



# Digitalna vezja UL, FRI

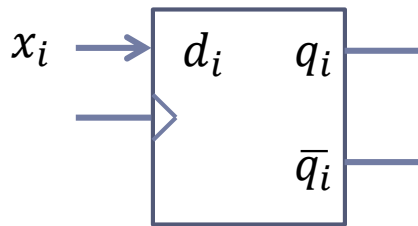


Vaja 10 Registri, Števci

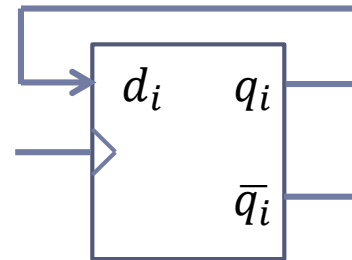
# Registri

## □ D pomnilna celica

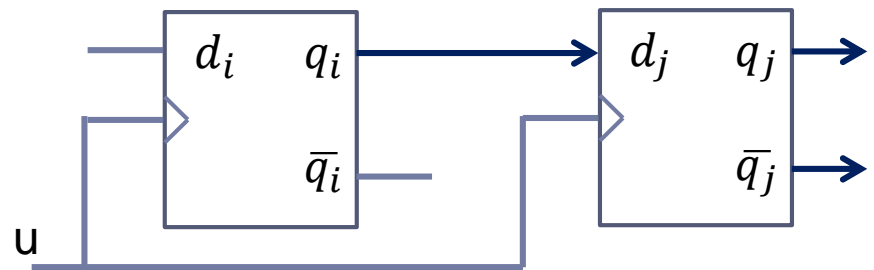
Vpis podatka v  $D_i$  ( $d_i = x_i$ )



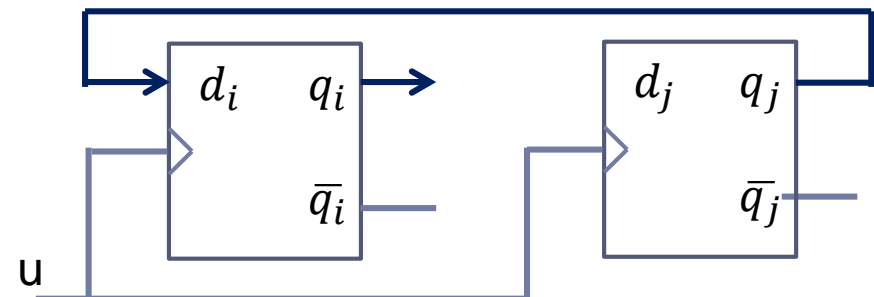
Ohranjanje stanja v  $D_i$  ( $d_i = q_i(t)$ )



□ Pomik desno iz  $D_i$  v  $D_j$  ( $d_j = q_i(t)$ )

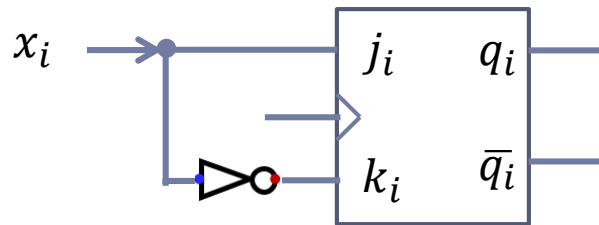


□ Pomik levo iz  $D_j$  v  $D_i$  ( $d_i = q_j(t)$ )

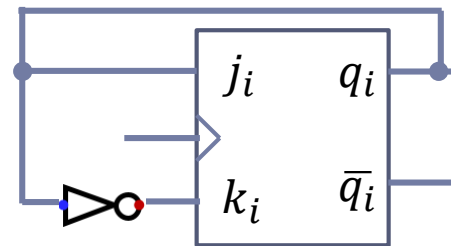


## JK pomnilna celica

Vpis podatka v  $JK_i$  ( $j_i = x_i, k_i = \bar{x}_i$ )

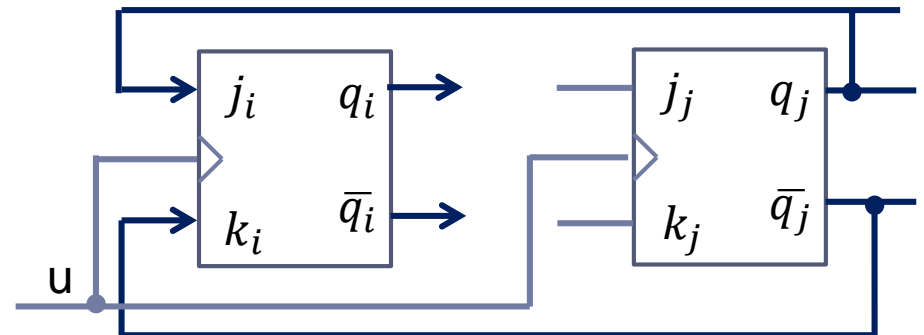


Ohranjanje stanja v  $JK_i$  ( $j_i = q_i, k_i = \bar{q}_i$ )



Pomik levo iz celice  $JK_j$  v  $JK_i$

$q_j$	$q_i$	$D^1 q_i$	$k_i$	$j_i$
0	0	0	?	0
0	1	0	?	1
1	0	1	1	?
1	1	1	0	?



$$j_i = q_j \quad k_i = \bar{q}_j$$

## Primer: 2-bitni register

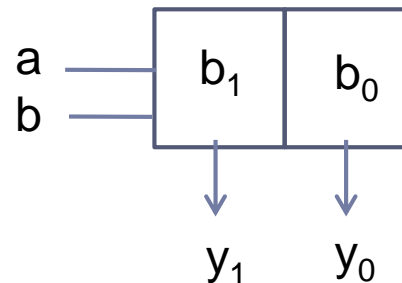
Realizirajte sinhronski 2-bitni register  $Y=(y_1, y_0)$ , ki ima naslednje načine delovanja:

- ❑ Reset: Brisanje pomnilnih celic  $Y(t+1) = 0$ .
- ❑ Hold: Vrednost pomnilnih celic se ohrani  $Y(t+1) = Y(t)$ .
- ❑ CPL: ciklični pomik levo:  $y_1(t+1) = y_0(t)$ ,  $y_0(t+1) = y_1(t)$ .
- ❑ PD: pomik desno za eno mesto, tako da se v celico  $y_1(t+1)$  zapiše konstanta 1.

Za realizacijo uporabite sinhronski JK pomnilni celici, če združite vhoda  $j = k$  in 4/1 MUX-je.

Načini delovanja:

a	b	
0	0	Reset
0	1	Hold
1	0	CPL
1	1	PD



# Tabela prehajanja stanj, izračun krmiljenja JK pomnilne celice, realizacija z MUX 4/1

a	b	$y_1(t)$	$y_0(t)$	$y_1(t+1)$	$y_0(t+1)$	$J_1=K_1$	$J_0=K_0$	MUX <sub>1</sub>	MUX <sub>0</sub>
0	0	0	0	0	0	0	0	$y_1$	$y_0$
0	0	0	1	0	0	0	1		
0	0	1	0	0	0	1	0		
0	0	1	1	0	0	1	1		
0	1	0	0	0	0	0	0	0	0
0	1	0	1	0	1	0	0		
0	1	1	0	1	0	0	0		
0	1	1	1	1	1	0	0		
1	0	0	0	0	0	0	0	$y_1 \oplus y_0$	$y_1 \oplus y_0$
1	0	0	1	1	0	1	1		
1	0	1	0	0	1	1	1		
1	0	1	1	1	1	0	0		
1	1	0	0	1	0	1	0	$\overline{y_1}$	$y_1 \oplus y_0$
1	1	0	1	1	0	1	1		
1	1	1	0	1	1	0	1		
1	1	1	1	1	1	0	0		

Vzbujevalna tabela JK celice:

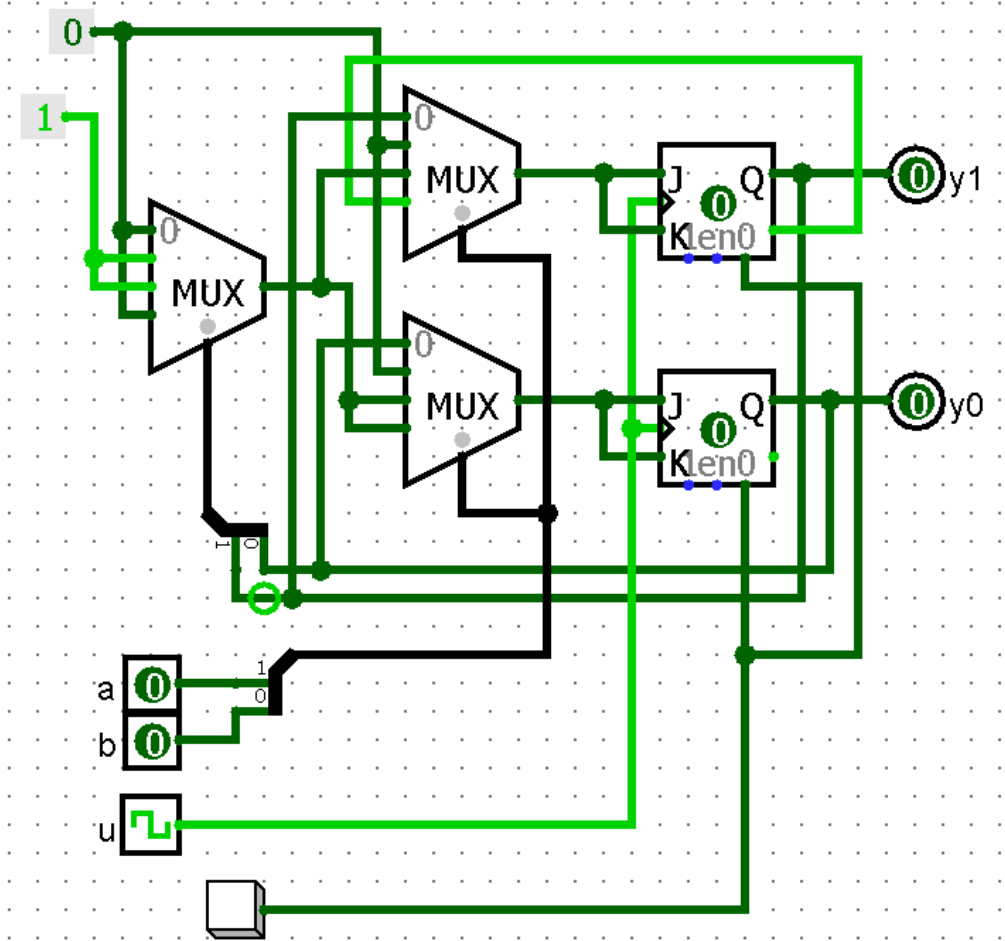
q	$D^1q$	k	j	$j = k$
0	0	?	0	0
0	1	?	1	1
1	0	1	?	1
1	1	0	?	0

Dodaten MUX 4/1

$y_1(t)$	$y_0(t)$	
0	0	0
0	1	1
1	0	1
1	1	0

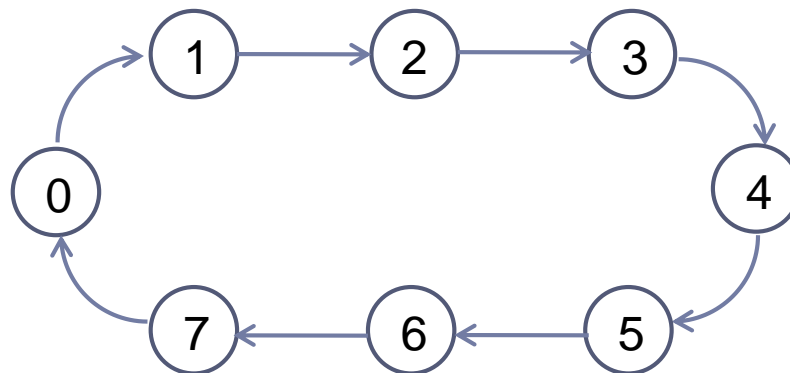


# Realizacija registra v Logisimu



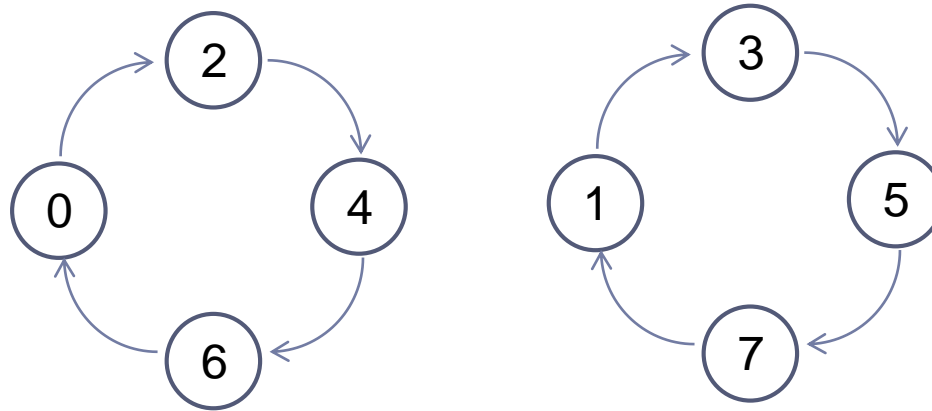
# Števci

- ❑ Sekvenčno vezje:  $Q_i$  je  $i$ -ti izhod števca
- ❑ **Modul štetja (M)** - število različnih izhodnih stanj števca:
- ❑ **Korak štetja** ( $k=1, 2, \dots$ )
- ❑ **Način štetja:**
  - Povečevanje vrednosti – Inkrement:  $Q(t+1) = Q(t) + k$
  - Zmanjševanje vrednosti – Dekrement:  $Q(t+1) = Q(t) - k$
  
- ❑ Primer:  $M=8, \text{INK}, k=1$



## Primer: Sinhronski števec (INC, $k=2$ )

- Realizirajte 3-bitni sinhronski števec za podano sekvenco z uporabo JK pomnilnih celic in logičnih vrat AND, OR, NOT.



Zaporedje stanj števca podamo kot inkrement za 2:

- soda števila: 0 – 2 – 4 – 6 – 0 – ...
  - liha števila: 1 – 3 – 5 – 7 – 1 –
- Z asinhronskim vhodom za brisanje ali postavljanje celice  $Q_0$  naj bo izvedeno preklapljanje iz sodega v liho zaporedje in obratno.
  - Spreminjanje stanj števca prikažite v digitalnem prikazovalniku v logisimu.



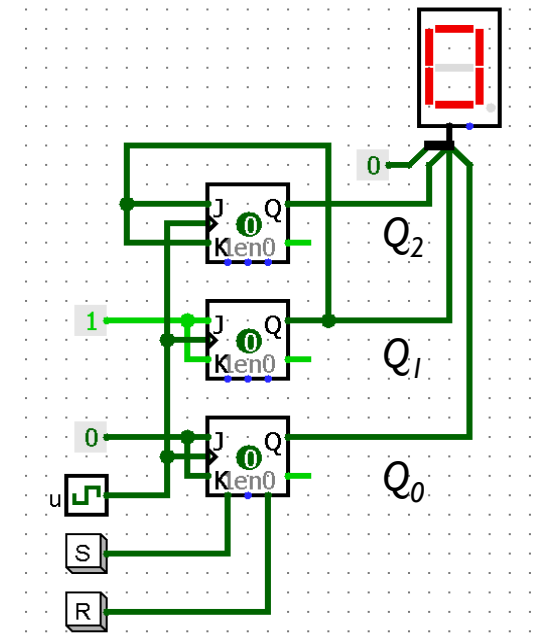
- Tabela stanj: t – trenutno stanje števca, t+1 – naslednje stanje števca
- Pomnilna celica JK – uporabimo rešitev za vhoda J=K

(t)			(t+1)			$J_2=K_2$	$J_1=K_1$	$J_0=K_0$
$Q_2$	$Q_1$	$Q_0$	$Q_2$	$Q_1$	$Q_0$			
0	0	0	0	1	0	0	1	0
0	0	1	0	1	1	0	1	0
0	1	0	1	0	0	1	1	0
0	1	1	1	0	1	1	1	0
1	0	0	1	1	0	0	1	0
1	0	1	1	1	1	0	1	0
1	1	0	0	0	0	1	1	0
1	1	1	0	0	1	1	1	0

$$J_2 = K_2 = Q_1$$

$$J_1 = K_1 = 1$$

$$J_0 = K_0 = 0$$



# Naloge

---

## □ 2-bitni register (Vpis, PL):

Realizirajte sinhronski 2-bitni register  $Y=(y_1, y_0)$ . Krmilni vhod  $a$  določa delovanje:

- ▶  $a = 0$ : Vpis:  $Y(t+1)=X$ , kjer je  $X=(x_1, x_0)$
- ▶  $a = 1$ : PL - pomik levo, tako da se na mesto  $y_0$  vpiše 0.

### Naloge:

- Zapišite tabelo prehajanja stanj delovanja registra.
- Zapišite krmilni funkciji za D pomnilni celici s 2/1 MUXi.
- Realizirajte register v logisimu.

## □ Sinhronski števec (M=4, INC/DEC):

Definirajte 2-bitni števec  $Q=(Q_1, Q_0)$ . Krmilni vhod  $A$  določa delovanje:

- ▶  $A=0$ :  $M=4$ , Dekrement,  $k=1$
- ▶  $A=1$ :  $M=4$ , Inkrement,  $k=1$

### Naloge:

- Zapišite tabelo prehajanja stanj delovanja števca.
- Zapišite krmilni funkciji za D pomnilni celici s 4/1 MUXi.
- Realizirajte števec v logisimu.
- Dodajte gumb za asinhronsko brisanje števca (Reset).

# Dodatna naloga

---

## □ **Sinhronski števec z vpisom začetnega stanja**

Definirajte 3-bitni števec  $Q=(Q_2, Q_1, Q_0)$ . Krmilni vhod A določa delovanje:

- ▶  $A=0$ : Vpis začetnega stanja 5,
- ▶  $A=1$ :  $M=6$ , Dekrement,  $k=1$

### **Naloge:**

- Zapišite tabelo prehajanja stanj delovanja števca z vpisom začetnega stanja.
- Zapišite krmilni funkciji za D pomnilni celici z MUXi (4/1 MUX, 2/1 MUX).
- Realizirajte števec v logisimu.
- Dodajte gumb za asinhronsko brisanje števca (Reset).