



Laboratorij za načrtovanje integriranih vezij

Univerza *v Ljubljani*
Fakulteta *za elektrotehniko*



Digitalni Elektronski Sistemi

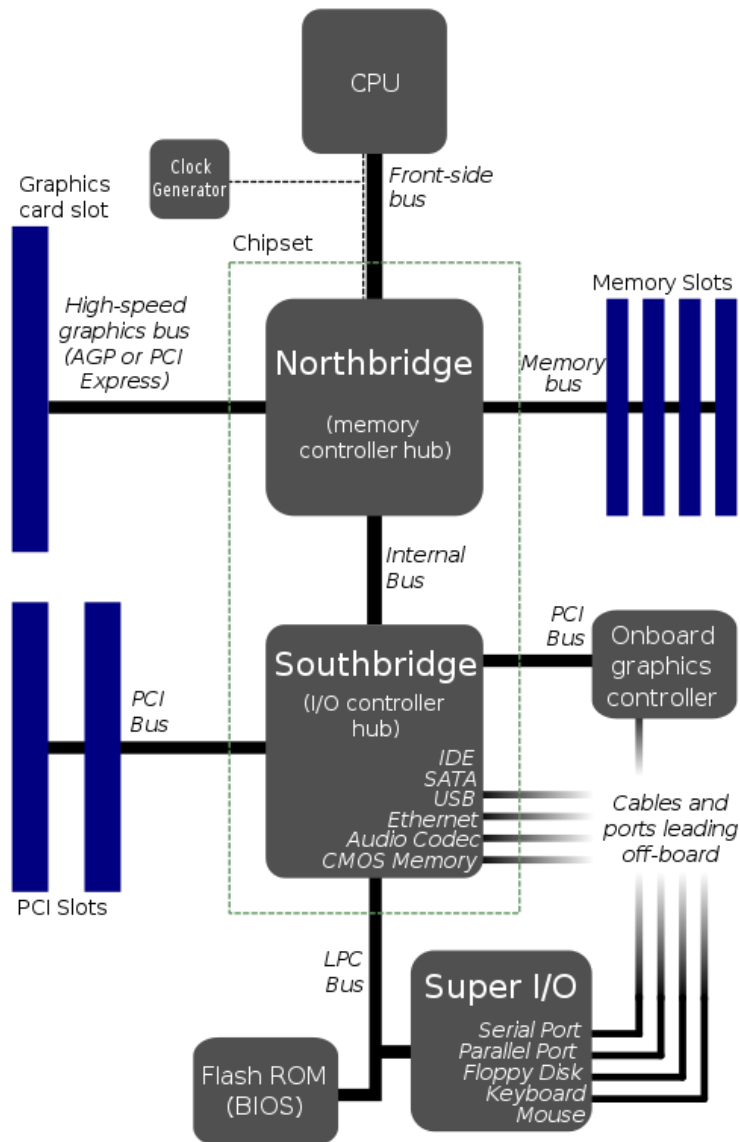
Zaporedni vmesniki

Vmesniki in vodila - 2

Razvoj komunikacije v digitalnih sistemih

- ▶ Vzporedni prenos podatkov po tiskanem vezju s standardnimi napetostnimi nivoji (5V TTL/CMOS)
- ▶ Paralelna vodila na plošči osebnega računalnika
 - ▶ vodilo ISA (Industry Standard Architecture, 1981/84)
 - ▶ IBM AT vodilo prenaša 16 bitov pri 8 MHz, TTL
 - ▶ paralelna vodila za disk (IDE/ATA)
- ▶ Vzporedno vodilo Centronics
 - ▶ prenos podatkov do tiskalnika, na krajše razdalje
- ▶ Zaporedna vodila
 - ▶ RS232, diferencialni nivoji omogočajo večje razdalje
- ▶ Hitra zaporedna vodila
 - ▶ USB, Ethernet, PCI Express

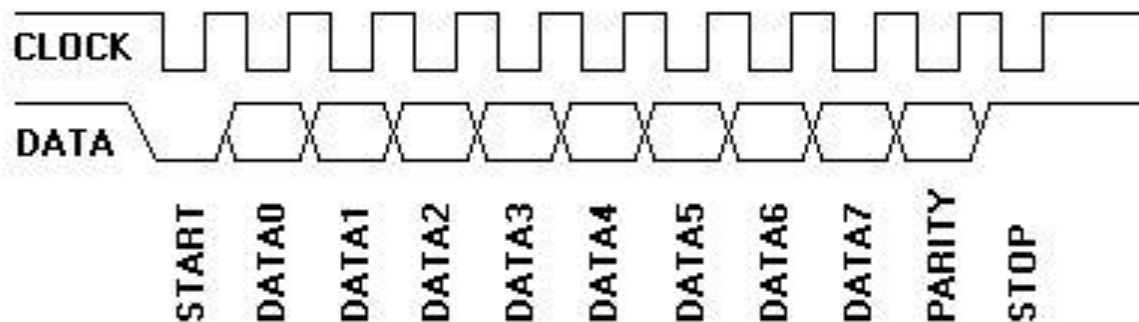
Vodila v osebnem računalniku



- ▶ **systemsko vodilo**
 - ▶ odvisno od procesorja
- ▶ **pomnilnik SDRAM**
 - ▶ 64 bitov, 133 MHz, SSTL 2.5V
 - ▶ do 533 MHz, SSTL 1.5V
- ▶ **vzporedno vodilo PCI**
 - ▶ 32 bitov, 33 MHz (133MB/s)
 - ▶ 64 bitov, 66 MHz (533MB/s)
- ▶ **hitri zaporedni vmesniki**
 - ▶ PCIe (1-16 GB/s)
 - ▶ SATA (1.5-6 GB/s)
- ▶ **vmesniki za zunanja vodila**
 - ▶ zaporedni USB, Ethernet
 - ▶ RS232, PS/2

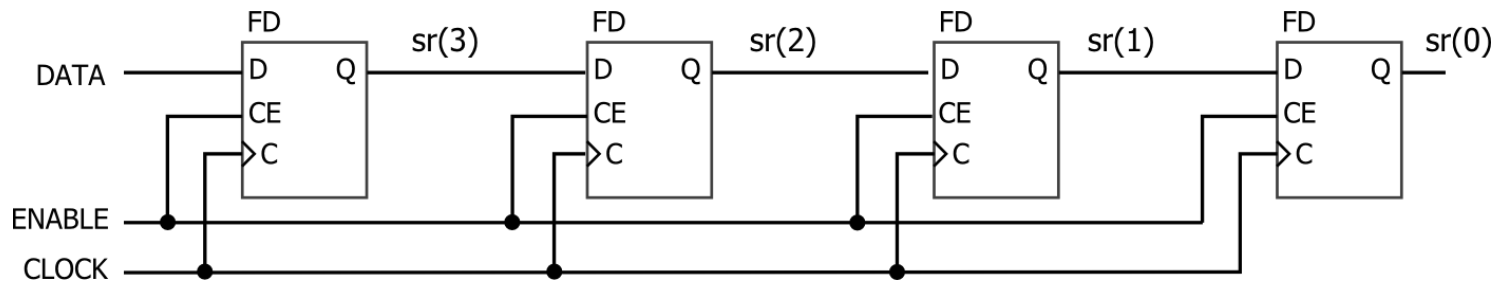
Zaporedni (serijski) vmesniki

- ▶ Najprimernejši za komunikacijo na daljše razdalje
- ▶ Asinhroni vmesniki
 - ▶ pošiljajo le podatke, uro mora sprejemnik rekonstruirati
- ▶ Sinhroni serijski vmesniki
 - ▶ pošiljajo podatke in uro
- ▶ Protokol določa vrstni red podatkovnih in kontrolnih bitov
 - ▶ Npr. vmesnik PS/2 pošlje najprej start ('0'), nato pa podatkovne bite od najnižjega (LSB) proti najvišjemu (MSB)...



Zaporedna pretvorba podatkov v vzporedno

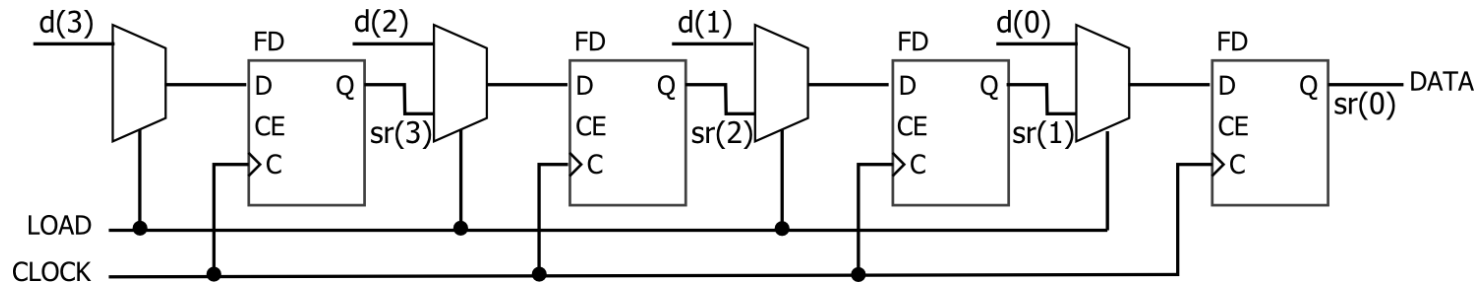
- ▶ Pretvorba s pomikalnim registrom (SIPO)
 - ▶ zaporedno vežemo toliko DFF, kolikor je dolg podatkovni paket
 - ▶ paziti moramo na zaporedje – prvi poslani bit je MSB ali LSB
- ▶ Npr. 4-bitni pomikalni register, prvi bit je LSB
 - ▶ podatki se pomikajo v desno, proti LSB



```
p: process (CLOCK)
begin
  if rising_edge(CLOCK) and ENABLE='1' then
    sr <= DATA & sr(3 downto 1);
  end if;
end process;
```

Vzporedna zaporedna pretvorba (oddajnik)

▶ Zaporedni oddajnik s pomikalnim registrom (PISO)

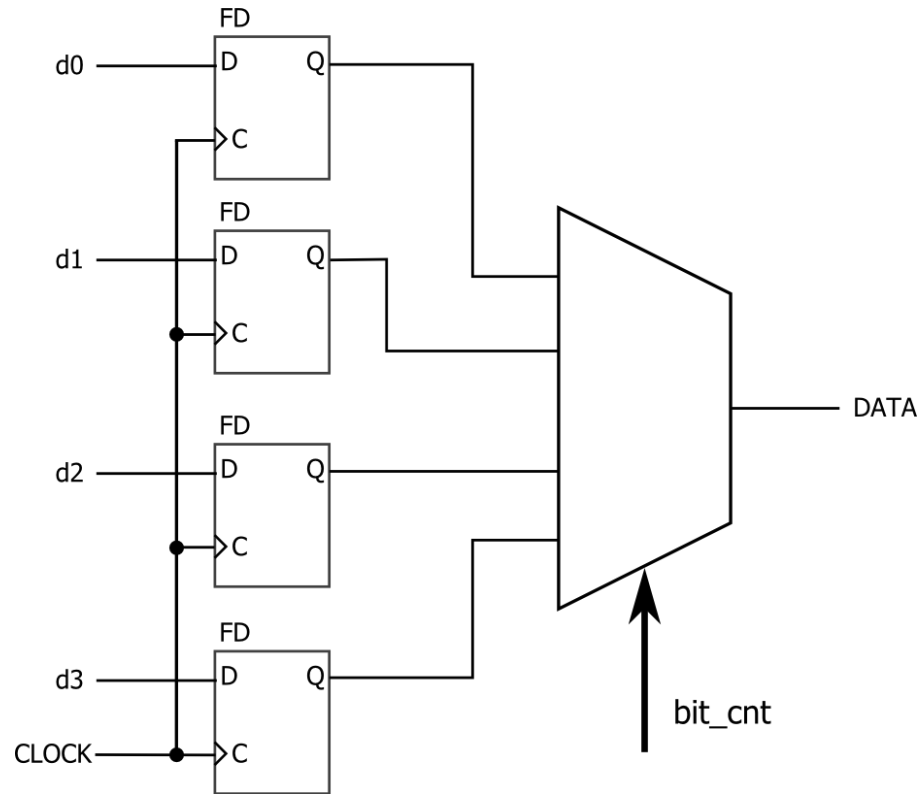


```
p: process (CLOCK)
begin
  if rising_edge(CLOCK) then
    if LOAD='1' then
      sr <= d;
    else
      sr <= '0' & sr(3 downto 1);
    end if;
  end if;
end process;

DATA <= sr(0);
```

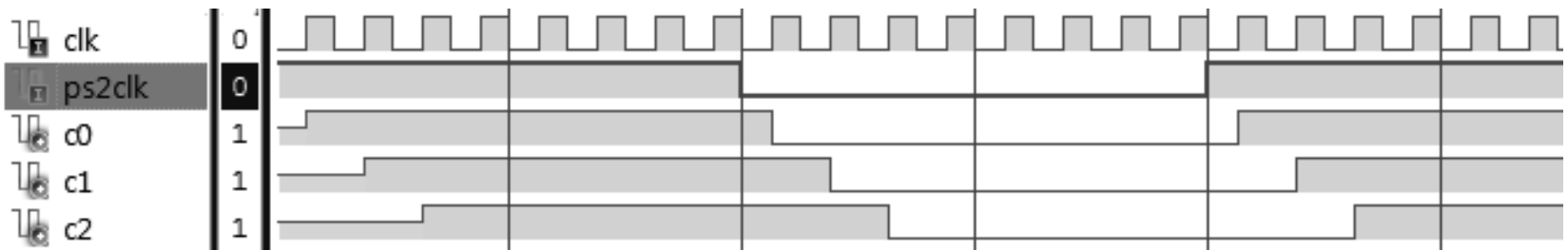
Pretvorba z izbiralnikom

- ▶ Zaporedno vzporedno pretvorbo lahko naredimo z registrom in izbiralnikom
 - ▶ števec bitov (`bit_cnt`) določa kateri podatek gre na izhod



Sinhronizacija sprejetih podatkov

- ▶ Digitalni sistem običajno dela s svojo uro
 - ▶ potrebujemo sinhronizacijo podatkov, ki se pomikajo z uro, ki jo določa hitrost prenosa
- ▶ Sinhronizacija z medpomnilniki
 - ▶ uporabimo pomnilnik FIFO (First-In First-Out) z dvema vrati, na eni strani vpisuje podatke sprejemnik, na drugi jih bere sistem
- ▶ Sinhronizacija ob vzorčenju
 - ▶ potrebujemo vsaj 2x višjo frekvenco kot je hitrost prenosa
 - ▶ vzorčimo uro in podatke ter detektiramo prehode ure za določanje trenutka zajemanja podatkov

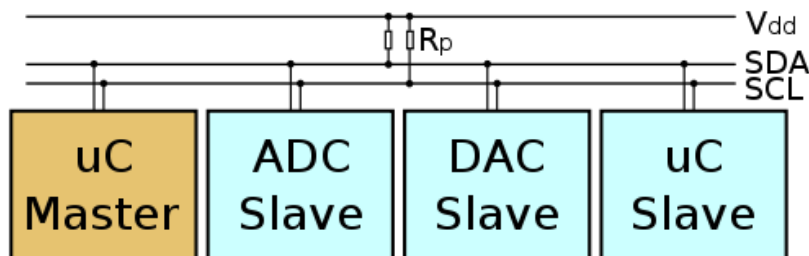


Lastnosti zaporednih vmesnikov

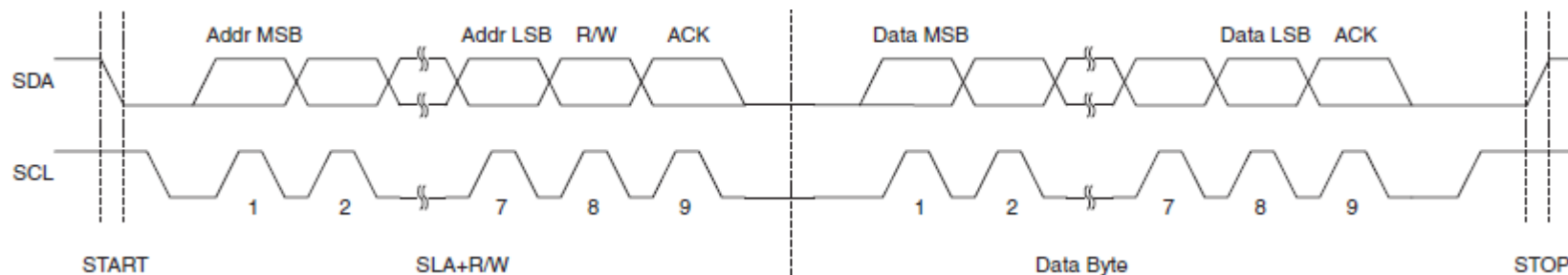
- ▶ Bitna hitrost = frekvenca s katero se prenašajo posamezni biti, določa periodo za posamezni bit ($T = f_{\text{bit}}$)
- ▶ Podatkovni okvir
 - ▶ podatkovne bite vedno spremlja nekaj kontrolnih bitov
 - ▶ podatkovni okvir – start, kontrolna vsota ali pariteta, stop
 - ▶ okvir omogoča usklajevanje sprejemnika in oddajnika in detekcijo ali odpravljanje napak pri prenosu
 - ▶ hitrost prenosa podatkov (podatkovni pretok) je zaradi okvirja manjša od bitne hitrosti
- ▶ Prenos poteka po povezavah z običajnimi logičnimi nivoji (npr. na tiskanem vezju) ali diferencialnih povezavah (za večje razdalje ali hitrosti prenosa)

Sinhroni zaporedni vmesnik – I2C

- ▶ Vodilo sestavlja ura (SCL) in podatkovna linija (SDA)
 - ▶ I2C – serijski EEPROM, D/A in A/D pretvorniki, senzorji
 - ▶ Omogoča povezavo več enot na vodilo, pasivna logična '1'

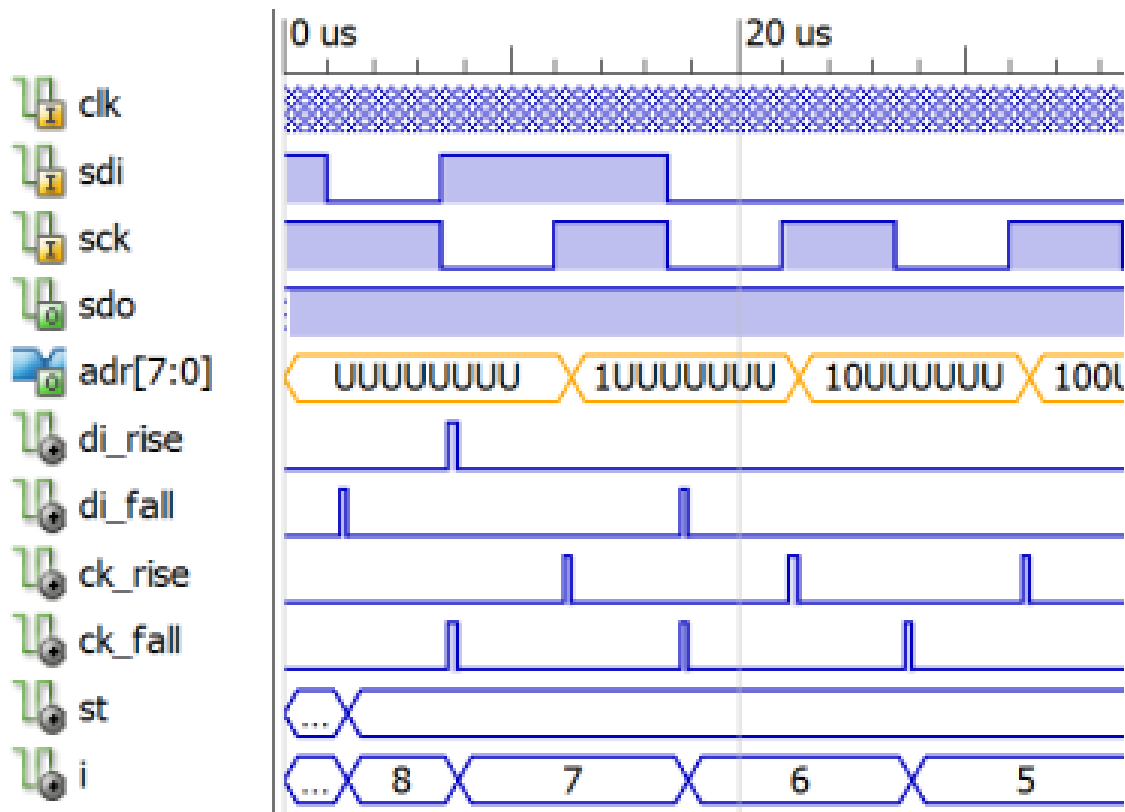


- ▶ Npr. mikroprocesor AVR ima vmesnik TWI (Two Wire Interface) za komunikacijo po I2C protokolu



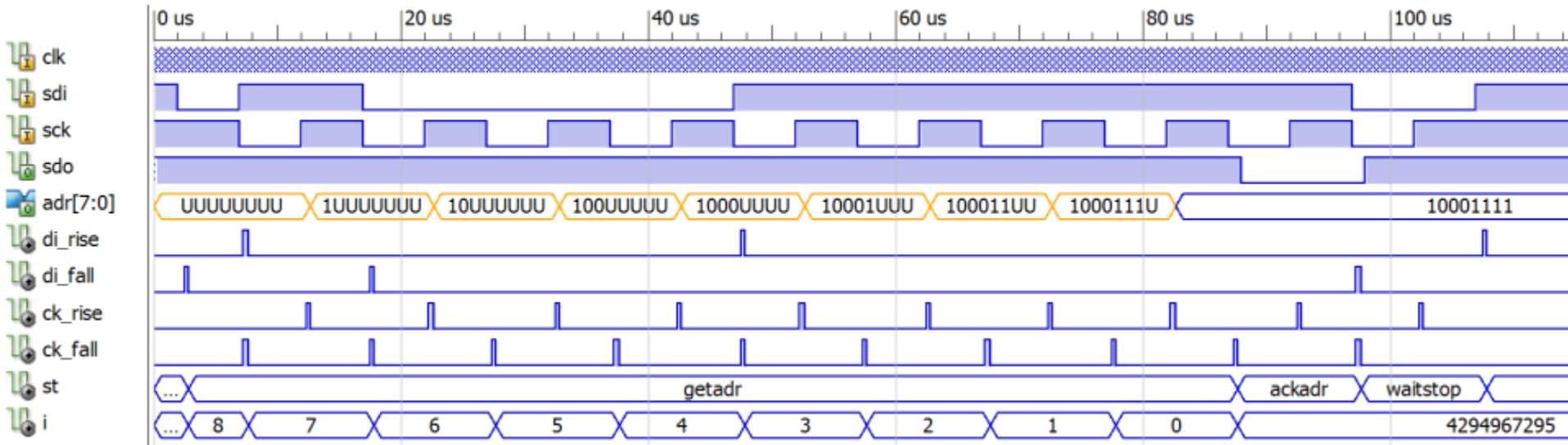
Izvedba serijskega protokola

- ▶ Kadar ni prenosa sta SDI in SCK na 1
 - ▶ ob SCK=0 se SDI nastavi (ck_fall)
 - ▶ ob SCK=1 se SDI ne spreminja – branje SDI (ck_rise)
- ▶ **START:** SCK=1 in SDI gre na 0 (di_fall)



Izvedba serijskega protokola

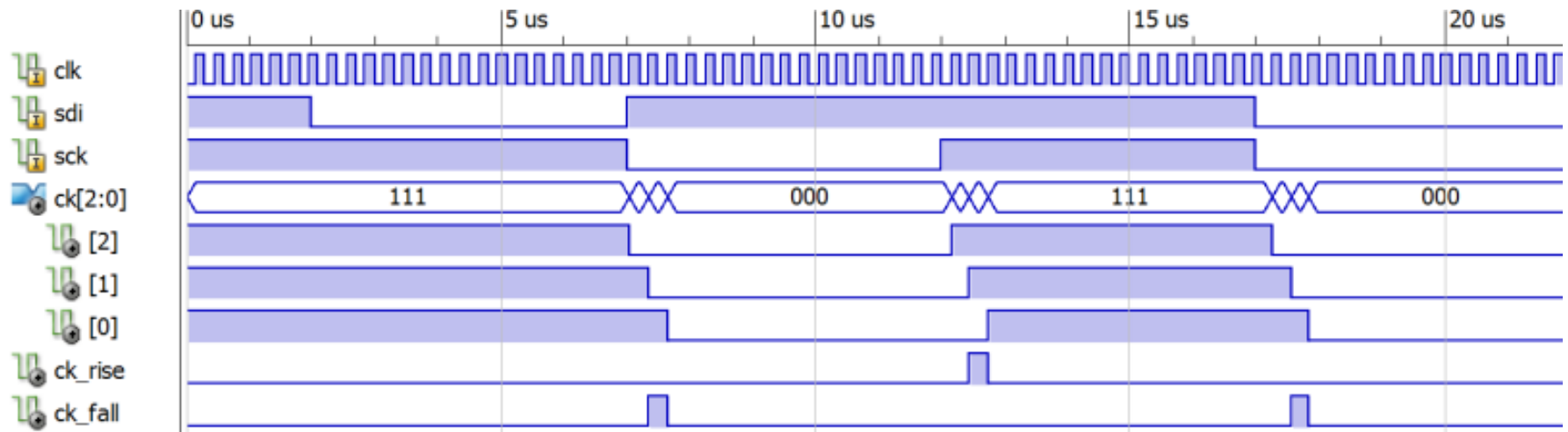
- ▶ po zadnjem bitu je še en cikel za prenos potrditve (ACK)
- ▶ STOP: SCK=1 in SDI gre na 0 (di_fall)



Vzorčenje ure in podatkov

▶ 3x vzorčenje s pomikalnimi registri (di, ck)

```
if rising_edge(clk) then
  di <= sdi & di(2 downto 1);
  ck <= sck & ck(2 downto 1);
end if;
```



```
ck_rise <= '1' when ck(1)='1' and ck(0)='0' else '0';
ck_fall <= '1' when ck(1)='0' and ck(0)='1' else '0';
```

Sekvenčni avtomat

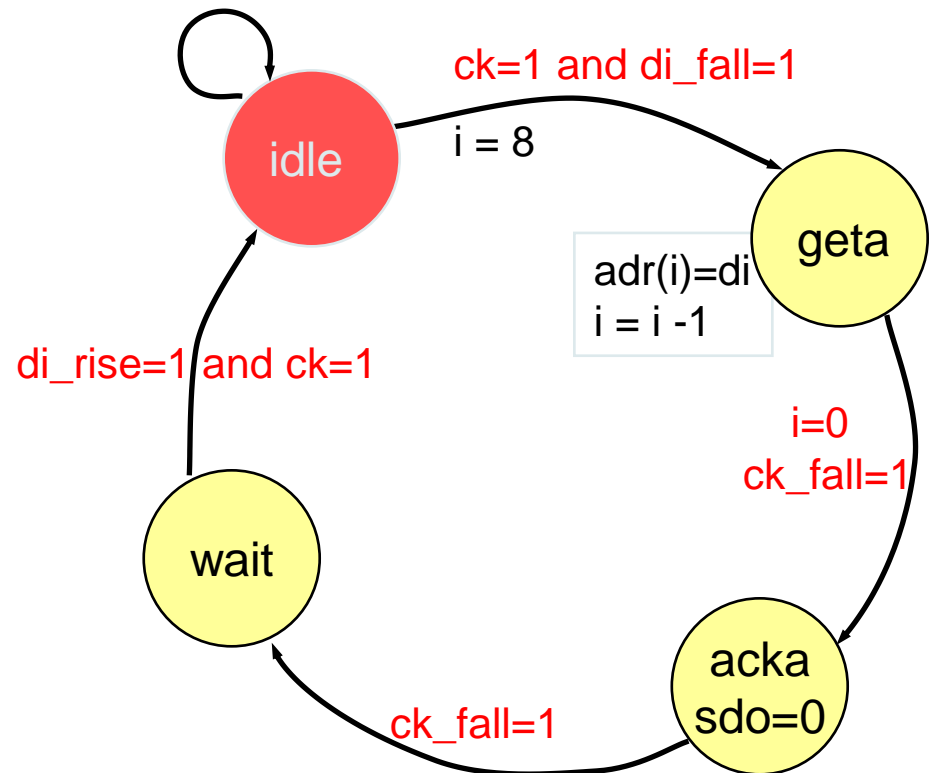
case st is

when idle =>

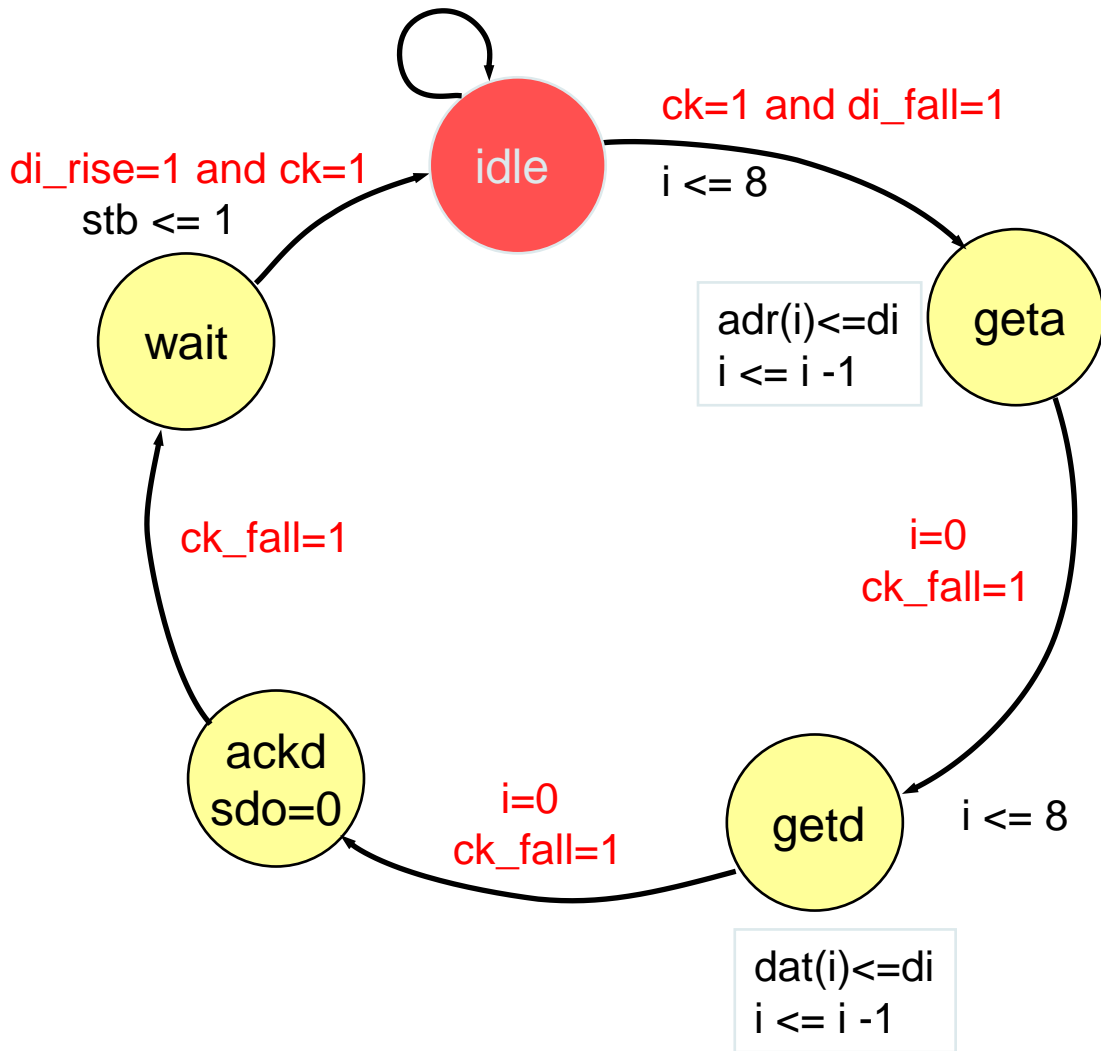
```
if ck(0)='1' and di_fall='1' then
  st <= geta; -- branje naslova
  i <= 8;
end if;
```

when geta =>

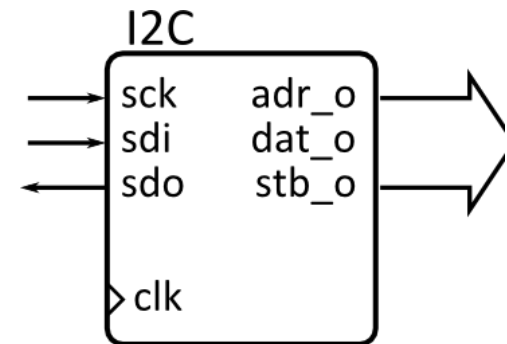
```
if ck_rise='1' then
  adr(i) <= di(0); -- beri nov bit
elsif ck_fall='1' then
  i <= i - 1;
  if i=0 then
    st <= acka; -- potrditev
  end if;
end if;
```



Končni diagram stanj

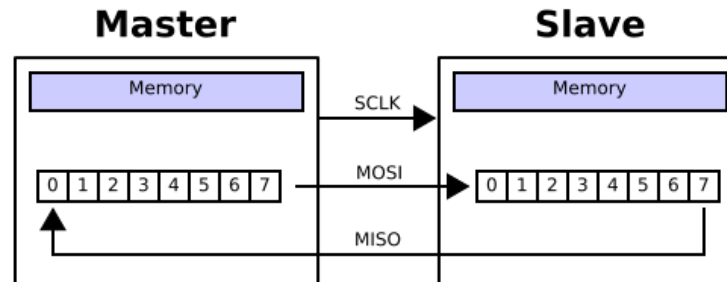


- Prenos 8-bitnega naslova in podatka



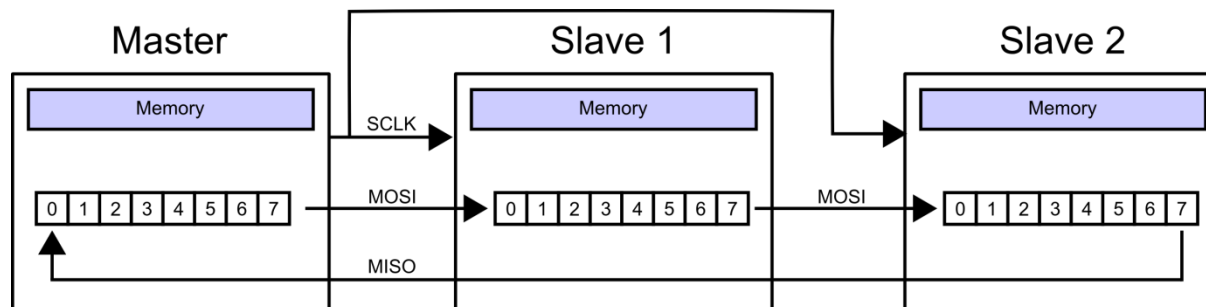
Sinhroni zaporedni vmesnik - SPI

- ▶ Vodilo sestavlja ura (SCLK), podatkovni izhod (MOSI) in vhod (MISO) v nadrejeno enoto



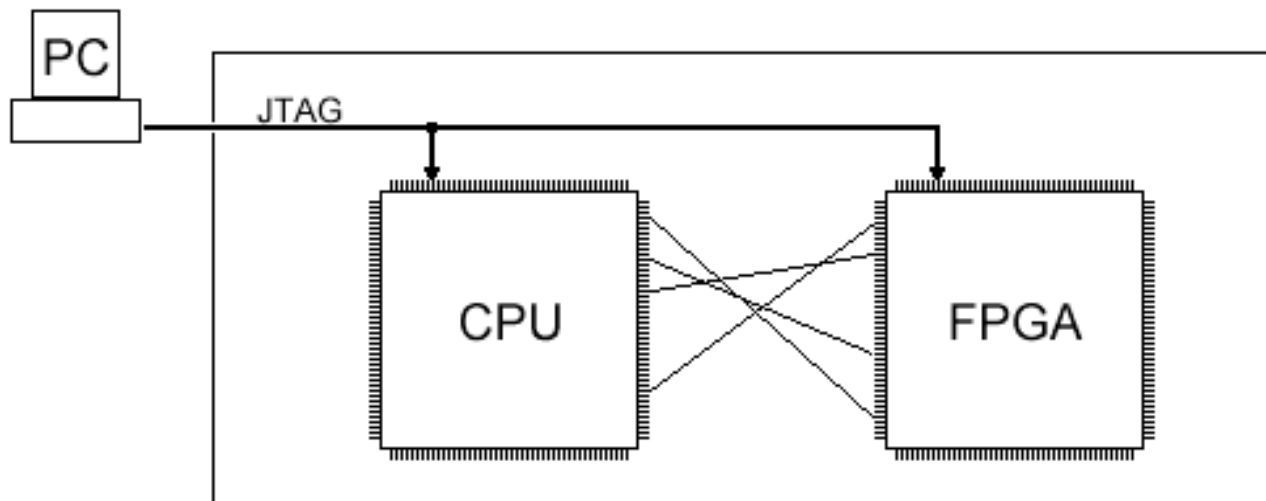
- ▶ Uporaba:

- ▶ MMC, SD kartice, senzorji, nalaganje mikrokrmilnikov
- ▶ prenos podatkov od MSB proti LSB
- ▶ več podrejenih enot lahko vežemo vzporedno ali v verigo

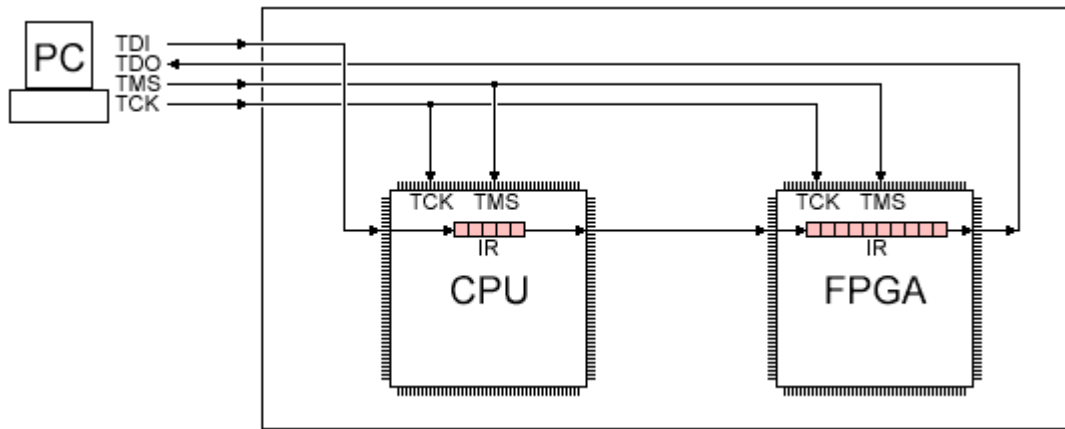


Sinhroni zaporedni vmesnik - JTAG

- ▶ vodilo sestavlja ura (TCK), kontrolni izhod (TMS), podatkovni izhod (TDO) in vhod (TDI) v nadrejeno enoto
- ▶ JTAG je zaporedni vmesnik po IEEE standardu (1149.1) za periferno testiranje vezij
 - ▶ proizvodno testiranje povezav med dvema integriranima vezjema
 - ▶ preko JTAG odklopimo priključke od notranjosti vezja, nastavljammo stanja na priključkih in beremo stanja na drugem vezju



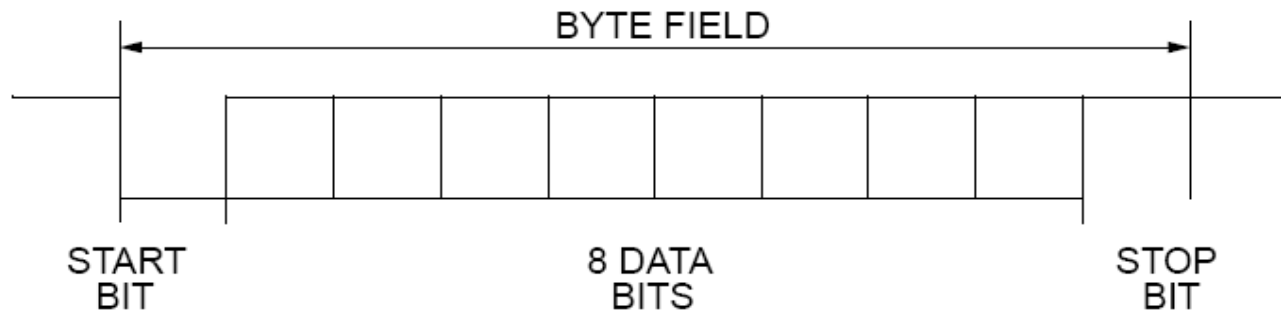
Testiranje in programiranje vezij z JTAG



- ▶ zaporedni prenos ukazov in podatkov
 - ▶ več vezij povežemo v JTAG verigo
-
- ▶ Poleg ukazov po standardu še dodatni ukazi za programiranje vezij CPLD ali FPGA
 - ▶ Programiranje in debugiranje mikrokontrolerov preko JTAG
 - ▶ ustavitev izvajanja ukazov in dostop do vseh registrov preko vmesnika JTAG

Asinhroni zaporedni vmesnik – RS232

- ▶ **Asinhroni sprejemnik in oddajnik (UART)**
 - ▶ asinhroni podatkovni paketi, sprejemna ura se sinhronizira za vsak paket – sinhronizacija paketov
 - ▶ vnaprej dogovorjene bitne frekvence
 - ▶ 9600, 38400, 115200 bit/s
 - ▶ npr. podatkovni paket 8N1:



Asinhroni zaporedni vmesnik – LIN

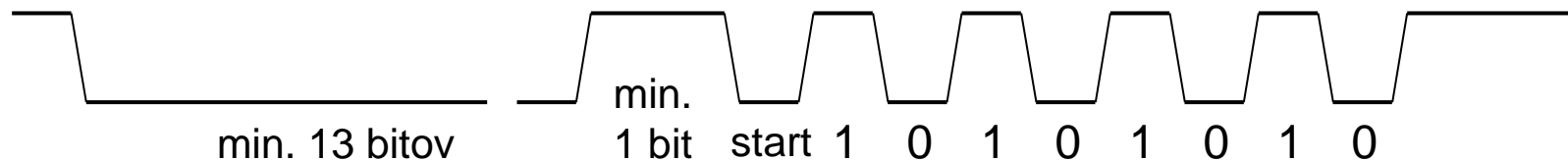
- ▶ **LIN (Local Interconnect Network)**
 - ▶ enostaven protokol, ki je podoben RS232
 - ▶ razvit za avtomobilsko industrijo
- ▶ V primerjavi z zmogljivejšim avtomobilskim vmesnikom **CAN** je počasnejši in enostavnejši
 - ▶ uporaben za monitoring senzorjev, prenos ukazov za odpiranje oken...
 - ▶ cenena izvedba vmesnika z mikrokrmilnikom
 - ▶ ne vsebuje robustnega odpravljanja napak
 - ▶ ni uporaben za varnostno kritične aplikacije



Prenos podatkov po protokolu LIN

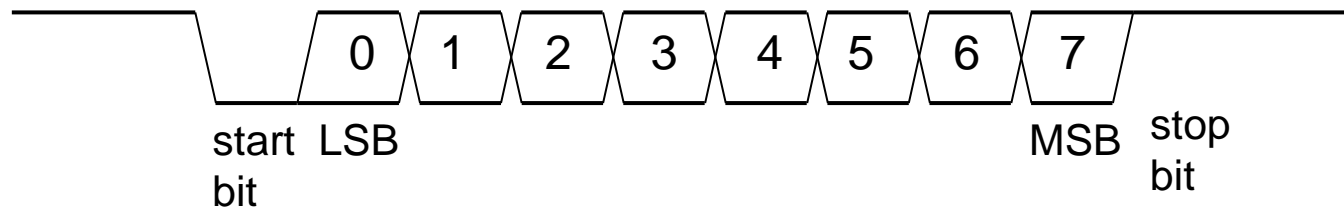
▶ Sinhronizacija sprejemnika

- ▶ min. 13 bitov logična 0, ki mu sledi 1 bit premora
- ▶ sledi start bit (logična 0) in kombinacija 55_{16}



▶ Prenos v serijski obliki 8N1

- ▶ start bit (logična 0), 8 podatkov in stop bit (log. 1)



Povzetek

- ▶ Kakšne so prednosti zaporednega (serijskega) prenosa podatkov pred vzporednim?
- ▶ Opiši kako naredimo z logičnim vezjem vzporedno-zaporedno pretvorbo podatkov.
- ▶ Navedi nekaj primerov zaporednih (serijskih) vmesnikov.
 - ▶ V čem se razlikujejo zaporedni vmesniki ?
 - ▶ Kje se uporabljajo ?