



Laboratorij za načrtovanje integriranih vezij

Univerza *v Ljubljani*  
Fakulteta *za elektrotehniko*



# Načrtovanje digitalnih elektronskih sistemov

Komunikacijski vmesniki in vodila

# Kaj prenašamo po vodilu

---

- ▶ podatke
  - ▶ prenašajo se po besedah (**byte**)
- ▶ ukaze oz. kontrole
  - ▶ izberi napravo / beri / piši
  - ▶ naslov
- ▶ signale, ki jih določa protokol prenosa
  - ▶ zahteve (**request**), prekinitve (**interrupt**)
  - ▶ potrditve (**acknowledge**)
  - ▶ časovno usklajevanje (sinhrono / asinhrono)

# Kako prenašamo podatke

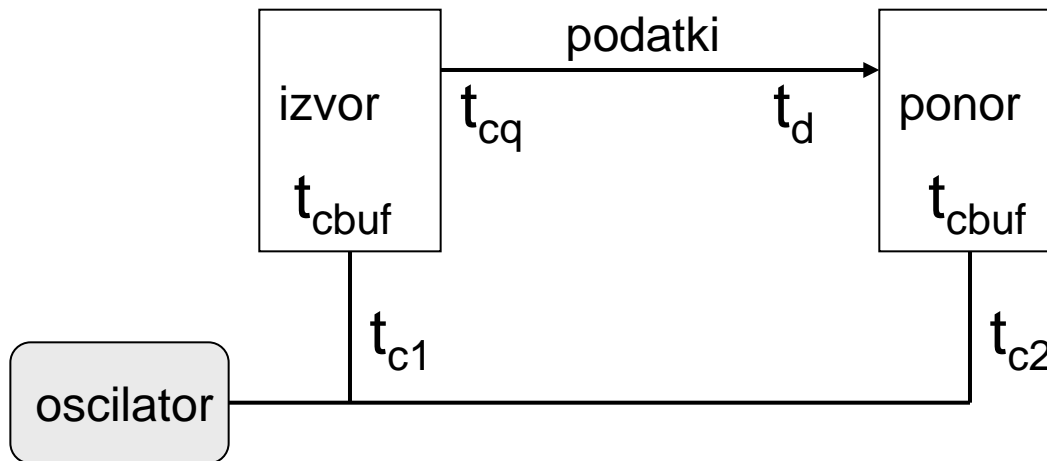
---

- ▶ po besedah (paralelno) ali bitih (serijsko)
- ▶ po smeri
  - ▶ enosmerno
  - ▶ dvosmerno (dvojno ali 3-stanjsko vodilo)
  - ▶ hkrati v obe smeri
- ▶ po protokolu
  - ▶ enostaven cikel
  - ▶ blokovni prenos (**burst**)
    - ▶ manj usklajevanja, večji izkoristek vodila

# Sinhronizacija prenosa podatkov

---

## ► Sistemska sinhronizacija

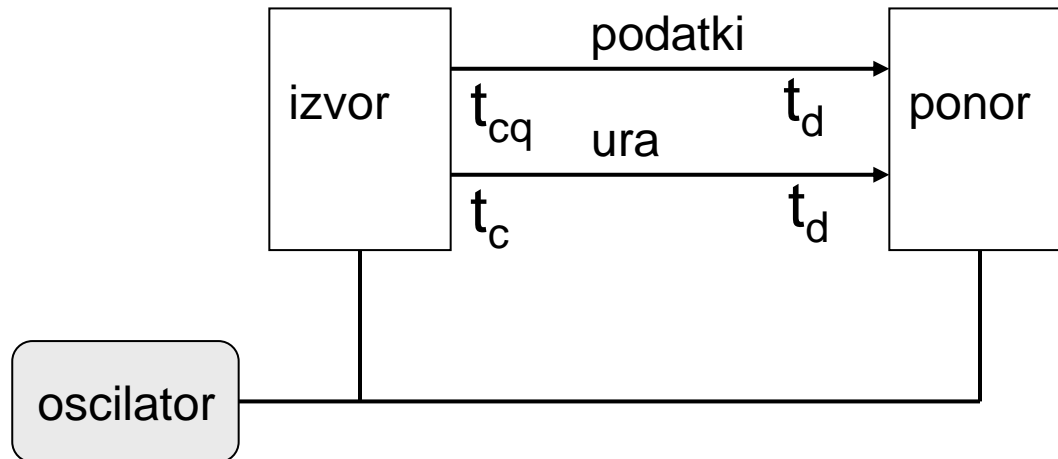


- upoštevati moramo vse zakasnitve!
  - pri nizkih hitrostih prenosa ni težav

# Sinhronizacija pri izvoru podatkov

---

- ▶ izvor pošilja podatke in uro
  - ▶ *source synchronous / clock forwarded* (SDRAM)

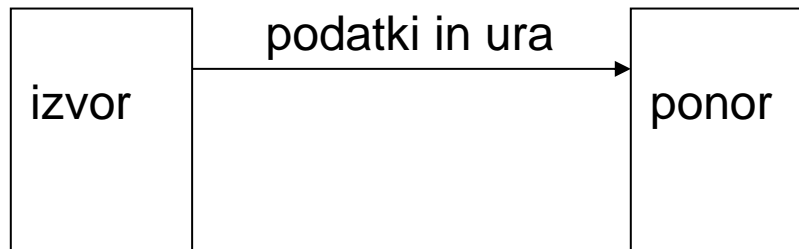


- dolžina linij za podatke in uro se mora ujemati
- v ponoru naredimo ponovno sinhronizacijo na sistemsko uro

# Lastna sinhronizacija podatkov

---

- ▶ podatkovni tok vsebuje podatke in uro
  - ▶ serijski prenos podatkov



- izvor dela paralelno/serijsko pretvorbo in kodiranje podatkov in ure
- ponor dela serijsko/paralelno pretvorbo in rekonstrukcijo ure s fazno sklenjeno zanko (PLL)

# Razvoj računalniškega vodila

---

## ▶ prva generacija

- ▶ 8-bitni paralelni prenos, **TTL/CMOS** nivoji
- ▶ sistemsko vodilo: naslovni, podatkovni, kontrolni signali
- ▶ omejitev hitrosti, prilagodljivost (jumperji za naslove)

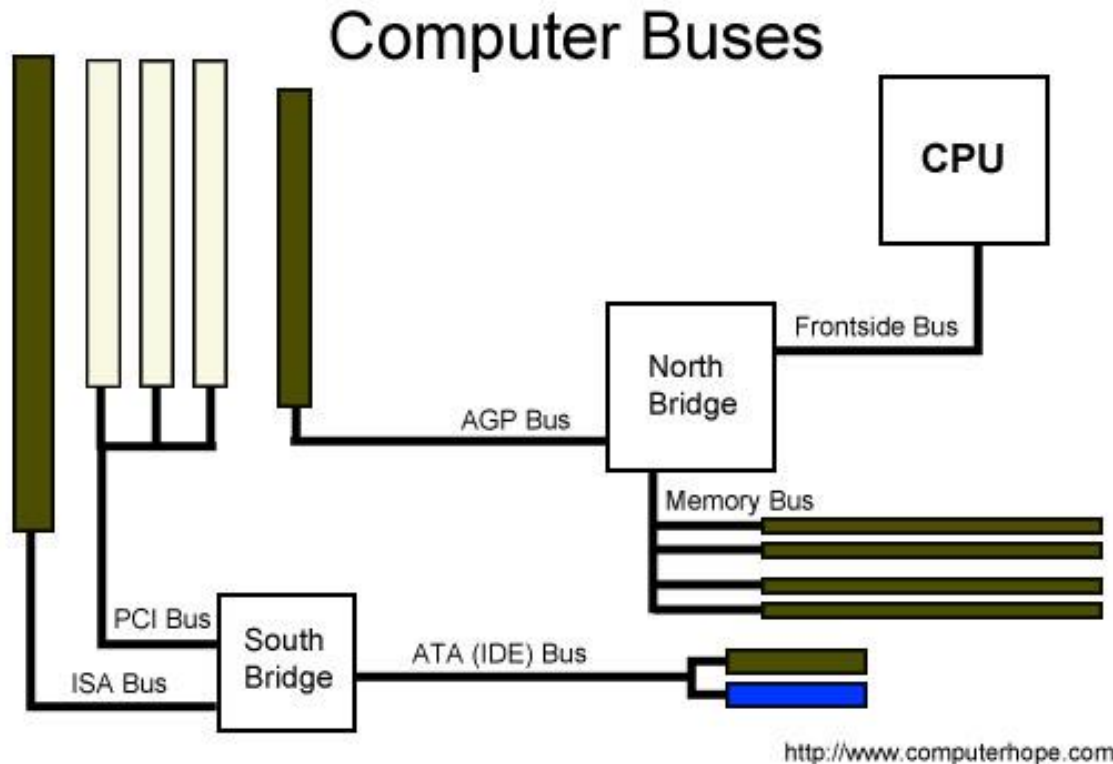
## ▶ druga generacija

- ▶ 16 ali 13-bitni prenos podatkov
- ▶ ločeno vodilo za pomnilnik in periferijo
- ▶ hitra paralelna vodila (**PCI**)

## ▶ sodobna vodila

- ▶ zelo hitra serijska vodila (**PCI Express**)
- ▶ omrežje s protokolom

# Razvoj vodil v osebnem računalniku



## ▶ PCI

- ▶ 32 bitov, 33 MHz, PCI

## ▶ ISA

- ▶ 16 bitov, 8 MHz, TTL

## ▶ ATA

## ▶ sistemsko vodilo

- ▶ odvisno od procesorja

## ▶ DDR SDRAM

- ▶ 64 bitov, 133 MHz, SSTL

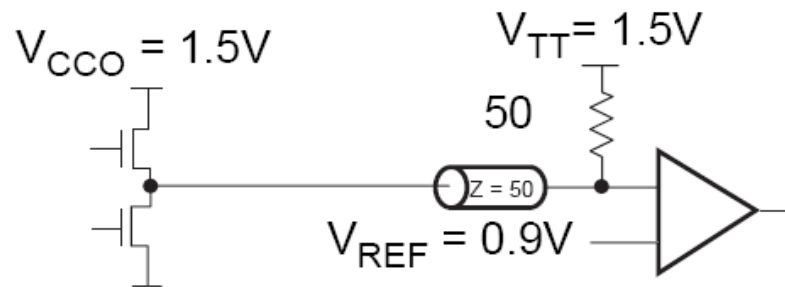


# Prenosne linije

---

- ▶ pri visokih frekvencah obravnavamo povezave kot prenosne linije (npr. mikrotrakasta linija)
- ▶ tehnologija zahteva nižje napetosti in zaključitve linij na tiskanem vezju
  - ▶ DDR SDRAM (SSTL 2.5V ali 1.8V)
  - ▶ QDR SDRAM (HSTL 1.5V)

HSTL Class III



# Hitra paralelna vodila

---

- ▶ problem časovne usklajenosti
  - ▶ čim manjša razlika v dolžinah linij
- ▶ problem hkratnega preklapljanja signalov (SSO)
  - ▶ tokovna obremenitev
- ▶ problem elektromagnetnih motenj
- ▶ rešitve: vodilo PCI uporablja odbiti val
  - ▶ oddajnik odda signal s  $\frac{1}{2}$  amplitudo
  - ▶ na koncu linije se signal odbije in prispe do sprejemnikov s polno amplitudo

# Hitra serijska vodila

---

- ▶ **problem organizacije podatkov**
  - ▶ paketi in okvirji, markerji označujejo začetek podatkov oz. kontrolnih informacij
- ▶ **problem sinhronizacije**
  - ▶ s kodiranjem (npr. 8b/10b) odpravimo dolga zaporedja ničel ali enic
  - ▶ FIFO medpomnilniki v sprejemniku in oddajniku
- ▶ **problem napak pri sprejetih bitih**
  - ▶ CRC polinomi

# Vodilo USB

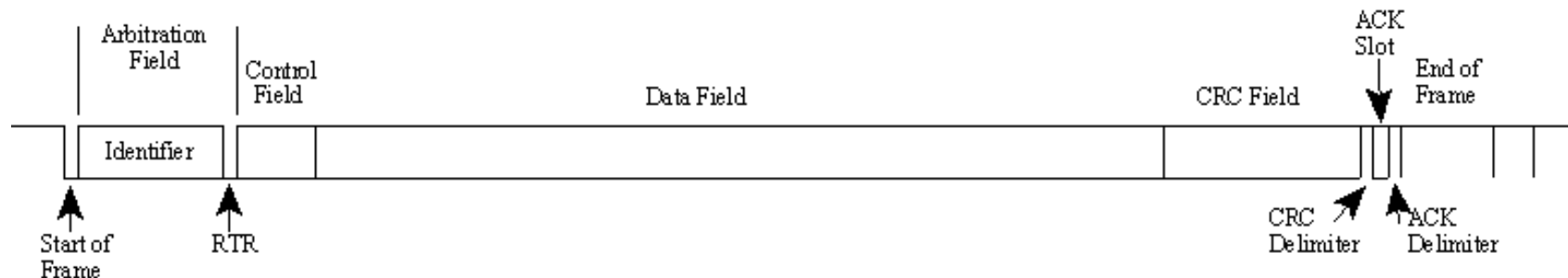
---

- ▶ krmilnik (**hub**) izvaja povpraševanje (**polling**)
- ▶ serijski prenos po paketih
  - ▶ NRZI kodiranje ('0' – sprememba nivoja, '1' – brez spremembe) in dodajanje '0' po zaporedju 6 enic
  - ▶ paketi se začnejo s SYNC kombinacijo in PID, na koncu paketa je CRC
- ▶ protokol določa vrste paketov in odziv naprav
  - ▶ standard USB definira protokolni sklad

# Vodilo CAN

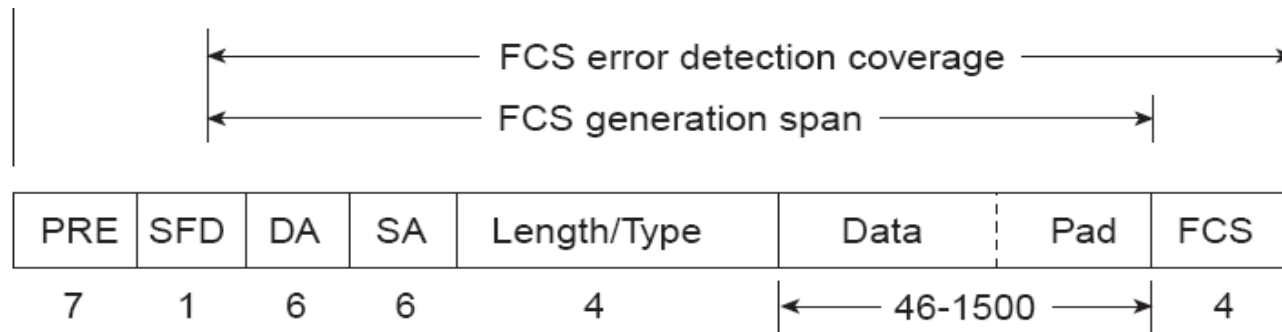
---

- ▶ razpršeno oddajanje (**broadcast**)
- ▶ princip dominantnih bitov (**wired and**)
- ▶ NRZ kodiranje paketov z dodajanjem bitov
- ▶ različni tipi okvirjev in možnost detekcije ter korekcije napak
  - ▶ podatkovni okvir vsebuje ID, 0-8 besed in potrditev



# Vodilo Ethernet

- ▶ CSMA/CD (*Carrier Sense, Multiple Access, Collision Detect*)
- ▶ protokolni sklad (MAC, MII)
- ▶ različni fizični vmesniki (10Base-T...1000Base-T)



Field length in bytes

PRE = Preamble  
SFD = Start-of-frame delimiter  
DA = Destination address  
SA = Source address  
FCS = Frame check sequence

# Prednosti serijskih vodil

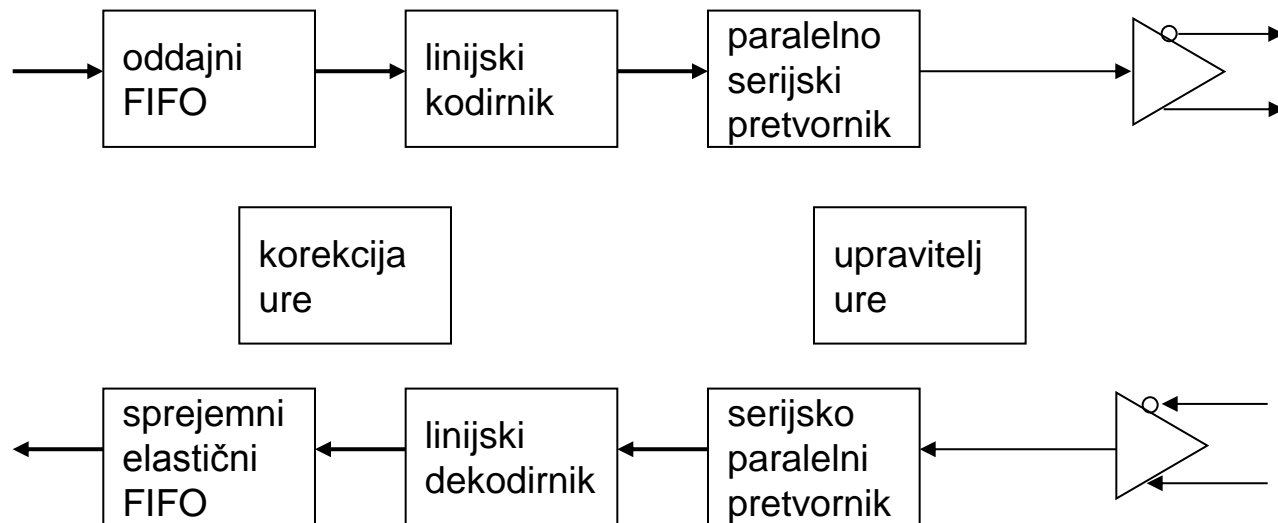
---

- ▶ večja pasovna širina (gigabitni prenos)
- ▶ manj priključkov na integriranem vezju
  - ▶ povezave točka s točko
  - ▶ ni potrebna delitev vodila med več uporabniki
- ▶ enostavna razširitev za različne protokole
  - ▶ PCI Express
  - ▶ Serial ATA
  - ▶ 1-10Gb Ethernet
  - ▶ FiberChannel

# Serijski pretvorniki

---

- ▶ serijski kodirnik in dekodirnik
  - ▶ npr. 1bit/1.5GHz na 20bitov/75MHz



- serijski signali se prenašajo diferencialno
- pretvornik dela z več fazami ure



# Linijski kodirnik

---

- ▶ odpravi dolga zaporedja ničel in enic
- ▶ uravnoteži povprečno število ničel in enic
- ▶ določa kontrolne kombinacije
  - ▶ sinhronizacijska ločila, poravnava kanalov...
- ▶ npr. 8b/10b kodiranje

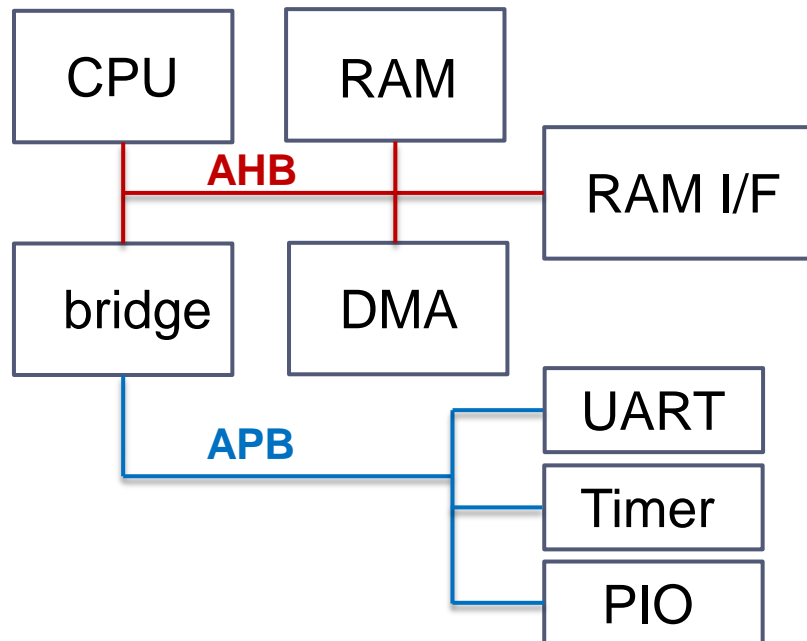
8b	10b RD-	10b RD+
00000000	1001110100	0110001011

- dekodirnik išče ločila v zaporedju bitov in opravi inverzno kodiranje

# Vodila v integriranem vezju

---

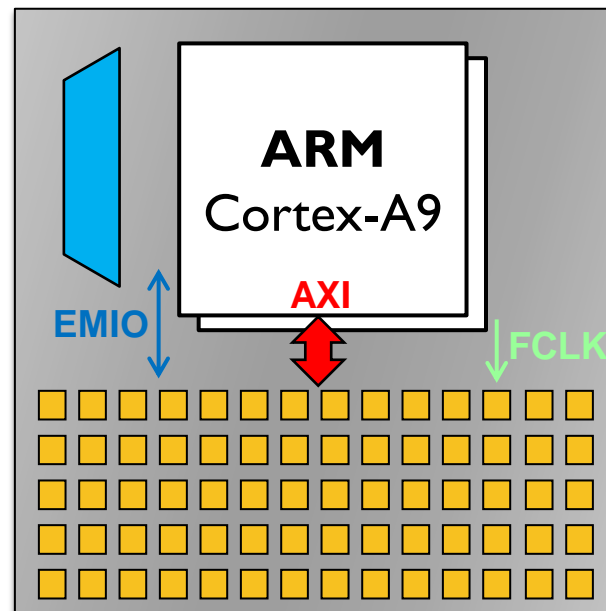
- ▶ vodilo je odvisno od procesorja oz. komponent
- ▶ npr. **ARM AMBA**
  - ▶ **AHB (Advanced High-speed Bus)** zmogljivo vodilo, cevljenje, blokovni prenos, več gospodarjev
  - ▶ **APB (Peripheral Bus)** enostaven vmesnik, ločeni poti za branje in pisanje



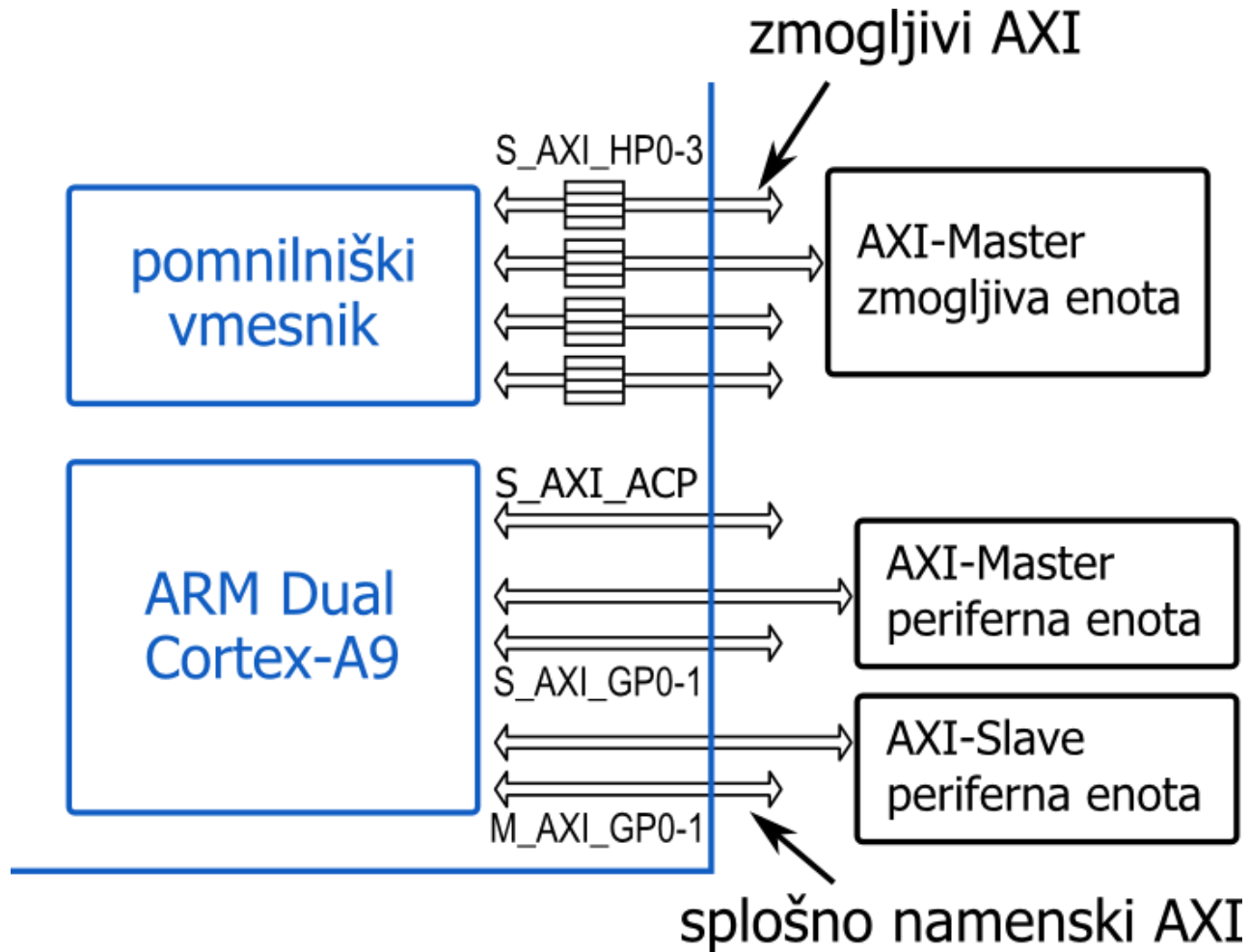
# Povezava ARM-FPGA

---

- ▶ vgrajene periferne enote (EMIO)
- ▶ pomnilniško preslikani vmesniki
  - ▶ Zynq uporablja AXI4 protokol in vmesnike
  - ▶ sistem nadrejenih enot (master-slave)



# Vmesniki AXI na procesorskem sistemu



# AXI4 (Advanced eXtensible Interface)

---

- ▶ pomnilniško preslikan vmesnik, preko AXI interconnect

- ▶ AXI4 (full)

- ▶ zmožljiv, kompleksen
    - ▶ do 256 podatkov/prenos

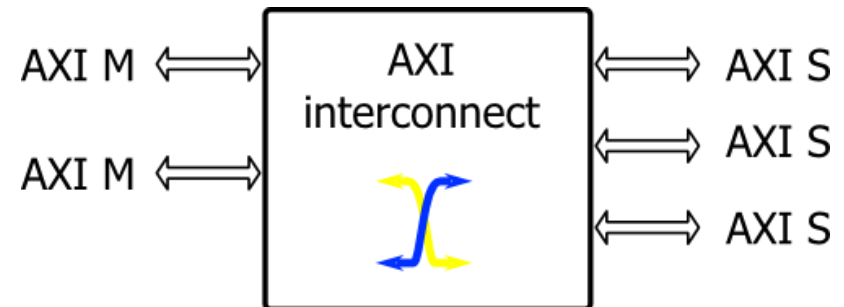
- ▶ AXI4-Lite

- ▶ manj signalov, enostavnejši
    - ▶ 1 podatek/prenos

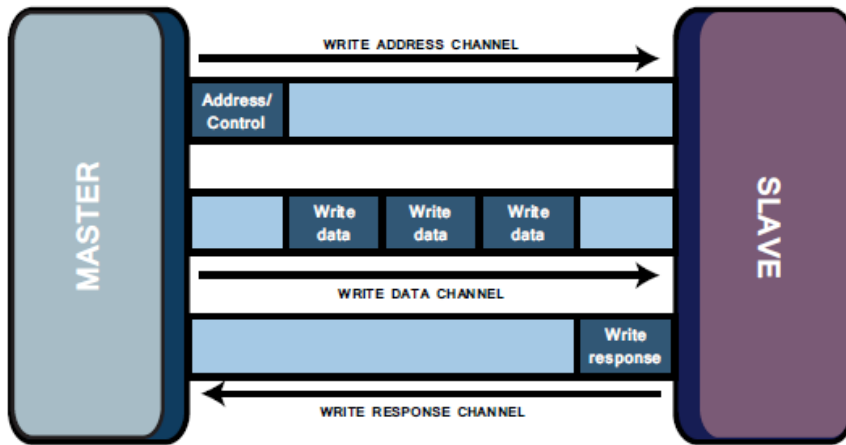
- ▶ pretočni vmesnik

- ▶ AXI4-Stream

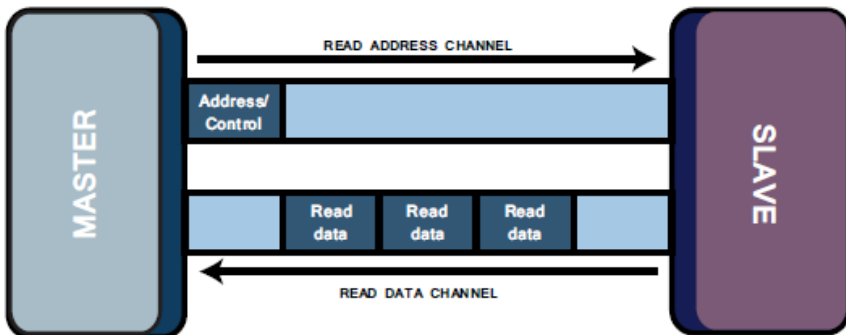
- ▶ pretočni protokol brez naslovov



# Vmesnik AXI4



- ▶ pisanje podatkov
  - ▶ ločena kanala za prenos naslova in podatka
  - ▶ potrditveni kanal

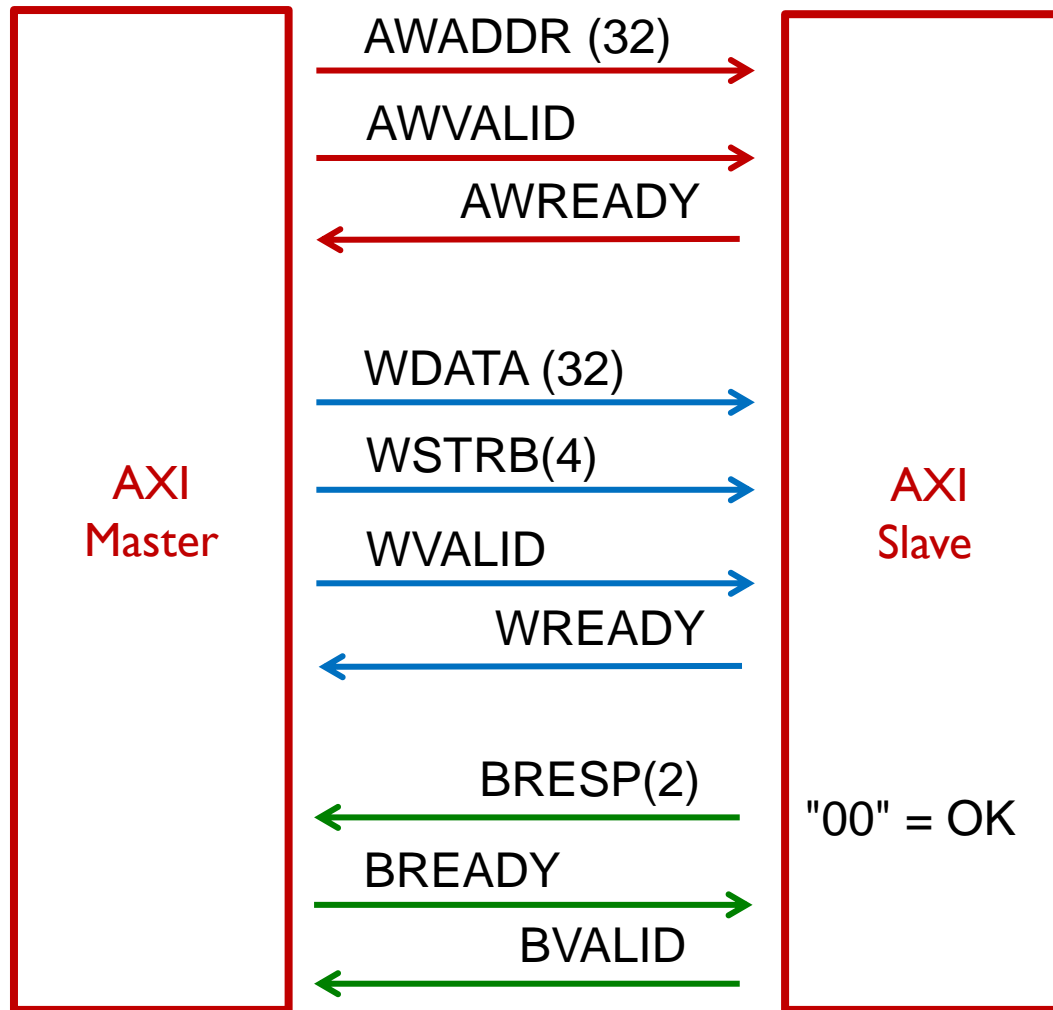


- ▶ branje podatkov
  - ▶ kanala za naslov in podatke

# Signali...

---

- ▶ pisanje podatkov AXI4-Lite



# Paketni prenos AXI4

- ▶ sočasno branje in pisanje



- ▶ poljuben vrstni red zaključkov
  - ▶ hitre podrejene enote zaključijo prej kot počasne enote

