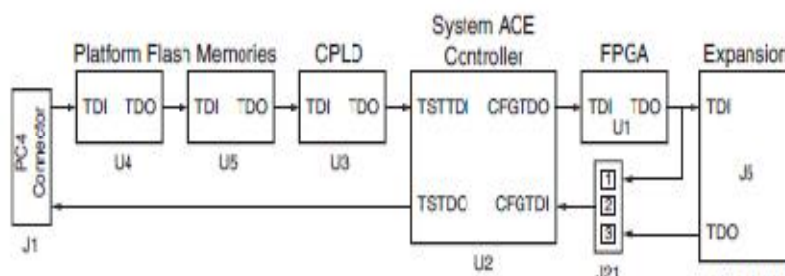
## FPGA

- Field programmable gate array
- Puno malih kombinacionih kola i veliki broj flip-flopova
- Struktura vodova za povezivanje je složena
- Postižu se velike brzine i paralelna struktura
- Postaju sve popularniji u realizaciji složenih digitalnih sistema



## Razvoj proizvoda na bazi CPLD i FPGA

- Opis zadatka se vrši uglavnom pomoću hardverskog jezika.
- Softverska sinteza i optimizacija logičke mreže.
- Programiranje preko JTAG konektora.
- Koristi se boundary scan standard.



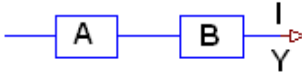
## Fizičke osobine digitalnih kola

Opšte teme koje se odnose podjednako na SSI, MSI, LSI i VLSI digitalna integrisana kola.

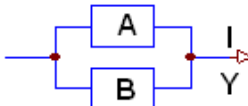
- Strujna logika - naponska logika
- Fizičke karakteristike
- Posledice kašnjenja: hazardi
- Tehnologije izrade digitalnih integrisanih kola

## Strujna logika

- Strujna logika je bila karakteristična **na počecima** digitalne elektronike (upravljanje mehaničkim prekidačima i elektromagnetnim prekidačima).
- **Prekidači**: dva stanja (provodi - 1, ne provodi - 0).
- Ostvarivanje **logičkih funkcija** kombinacijom prekidača.

- I funkcija
 

$$Y=A \cdot B$$

- Ili funkcija
 

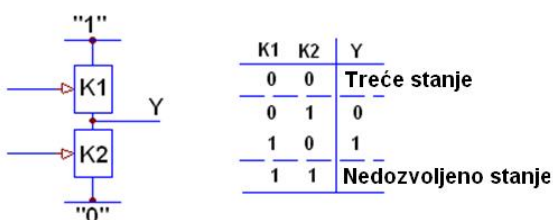
$$Y=A+B$$

- I relativno **složeni zadaci automatizacije** su rešavani na ovaj način, ponekad se primenjuje čak i danas.



## Naponska logika

- Kod savremenih digitalnih kola uglavnom se primenjuje naponska logika.
- **Formiraju se određeni naponski nivoi** korišćenjem izvora napona i elektronskih prekidača (tranzistori).



- Pored **dva dobro definisana naponska nivoa (0, 1)** ponekad se koristi i treće (neodređeno) stanje.

145

## Prekidači u logičkim (digitalnim) kolima

- Bipolarni tranzistori
- MOSFET-ovi
- **Konačna vremena prebacivanja** - kašnjenja u promeni logičkih nivoa
- **Konačne otpornosti** u uključenom i isključenom stanju - dolazi do pomeranja logičkih nivoa.

146

## Prenosna karakteristika

**Idealna** karakteristika:

- idealni prekidači koji rade u protivfazi.

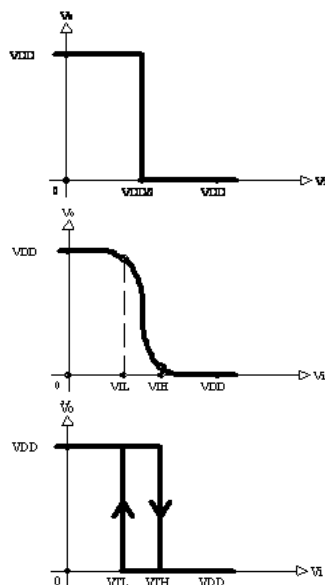
**Realna** karakteristika:

- realni prekidači, postepeni prelaz.

Karakteristika sa

**histerezisom:**

- dvoznačna zavisnost, skokoviti prelaz.

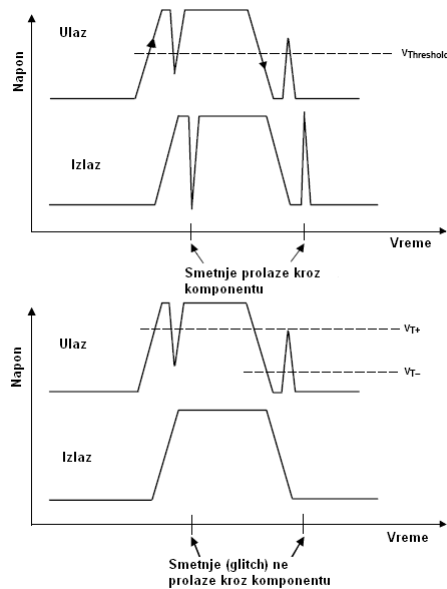


147

## Uticaj histereze

1. Bez histereze **smetnje** sa ulaza **prenose se na izlaz**, ponekad čak pojačano.

2. Primenom histereze, smetnje do određenog nivoa (pragovi) **ne utiču na izlazni signal**.



148

## Logički nivoi

- Izlazni logički nivoi:  $V_{OL}$ ,  $V_{OH}$ .
- Ulazni logički nivoi:  $V_{IL}$ ,  $V_{IH}$ .
- **Pravilni odnosi** između ulaznih i izlaznih nivoa:  
 $V_{OL} < V_{IL}$ ,  $V_{OH} > V_{IH}$ .
- **Kaskadna veza**: izlaz jednog kola upravlja ulazom drugog kola.
- Logički nivoi u nekoj meri zavise od napona napajanja, opterećenja i temperature.
- Manja odstupanja postoje i između različitih primeraka kola istog tipa, čak i pod istim uslovima.

149

## Margine smetnji

- Na istoj skali se predstavljaju **ulazni i izlazni logički nivoi** (opsezi nivoa):



- U prikazanom slučaju **dijagram je simetričan** u odnosu na tačku  $V_{DD}/2$ , što ne mora biti slučaj kod svakog kola.
- Margina smetnji za nizak logički nivo (0):

$$NM_0 = V_{ILMAX} - V_{OLMAX}$$

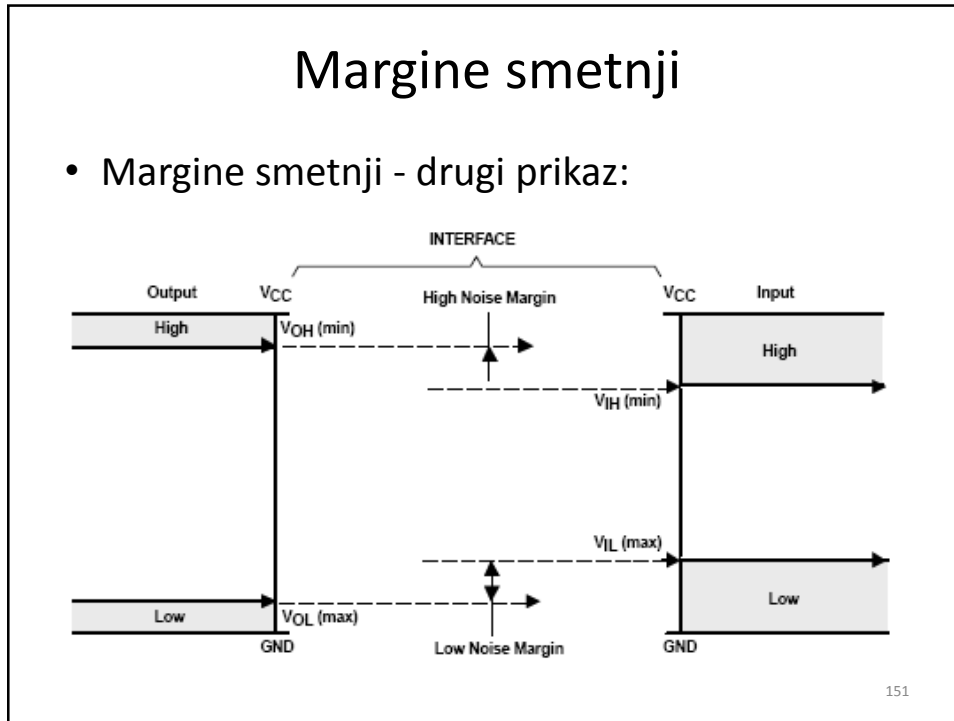
- Margina smetnji za visoki logički nivo (1):

$$NM_1 = V_{OHMIN} - V_{IHMIN}$$

150

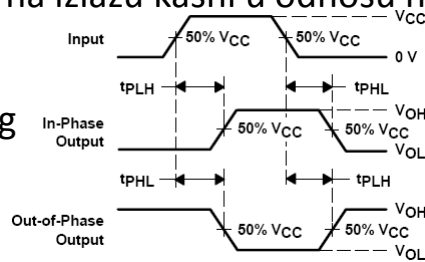
## Margine smetnji

- Margine smetnji - drugi prikaz:



## Kašnjenja digitalnih kola

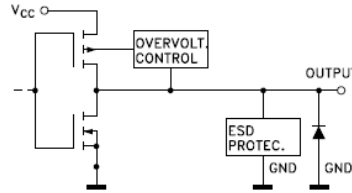
- Za promenu stanja prekidača (tranzistori) **potrebno je izvesno vreme.**
- Formiranje logičkog nivoa na izlazu kasni u odnosu na skok ulaznog napona.
- Umesto konkretnih vrednosti zadaju se opsezi zbog **varijacija parametara.**
- Kašnjenja se javljaju i na **linijama prenosa.**
- Pri proučavanju uticaja smetnji treba uzimati najnepovoljniji slučaj (najveće kašnjenje).



PARAMETER	FROM (INPUT)	TO (OUTPUT)	LOAD CAPACITANCE	T <sub>A</sub> = 25°C		SN54AHC08		SN74AHC08		UNIT
				MIN	TYP	MIN	MAX	MIN	MAX	
t <sub>PLH</sub>	A or B	Y	C <sub>L</sub> = 15 pF	4.3*	5.9*	1*	7*	1	7	ns
t <sub>PHL</sub>				4.3*	5.9*	1*	7*	1	7	
t <sub>PLH</sub>	A or B	Y	C <sub>L</sub> = 50 pF	5.8	7.9	1	9	1	9	157 ns
t <sub>PHL</sub>				5.8	7.9	1	9	1	9	

## Opteretljivost izlaza

- Izlazni signal se redovno formira pomoću dva tranzistora.
- Pad napona** na tranzistorima **zavisi od veličine i smera struje**.
- Pri preopterećenju poremete se logički nivoi.
- Treba računati i na **povećanje kašnjenja**.
- Opteretljivost se može zadati preko broja standardnih ulaza koji se može vezati na dati izlaz (**fan out**) (uvek  $\geq 1$ ), ili preko konkretnih vrednosti struja  $\downarrow$ .



PARAMETER	TEST CONDITIONS	V <sub>CC</sub>	T <sub>A</sub> = 25°C			SN54AHC08		SN74AHC08		UNIT
			MIN	TYP	MAX	MIN	MAX	MIN	MAX	
V <sub>OH</sub>	I <sub>OH</sub> = -50 μA	2 V	1.9	2		1.9	1.9		V	
		3 V	2.9	3		2.9	2.9			
		4.5 V	4.4	4.5		4.4	4.4			
	I <sub>OH</sub> = -4 mA	3 V	2.58		2.48	2.48				
V <sub>OL</sub>	I <sub>OL</sub> = 50 μA	2 V			0.1		0.1	0.1	V	
		3 V			0.1		0.1	0.1		
		4.5 V			0.1		0.1	0.1		
	I <sub>OL</sub> = 4 mA	3 V			0.36		0.5	0.44		
	I <sub>OL</sub> = 8 mA	4.5 V			0.36		0.5	0.44		

## Potrošnja digitalnih kola

- Za regularan rad kola potrebna je **struja iz izvora napajanja** (V<sub>CC</sub> ili V<sub>DD</sub>).
- Statički gubici** (struje polarizacije) - redovno su male vrednosti.
- Uzroci **dinamičkih gubitaka**:
  - preklapanje između vremena provođenja gornjeg i donjeg tranzistora,
  - kapacitivno opterećenje na izlazu (sopstvena i spolja priključena kapacitivnost)
- Dinamički gubici se zadaju preko **ekvivalentnog kapacitivnog opterećenja** (C<sub>PD</sub>):  $I_{CCDYN} = V_{CC} * C_{PD} * f$ .

Symbol	Parameter	Test Condition	Value						Unit	
			T <sub>A</sub> = 25°C			-40 to 85°C		-55 to 125°C		
			Min.	Typ.	Max.	Min.	Max.	Min.		Max.
C <sub>IN</sub>	Input Capacitance	V <sub>CC</sub> (V)	5.0	5	10		10		10	pF
C <sub>PD</sub>	Power Dissipation Capacitance	V <sub>CC</sub> (V)	5.0	40						pF

154

## Temperaturni opsezi

- Opsezi: lagerovanje (storage), radni (operating)
- Prema području primene: komercijalni (commercial), industrijski (industrial), vojni (military)

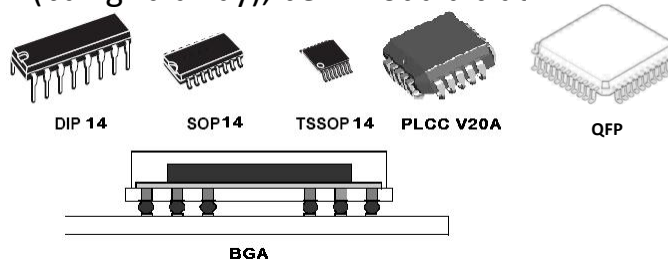
Temperatura	Komercijalni	Industrijski	Vojni
Lagerovanje	-65...+150°C	-65...+150°C	-65...+150°C
Radni opseg	0...70°C	-25...85°C	-55...+150°C

- Primer iz kataloga:

T <sub>stg</sub>	Storage Temperature	-65 to +150	°C
T <sub>op</sub>	Operating Temperature	-55 to 125	°C

## Kućišta

- **DIL** ili **DIP** (dual in-line package) - najstariji, razmak nožica (izvoda) u redu je 1/10 inča, između redova 3/10 inča.
- **SOP** (small outline package), razmak nožica 1/20 inča.
- **TSSOP** (thin shrink small outline package), izvodi su još gušće postavljeni ( $\leq 0,65\text{mm}$ ).
- **PLCC** (plastic chip carrier), izvodi sa četiri strane
- **QFP** (quad flat pack), izvodi sa četiri strane
- **BGA** (ball grid array), bez izvoda u tradicionalnom smislu.



156

## Posledice kašnjenja: hazardi

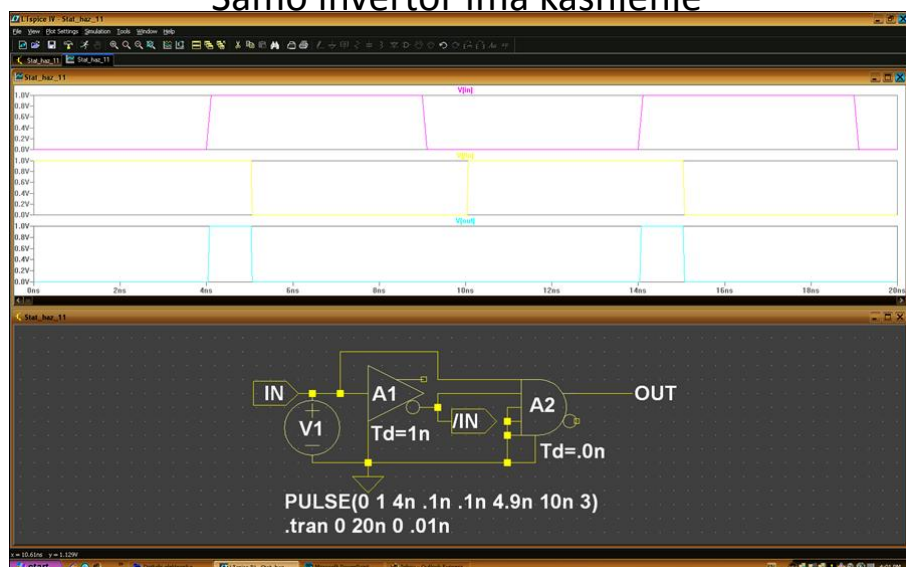
- Zbog neizbežnih kašnjenja, u nekim kratkim intervalima (privremeno) izlazi logičkih kola ne daju logičke nivoe koji slede iz realizovane logike.
- **Pogrešni logički nivoi mogu da dovedu do pogrešnih reagovanja** u preostalom delu kola.
- Greška može biti **prolazna** ili **trajna**.
- **Primer** (sledeći slajd): izlaz / kola, prema datoj logici, uvek bi trebao da bude na logičkoj nuli, ipak se kratkotrajno **pojavljuje logička jedinica**.

$$Y = X \cdot \overline{X} = 0$$

157

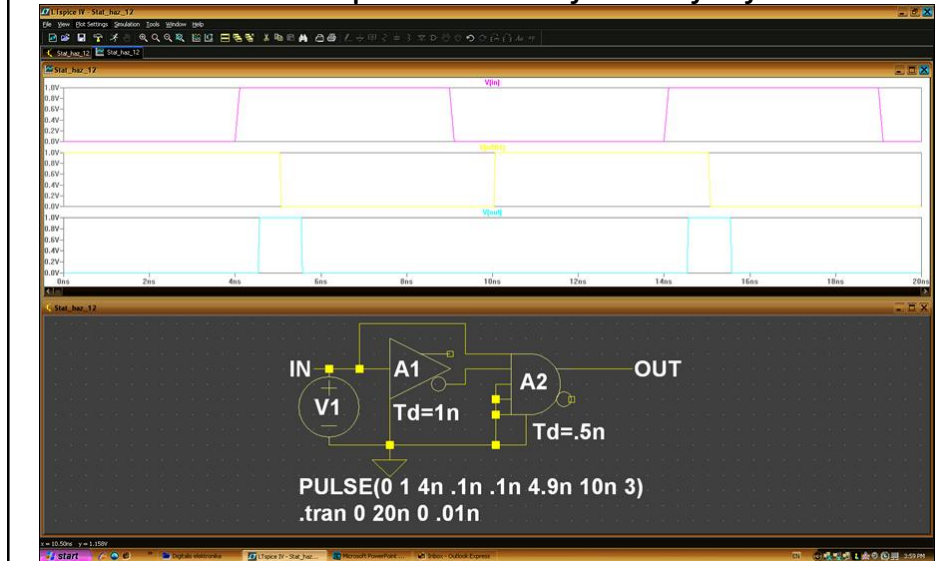
## Hazardno ponašanje usled kašnjenja u jednom prostom kolu

Samo invertor ima kašnjenje



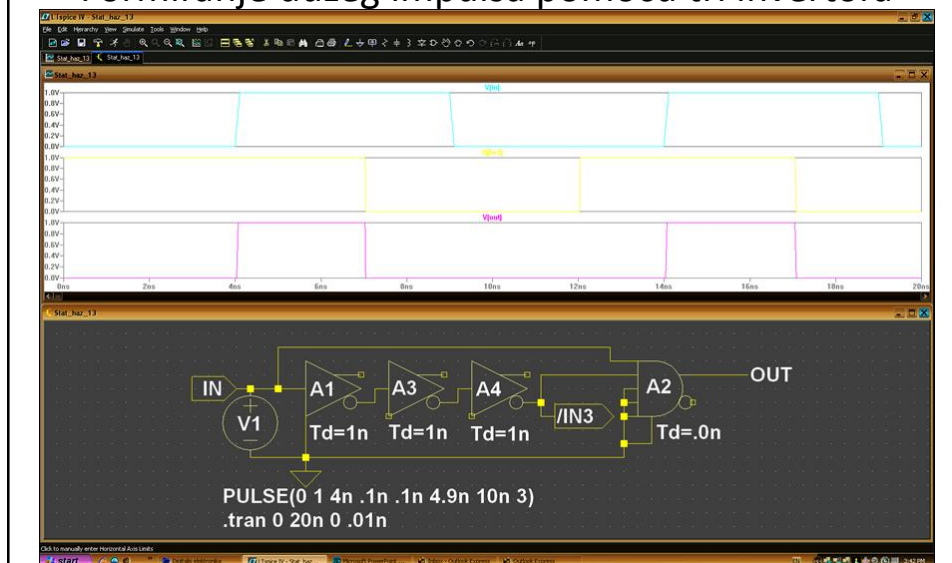
## Hazardno ponašanje usled kašnjenja u jednom prostom kolu

Obe komponente imaju kašnjenje



## Hazardno ponašanje usled kašnjenja u jednom prostom kolu

Formiranje dužeg impulsa pomoću tri invertora



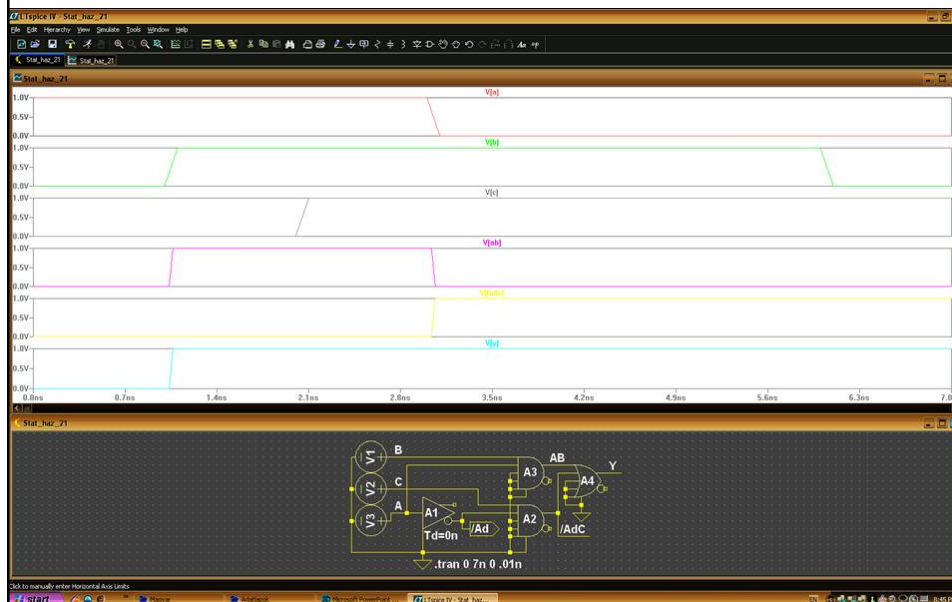


## Statički hazard

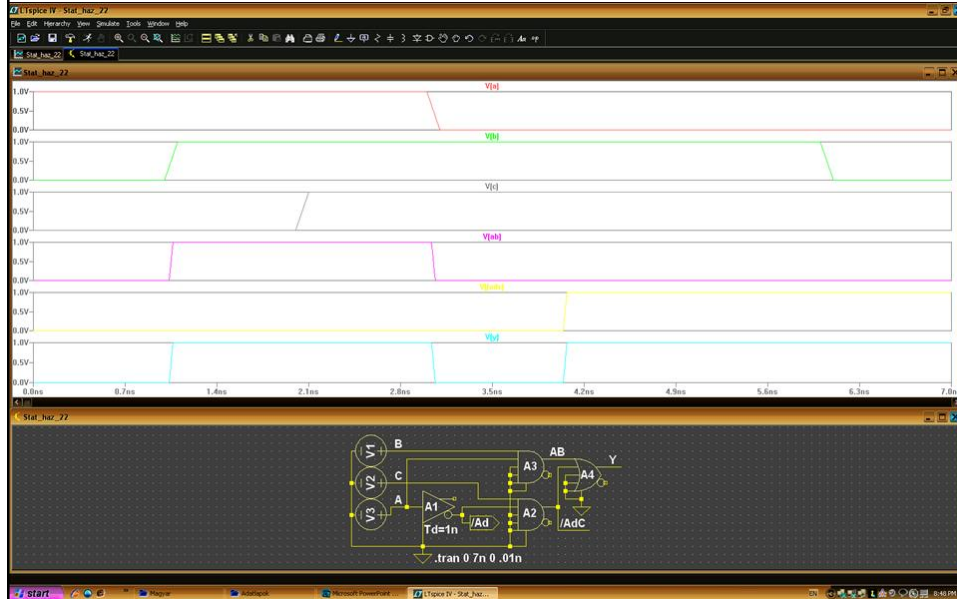
- Pod statički hazardom podrazumevamo situaciju kada, umesto da stoji na fiksnom logičkom nivou, logički signal **kratkotrajno uzima suprotnu vrednost (glitch)**.
- Prethodni primeri su bili statički hazardi.
- Na sledećim slajdovima analiziraće se **statički hazard prisutan u složenijem kolu**.

161

## Nema kašnjenja - nema hazarda



## Nastupa hazard zbog kašnjenja invertora



## Eliminacija statičkog hazarda

Pošto u jednom kratkom intervalu na oba ulaza ILI kola dolazi logička nula, i izlaz će da padne na nulu.

**Dodavanjem jednog redundantnog bloka** (još jedan logički proizvod) **ne menja se logička funkcija** ali se **eliminiše hazard**. Nije uvek minimizirano kolo ujedno i optimalno kolo!

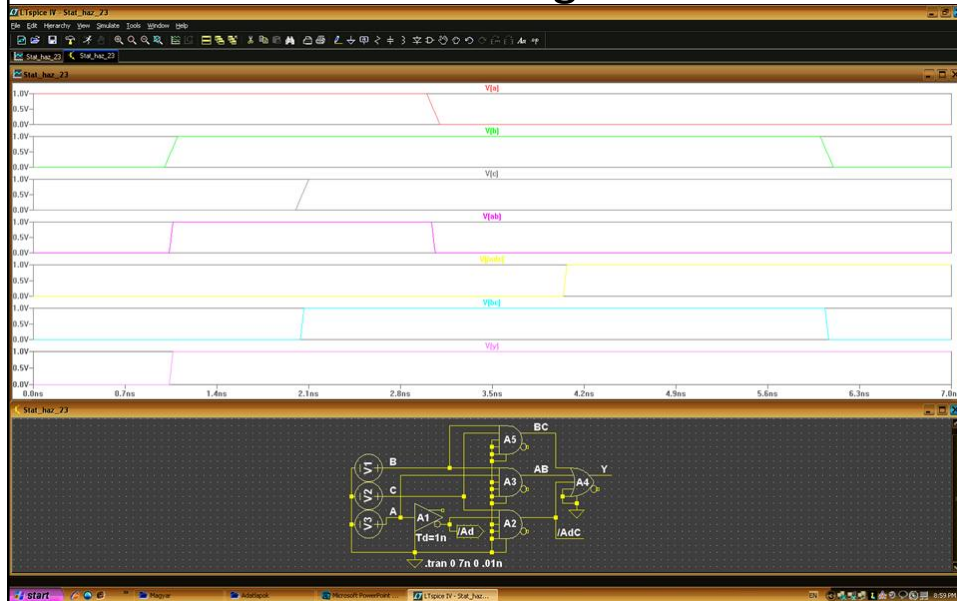
A\BC	00	01	11	10
0	0	1	1	0
1	0	0	1	1

/AC  
AB

A\BC	00	01	11	10
0	0	1	1	0
1	0	0	1	1

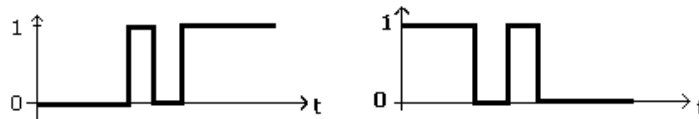
BC

## Eliminacija hazarda primenom redundantnog bloka



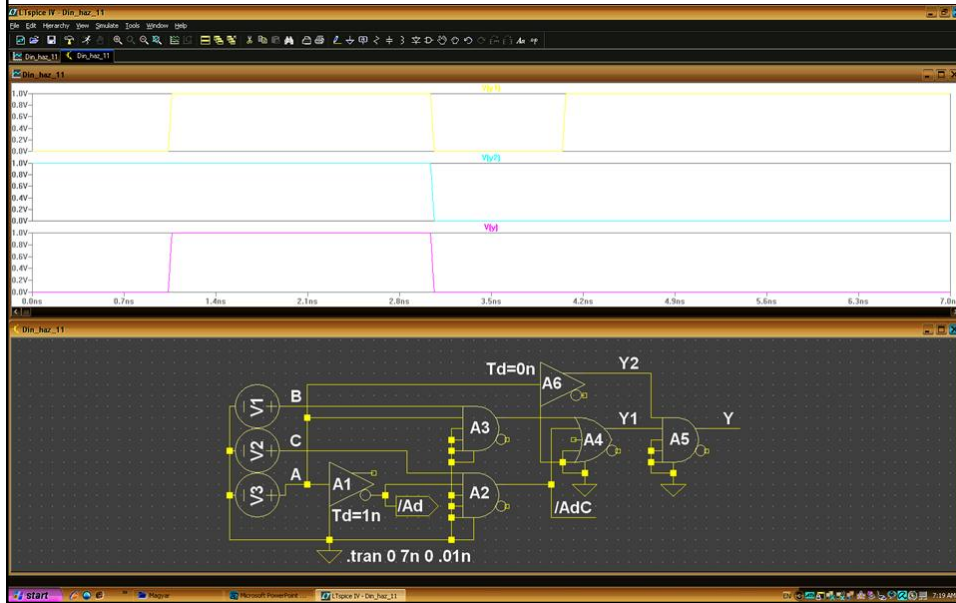
## Dinamički hazard

- Pojavljuje se u situacijama kada se **menja logička vrednost** na izlazu digitalnog kola (u skladu sa funkcijom koja se realizuje).
- Ako se promena nivoa ne odigrava pravilno, već se javlja **višestruki skok** gore-dole, reč je o dinamičkom hazardu.

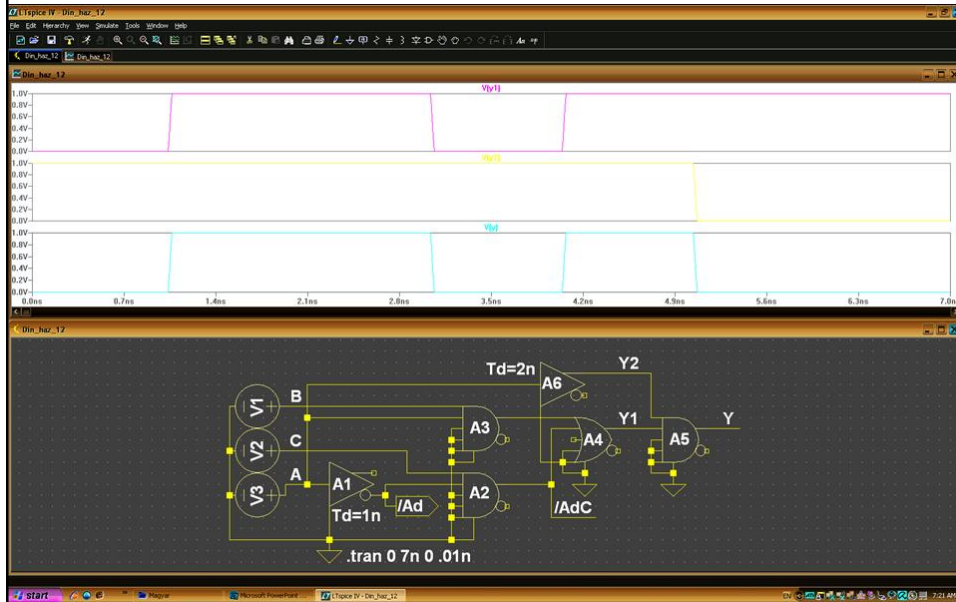


- Primer: Ranije prikazano kolo sa statičkim hazardom, dopunjeno sa I kolom i sa jednim kašnjenjem. 166

Dinamički hazard - Ako A6 nema kašnjenje, izlaz Y odmah skače na konačan logički nivo, bez obzira na statički hazard kod Y1



Dinamički hazard - Zbog kašnjenja kod A6 i statičkog hazarda kod Y1, izlaz Y skače gore-dole pre nego što nastaje konačan logički nivo



## Funkcionalni hazard

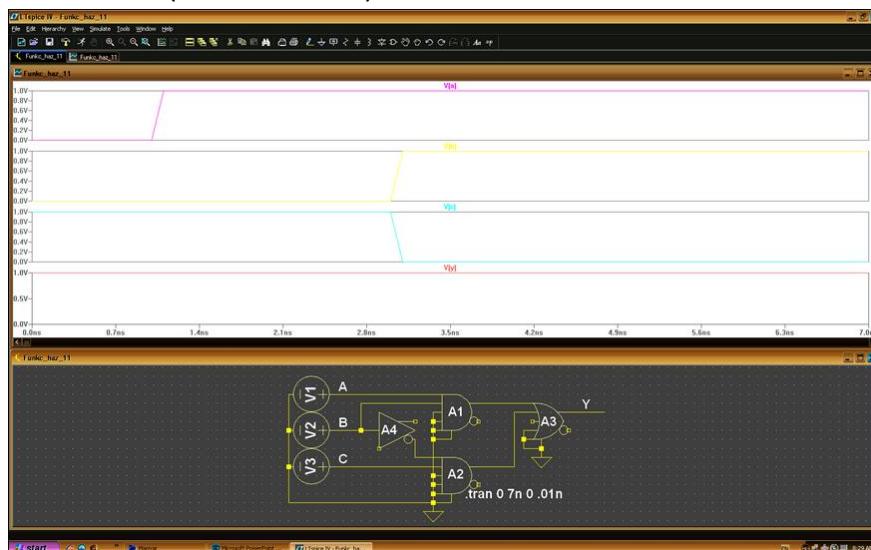
- **Više ulaza** se menja približno istovremeno
- **Izlaz zavisi od redosleda promena** - javljaju se kratkotrajni skokovi (glitch).
- Moguća rešenja:
  - uvođenje namernih kašnjenja,
  - sinhronizacija ulaza.

169

## Funkcionalni hazard

- ABC: 101 → 110
- B i C se menjaju istovremeno
- Y: 1 → 1 (nema hazarda)

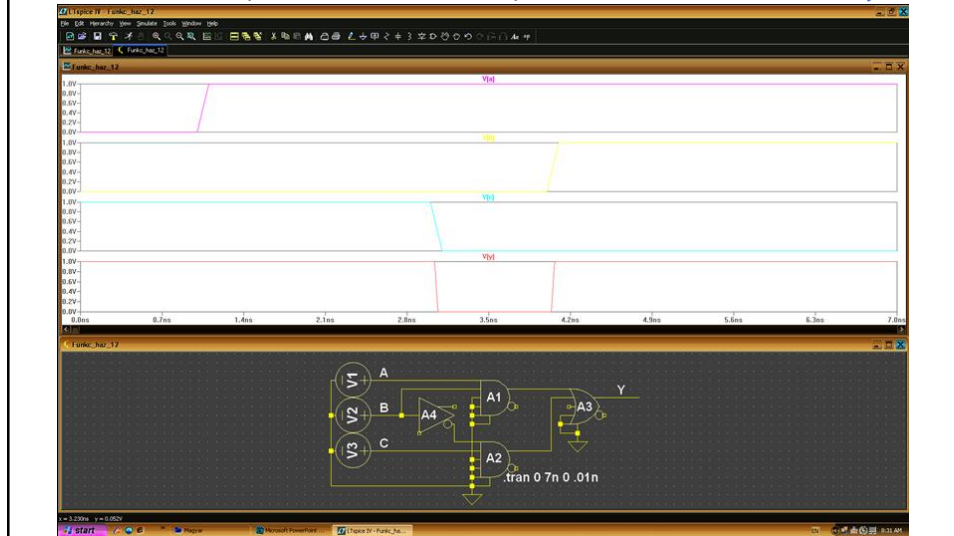
A\BC	00	01	11	10
0	0	1	0	0
1	0	1	1	1



## Funkcionalni hazard

- ABC: 101→100→110
- **prvo se menja C, zatim B**
- Y: 1→0→1 (funkcionalni hazard)

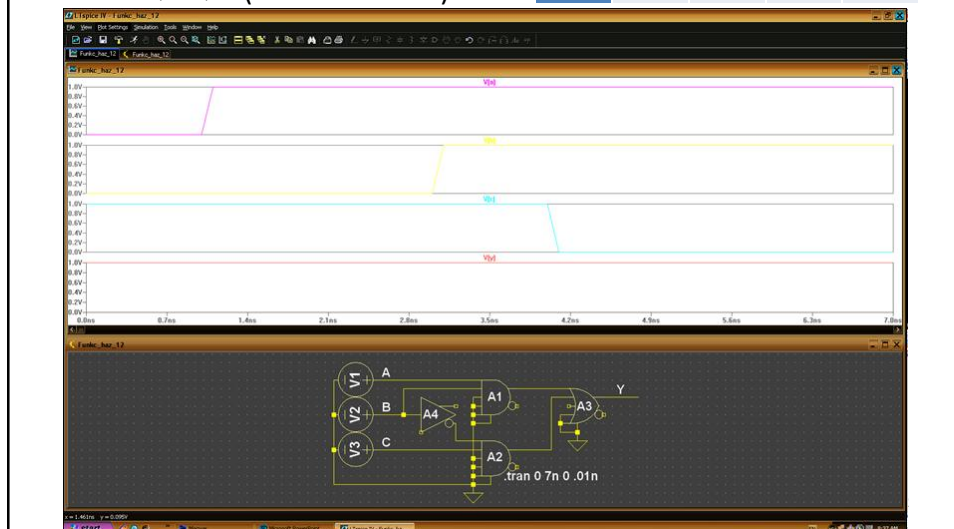
A\BC	00	01	11	10
0	0	1	0	0
1	0	1	1	1



## Funkcionalni hazard

- ABC: 101→111→110
- **prvo se menja B, zatim C**
- Y: 1→1→1 (nema hazarda)

A\BC	00	01	11	10
0	0	1	0	0
1	0	1	1	1



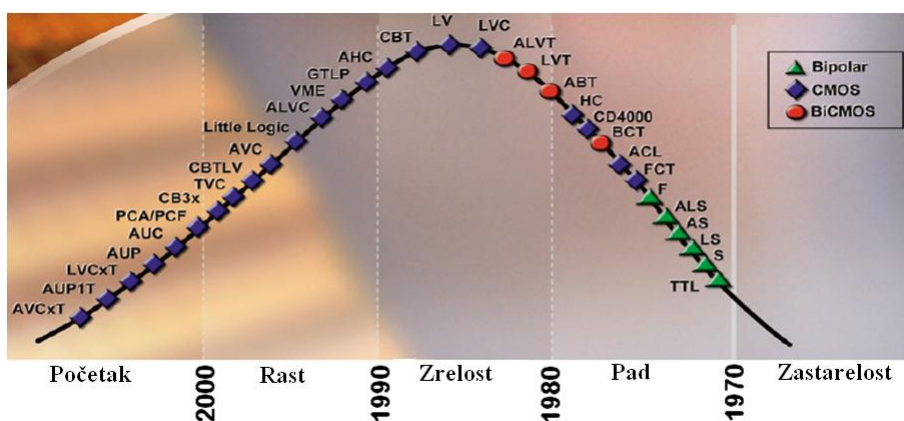
## Tehnologije izrade integriranih kola

- Iste logičke funkcije
- Različite tehnologije (primenjene komponente i postupci)
- Osnovne tehnologije: **bipolarna, CMOS i mešovita (BiCMOS)**
- Moguće je **optimizirati** određene parametre
- Pravci razvoja:
  - **smanjenje kašnjenja** - redovno povlači za sobom povećanje gubitaka,
  - **smanjenje napona napajanja** - redovno se kvare margine smetnji.

173

## Popularnost i životni ciklus familija logičkih kola

Dijagram popularnosti kola proizvodnje **Texas Instruments**

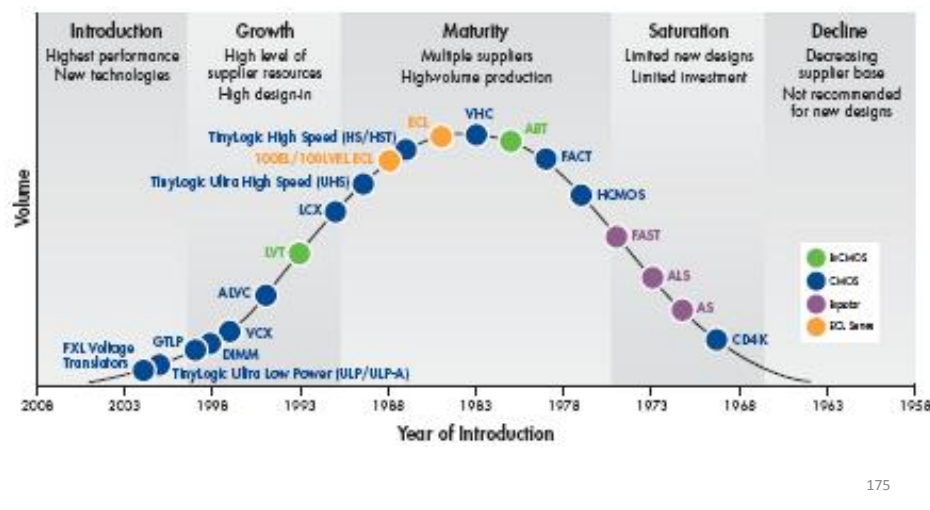


**Na desnoj strani** - zastarela kola, samo za potrebe servisiranja,  
**Na sredini** - kola razvijena osamdesetih godina, danas su najviše primenjena,  
**Levo** - familije kola koja ovih godina osvajaju tržište.

174

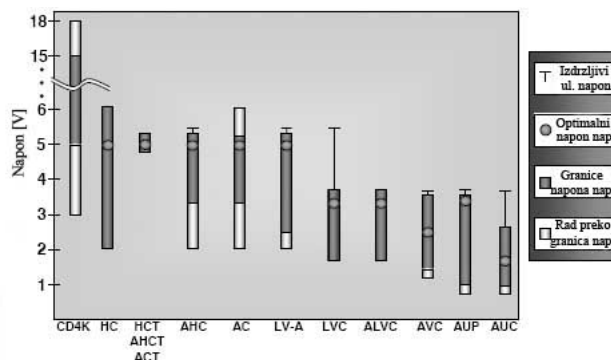
## Popularnost i životni ciklus familija logičkih kola

Dijagram popularnosti logičkih kola firme **Fairchild** (ta firma je proizvela prva integrisana kola pre pedeset godina).



## Podela prema naponu napajanja

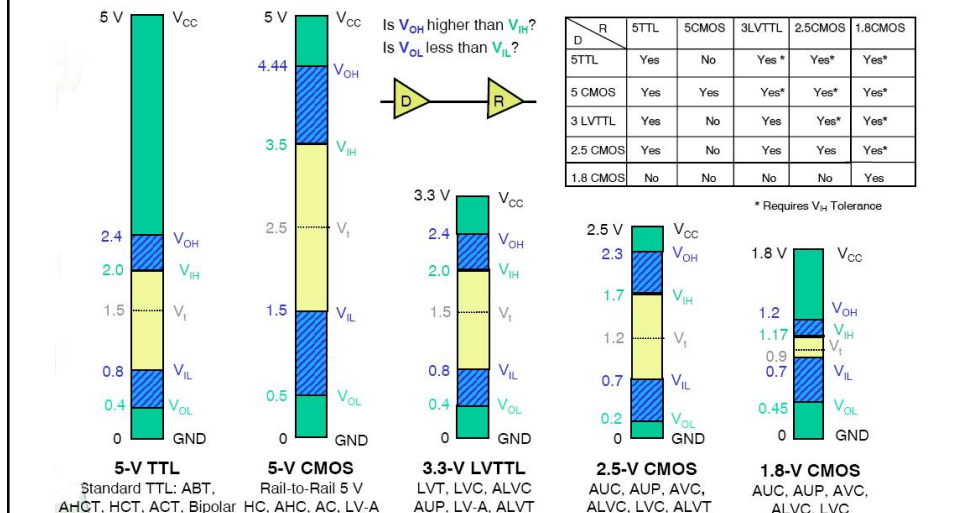
- Neke familije logičkih kola su projektovane za **fiksne vrednosti napona napajanja**, druge za **širi opseg**.
- Optimalna vrednost, nazivna vrednost, tolerancija u odnosu na napon, rad van radnog opsega (sa promenjenim parametrima).





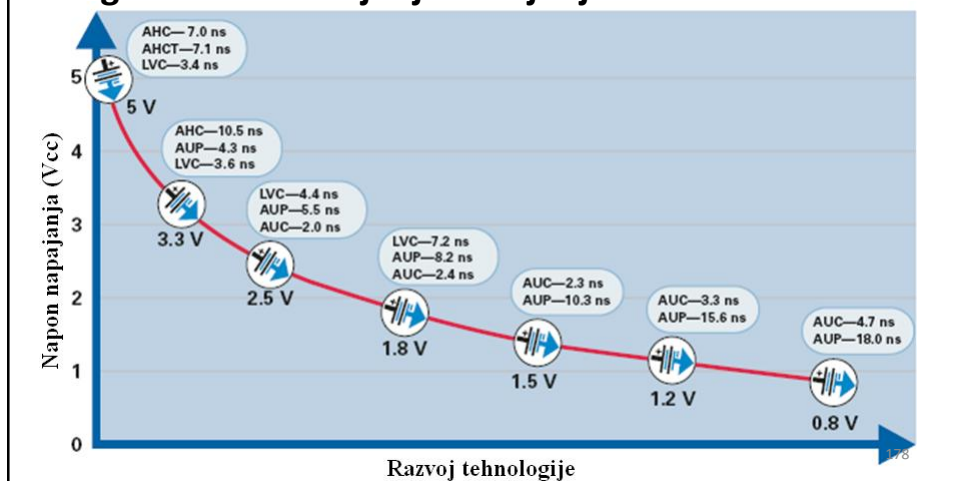
## Kompatibilnost logičkih nivoa

- Unutar jedne familije kola su uvek kompatibilna (mogu se vezati ulaz(i) na izlaz).
- Koja kola sa čime se mogu pobuditi (u slučaju različitih familija)?



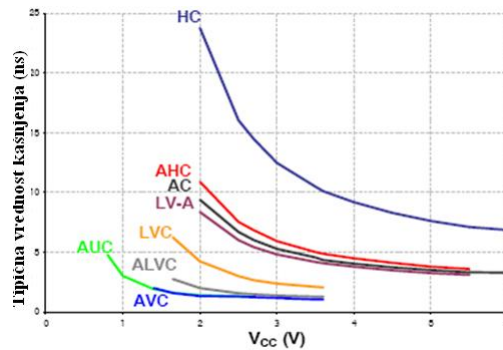
## Zavisnost kašnjenja od napona napajanja

- Texas Instruments Logic Selection Guide 2007.
- Cilj smanjenja napona napajanja je smanjenje gubitaka i smanjenje kašnjenja.



## Zavisnost kašnjenja od napona napajanja

- Pojedine familije logičkih kola se ne projektuju za fiksni napon napajanja već za neki opseg napona.
- Unutar iste familije, **pri smanjenju napona napajanja rastu kašnjenja.**



179

## Raspoloživi izbor logičkih funkcija u pojedinim familijama logičkih kola

### Primer 1: Kola za sprezanje proizvodnje Texas Instruments

Buffers, Drivers and Transceivers

✓ Product available in technology indicated    + New product planned in technology indicated.  
 OP = open pin    OC = open collector    OD = open drain    PP = push-pull    TP = tri-state    SS = 3-state

DESCRIPTION	OUTPUT	TYPE	TECHNOLOGY																							
			80	AC	ACT	ANC	ANCT	ALB	ALVC	ALVT	8	AC	AMP	ACT	CMR	P	PP	OC	OD	OP	PP	TP	SS	8	TL	
Single Buffers/Drivers	OD	1006																								
	SS	10240																								
Single Inverters		1004				✓	✓																			
Single Schmitt-Trigger Inverters		1014				✓	✓																			
Unbuffered Single Inverters		10L04				✓																				
		2006																								
Dual Buffers/Drivers	SS	20240																								
		2004																								
Dual Inverters		2004																								
		20L04																								
Dual Schmitt-Trigger Inverters		2014																								
Triple Buffers/Drivers	OD	3006																								
		3004																								
Triple Inverters		3004																								
		3014																								
Unbuffered Triple Inverters		30L04																								
		3004																								
Hex Buffers/Drivers	OC	06																								
	OD	06																								
	OC	16																								
	SS	306																								
		306																								
Hex Buffers/Converters		1005																								
		4000																								
Hex Buffers/Converters		4049																								
		1004																								



## Digitalne komunikacije

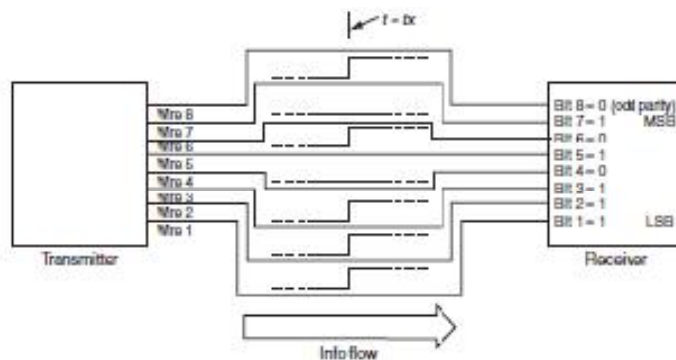
Na početku digitalna elektronika je korišćena u računarskoj i regulacionoj tehnici. Danas je značajna primena u komunikacione svrhe.

Komunikacija može biti:

- Paralelna
- Serijska

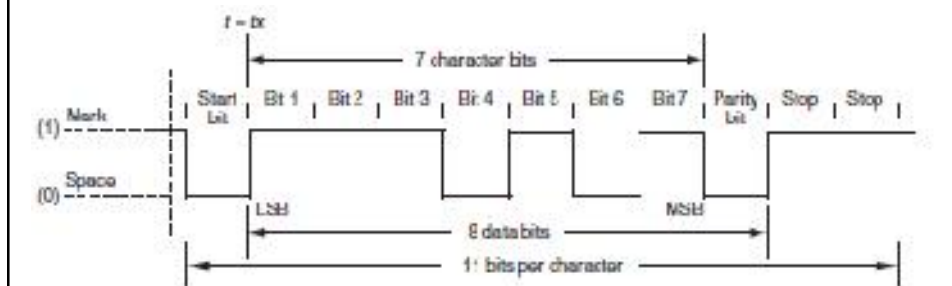
## Paralelna komunikacija

- Bitovi koji treba da budu prenešeni se pojavljuju na prenosnim linijama istovremeno.
- Postiže se velika brzina.
- Potreban je veliki hardver, zato je skup.
- Koristi se samo na malim udaljenostima, na pr. između procesora i memorije.



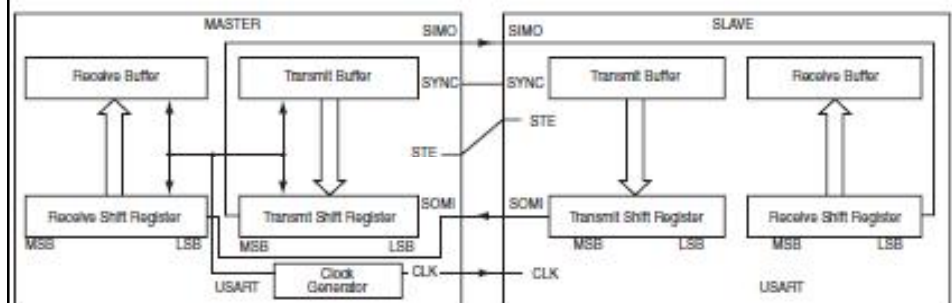
## Serijska komunikacija

- Koristi samo jednu liniju za prenos (to može da znači jednu paricu – dve žice, ili jedan radio link ili...).
- Bitovi idu jedan za drugim u vremenu.
- Ostvaruje se sporiji prenos.
- Može da se koristi i za velika rastojanja i za mala.
- Manji hardver – jeftinija izvedba u odnosu na paralelni prenos.



## Sinhrona serijska komunikacija

- Pored podataka prosleđujemo i takt signal.

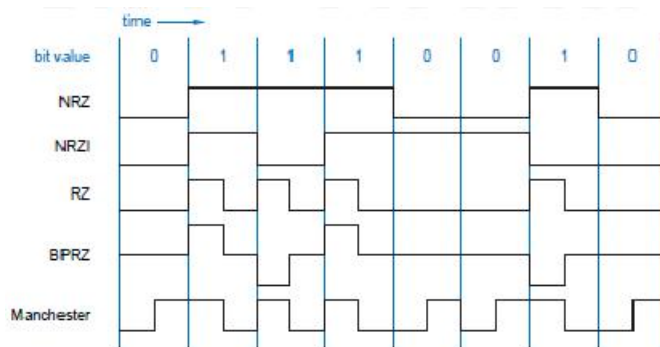


## Asinhrona serijska komunikacija

- Takt se ne prenosi od izvora informacije prema korisniku.
- Prenešeni signal ipak posredno sadrži informaciju o taktu.
- Za detekciju bitova potrebno je rekonstruisati takt – da bi znali u kom momentu treba da gledamo primljeni signal.
- Potrebno je ubaciti dodatne informacije (start bit, stop bit, bit parnosti) da bi znali gde počinje byte i da se vrši detekcija greške.
- Ako se vrši prenos na daljinu, na prijemu je prvo potrebno pojačanje i filtracija signala, posle se detektuju bitovi.

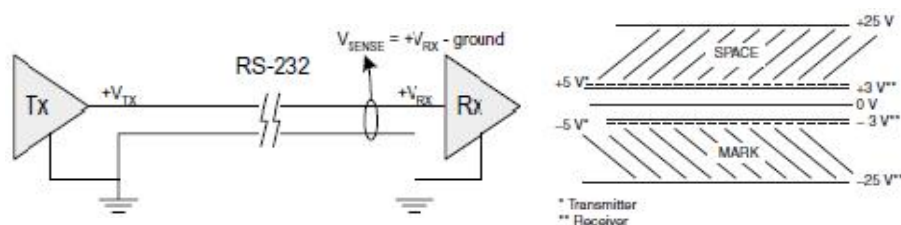
## Kodiranje

- Pri serijskoj komunikaciji podaci se redovno ne prenose u osnovnoj formi – primenjuje se kodiranje.
- Kodiranje karaktera: karakteri se zamenjuju sa serijom bitova.
- Prenosni kodovi: kako predstavljamo nule i jedinice u kodu.



### Standardi za serijski prenos na daljinu – RS-232

- Pri slanju formiraju se pozitivni i negativni naponski nivoi.
- Prenosi se signalni vod i masa (tzv. asimetrični prenos)
- Naponski nivoi se slabe pri prenosu. Za detekciju logičkog nivoa na prijemu koriste se pragovi bliže nuli.



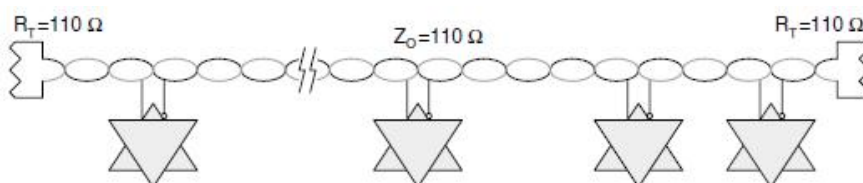
### Standardi za serijski prenos na daljinu – RS-422

- Koriste se jedna parica za prenos u jednom smeru.
- Slanje jedne logičke vrednosti se vrši primenom pozitivnog signala na jedan signalni vod i negativnog signala na drugi (simetrični prenos).
- Na prijemu se detektuje polaritet razlike dva signala.
- Za izbegavanje refleksije kraj voda treba zatvoriti sa otpornikom jednakim talasnoj impedansi kablova.



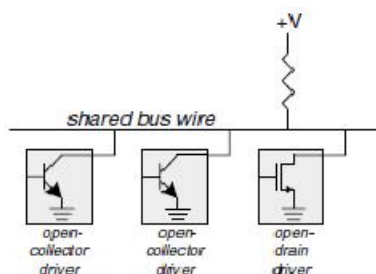
### Standardi za serijski prenos na daljinu – RS-485

- Koristi simetričan prenos preko parice kao RS-422.
- Moguće je spajanje više prijemnika i odašiljača.
- Definisan je protokol po kome se prijemnici i odašiljači dogovore ko će u datom intervalu koristiti zajednički prenosni put.
- Danas je ovaj standard najviše rasprostranjen za internet i industrijske komunikacije.



### Standardi za serijski prenos na mala rastojanja – I<sup>2</sup>C

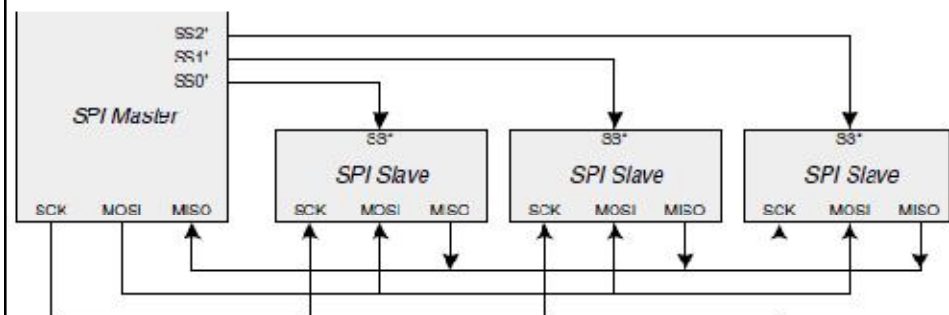
- Uveo Philips za komunikaciju između mikrokontrolera i njegovih periferija.
- Uređaji (integrisana kola) se prave sa open drain izlazom.
- Možemo povezati veći broj uređaja ali istovremeno komuniciraju samo dva.
- Prenose se i podaci i takt - posebnim linijama.





## Standardi za serijski prenos na mala rastojanja – SPI

- Koriste se posebni vodovi za prenos u dva smera (MSI, MISO).
- Centralna jedinica šalje i takt na svaku periferiju.
- Centralna jedinica šalje i selekzione signale za pojedine periferije.



Kraj 2. glave

(KOLA DIGITALNE  
ELEKTRONIKE)