Digitalna radionica: uvod u digitalnu elektroniku

Miloš Stanojević

Matematička gimnazija, Nedelja informatike

3. april 2015.

Neke definicije

- Kola koja implementiraju kombinacionu logiku ne čuvaju nikakva unutrašnja stanja, odnosno nemaju nikakvu memoriju;
- Osnovna logička struktura koja može imati jedan ili više ulaza i jedan izlaz zove se **kapija** (gate);
- Kapije su osnovni gradivni blokovi mnogo komplikovanijih digitalnih kola.

NOT

simbol

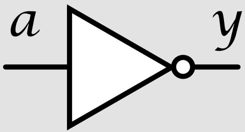


tabela preslikavanja

a	y
0	1
1	0

logički izraz

$$y = \overline{a}$$

AND

simbol

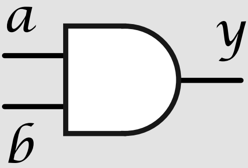


tabela preslikavanja

a	b	y
0	0	0
0	1	0
1	0	0
1	1	1

logički izraz

$$y = a \cdot b$$

OR

simbol

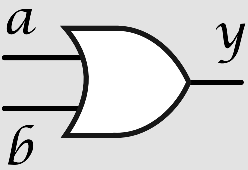


tabela preslikavanja

a	b	y
0	0	0
0	1	1
1	0	1
1	1	1

logički izraz

$$y = a + b$$

XOR (Exclusive OR)

simbol

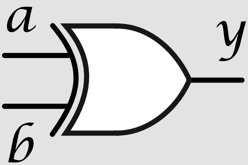


tabela preslikavanja

a	b	y
0	0	0
0	1	1
1	0	1
1	1	0

logički izraz

$$y = a \oplus b$$

NAND (Not AND)

simbol

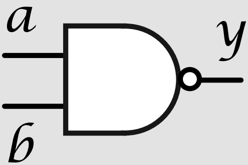


tabela preslikavanja

a	b	y
0	0	1
0	1	1
1	0	1
1	1	0

logički izraz

$$y = \overline{a \cdot b}$$

NOR (Not OR)

simbol

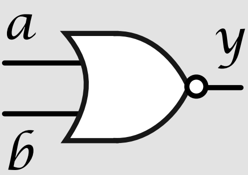


tabela preslikavanja

a	b	y
0	0	1
0	1	0
1	0	0
1	1	0

logički izraz

$$y = \overline{a + b}$$

Bulova algebra

OR	AND
$a + 0 = 0$	$a.0 = 0$
$a + a = a$	$a.a = a$
$a + 1 = 1$	$a.1 = a$
$a + \overline{a} = 1$	$a.\overline{a} = 0$

- AND ima prednost u odnosu na OR, npr.

$$a.b + c.d = (a.b) + (c.d)$$

Bulova algebra, cont'd

- Komutativnost
  $a + b = b + a$
  $a.b = b.a$
- Asocijativnost
  $(a + b) + c = a + (b + c)$
  $(a.b).c = a.(b.c)$
- Distributivnost
  $a.(b + c + \dots) = (a.b) + (a.c) + \dots$
  $a + (b.c \dots) = (a + b).(a + c) \dots$ **NOVO**
- Absorpcija
  $a + (a.c) = a$ **NOVO**
  $a.(a + c) = a$ **NOVO**

Primeri

Pokažimo da važi $a.(\overline{a} + b) = a.b$. Rešenje:

$$a.(\overline{a} + b) = a.\overline{a} + a.b = 0 + a.b = a.b$$

Pokažimo da važi $a + (\overline{a}.b) = a + b$. Rešenje:

$$a + (\overline{a}.b) = (a + \overline{a}).(a + b) = 1.(a + b) = a + b$$

De Morganova teorema

Važe sledeća pravila:

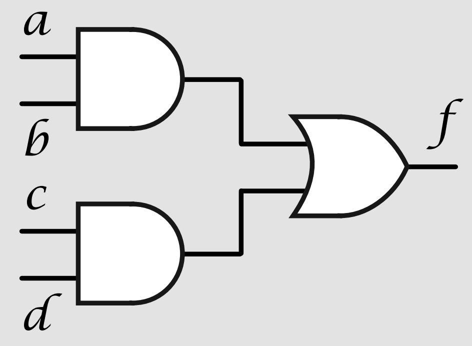
- $\overline{a+b+c+\dots} = \overline{a}.\overline{b}.\overline{c}.\dots$
- $\overline{a.b.c.\dots} = \overline{a}+\overline{b}+\overline{c}+\dots$

Odnosno:

- $a+b+c+\dots = \overline{\overline{a}.\overline{b}.\overline{c}.\dots}$
- $a.b.c.\dots = \overline{\overline{a}+\overline{b}+\overline{c}+\dots}$

De Morganova teorema: kapije

logički izraz



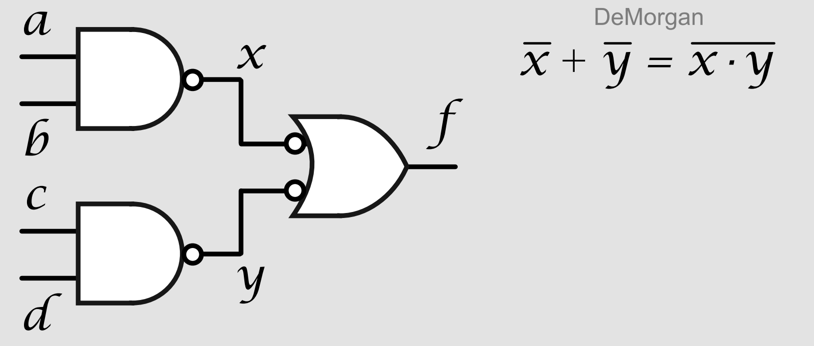
logički izraz

$$f = a \cdot b + c \cdot d$$

Želimo da dati izraz predstavimo pomoću samo NAND ili samo NOR kapija.

De Morganova teorema: kapije

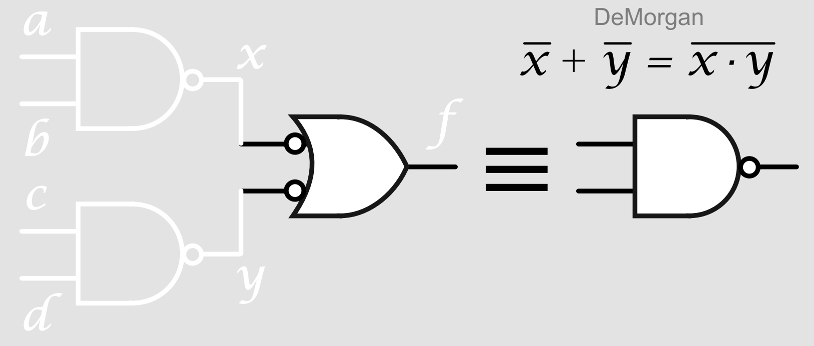
dodajemo "mehuriće"



U tu svrhu koristićemo logiku "mehurića", za koju važi da se dva uzastopna mehurića poništavaju (mehurić predstavlja negaciju).

De Morganova teorema: kapije

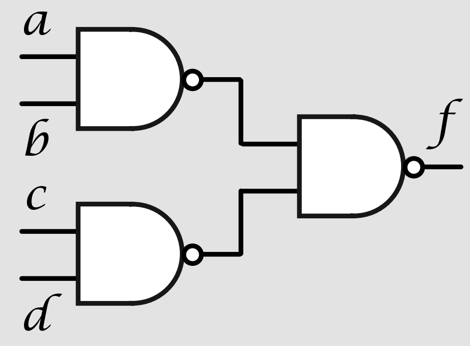
primetimo



Primetimo da su onda markirane(crnom bojom) kapije ekvivalentne.

De Morganova teorema: kapije

i konačno



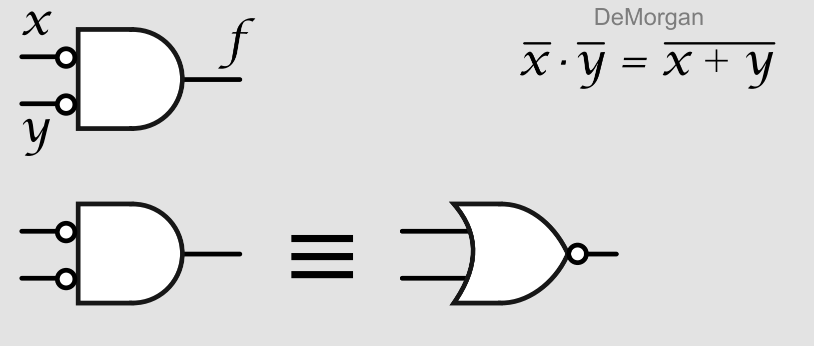
logički izraz

$$f = a \cdot b + c \cdot d$$

$$f = \overline{(\overline{a \cdot b}) \cdot (\overline{c \cdot d})}$$

NB Veoma korisno kada želimo da koristimo samo NAND kapije!

De Morganova teorema: kapije



Primetimo da se ista tehnika može primeniti i ako želimo da dobijemo NOR kapije.

RS latch

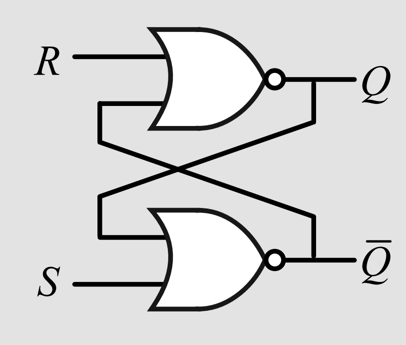
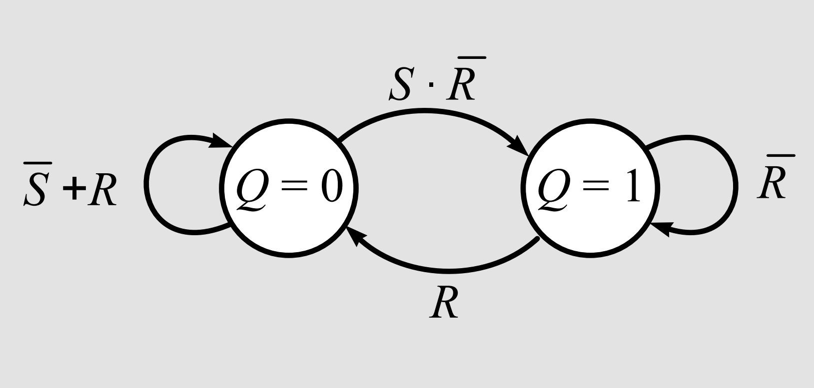


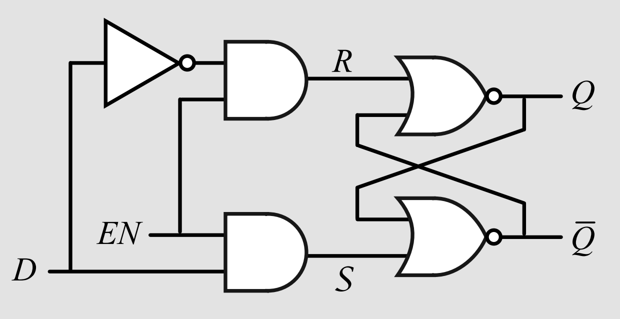
tabela preslikavanja

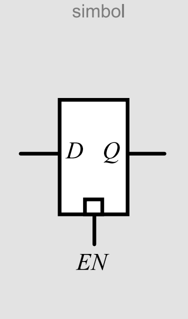
S	R	Q'	$\overline{Q}'$	
0	0	Q	$\overline{Q}$	*hold*
0	1	0	1	*reset*
1	0	1	0	*set*
1	1	0	0	*illegal*

Mašina stanja RS latcha

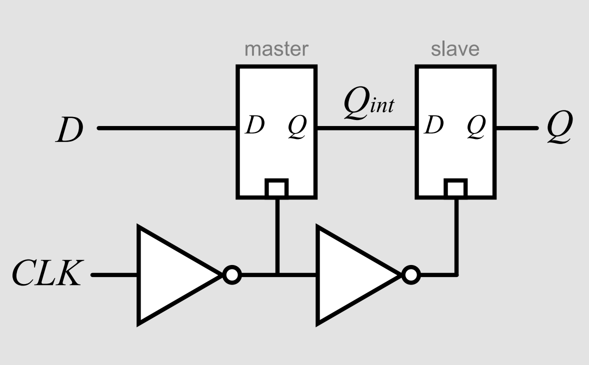


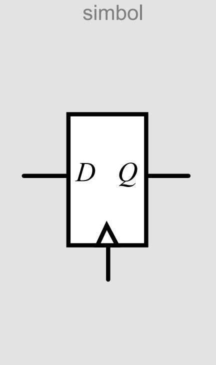
Transparentni D latch



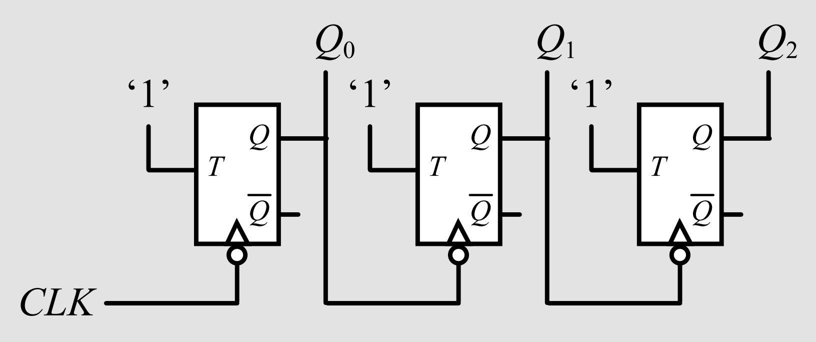


Master-Slave D Flip-Flop

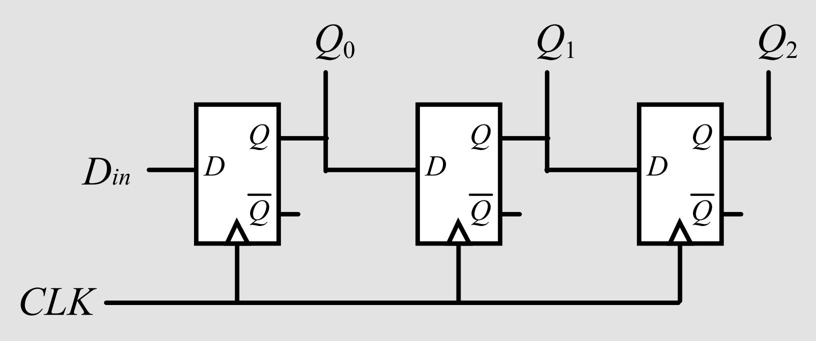




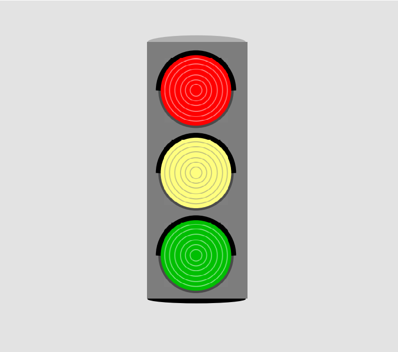
Ripple Counter



Shift Register



Semafor



Današnji zadatak je implementacija jednostavnog semafora na protobordu.

Plan vežbe

- Razumeti način na koji se došlo do šeme kola;
- Napraviti kolo sa date šeme (i pomoliti se da proradi);
- Pokušati dodatne zadatke (ako ostane vremena!).

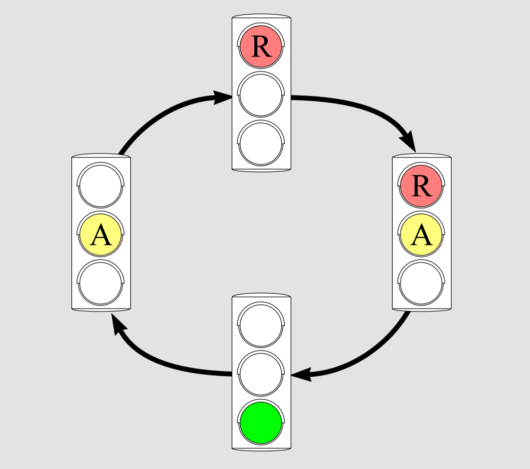
Postupak dolaska do kola

Prvo pravimo mašinu stanja, gde stanja obeležavamo na sledeći način:

$MN = 00$	zeleno svetlo
$MN = 01$	žuto svetlo
$MN = 10$	crveno svetlo
$MN = 11$	crveno i žuto svetlo

Dakle, svako stanje našeg semafora jedinstveno je definisano binarnim brojem MN!

Automat
korak: 1



Postupak dolaska do kola
korak: 2

Sledeće pravimo **tabelu prelaska između stanja**:

M	N	M'	N'
0	0	0	1
0	1	1	0
1	0	1	1
1	1	0	0

Iz tabele prelaska zaključujemo da važe sledeće relacije za M i N:

$$M' = \overline{M}.N + M.\overline{N}$$

$$N' = \overline{M}.\overline{N} + M.\overline{N}$$

Bitno je da razumemo da su M' i N' u stvari vrednosti koje treba da damo našim flip-flopovima na ulazu, tj. $D_M = M'$ i $D_N = N'$!

Postupak dolaska do kola

korak: 3

Sada transformišemo ove izraze koristeći De Morganova pravila (jer na raspolaganju imamo samo NAND kapije):

$$M' = \overline{M}.N + M.\overline{N} = \overline{(\overline{M}.N).(M.\overline{N})}$$

$$N' = \overline{M}.\overline{N} + M.\overline{N} = \overline{(\overline{M}.\overline{N}).(M.\overline{N})}$$

Postupak dolaska do kola

korak: 4

Primetimo, takođe, da je crveno svetlo uključeno kada god važi da je $M = 1$, kao i da je žuto svetlo uključeno kada god važi $N = 1$. Slično, zeleno svetlo je uključeno samo kada je $MN = 00$, odnosno kada je $\overline{M}.\overline{N} = 0$. Stoga važi:

$$R = M$$

$$Y = N$$

$$G = \overline{M}.\overline{N} = \overline{\overline{\overline{M}.\overline{N}}}$$

Gde R, Y i G označavaju da su upaljena crveno, žuto i zeleno svetlo, respektivno.

Kolo

Konačno, korišćenjem sledećih izraza:

- $D_M = \overline{(\overline{M}.N).(M.\overline{N})}$
- $D_N = \overline{(\overline{M}.\overline{N}).(M.\overline{N})}$
- $R = M$
- $Y = N$
- $G = \overline{\overline{\overline{M}.\overline{N}}}$

Dolazimo do konačnog kola, za koje nam treba dva flip-flopa i **šest** NAND kapija (zašto?).