

Digitalna vezja

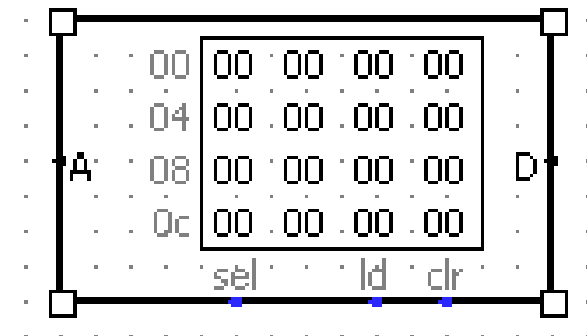
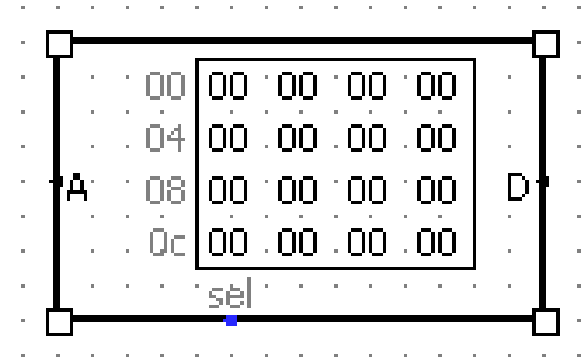
UL, FRI

Vaja 11, Pomnilniki: ROM, RAM

Pomnilnik (ROM, RAM)

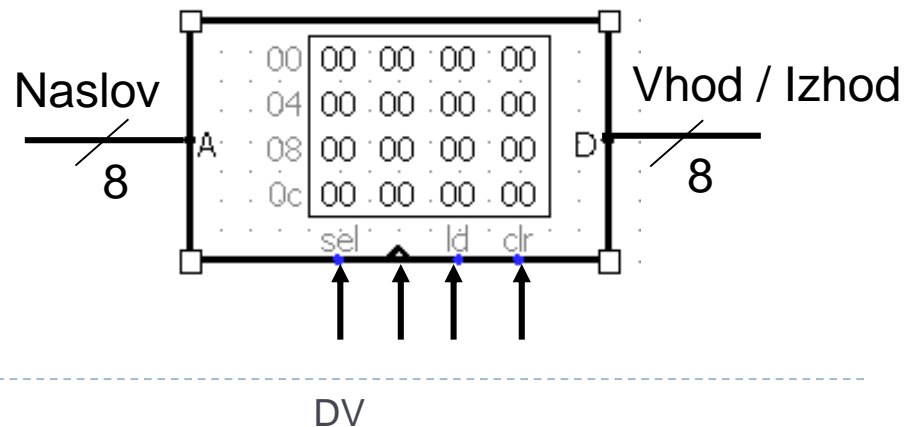
Logisim

- ROM:
 - Naslovni vhodi (A - Address)
 - Onemogoči (sel - chip select)
 - Izhodi (D - Data)
- RAM:
 - Naslovni vhodi (A - Address)
 - Onemogoči (sel - chip select)
 - Podatki na izhod (ld - Load)
 - Briši (clr - Clear)
 - Vhodi/Izhodi (D - Data)

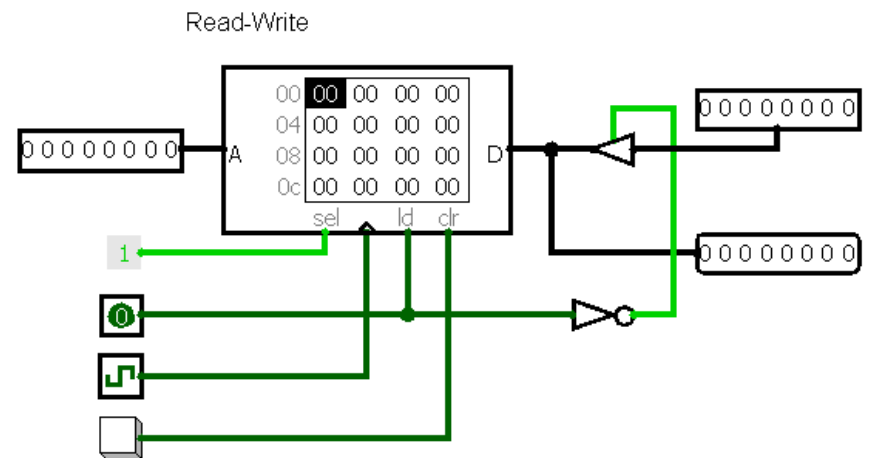
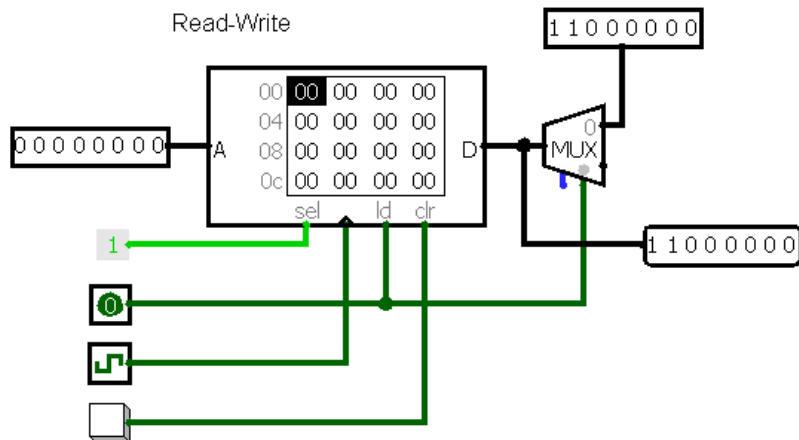
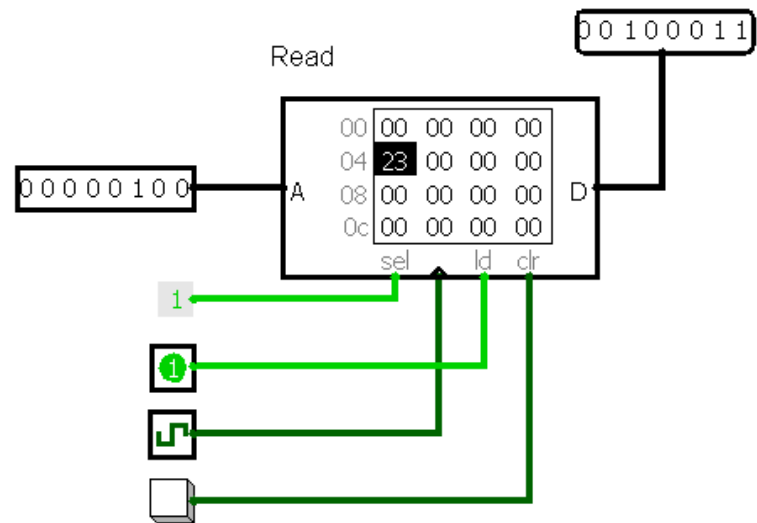
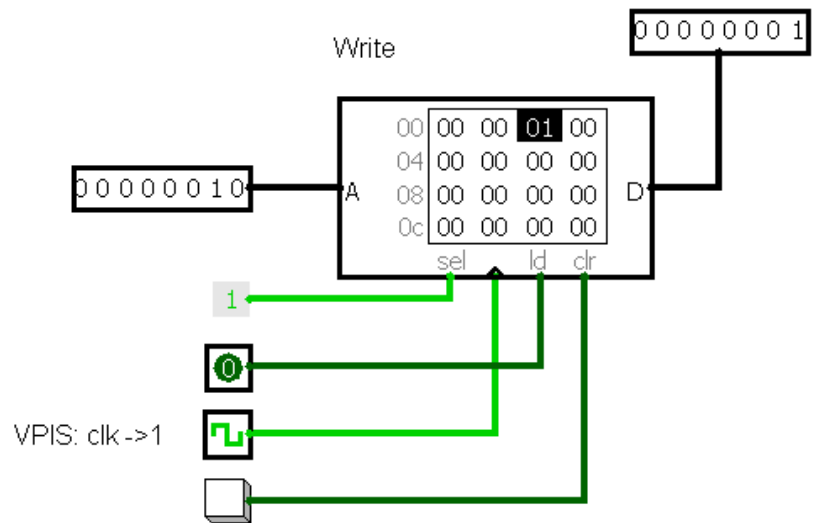


Vaja 1: Pomnilnik RAM

- Naslov: A
- Podatkovni vhod/izhod: D
- Krmilni signali:
 - sel (Chip Select): 0 – onemogoči delovanje RAMa
 - ld (Load): 1 – omogoči podatke na izhod RAMa
 - clr (Clear): 1 – asinhronsko brisanje RAMa
 - Clk – urin signal
- Realizirajte:
 - Branje pomnilnika (D→Izhod)
 - Pisanje v pomnilnik (D→Vhod)
 - Branje/Pisanje (D→Vhod/Izhod)



Rešitev:

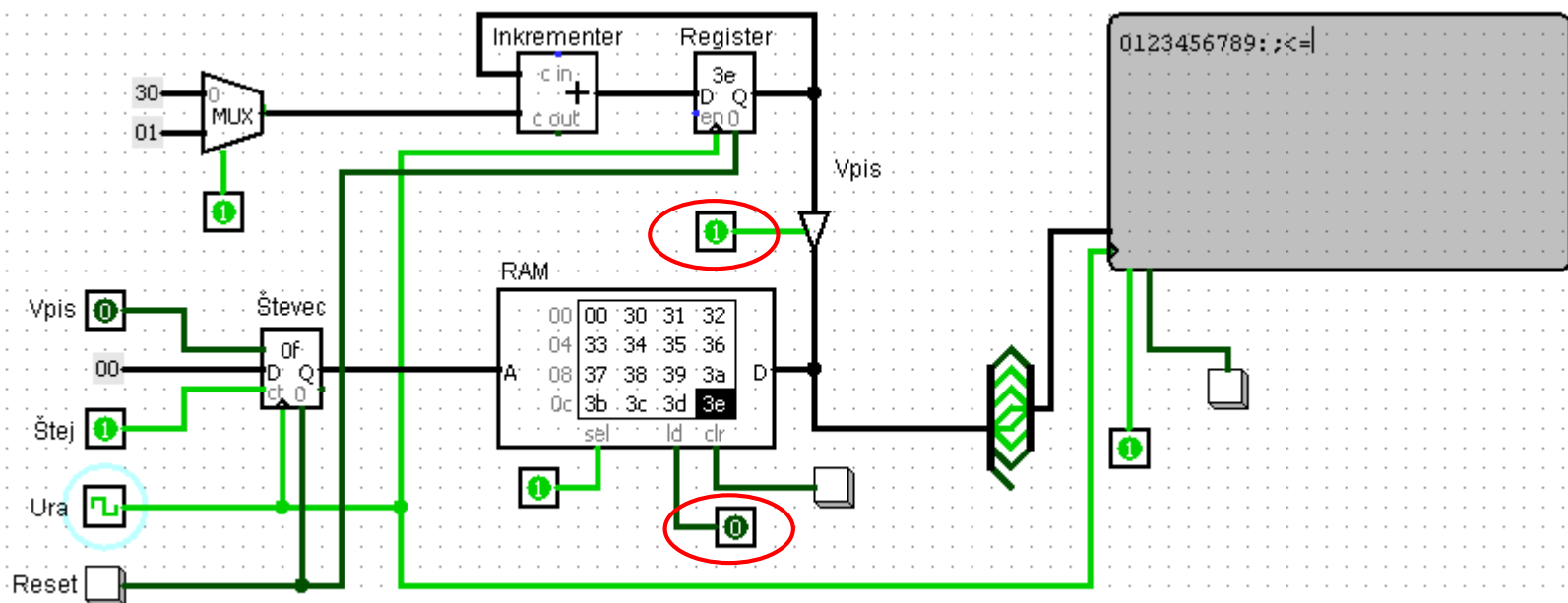


Vaja 2: Vpis/branje ASCII znakov

Program: logisim-win-2.3.1

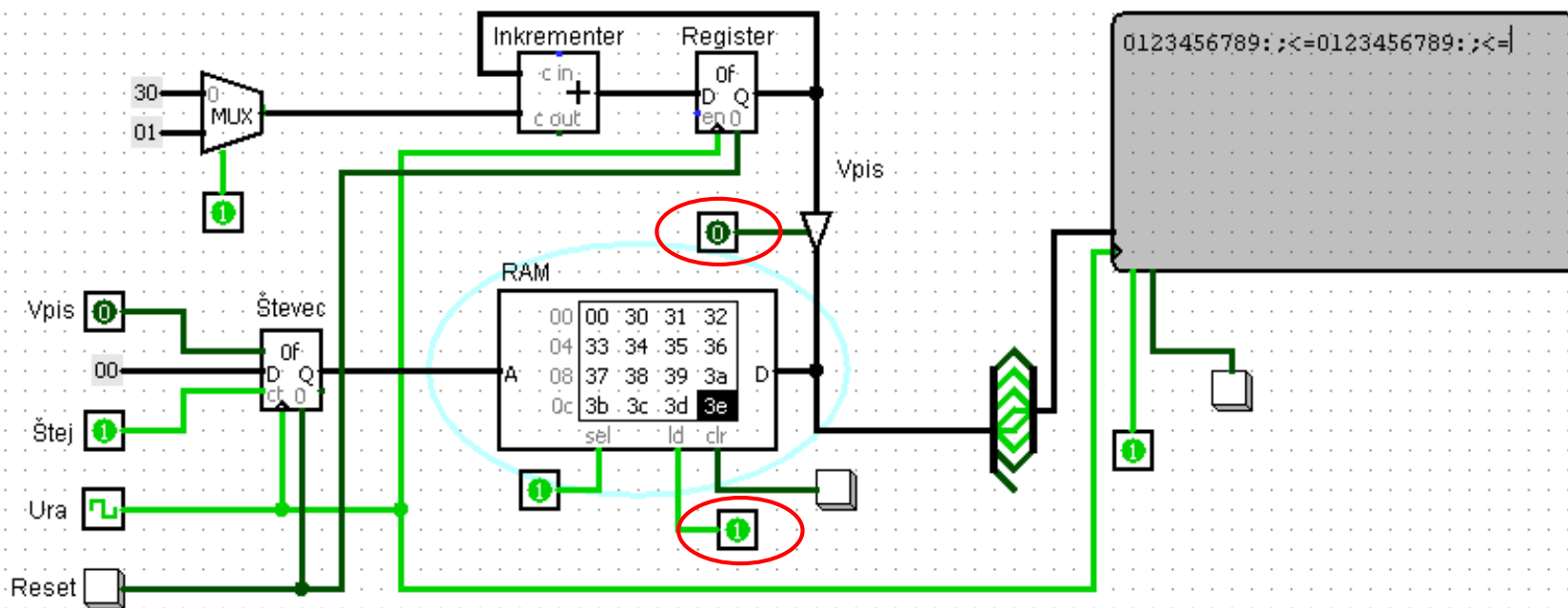
Vpis v RAM:

- ▶ Signala za določanje vpisa: $Vpis = 1$, $RAM\ Id = 0$
- ▶ $MUX = 0$, Vpiše se začetna koda ASCII (število 0 = 30_H)
- ▶ $MUX = 1$, s signalom Ura se potem vpisujejo kode vse do $3e_H$



Branje:

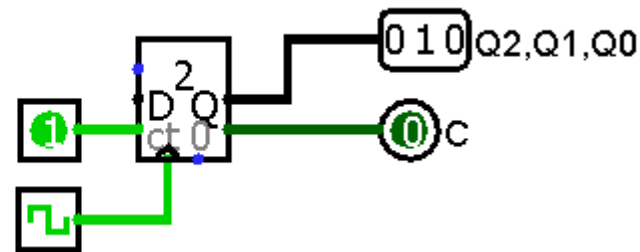
- Signala za določanje branja: $V_{pis} = 0$, $RAM\ Id = 1$



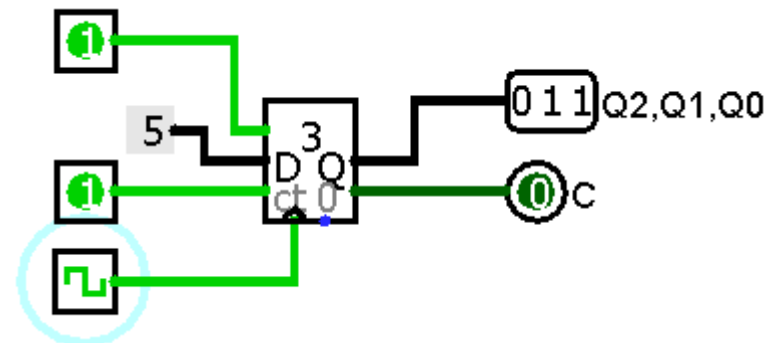
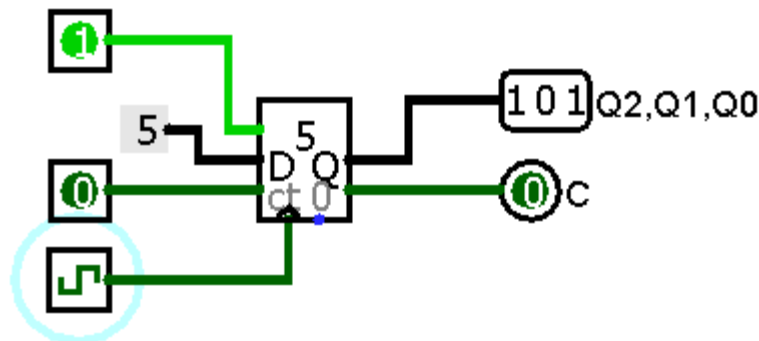
Števec

- ▶ Nastavitev maksimalne vrednosti: 0x5
- ▶ Vhodi: clk, ct = 1 - inkrement
- ▶ Izhodi: $Q_2 Q_1 Q_0$ – stanja, $C = 1$, ko števec doseže maksimalno vrednost

Circuit: main	
Circuit Name	main
Shared Label	
Shared Label Facing	East
Shared Label Font	SansSerif Plain 12



- ▶ Vpis maksimalne vrednosti in dekrement:



Vaja 3: Realizacija števca

- ❑ Števec ima stanja:
 - ❑ 0, 1, 2, 3, 4
- ❑ Operacija:
 - ❑ inkrement
- ❑ Vhod:
 - ❑ $B = 1$ – izvede se asinhronsko brisanje števca = Reset,
- ❑ Izhod:
 - ❑ $P = 1$, če je števec v stanjih 0,1,2,3
 - ❑ $P = 0$, če je števec v stanju 4 .
- ❑ Realizacija - logisim:
 - ❑ Register stanj: JK pomnilne celice
 - ❑ Vhodna/izhodna logika: ROM
 - ❑ Prikaz stanj: Hex Digit Display, P: LED

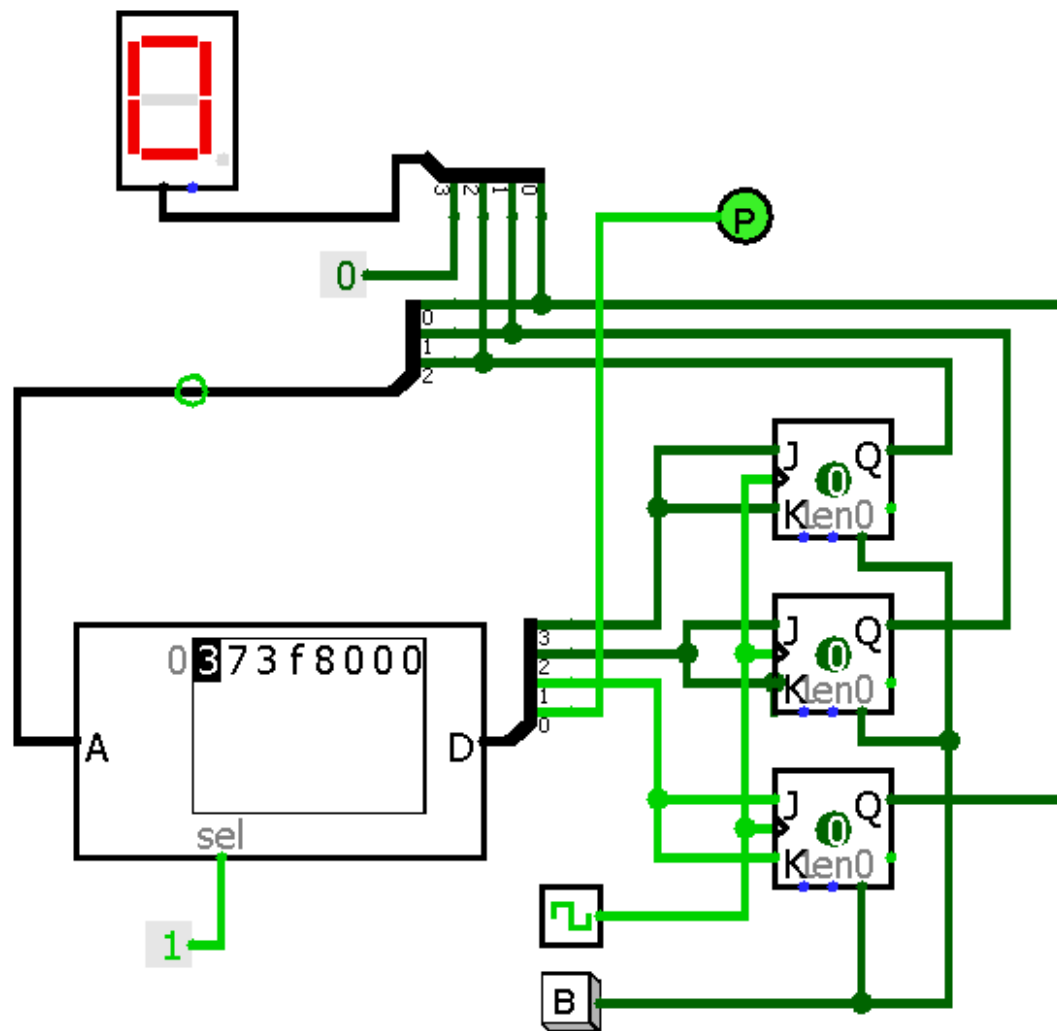
Rešitev: Pravilnostna tabela

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

► ROM 8 x 4

Naslov	Podatki: JK, P
0	3
1	7
2	3
3	F
4	8

Rešitev - logisim



Vaja 4: Realizacija semaforja

Za primer semaforja (Vaja 10) izdelajte krmilnik, ki naj reagira na vhodni signal B, ki ga upravlja pešec.

- ❑ Pri realizaciji uporabite stanja z izhodi: Rg, Rr, Yr, Gr
- ❑ Na semaforju za pešca naj gori zelena luč 4 urine periode.
- ❑ Pri določanju stanj pazite na varnost pešcev !
 - ❑ Stanja, ki so varna za pešca
 - Rr (rdeča luč avto, rdeča luč pešec)
 - Rg (rdeča luč avto, zelena luč pešec)
 - Yr (rumena luč avto, rdeča luč pešec)
 - Gr (zelena luč avto, rdeča luč pešec)
 - ❑ Stanji, ki sta nevarni za pešca
 - Yg (rumena luč avto, zelena luč pešec)
 - Gg (zelena luč avto, zelena luč pešec)
- ❑ Oznake vhodov in izhodov: B – gumb (1 – pritisnjen, 0 – ni pritisnjen), R – rdeča luč avto, Y – rumena luč avto, G – zelena luč avto, r – rdeča luč pešec, g – zelena luč pešec,

1. Narišite diagram prehajanja stanj in izpolnite aplikacijsko tabelo.
2. Za register stanj uporabite dve JK pomnilni celici.
3. Za časovno izvedbo kombinacije Rg uporabite števec.
4. Za realizacijo funkcij za določanje naslednjega stanja $S(t+1)$ in izhodov za prižiganje luči uporabite ROM.

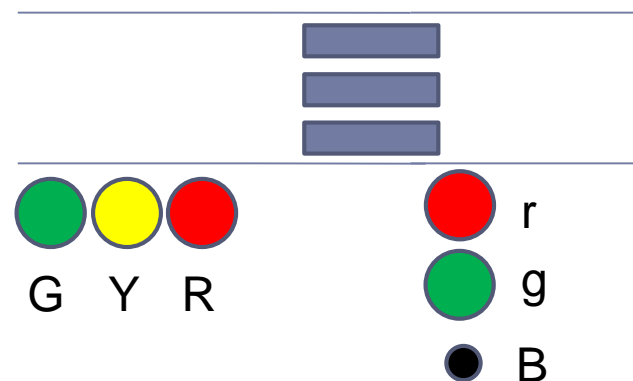
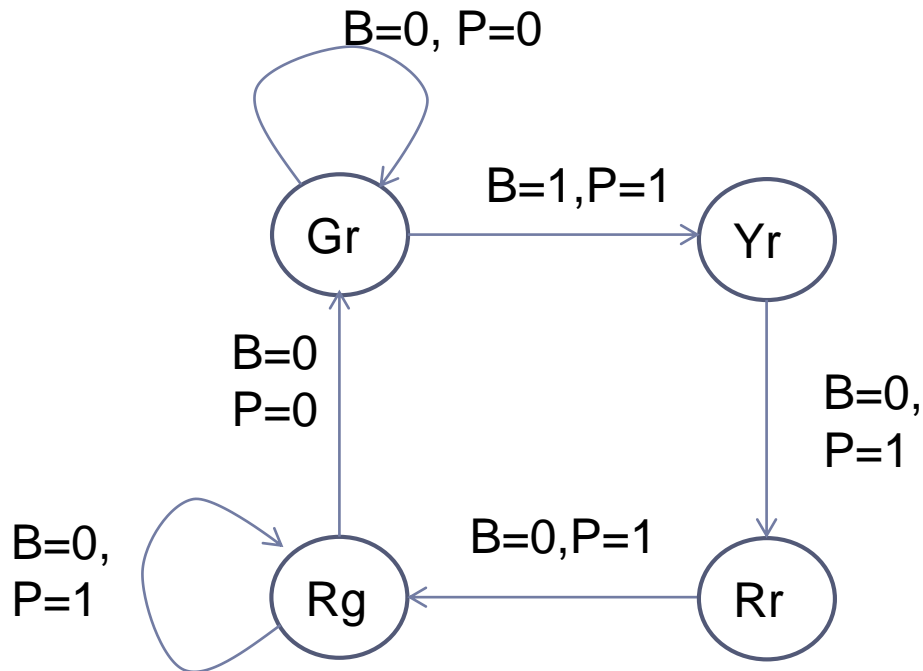


Diagram prehajanja stanj

□ Rešitev:

- $B = 1$ (semafor gre v stanje Yr)
- $P = 1$ (čas delovanja semaforja za pešca), postavi se na 1, ko je pešec pritisnil tipko ($B = 1$) in potem traja 3 periode (uporabimo števec)



	$Q_1(t)$	$Q_0(t)$
Gr	0	0
Yr	0	1
Rr	1	0
Rg	1	1

Števec (D pomnilna celica)

- ❑ B = Reset, izhod P= 1, če je števec: 0,1,2,3.
- ❑ Realizacija: D pomnilne celice, logična vrata: AND, OR, NOT

Q2(t)	Q1 (t)	Q0(t)	Q2(t+1)	Q1 (t+1)	Q0(t+1)	P
0	0	0	0	0	1	1
0	0	1	0	1	0	1
0	1	0	0	1	1	1
0	1	1	1	0	0	1
1	0	0	0	0	0	0
1	0	1				
1	1	0				
1	1	1				

$$D2 = Q2(t + 1) = \overline{Q2}.Q1.Q0$$

$$D1 = Q1(t + 1) = \overline{Q2}.\overline{Q1}.Q0 \vee \overline{Q2}.Q1.\overline{Q0}$$

$$D0 = Q0(t + 1) = \overline{Q2}.\overline{Q1}.\overline{Q0} \vee \overline{Q2}.Q1.\overline{Q0}$$

$$P = \overline{Q2} \vee Q1 \vee Q0$$

Števec (JK pomnilna celica)

- Realizacija: JK pomnilne celice, logična vrata: AND OR; NOT

Q2(t)	Q1 (t)	Q0(t)	Q2(t+1)	Q1 (t+1)	Q0(t+1)	J2=K2	J1=K1	J0=K0	P
0	0	0	0	0	1	0	0	1	1
0	0	1	0	1	0	0	1	1	1
0	1	0	0	1	1	0	0	1	1
0	1	1	1	0	0	1	1	1	1
1	0	0	0	0	0	1	0	0	0
1	0	1							
1	1	0							
1	1	1							

$$J2 = K2 = \overline{Q2}.Q1.Q0 \vee Q2.\overline{Q1}.\overline{Q0}$$

$$J1 = K1 = \overline{Q2}.Q0$$

$$J0 = K0 = \overline{Q2} \vee Q1 \vee Q0$$

$$P = \overline{Q2} \vee Q1 \vee Q0$$

Števec (JK pomnilna celica) - minimizacija

	Q2			
Q1	x	x	1	
	1	x		
	Q0			

$$J2 = K2 = Q2 \vee Q1.Q0$$

	Q2			
Q1	x	x	1	
		x	1	
	Q0			

$$J1 = K1 = Q0$$

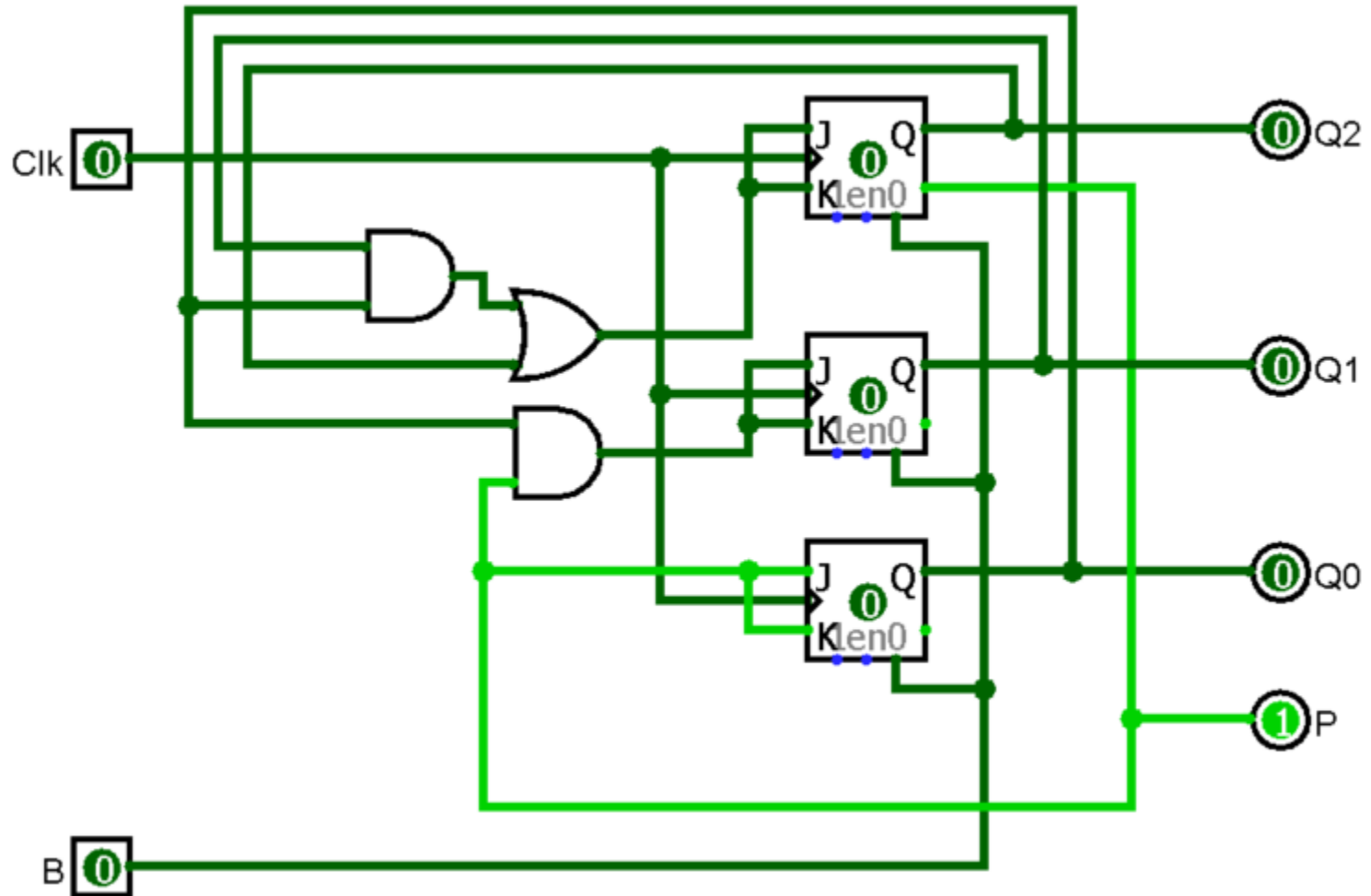
	Q2			
Q1	x	x	1	1
		x	1	1
	Q0			

$$J0 = K0 = \overline{Q2}$$

	Q2			
Q1	x	x	1	1
		x	1	1
	Q0			

$$P = \overline{Q2}$$

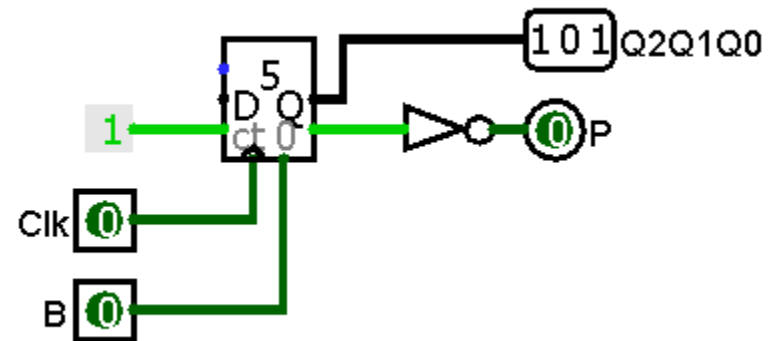
Števec - logisim



Števec - logisim

- ❑ Uporabimo števec, kjer določimo:
 - Maksimalno vrednost 5
 - $P = 1$ (izhod števca se postavi na 1, ko doseže maksimalno vrednost 5)

Counter	
Data Bits	3
Maximum Value	0x5
Action On Overflow	Wrap around
Trigger	Rising Edge
Label	
Label Font	SansSerif Plain 12



Register stanj

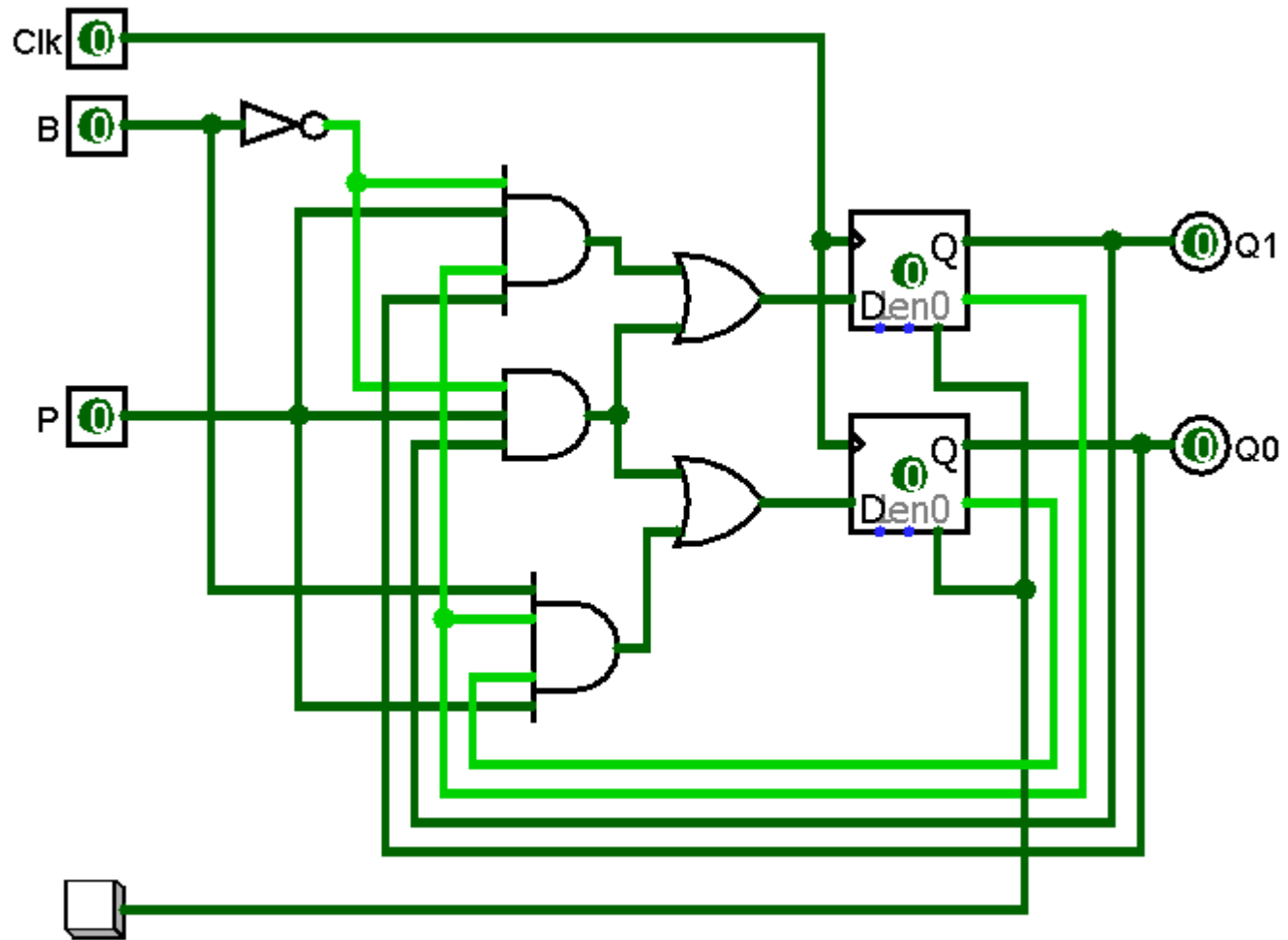
- Definiramo samo prehode, ki so podani v diagramu

Trenutno stanje		Vhod		Naslednje stanje	
Q1(t)	Q0(t)	B	P	Q1(t+1)	Q0(t+1)
0	0	0		0	0
0	0	1	1	0	1
0	1				
0	1	0	1	1	0
1	0				
1	0	0	1	1	1
1	1	0	0	0	0
1	1	0	1	1	1

$$D1 = \overline{Q1}.Q0.\overline{B}.P \vee Q1.\overline{B}.P$$

$$D0 = \overline{Q1}.\overline{Q0}.B.P \vee Q1.\overline{B}.P$$

Register stanj - logisim



Izhodi – prižiganje luči

stanje			semafor avto			semafor pešec	
	Q_1	Q_0	R	Y	G	r	g
Gr	0	0	0	0	1	1	0
Yr	0	1	0	1	0	1	0
Rr	1	0	1	0	0	1	0
Rg	1	1	1	0	0	0	1

$$R = Q_1 \qquad r = \overline{Q_1} \cdot \overline{Q_0} = \overline{Q_1} \vee \overline{Q_0}$$

$$Y = \overline{Q_1} \cdot Q_0 \qquad g = Q_1 \cdot Q_0$$

$$G = \overline{Q_1} \cdot \overline{Q_0}$$

Semafor - logisim

- Signal B =1 toliko časa, da se prižge Y, potem ga postavimo na 0)

