



Digitalna vezja UL, FRI



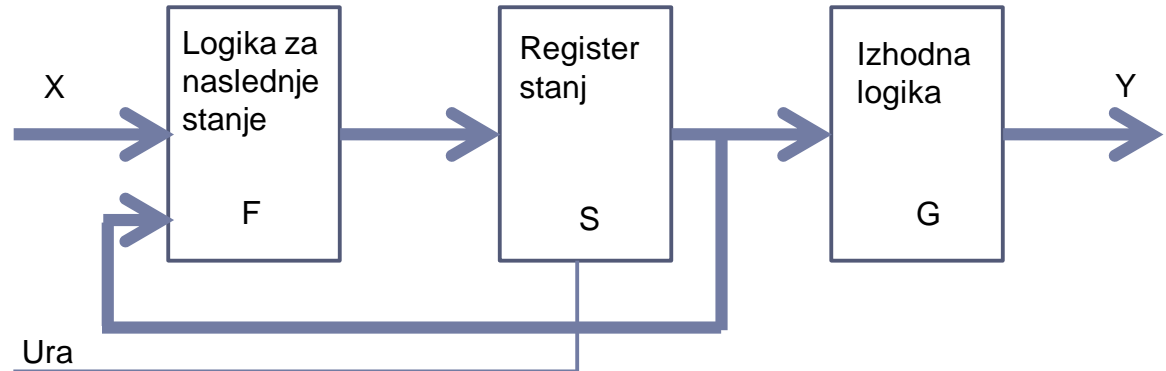
Vaja 10, Avtomat za razpoznavanje niza črk

Avtomat – Končni stroj stanj

Moore (MO):

$$S(t+1) = F[X, S(t)]$$

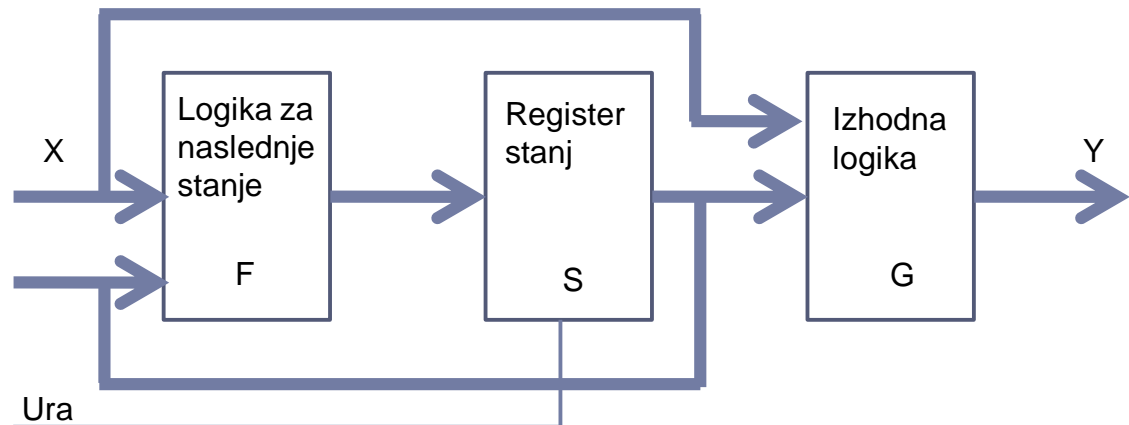
$$Y = G[S(t)]$$



Mealy (ME):

$$S(t+1) = F[X, S(t)]$$

$$Y = G[X, S(t)]$$

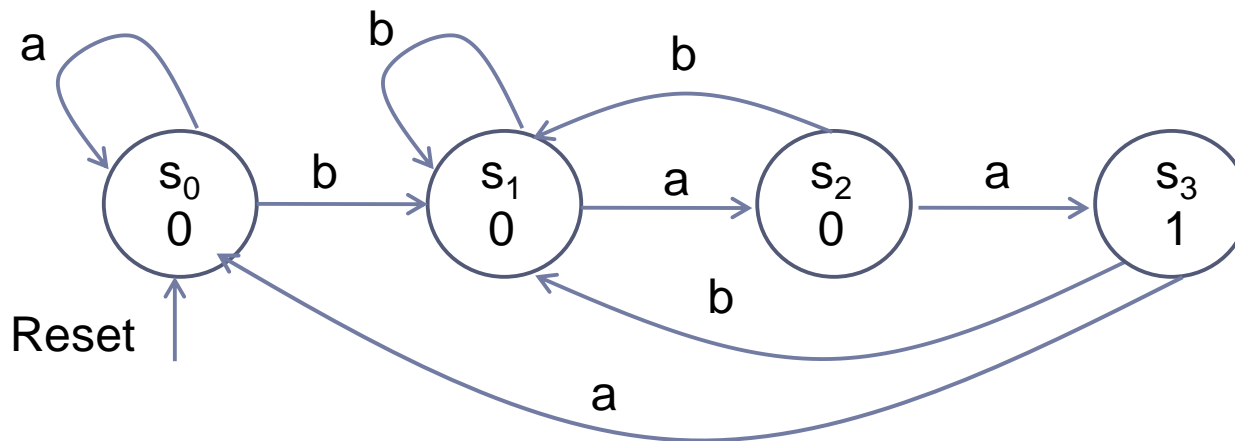


Avtomat za razpoznavanje niza črk

I. Opis problema

- ❑ Realizirajte Mooreov avtomat za razpoznavanje niza črk s podano vhodno množico $X=\{a,b\}$ in izhodno množico $Y=\{0,1\}$.
- ❑ Izhod y je enak 1, če smo v vhodnem zaporedju črk na tipkovnici razpoznali niz 'baa' (prva črka je b), sicer je izhod y enak 0.
- ❑ Za vpis niza črk v logisimu uporabimo tipkovnico, na izhodu tipkovnice je potrebno vsako črko kodirati oz. pretvoriti v binarno vrednost ($a=1, b=0$) in jo posredovati v vhodni pomikalni register (VH). Izhod iz VH je označen kot vhod avtomata $x=\{0,1\}$, izhod $y=\{0,1\}$ je rezultat razpoznavanja in se shranjuje v izhodni pomikalni register (IZ).
- ❑ Vezje ima na vhodu urin signal Clk za sinhronski vpis podatkov v registre in mora imeti omogočeno brisanje vseh modulov s signalom Reset.
- ❑ Primer razpoznavanja za podanih 8 črk:
Tipkovnica: aab**baa**ba (→ smer branja vhodnih črk)
IZ (Y): 00000**1**00

2. Diagram prehajanja stanj (DPS)



□ Kodiranje

vhodna množica (X)

Črka	x
a	1
b	0

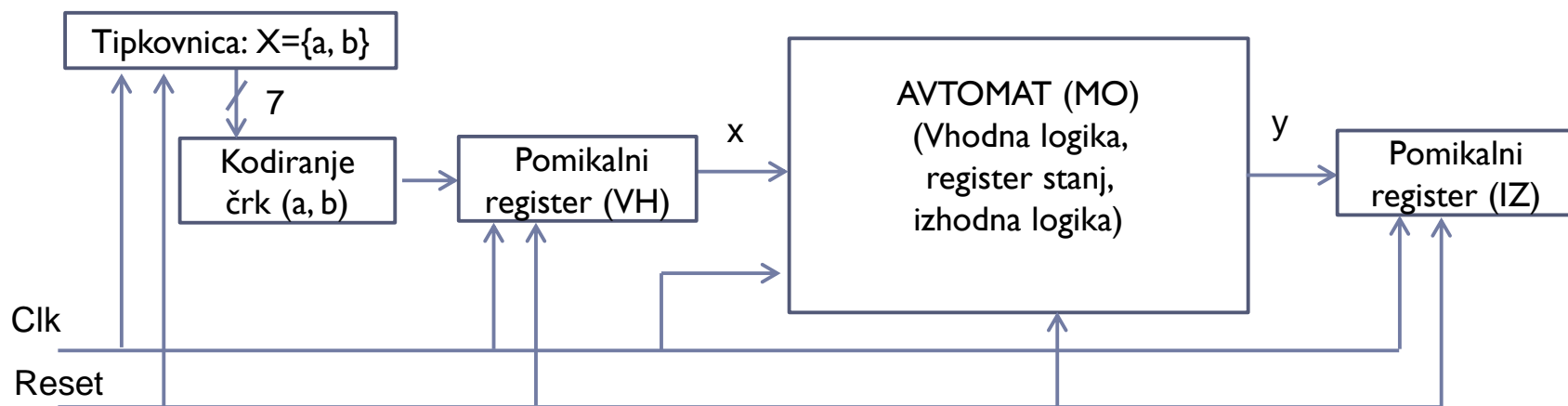
množica stanj (TS)

Stanje	Q ₁	Q ₀
s ₀	0	0
s ₁	0	1
s ₂	1	0
s ₃	1	1

izhod y za TS

Stanje	y
s ₀	0
s ₁	0
s ₂	0
s ₃	1

3. Blok shema vezja z Mooreovim avtomatom



4. Gradniki


- ❑ Tipkovnica in pomikalni registri.
- ❑ Za register stanj uporabite JK pomnilne celice, kjer je $J=K$.
- ❑ Za vhodno in izhodno logiko uporabite gradnike, ki omogočajo najbolj enostavno vezje, če imate na voljo logična vrata (NOT, AND, OR, NAND, NOR) in multiplekserje (2/1 MUX, 4/1 MUX).

5. Vhodna logika, register stanj, izhodna logika

- Binarna aplikacijska tabela prehajanja stanj, izhod y in krmilni signali za JK pomnilni celici ($J_1=K_1$ in $J_0=K_0$).

x	$Q_1(t)$	$Q_0(t)$	$Q_1(t)$	$Q_0(t)$	y	$J_1 = K_1$	$J_0 = K_0$
0	0	0	0	1	0	0	1
0	0	1	0	1	0	0	0
0	1	0	0	1	0	1	1
0	1	1	0	1	1	1	0
1	0	0	0	0	0	0	0
1	0	1	1	0	0	1	1
1	1	0	1	1	0	0	1
1	1	1	0	0	1	1	1


Vzbujanje pomnilne celice $J_1 = K_1$


Vzbujanje pomnilne celice $J_0 = K_0$

- Vhodna logika za določanje naslednjega stanja:
 - Krmilna funkcija $J_1=K_1$ je realizirana z logičnimi vrati NAND
 - Krmilna funkcija $J_0=K_0$ je realizirana z 2-naslovnim MUXjem
- Izhodna logika: y je realiziran z logičnimi vrati AND

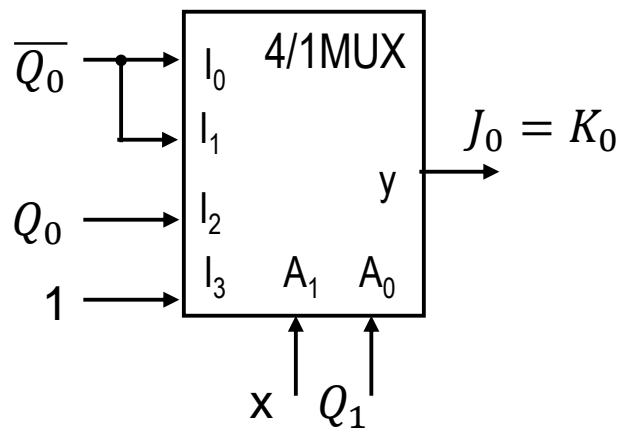
$$J_1 = K_1$$

	$\overline{Q_0}$	Q_0
$\overline{x} \overline{Q_1}$		
$\overline{x} Q_1$	1	1
$x Q_1$		1
$x \overline{Q_1}$		1

$$J_1 = K_1 = \overline{x} \cdot Q_1 \vee x \cdot Q_0$$

$$= (\overline{x} \uparrow Q_1) \uparrow (x \uparrow Q_0)$$

$$J_0 = K_0$$



$$y$$

	$\overline{Q_0}$	Q_0
$\overline{x} \overline{Q_1}$		
$\overline{x} Q_1$		1
$x Q_1$		1
$x \overline{Q_1}$		

$$y = Q_1 \cdot Q_0$$

6. Implementacija - Rešitev a)

□ Pretvorba črk (7-bitna koda ASCII) v binarno vrednost 0 in 1:

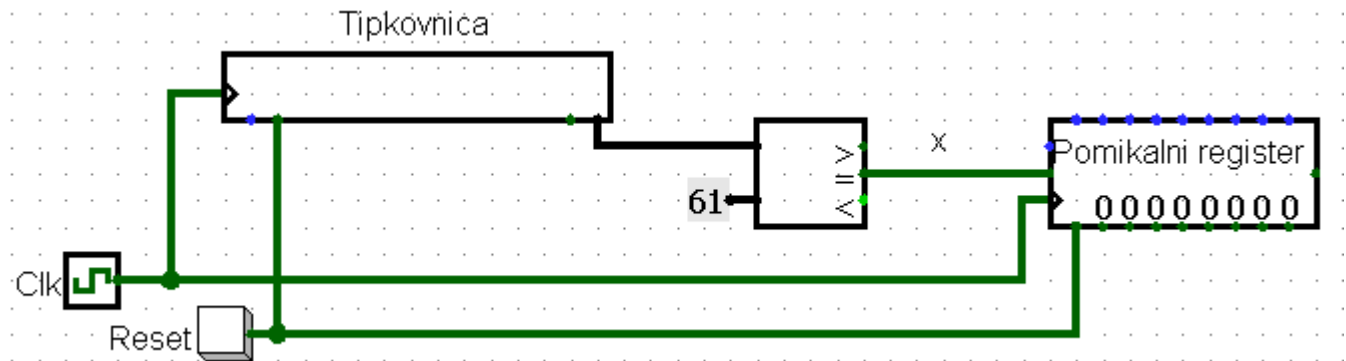
- črka a = 1100001 = 61_h,
- črka b = 1100010 = 62_h

□ Pseudokoda pretvorbe, kjer preverjamo ali je na izhodu tipkovnice črka a:
if ASCII (a) == 61_h Logično vezje: primerjalnik ima na izhodu 1

 x=1

else

 x=0



Pomembno:

Predpostavljamo, da sta na tipkovnici uporabljeni samo črki podane abecede $X=\{a,b\}$, vhod $x = 1$ za črko 'a' in vhod $x = 0$ za črko 'b'.

- ❑ Realizacija v logisimu – vhodni niz je shranjen v pomikalnem registru.
- ❑ Kodiranje znakov – Rešitev a)
- ❑ Tipkovnica: aab**ba**ba

