



Laboratorij za načrtovanje integriranih vezij

Univerza *v Ljubljani*  
Fakulteta *za elektrotehniko*



Digitalni Elektronski Sistemi

Model vezja

Računalniški model in realno vezje

# Model logičnega negatorja

---

Načini predstavitve (modeliranja) negatorja:

- ▶ enačba
- ▶ električni simbol
- ▶ pravilnostna tabela

$$y = \text{NOT } (a)$$

značilen simbol:

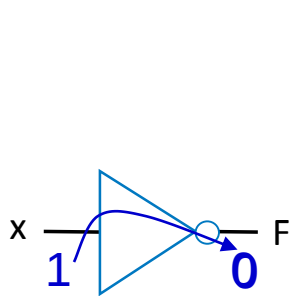


IEEE simbol:

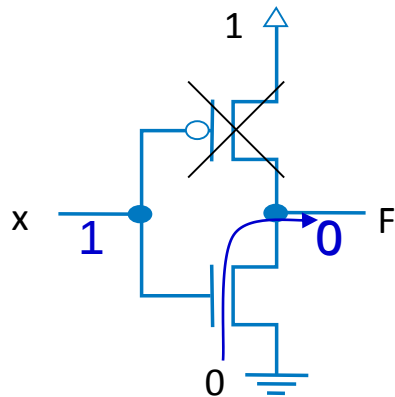


| a | y |
|---|---|
| 0 | 1 |
| 1 | 0 |

# Delovanje negatorja



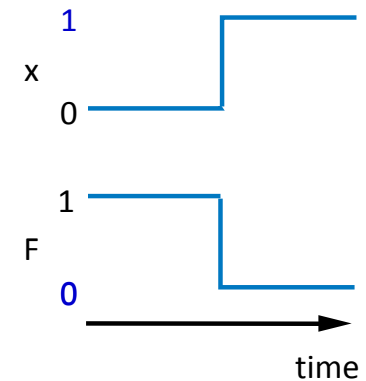
električni  
simbol



tranzistorska izvedba  
vezja

| x | F |
|---|---|
| 0 | 1 |
| 1 | 0 |

pravilnostna  
tabela



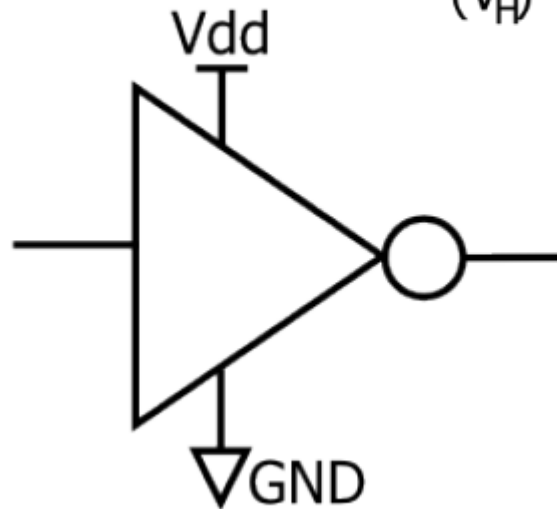
časovni diagram

vrednost vhoda x : 1 povzroči izhod F : 0

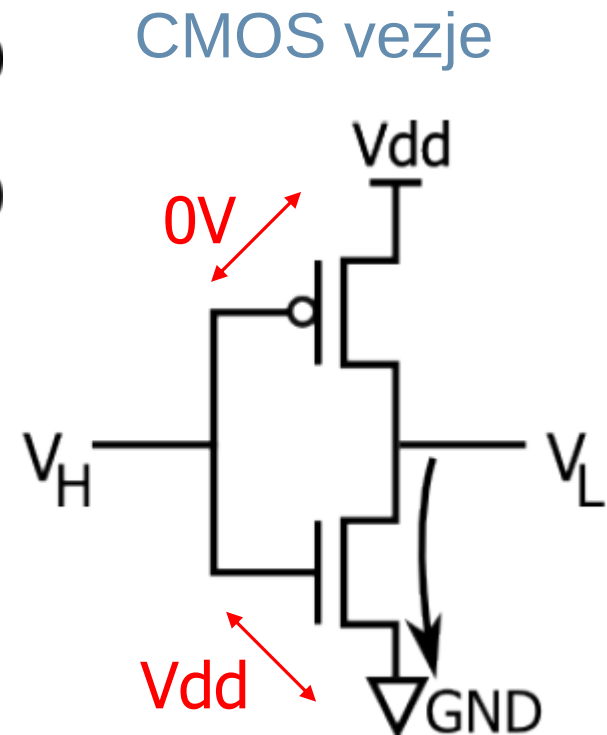


# Logični negator: model in realno vezje

- V vezju so namesto 0 in 1 različni potenciali

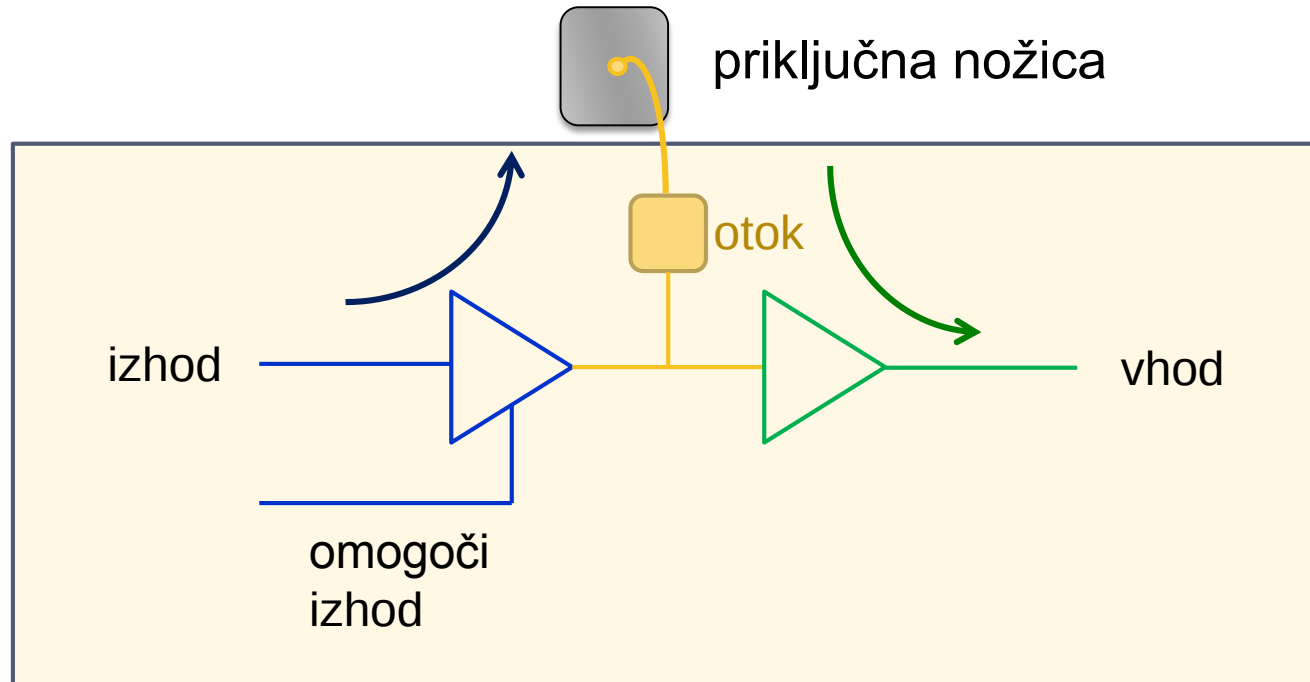


|         | $U_{IN} [V]$ | $U_{OUT} [V]$ |         |
|---------|--------------|---------------|---------|
| $(V_L)$ | 0.0          | 3.3           | $(V_H)$ |
|         | 0.5          | 3.3           |         |
|         | ---          | ---           |         |
| $(V_H)$ | 3.0          | 0.0           | $(V_L)$ |
|         | 3.3          | 0.0           |         |



# Vhodno-izhodni priključki integriranega vezja

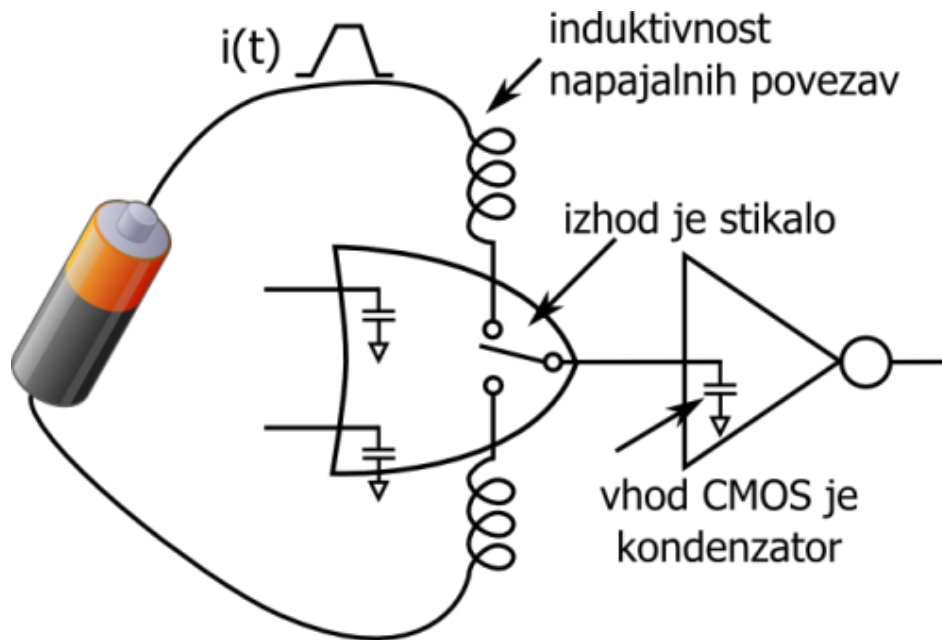
- ▶ vhodni ojačevalnik (buffer)
- ▶ izhodni tri-stanjski ojačevalnik
  - ▶ logični izhod ali izklopljeno stanje (visoka impedanca)



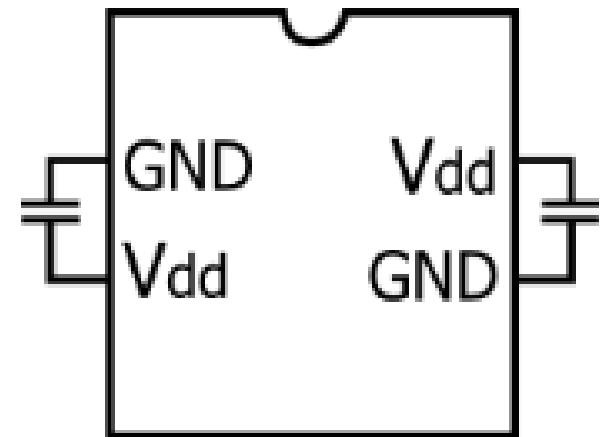
# Model vhodnih in izhodnih priključkov

## Enostaven model tehnologije CMOS

- ▶ vhod = kondenzator
- ▶ izhod = stikalo

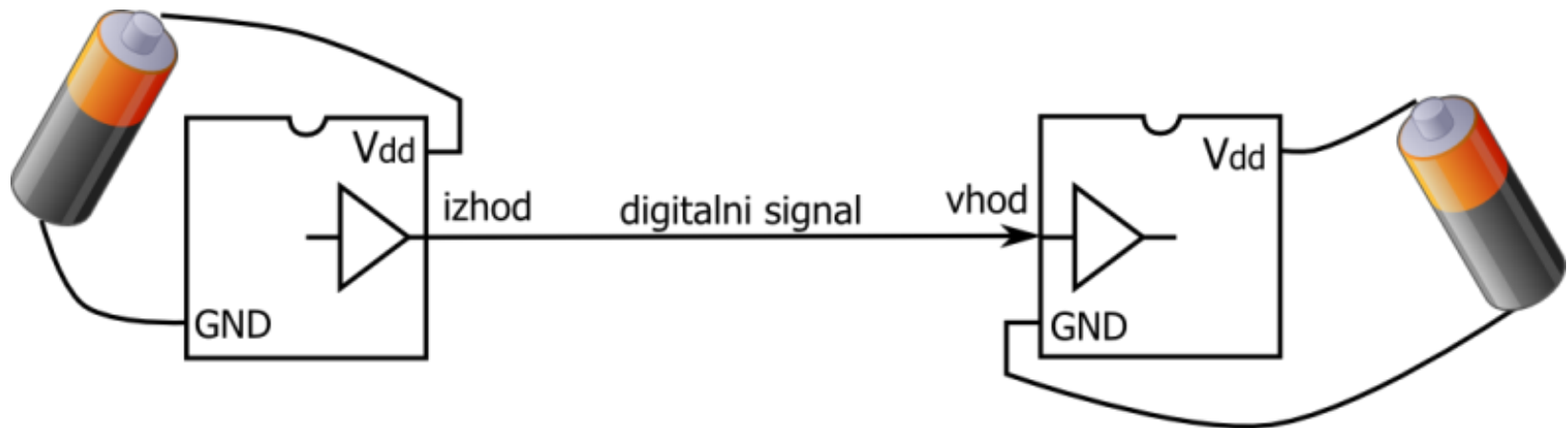


kondenzatorji na  
napajal. priključkih



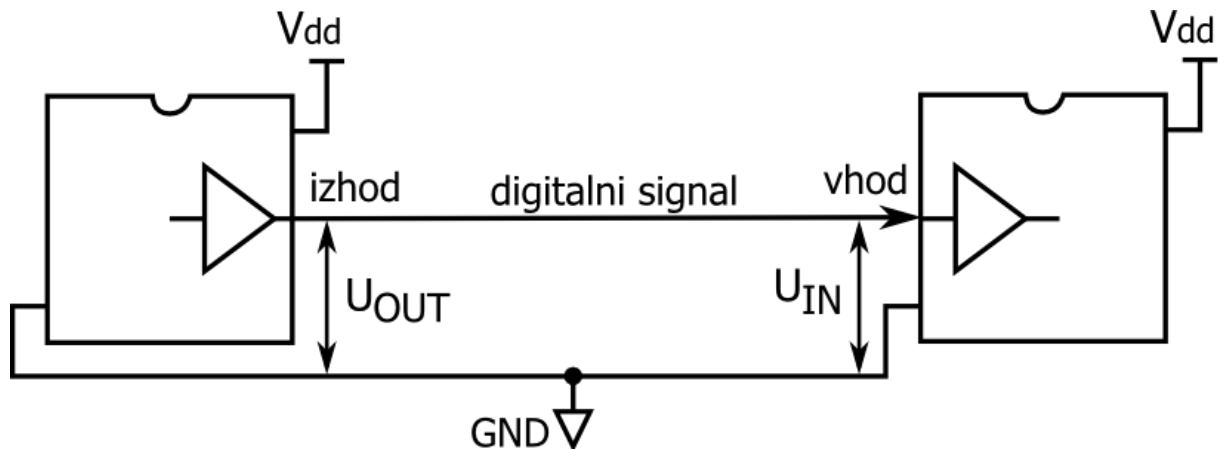
# Prenos signalov v digitalnem vezju

---



# Prenos signalov v digitalnem vezju

---



# Območje potencialov: 0 ... Vdd/2 ... Vdd

- Kako interpretiramo signal s potencialom  $V_{dd} / 2$  ?

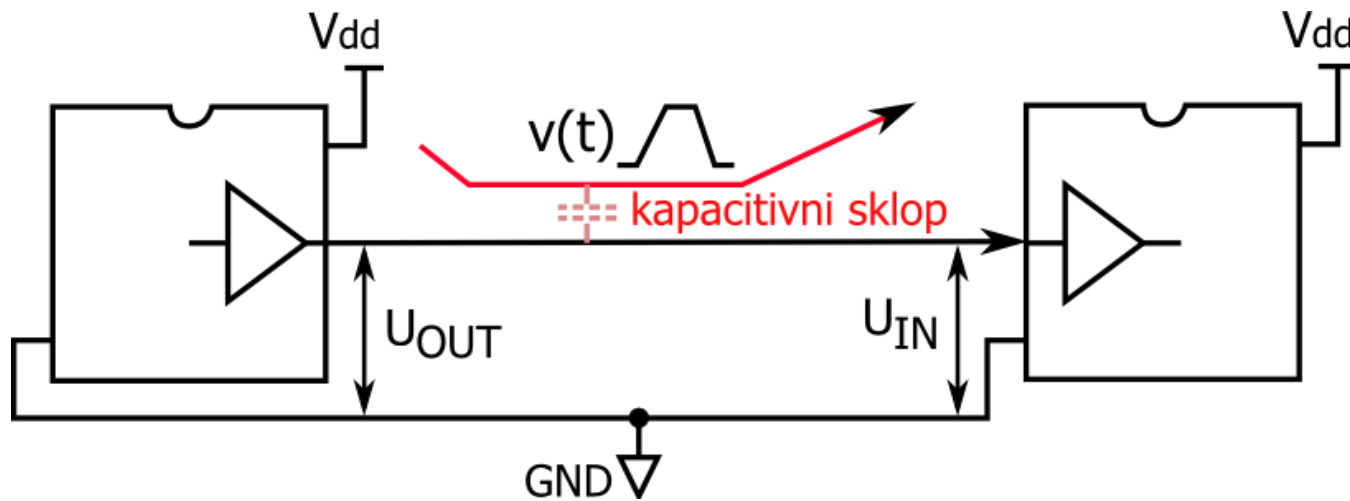
logična 0:  $0V \leq V_L \leq 1V$

logična 1:  $3V \leq V_H \leq 5V$



# Šum na signalnih povezavah

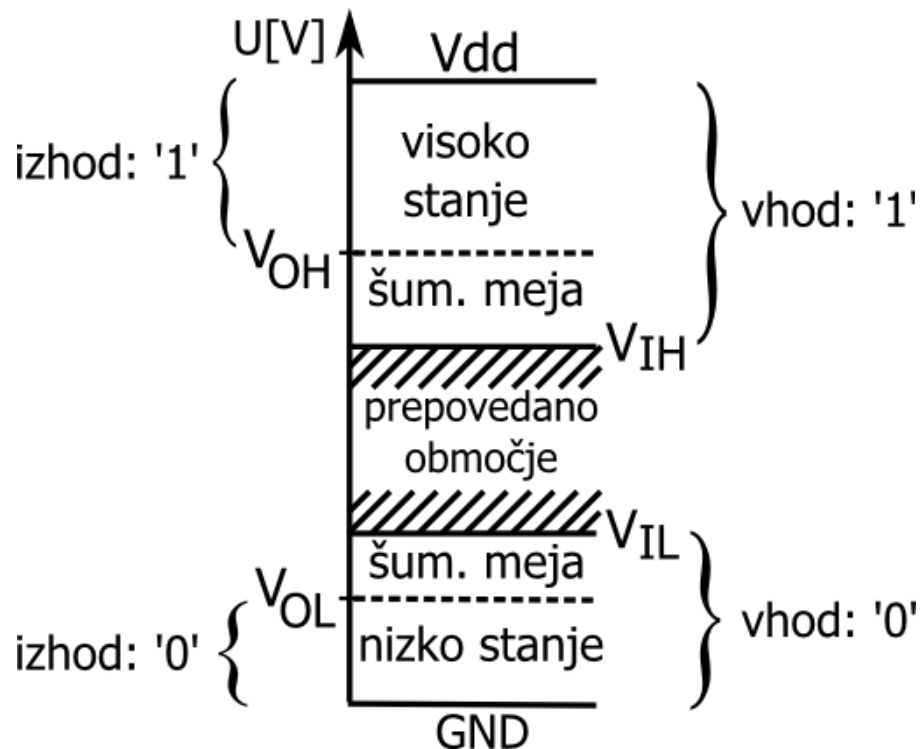
---



# Rešitev: statični red

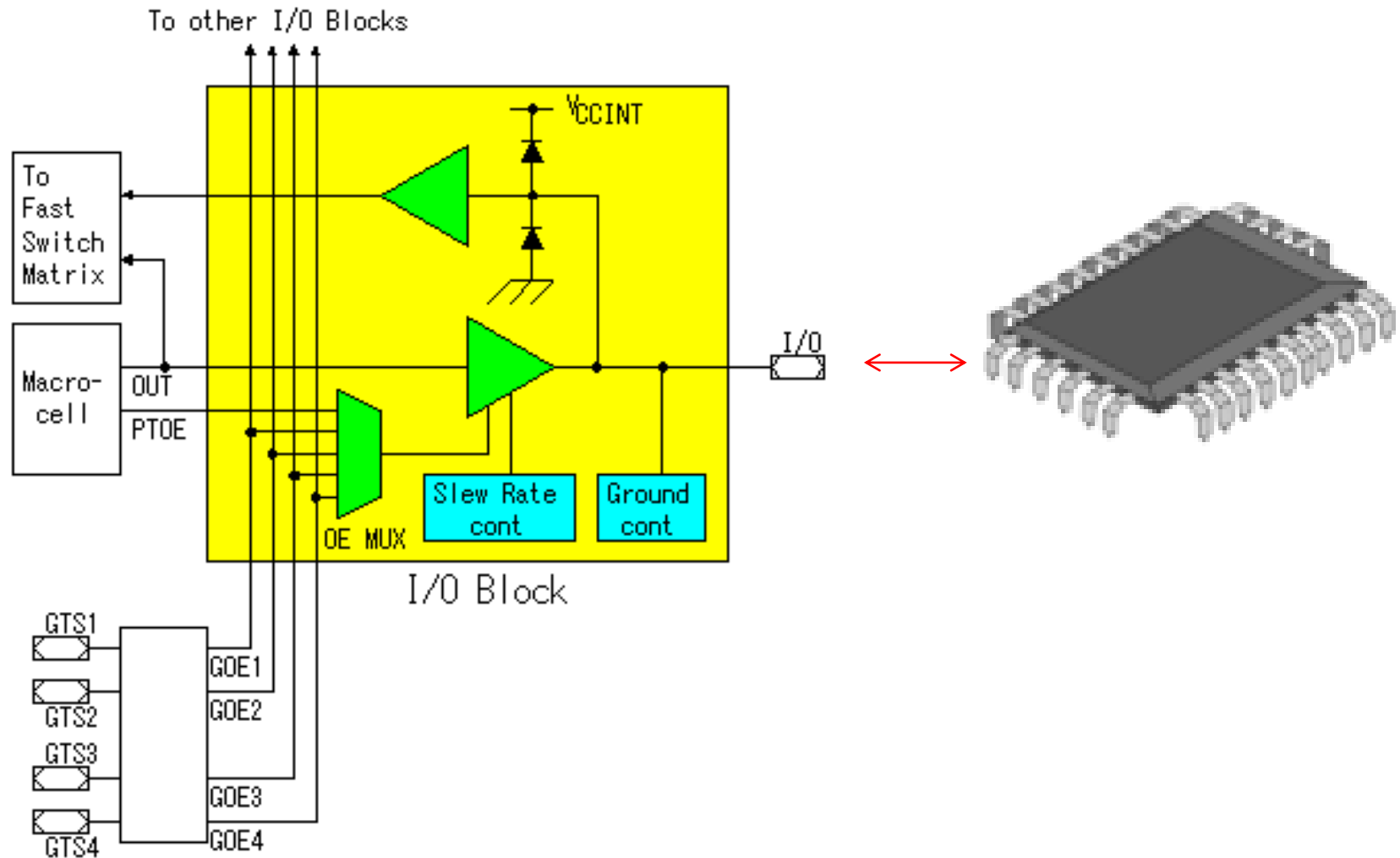
- ▶ logični izhod ima manjše območje kot za vhod

- ▶ Npr. podatki za 5V CMOS:



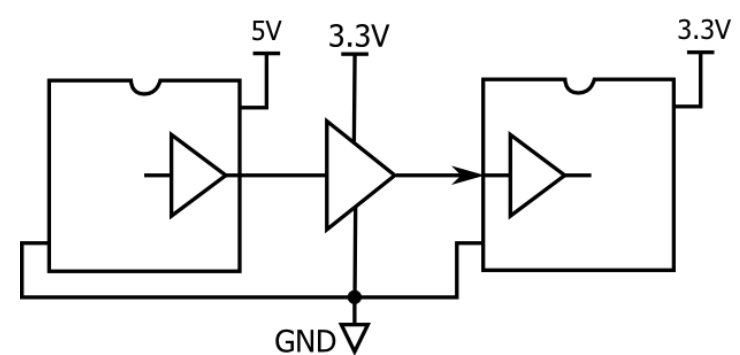
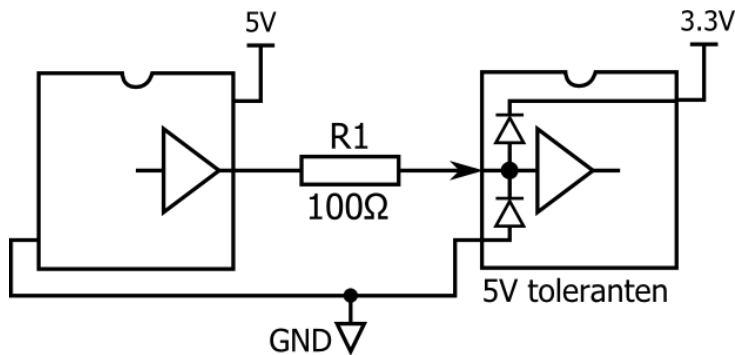
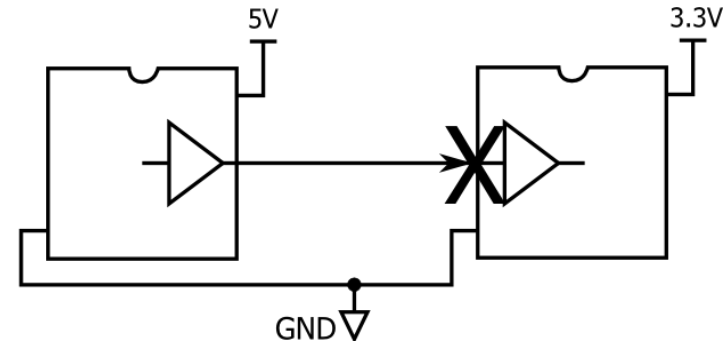
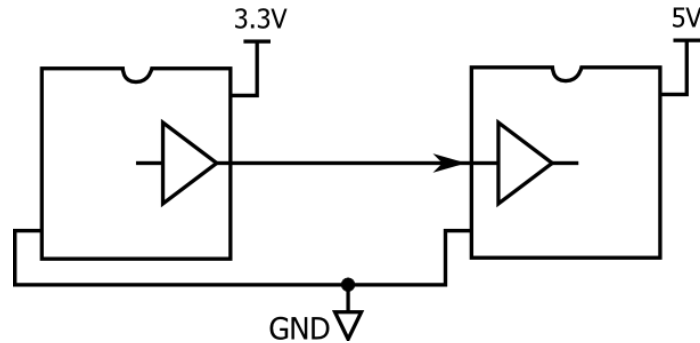
| oznaka   | pomen              | napetost[ V] |
|----------|--------------------|--------------|
| $V_{IH}$ | vhodni visok nivo  | 3            |
| $V_{IL}$ | vhodni nizek nivo  | 1            |
| $V_{OH}$ | izhodni visok nivo | 3.1          |
| $V_{OL}$ | izhodni nizek nivo | 0.2          |

# Priključki programirljivega vezja

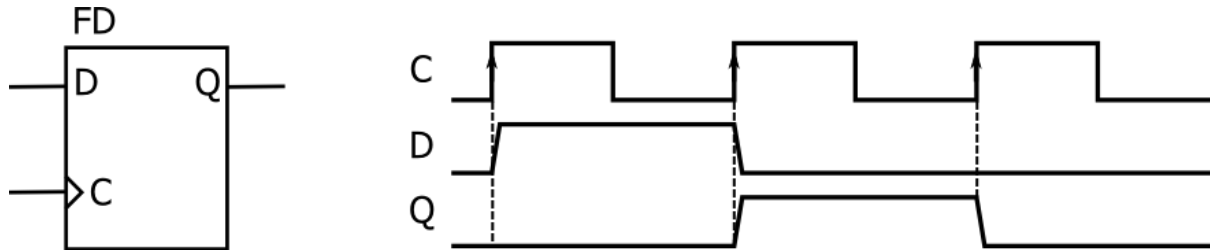


# Vezava signalov CMOS (5V) in LVCMOS (3.3V)

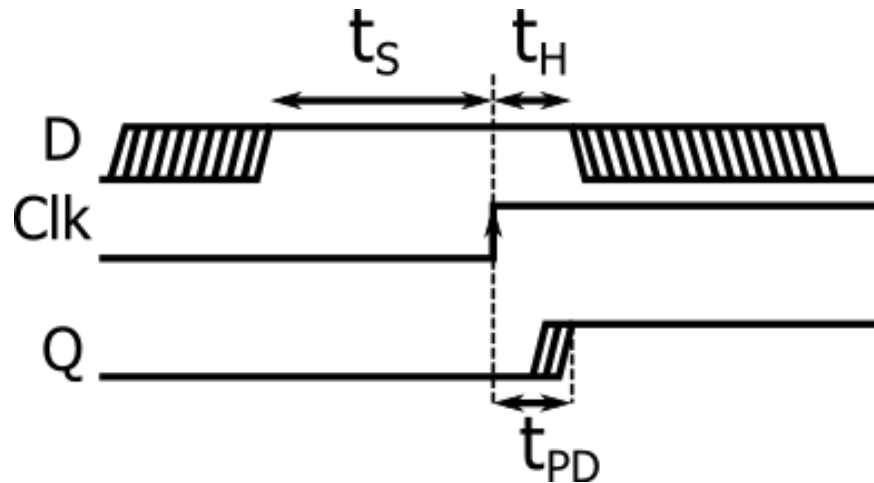
- ▶ Preveriti je potrebno statični red!
  - ▶ iz 3.3V na 5V gre, v obratni smeri pa je potrebna prilagoditev



# Realni pomnilni gradniki (flip-flopi)

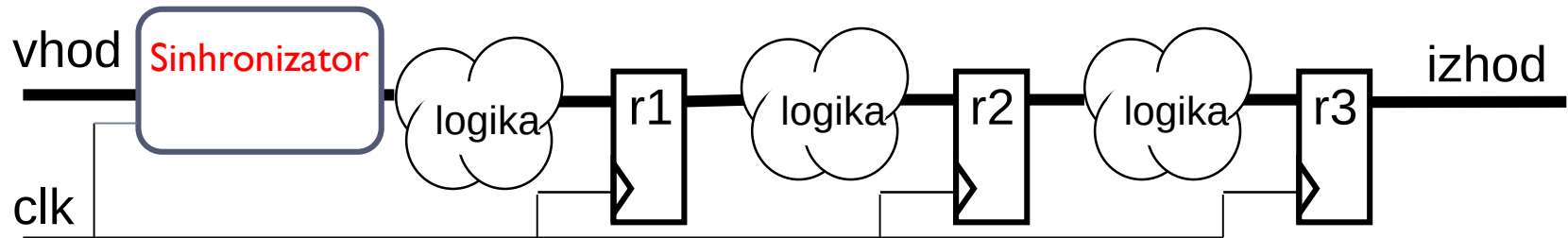


► Realni časovni diagram:



# Dinamični red

---



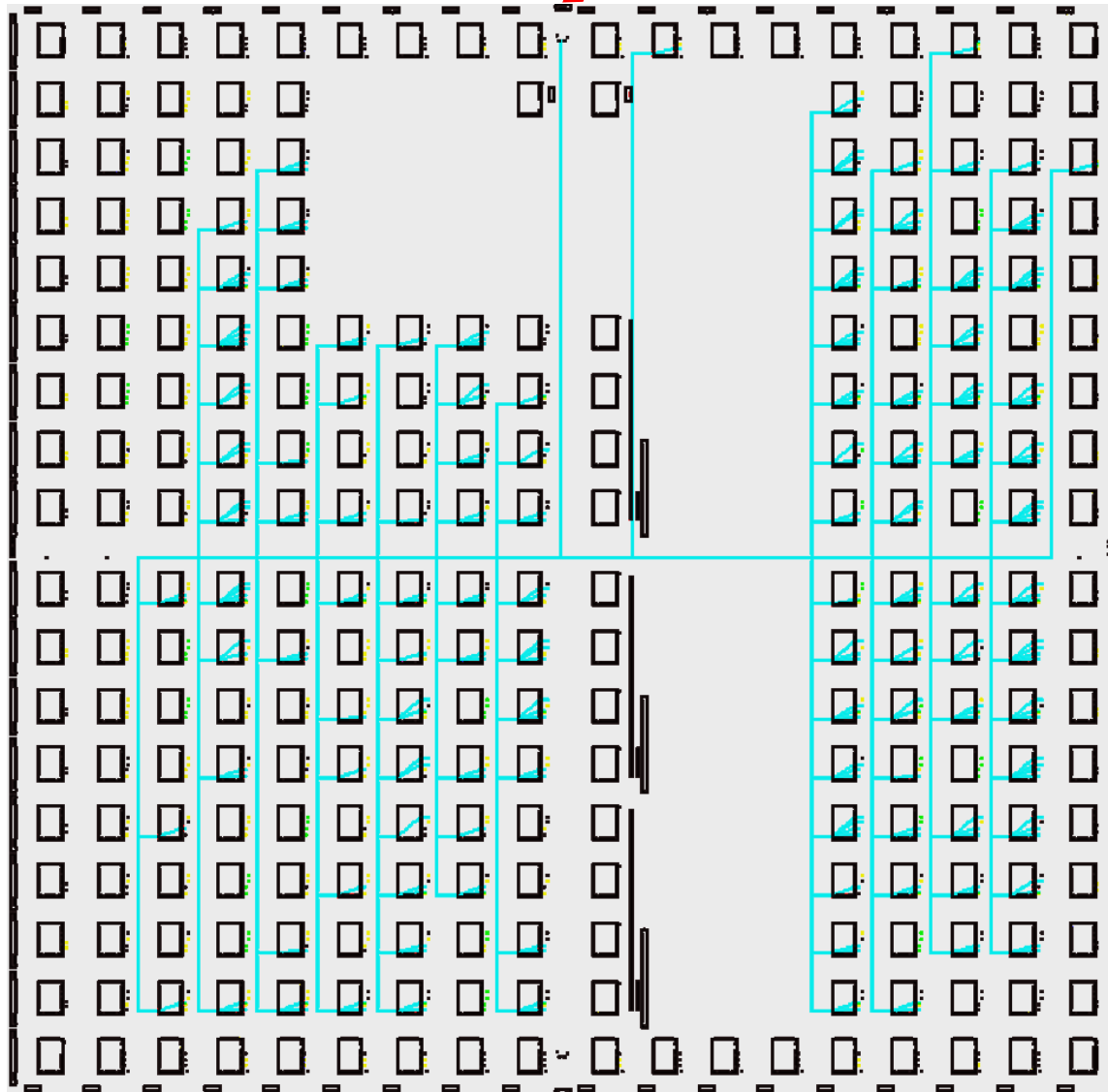
S skrbnim načrtovanjem bo vedno izpolnjen dinamični red

- ▶ ura mora priti do vseh pomnilnih elementov istočasno
- ▶ sinhrono sekvenčno vezje

Na asinhronem vhodu ne moremo upoštevati pravil !

- ▶ vhodi se ne spreminjajo po zakonitostih naše ure in ne moremo zagotavljati dinamičnega reda ( $t_s$  in  $t_H$ )

vhod clk



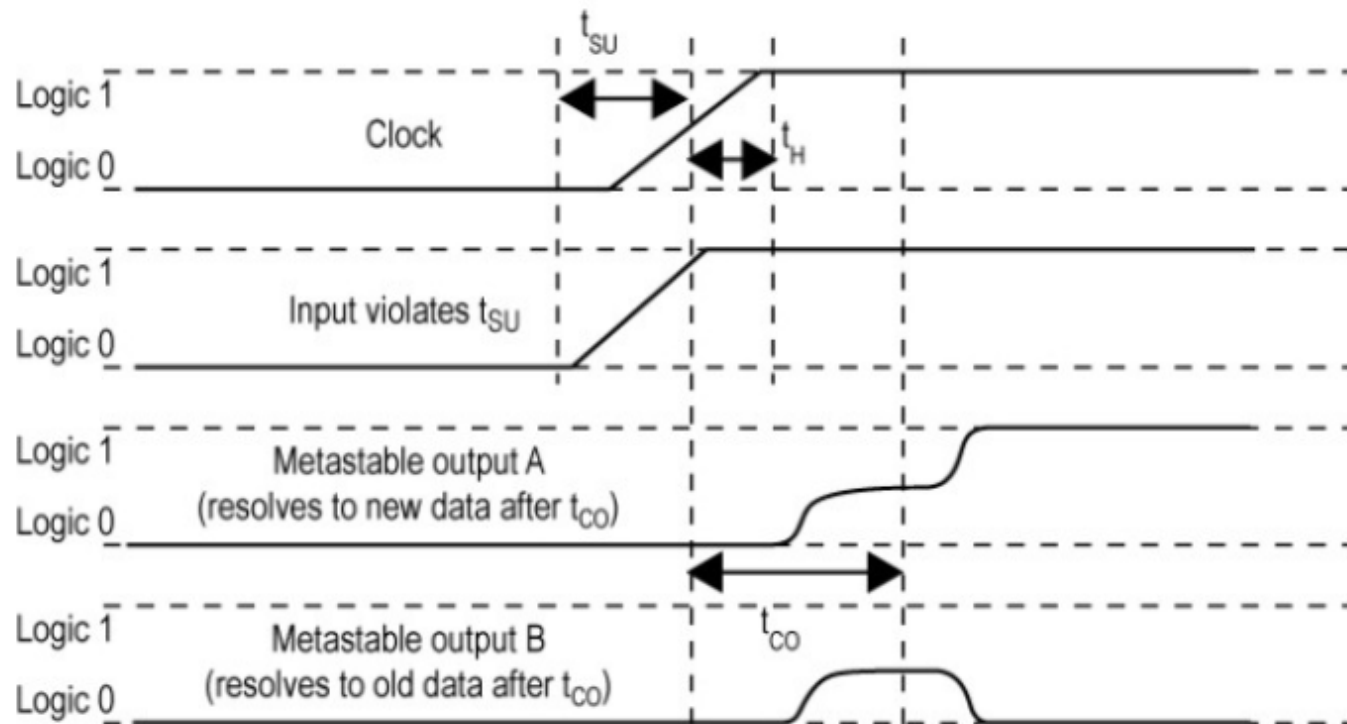
## Vezava ure v čipu

Struktura H povezav  
zagotavlja enake  
zakasnitve do vseh celic

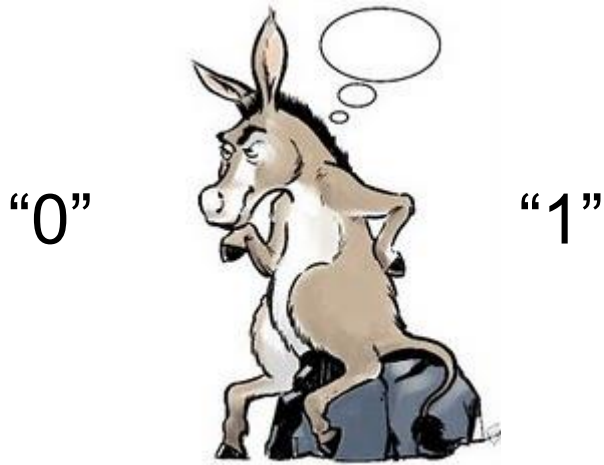
# Vezje za sinhronizacijo

## Poskus rešitve: uporabimo D flip-flop

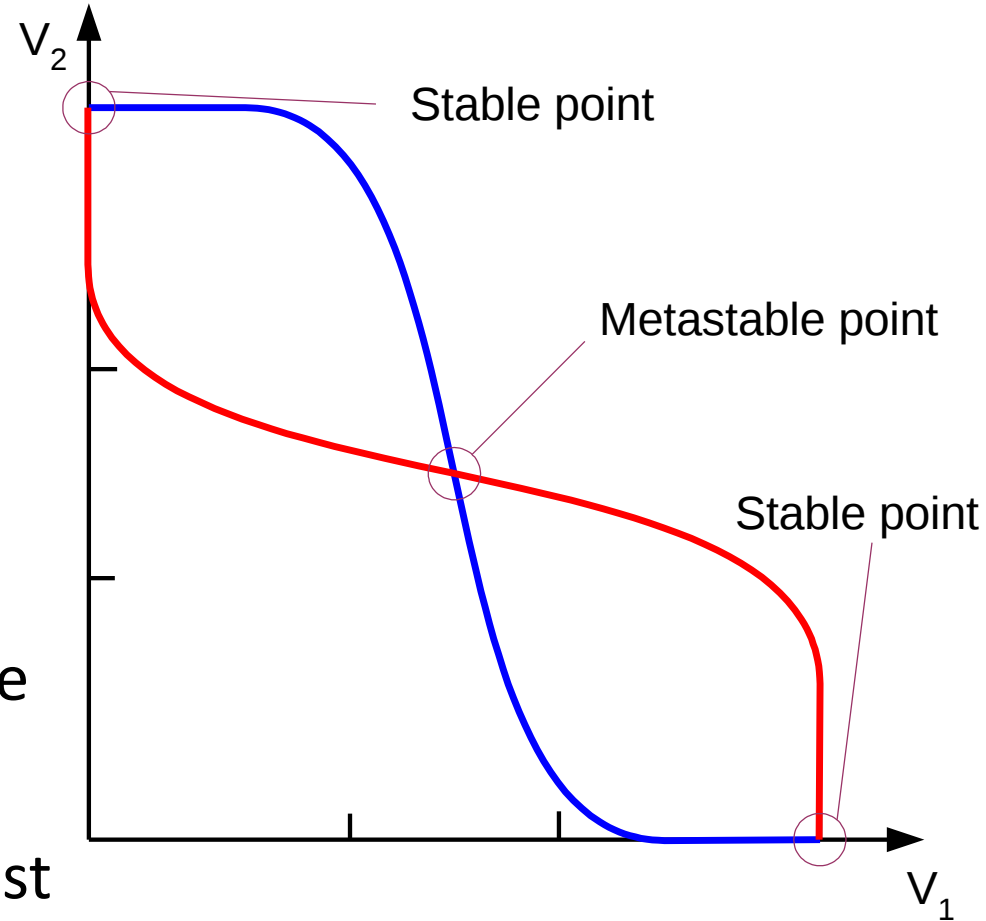
- ▶ obstaja možnost, da gre v metastabilno stanje
- ▶ čez (nedoločen) čas gre izhod v eno ali drugo stabilno stanje



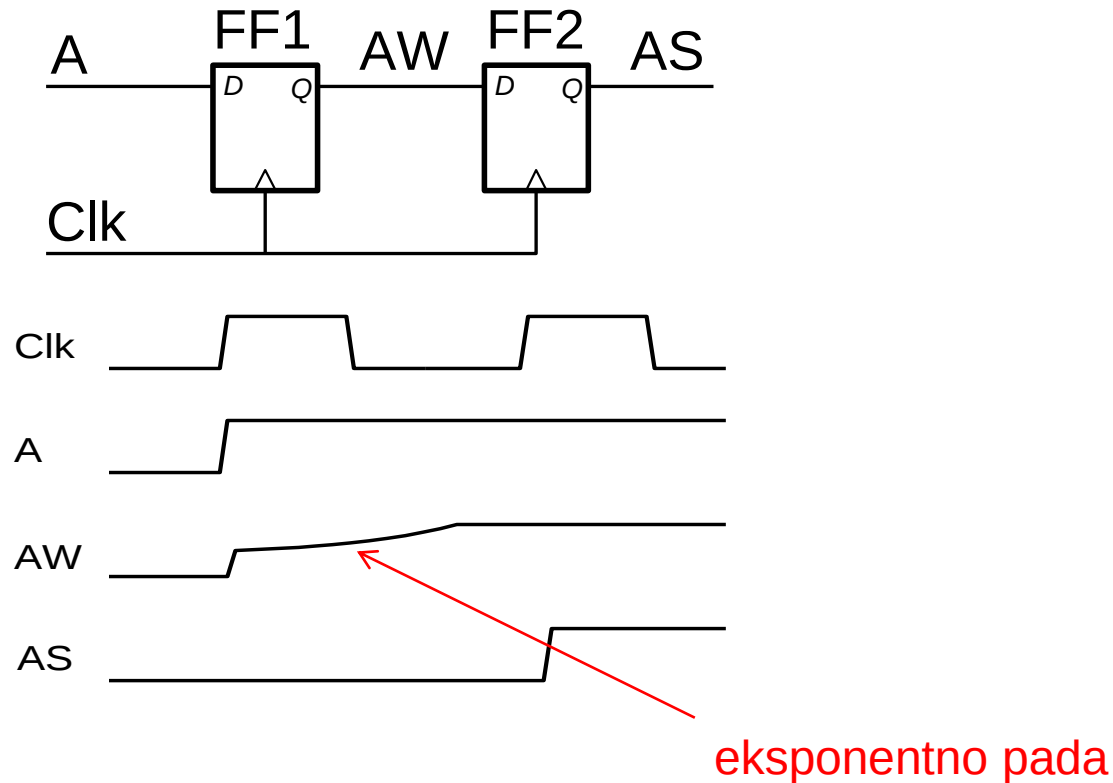
# Metastabilno stanje



- ▶ lastnost bistabilnih vezij
- ▶ čez čas gre v stabilno stanje
- ▶ nedoločen čas okrevanja
- ▶ verjetnost za metastabilnost eksponentno pada s časom



# Sinhronizacija z več D flip-flopi

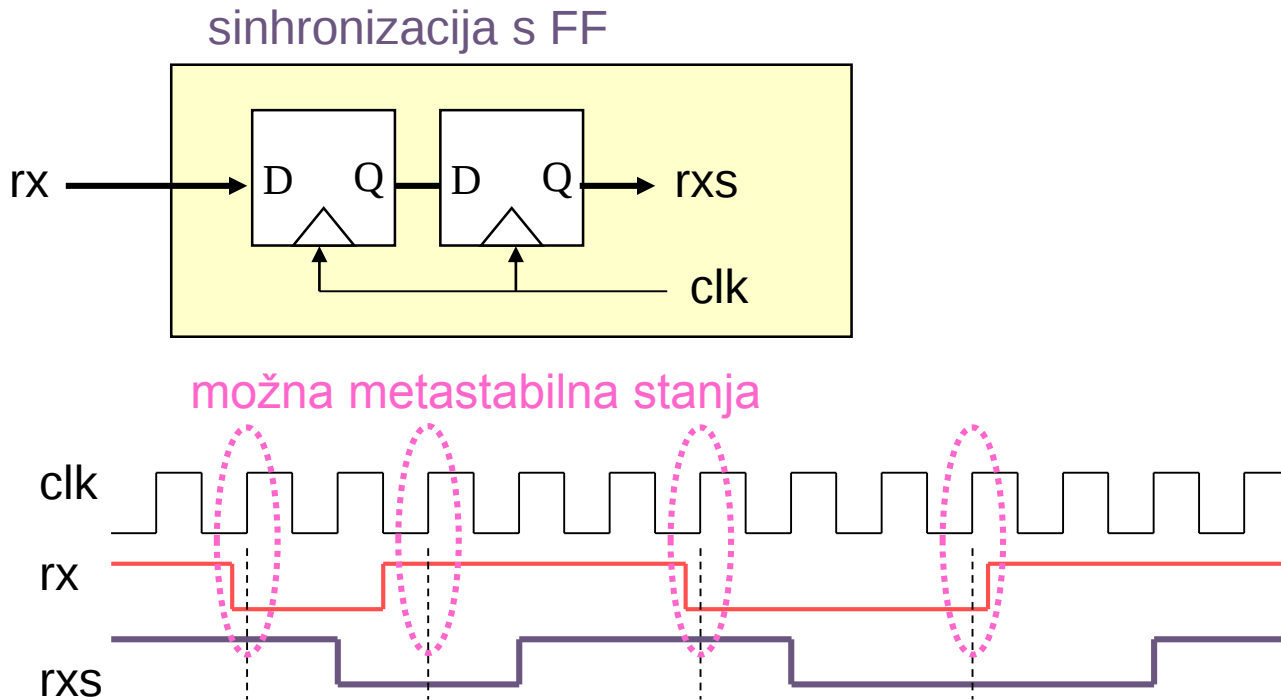


$$P(\text{napake}) = P(\text{metastab stanja}) \times P(\text{ni še stabilno po } t_w)$$

- Zakasnitev signala je cena za sinhronizacijo, ki se ji ne moremo izogniti !

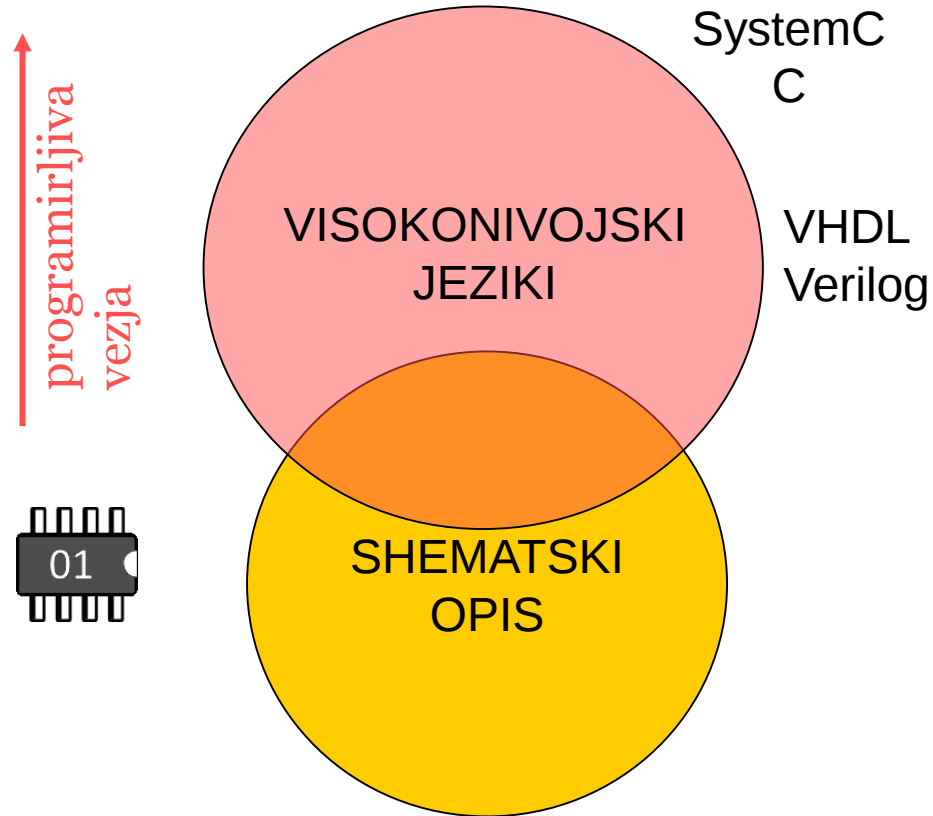
# Načrtovanje vmesnikov

- ▶ problem komunikacijskih vmesnikov je **sinhronizacija**
  - ▶ asinhronne signale vzorčimo z višjo frekvenco ure in jih peljemo čez sinhronizacijsko vezje
- ▶ težav z metastabilnostjo ne vidimo na simulaciji !



# Ravni (nivoji) modeliranja digitalnih vezij

- ▶ specifikacija
  - ▶ postopkovni (behavioral)
  - ▶ funkcijski (dataflow, RTL)
  - ▶ logični
- 
- ▶ transistorska shema
  - ▶ geometrija vezja (layout)
- 
- ▶ Standardizirani jeziki (IEEE)
    - ▶ VHDL
    - ▶ Verilog, System Verilog
    - ▶ SystemC



# Načrtovanje s programirljivimi vezji

1. Opis vezja in simulacija (Design Entry Utilities)
2. Sinteza logičnega vezja (Synthesize)
3. Prevajanje in tehnološka preslikava
  - določimo lokacije priključkov (User Constraints)
4. Izdelava prog. datotek in nalaganje vezja

## Proces za CPLD

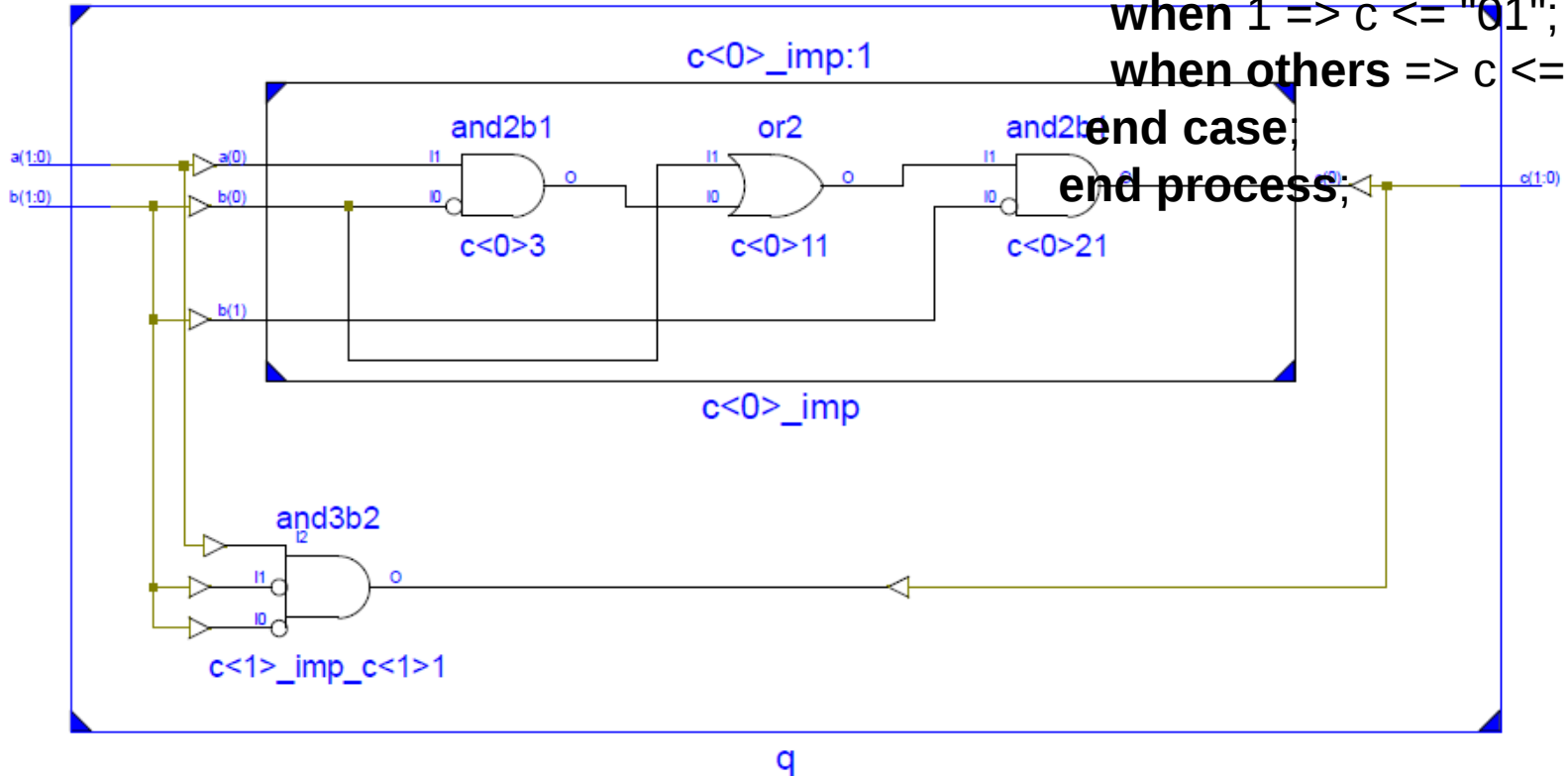


## Proces za FPGA



## Kaj dela opisano vezje?

**always @ (a or b)**  
**case (b)**



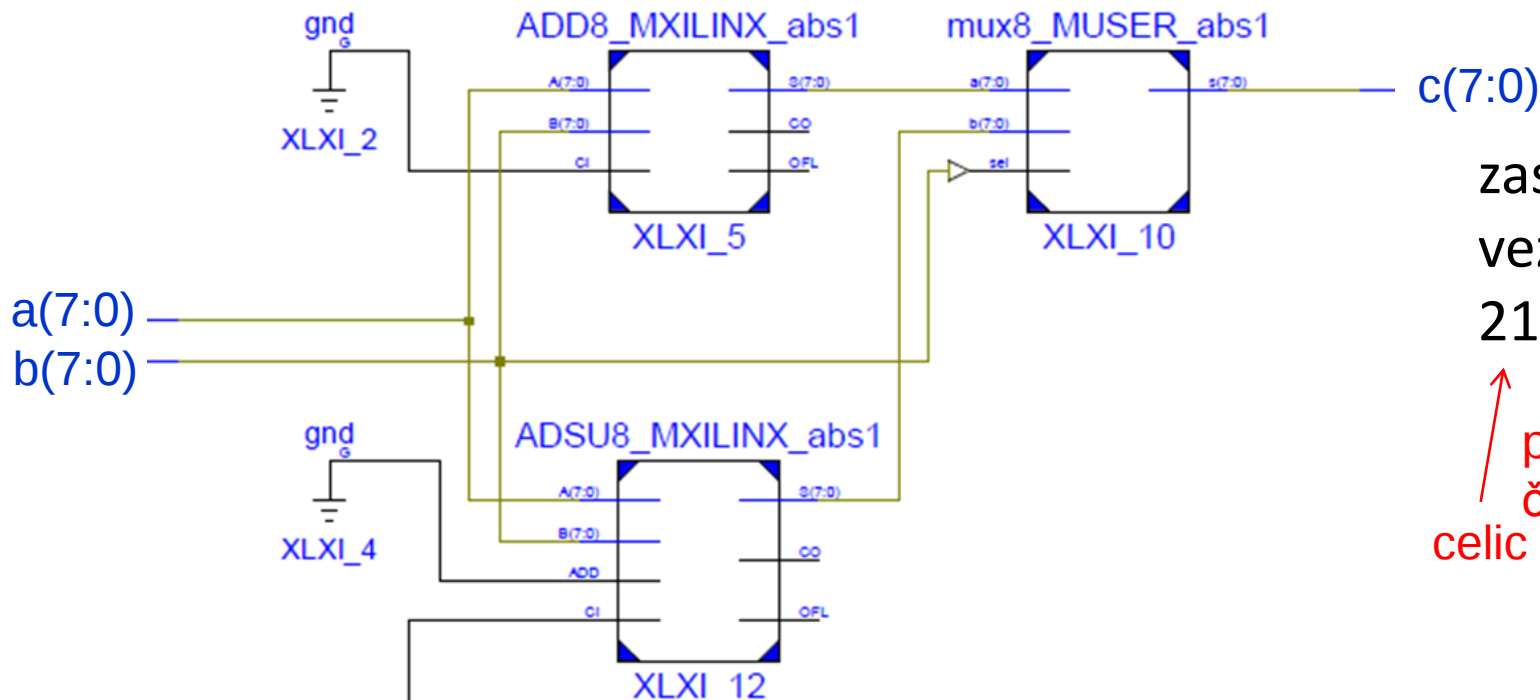
```
process (a, b)
begin
  case b is
    when 0 => c <= a;
    when 1 => c <= "01";
    when others => c <= "00";
  end case;
end process;
```

# Sinteza vezja iz opisa v jeziku VHDL

`c <= a+b when b>0 else a-b;`

► Kaj naredi program za sintezo vezja?

- operatorji +, - so kombinacijska vezja: seštevalnik, odštevalnik
- izbirni stavek when ... else je izbiralnik



zasedenost  
vezja CPLD

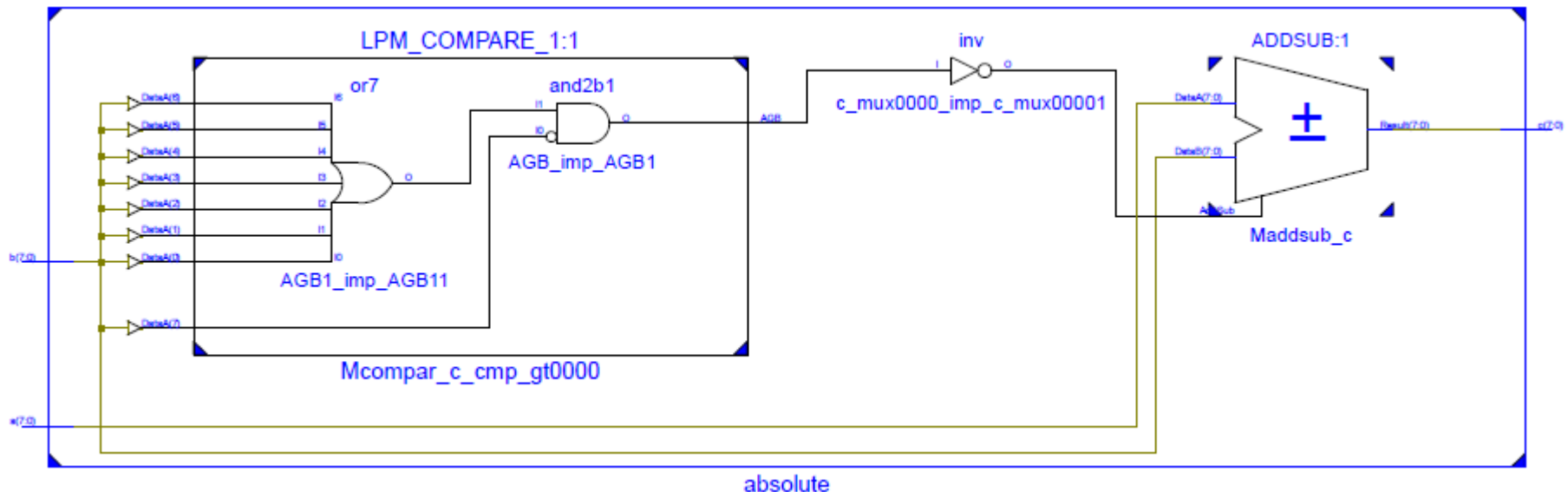
21 / 63 / 0

celic      produkt.  
                 členov  
                 flip-flopov

# Postopek sinteze vključuje optimizacijo

$c \leq a+b$  **when**  $b > 0$  **else**  $a-b$ ;

- ▶ primerjalnik (or7, and2) in ADDSUB blok
- ▶ zasedenost 19 / 55 / 0



▶ Drug zapis:  $c \leq a+b$  **when**  $b \geq 0$  **else**  $a-b$ ;

- ▶ zasedenost 17 / 53 / 0

# Načrtovanje na nivoju registrov

- ▶ **RTL**: delitev na krmilni in podatkovni del
- ▶ določitev zaporedja operacij (avtomat)
- ▶ opis gradnikov na podatkovni poti

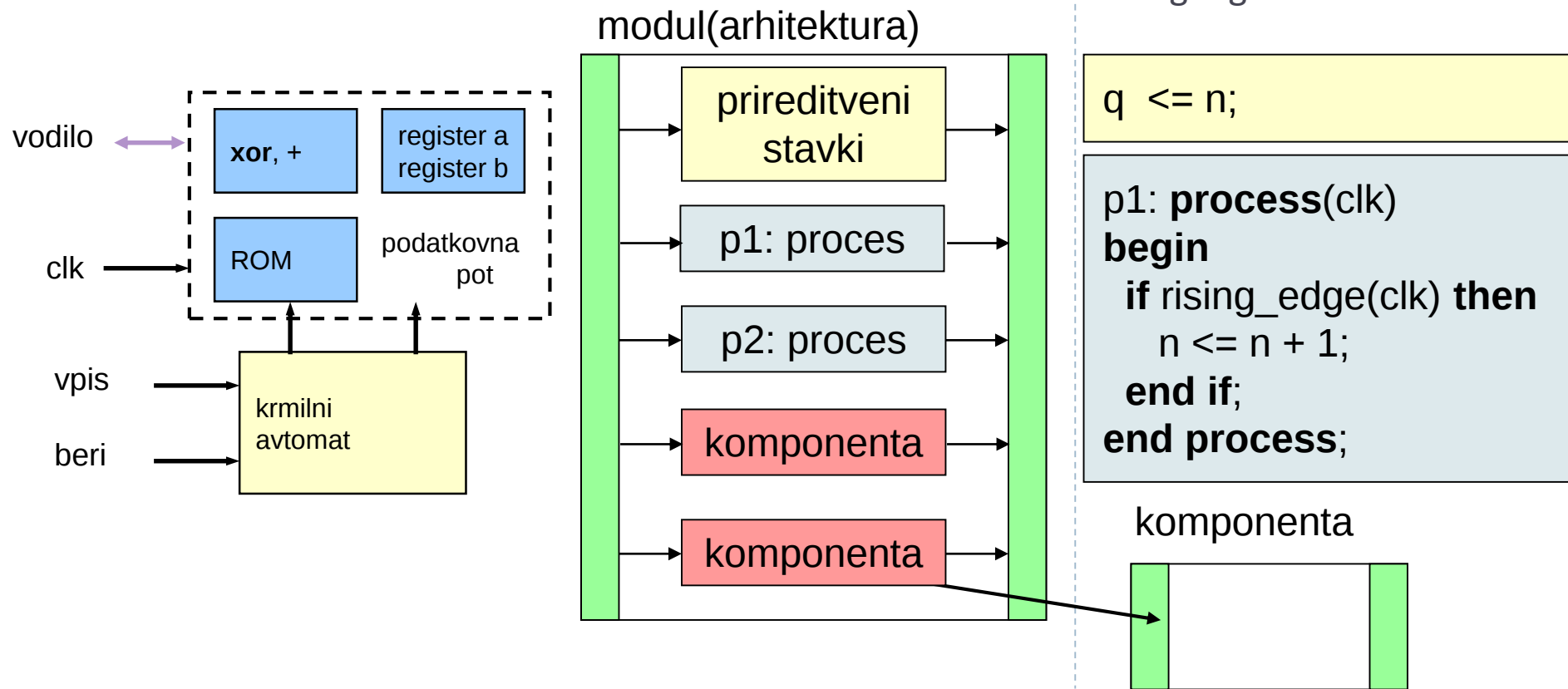
## VHDL

Very high-speed IC

Hardware

Description

Language

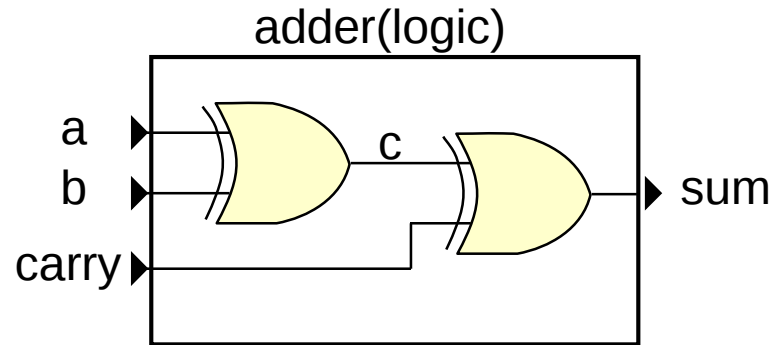


# Funkcijski opis vezja v jeziku VHDL

- ▶ stavki opisujejo gradnike vezja
  - ▶ stavki za opis vezja se izvajajo paralelno
  - ▶ vrstni red stavkov ni pomemben (sočasni stavki)

```
entity adder is  
  port ( a, b : in std_logic;  
         carry : in std_logic;  
         sum : out std_logic);  
end adder;
```

```
architecture logic of adder is  
  signal c : std_logic;  
begin  
    sum <= c xor carry;  
    c <= a xor b;  
end one;
```



dekларacija notranjega signala

# Postopkovni opis vezja v jeziku VHDL

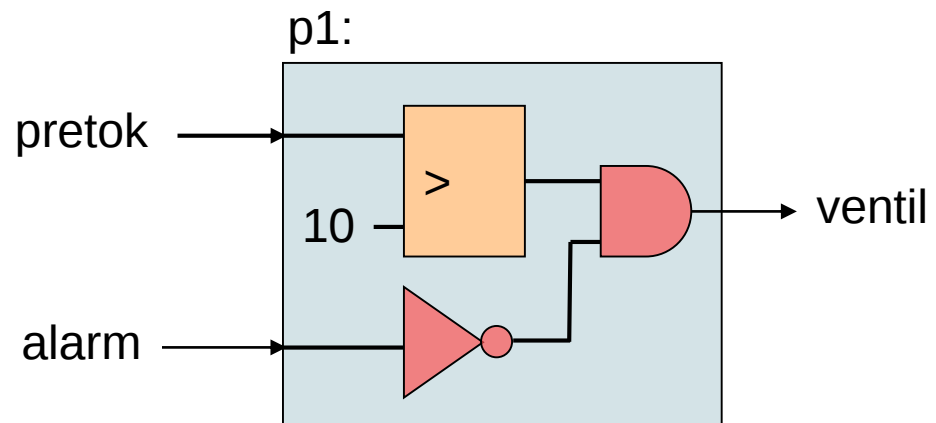
- ▶ v procesu opišemo delovanje vezja
  - ▶ zgradbo vezja določi program za sintezo vezij
  - ▶ vrstni red stavkov je pomemben (sekvenčni stavki)

arhitektura

**p1: process**

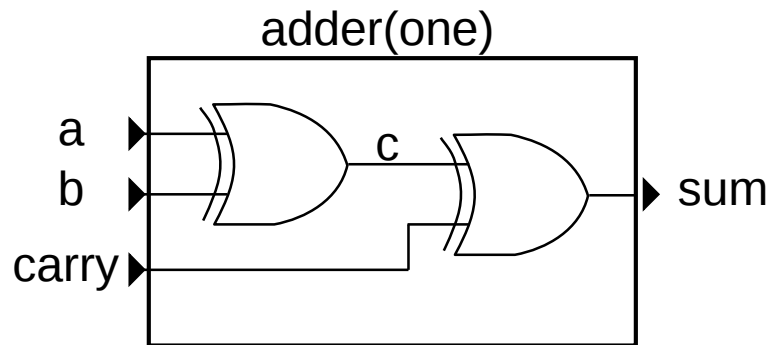
```
ventil <= '0';  
if pretok > 10 then  
    ventil <= '1';  
end if;  
if alarm = '1' then  
    ventil <= '0';  
end if;
```

**end process;**

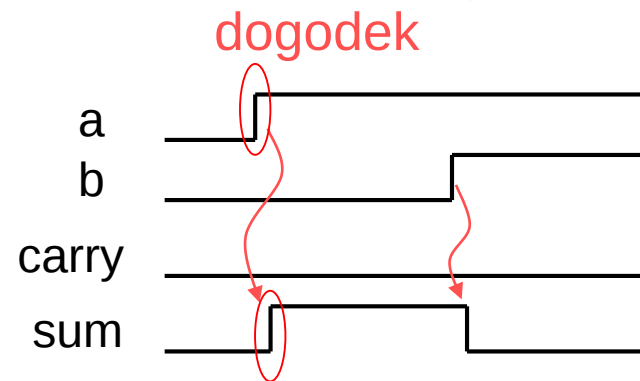


# Obravnava VHDL modelov vezij

```
architecture logic of adder is  
  signal c : std_logic;  
begin  
  sum <= c xor carry;  
  c <= a xor b;  
end one;
```

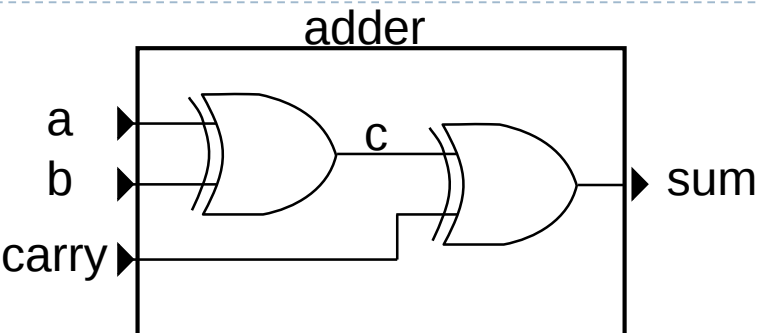


Sintetizirano vezje



Graf simulacije (waveform)

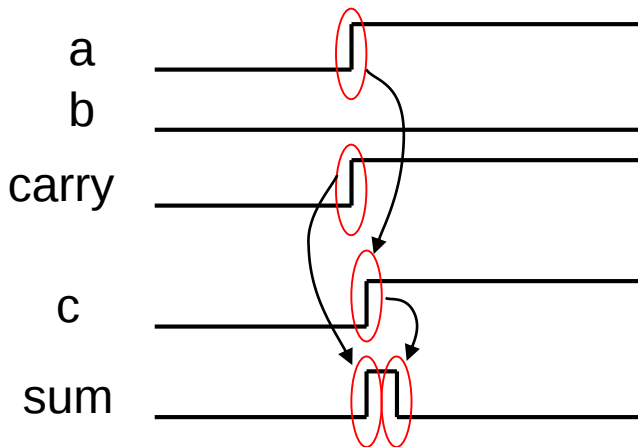
# Potek simulacije



seznam dogodkov:

a: 0, 1 (T)    carry: 0, 1 (T)

dogodek

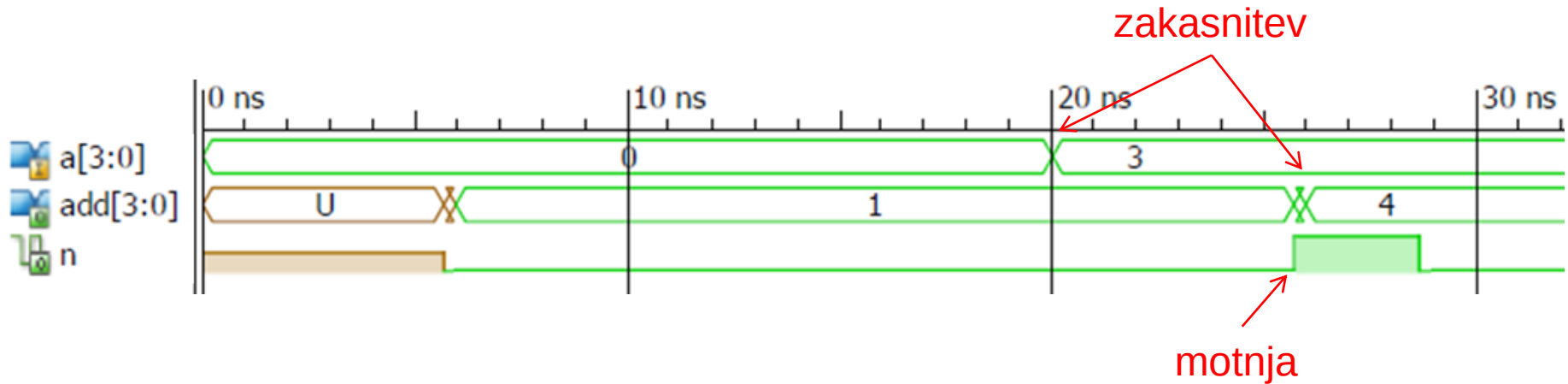


| korak     | čas         | a       | b | carry   | sum            | c             |
|-----------|-------------|---------|---|---------|----------------|---------------|
| seznam    | 0           | 0, 1(T) | 0 | 0, 1(T) | 0              | 0             |
| izvrši    | T           | 1       | 0 | 1       | 0              | 0             |
| izračunaj | T           | 1       | 0 | 1       | $1(T+\Delta)$  | $1(T+\Delta)$ |
| izvrši    | $T+\Delta$  | 1       | 0 | 1       | 1              | 1             |
| izračunaj | $T+\Delta$  | 1       | 0 | 1       | $0(T+2\Delta)$ | 1             |
| izvrši    | $T+2\Delta$ | 1       | 0 | 1       | 0              | 1             |

# Lastnosti realnih kombinacijskih vezij

## Primer: seštevalnik in primerjalnik

```
add <= a + 1;  
n <= '1' when add=0 else '0';
```



# Povzetek

---

- ▶ Opiši model priključkov digitalnega vezja v izvedbi CMOS.
  - ▶ Zakaj potrebujemo kondenzatorje ?
  - ▶ Kako delujejo dvosmerni priključki ?
- ▶ Razloži prenos logičnih vrednosti med digitalnima vezjema.
  - ▶ Kaj določa statični red?
- ▶ Pojasni razlike med logičnim modelom in realnim vezjem.
  - ▶ Opiši delovanje realnih gradnikov.
  - ▶ Kaj je potrebno zagotoviti za zanesljivo delovanje flip-flopa?
- ▶ Razloži ravni (nivoje) modeliranja digitalnih vezij.
  - ▶ Kaj je VHDL in katere ravni obsega?