



# OSNOVE RAČUNALNIŠKE ARHITEKTURE I

## 3 Osnovni principi delovanja



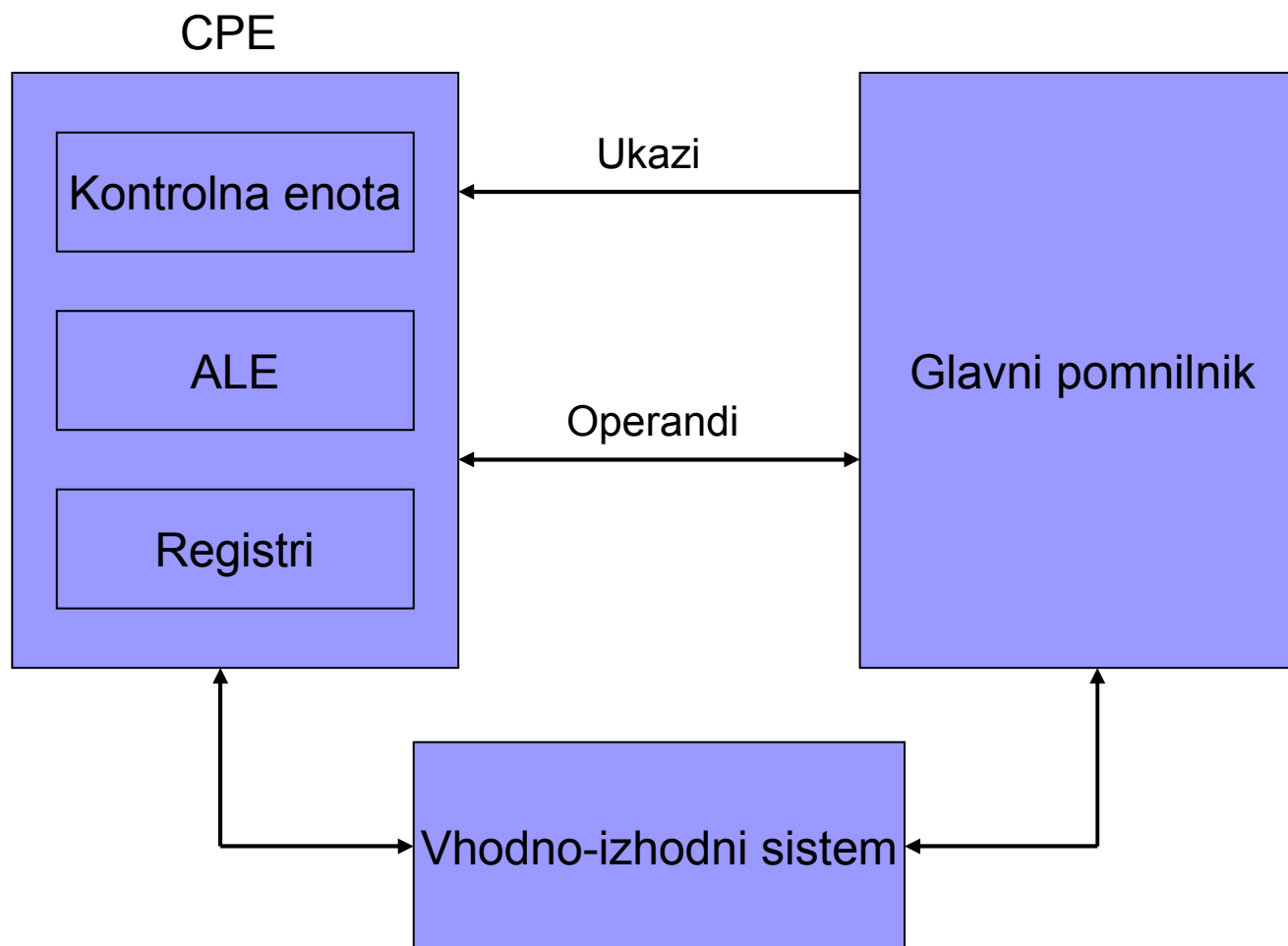
- Turingov stroj: matematični (miselni) model računanja – to ni resničen stroj
  
- Von Neumannov model: dvojni pomen
  - Matematični model računanja
  - Resničen stroj
  
- Von Neumannov računalniški model je ekvivalenten Turingovemu stroju  $\Rightarrow$  na njem se da izračunati vse kar je izračunljivo



## 3.1 Von Neumannov računalniški model

- Sestavljajo ga trije osnovni deli:
  - CPE (Centralna procesna enota)
  - Glavni pomnilnik
  - V/I sistem
- Je stroj s shranjenim programom, ki je shranjen v glavnem pomnilniku in vodi delovanje stroja
- CPE jemlje ukaze iz glavnega pomnilnika in jih izvaja enega za drugim

# Zgradba tipičnega von Neumannovega računalnika






# Delovanje von Neumannovega računalnika


- Njegovo delovanje popolnoma določajo ukazi (strojni ukazi), ki jih CPE jemlje iz glavnega pomnilnika enega za drugim
- Na nek način je določeno, iz katerega naslova se vzame prvi ukaz po vklopu računalnika ali po pritisku na tipko RESET



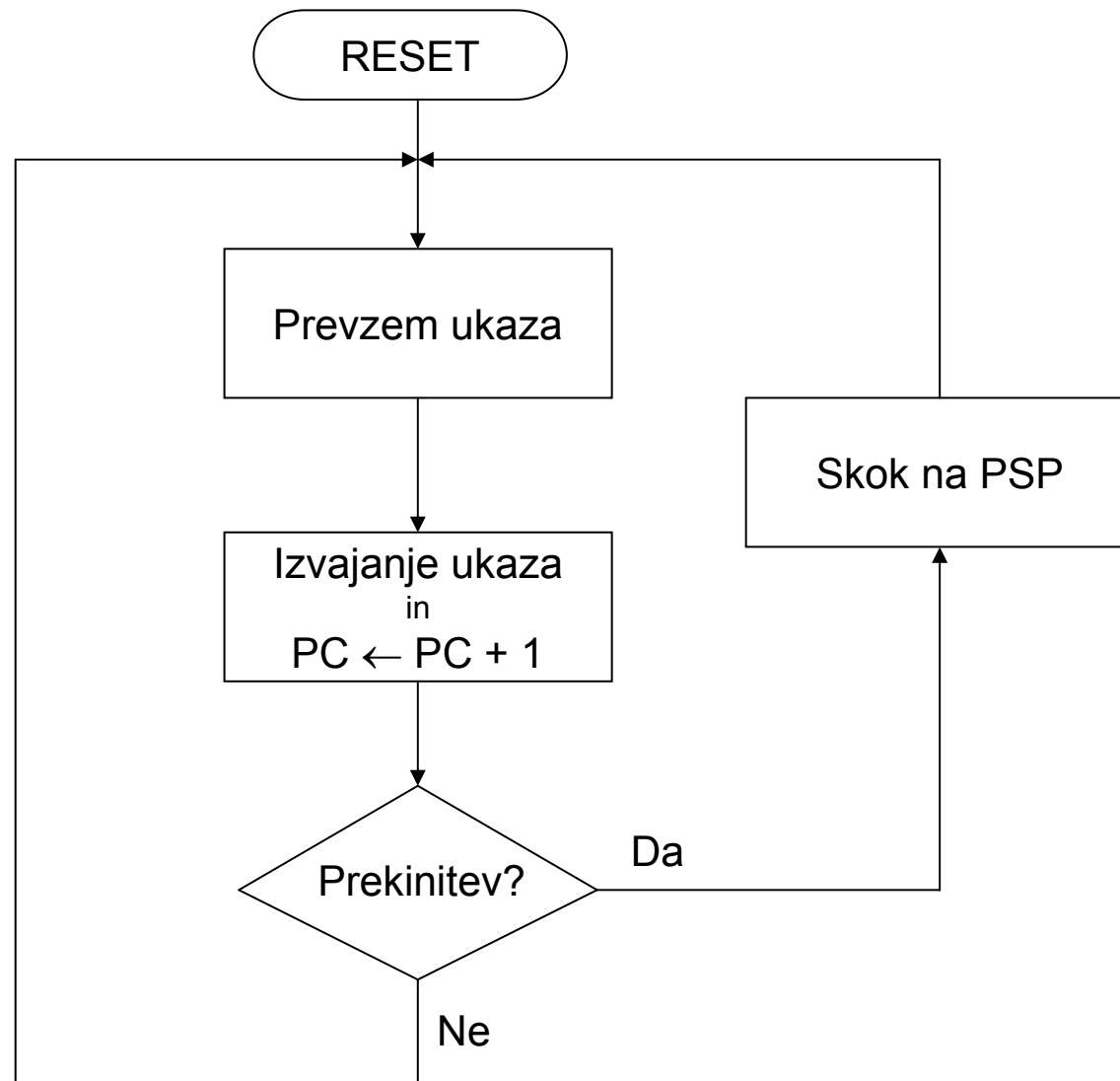
## Pri vsakem ukazu razlikujemo dva koraka

- 1. korak: Jemanje ukaza iz pomnilnika  
(tudi branje ali prevzem ukaza)  
– angl. Fetch
- V CPE je poseben register – programski števec (PC) – ki vedno vsebuje pomnilniški naslov na katerem je v pomnilniku shranjen naslednji ukaz

- 
- 2. korak: Izvrševanje v koraku 1 prevzetega ukaza  
– angl. execute
  - Vsak ukaz vsebuje informacije o operaciji in operandih, nad katerimi naj se operacija izvrši
  - CPE operacijo izvrši in poskrbi, da je v PC naslov naslednjega ukaza
  - Pravilo: ukazi v pomnilniku so shranjeni po naraščajočih naslovih zato  $PC \leftarrow PC + 1$
  - Izjema: Skočni ukazi, s katerimi lahko v PC zapišemo poljuben naslov

- 
- Po zaključku koraka 2 prične CPE zopet s korakom 1
  - Ta dva koraka se ponavljata dokler računalnik deluje
  - **Izjema: prekinitev ali past**  
CPE po koraku 2 ne prevzame ukaza po pravilu  $PC \leftarrow PC + 1$ , temveč začne izvajati drug program - prekinitveno servisni program (PSP)







- Zaporedno izvajanje ukazov je počasno in predstavlja osnovno slabost von Neumannovih računalnikov
  
- Razširitve osnovnega von Neumannovega modela so zajete v Flynnovi klasifikaciji iz leta 1966
  
- Osnovna kriterija:
  - Število ukazov, ki se izvršujejo naenkrat
  - Število operandov, ki jih en ukaz naenkrat obdeluje



# Flynnova klasifikacija

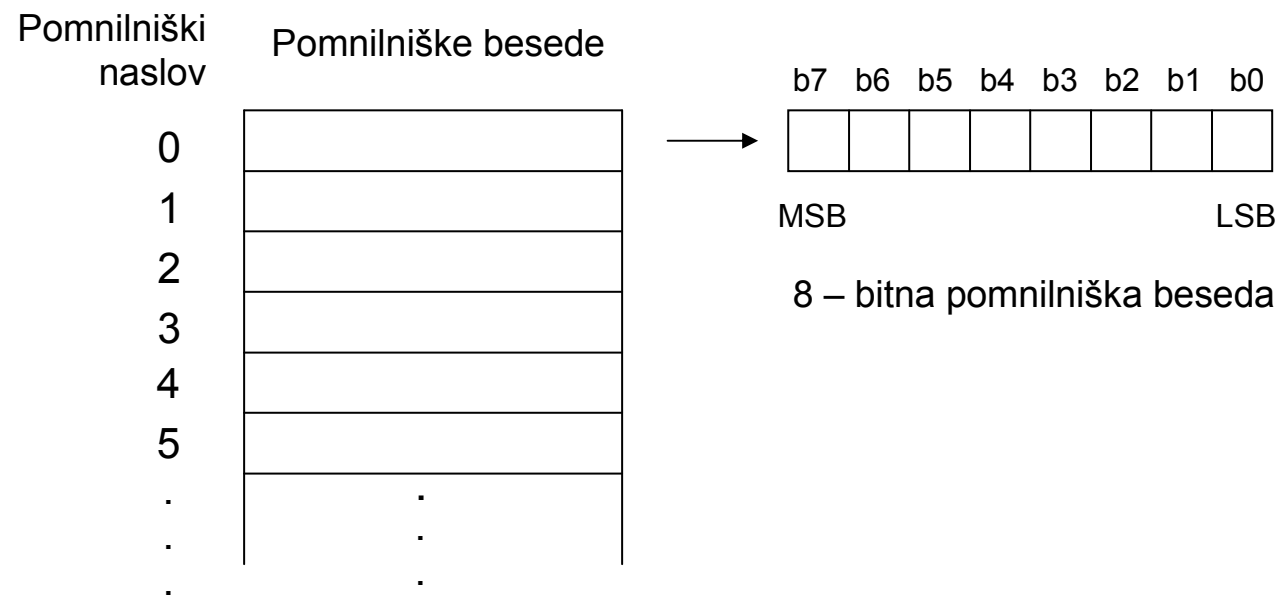
- Vsak računalnik spada v enega od štirih razredov:
  - 1 SISD (Single Instruction Single Data)
  - 2 SIMD (Single Instruction Multiple Data)
  - 3 MISD ! (Multiple Instruction Single Data)
  - 4 MIMD (Multiple Instruction Multiple Data)
    - SIMD in MIMD - paralelni računalniki



## 3.2 Glavni pomnilnik v von Neumannovem računalniku

- Glavni pomnilnik je pasivna naprava in služi kot skladišče za shranjevanje ukazov in operandov
- Osnovna celica je enobitna pomnilniška celica
- Glavni pomnilnik sestavljajo pomnilniške besede
- Pomnilniška beseda je definirana kot najmanjše število bitov, ki imajo svoj naslov

- Vsaka pomnilniška beseda ima svoj enoveljaven pomnilniški naslov
- Dolžina pomnilniške besede je število enobitnih celic v besedi
- Običajna dolžina pomn. besede 8 bitov = 1 bajt





# Pomnilniški naslov

- Naslov pomnilniške besede je nespremenljiv, gledano iz CPE ima ista beseda vedno isti naslov
- Dolžina pomnilniškega naslova: število bitov s katerimi je podan naslov
- Dolžina naslova določa velikost pomnilniškega prostora ali naslovni prostor
- Velikost pomnilniškega prostora: največja možna velikost glavnega pomnilnika, kot jo vidi CPE
- 32-bitni naslov  $\Rightarrow$  velikost pomn. prostora  $2^{32} = 4G$  pomnilniških besed



Pomen oznak kilo, mega, giga, ..., je samo  
pri glavnih pomnilnikih drugačen !

- 1k (kilo) =  $2^{10}$  = 1 024 (1kB = 1024B)
- 1M (mega) =  $2^{20}$  = 1 048 576
- 1G (giga) =  $2^{30}$  = 1 073 741 824
  
- Druga področja (frekvenca, hitrost prenosa, ...)
- 1k =  $10^3$  = 1 000 (1kHz = 1 000 Hz)
- 1M =  $10^6$  = 1 000 000
- 1G =  $10^9$  = 1 000 000 000



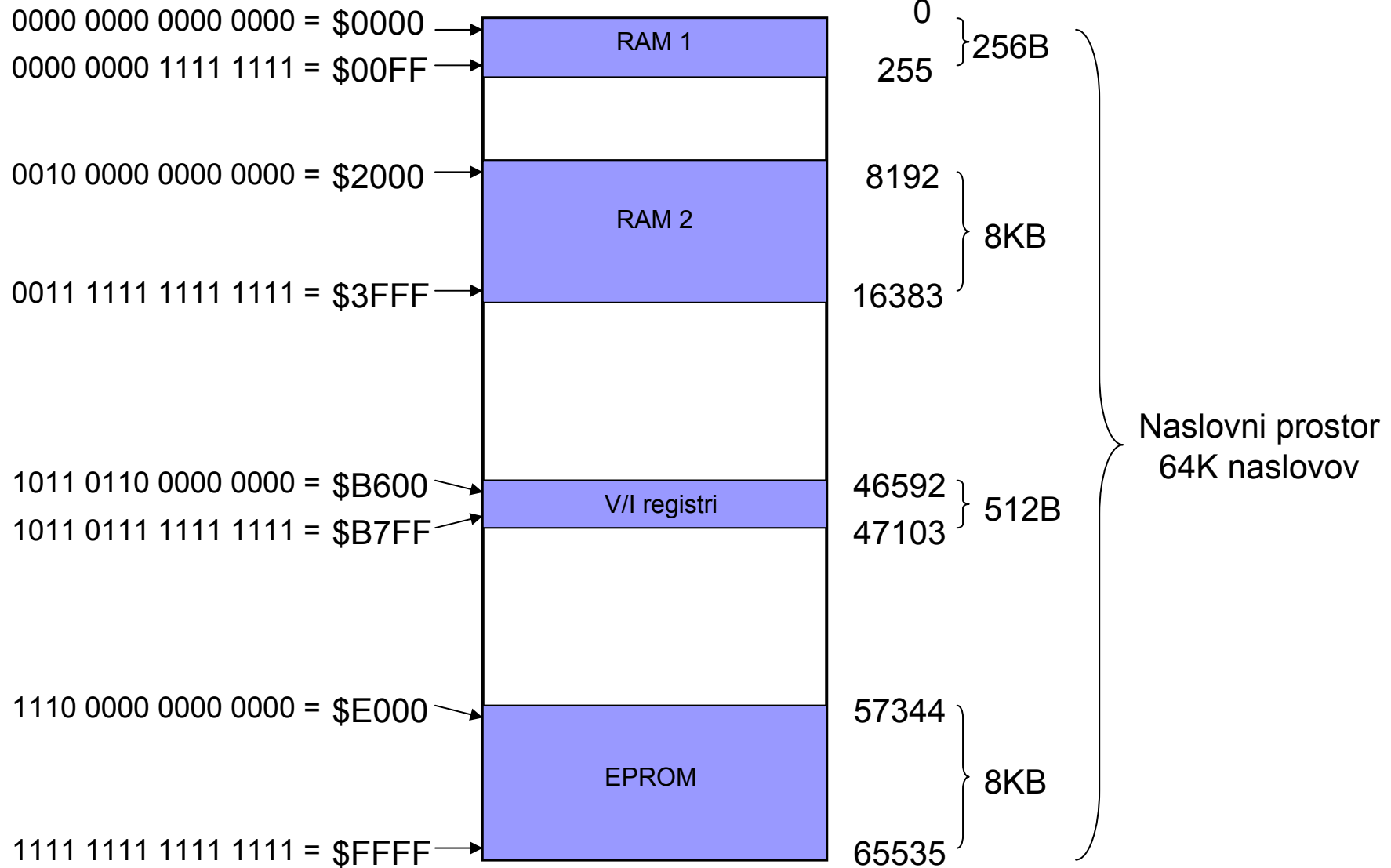
## Vsebina pomnilniške besede

- Vsebina pomnilniške besede se lahko spreminja. V 8-bitno pomnilniško besedo lahko shranimo  $2^8 = 256$  različnih vsebin
- Število pomnilniških besed v glavnem pomnilniku ni nujno enako velikosti naslovnega prostora
- Deli naslovnega prostora so lahko prazni (vsi naslovi niso uporabljeni)  $\Rightarrow$  glavni pomnilnik je manjši od največje možne velikosti



## Slika pomnilnika pri 68HC11 (memory map)

16-bitni pomnilniški naslov

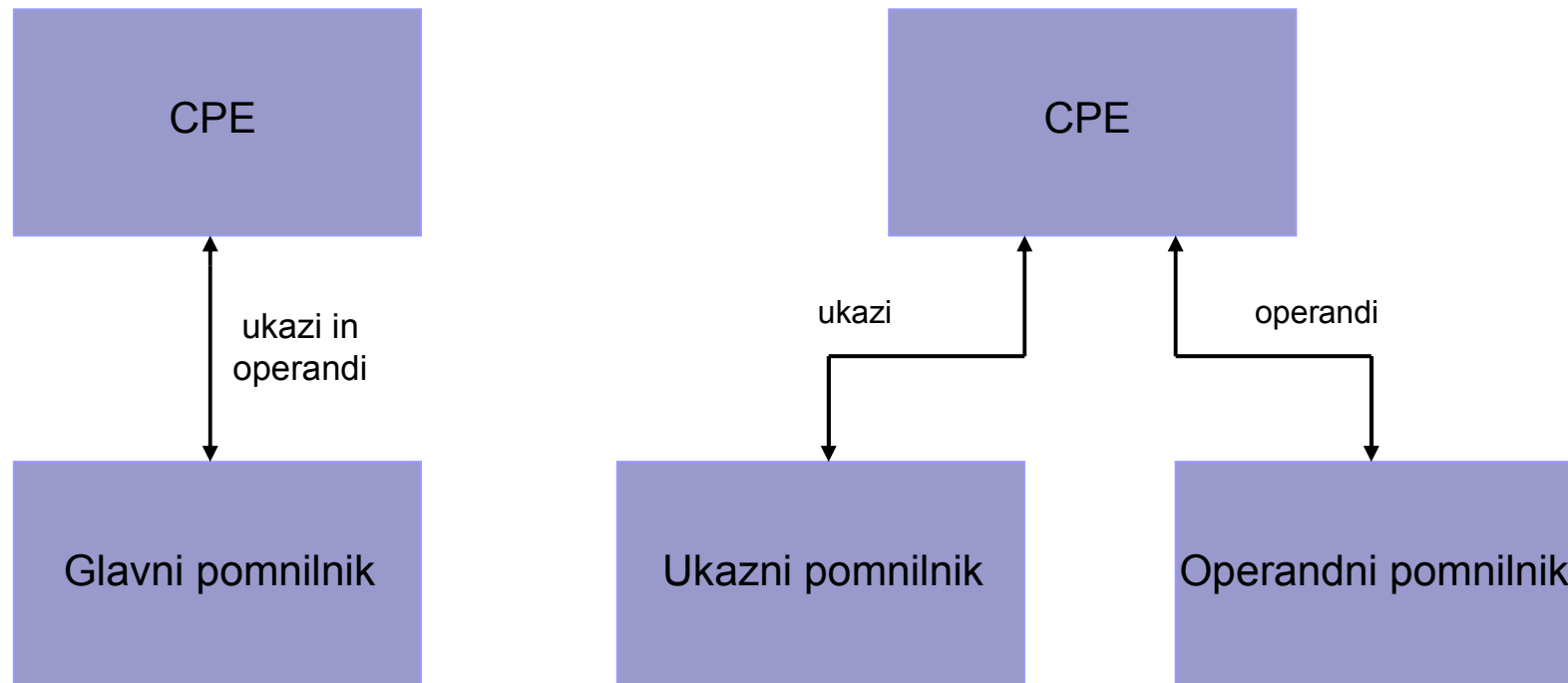




# Dostop do pomnilnika

- CPE uporablja glavni pomnilnik tako, da pošlje v pomnilnik naslov pomnilniške besede in informacije o vrsti prenosa (smer prenosa, število pomn. besed, ki naj se prenesejo)
- Smer prenosa
  - CPE ← gl.pomnilnik – branje (bralni dostop)
  - CPE → gl.pomnilnik – pisanje (pisalni dostop)
- Prenosi CPE ↔ gl.pomnilnik – promet
- Von Neumannovo ozko grlo – povezava med CPE in glavnim pomnilnikom

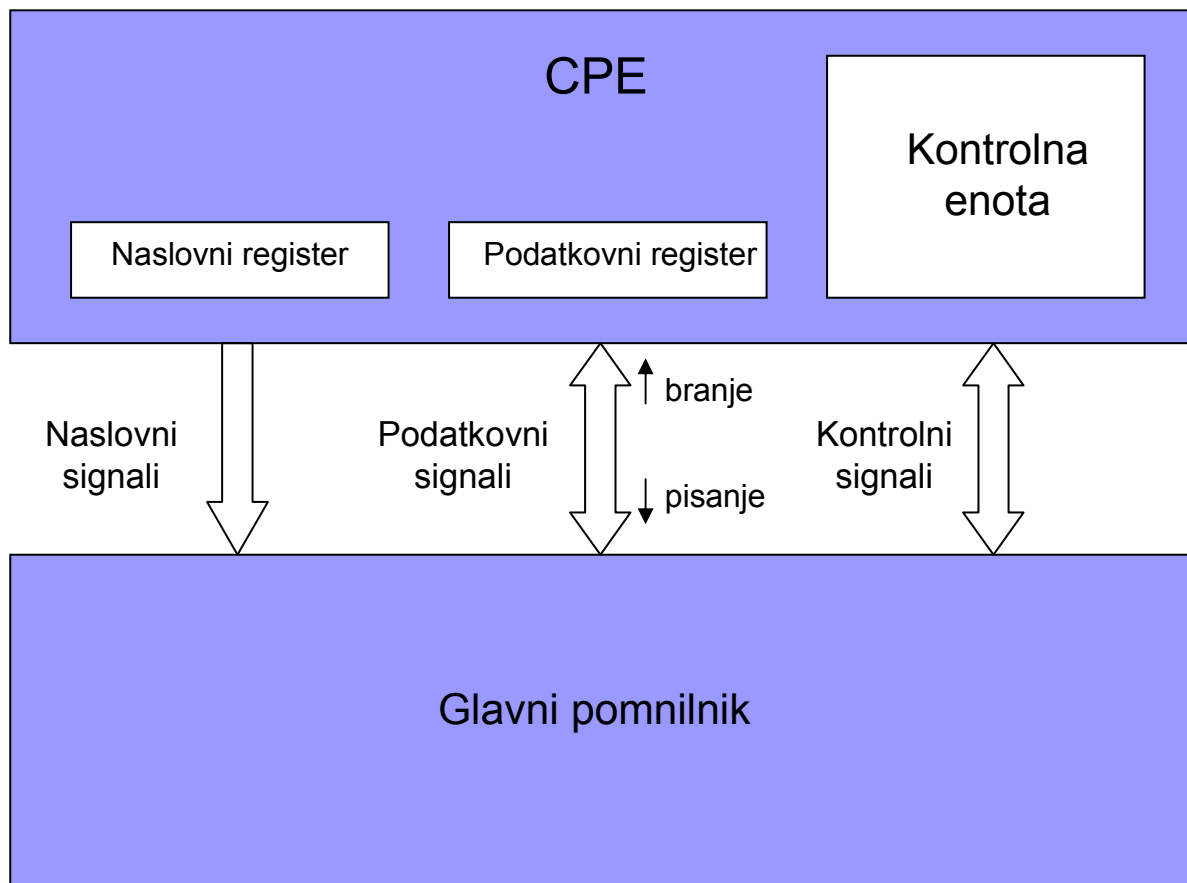
# Razširitev von Neumannovega ozkega grla



Princetonska arhitektura

Harvardska arhitektura

# Povezava med CPE in glavnim pomnilnikom





## Povzetek lastnosti glavnega pomnilnika v von Neumannovem računalniku

- Pomnilnik je enodimenzionalen in organiziran v besede
- Ni razlike med ukazi in operandi
- Pomen ni sestavni del operandov
- Veliko več bralnih kot pisalnih dostopov
  - Razmerje: okrog 80% branj, 20% pisanj
  - Zakaj?



Kombinacija 8 bitov v pomnilniku  
npr. 1000 1011 lahko predstavlja:

- število brez predznaka: 139(Desetiško)
- število s predznakom: - 11(Desetiško)
- znak v razširjeni ASCII abecedi: <
- Strojni ukaz (op. koda 68HC11): ADDA
- pomnilniški naslov: 139(Desetiško)
- kombinacijo bitov



## 3.3 Vhodno izhodni sistem v von Neumannovem računalniku

- V/I naprave, ki služijo pretvarjanju informacij iz ene oblike v drugo.
- V/I naprave za shranjevanje informacij – pomožni pomnilniki.
- Osnovni način delovanja V/I sistema: prenos podatkov med V/I sistemom in glavnim pomnilnikom.



## Izvedbe V/I prenosa

- Programski vhod/izhod (programmed I/O – PIO)
  - Z V/I napravo komunicira CPE
  - Potek prenosa V/I → CPE → pomnilnik ali pomnilnik → CPE → V/I
  
- Neposreden dostop do pomnilnika (direct memory access – DMA)
  - Potreben DMA krmilnik
  - Potek prenosa V/I → pomnilnik (preko DMA krmilnika) ali obratno





## Krmilnik V/I naprave

- Vsaka V/I naprava je priključena preko krmilnika naprave.
- Gledano iz CPE je krmilnik videti kot določeno število registrov, v katere se lahko piše ali iz njih bere.
- Pisanje (lahko tudi branje) v register lahko sproži operacijo v V/I napravi (ukaz napravi).
- Branje iz registra odraža stanje naprave po operaciji (status naprave).



## Naslovi registrov v krmilniku

- Pomnilniško preslikan vhod/izhod (memory mapped I/O) – registri krmilnikov so iz CPE videti enako kot pomnilniške lokacije.
- Ločen vhodno/izhodni naslovni prostor – potrebni so posebni vhodno/izhodni ukazi za dostop do registrov krmilnikov.
- Posredno naslavljanje preko vhodno/izhodnih procesorjev – registri krmilnikov so v posebnem naslovnem prostoru, do katerega imajo dostop samo V/I procesorji.



## 3.4 Lokalnost pomnilniških dostopov

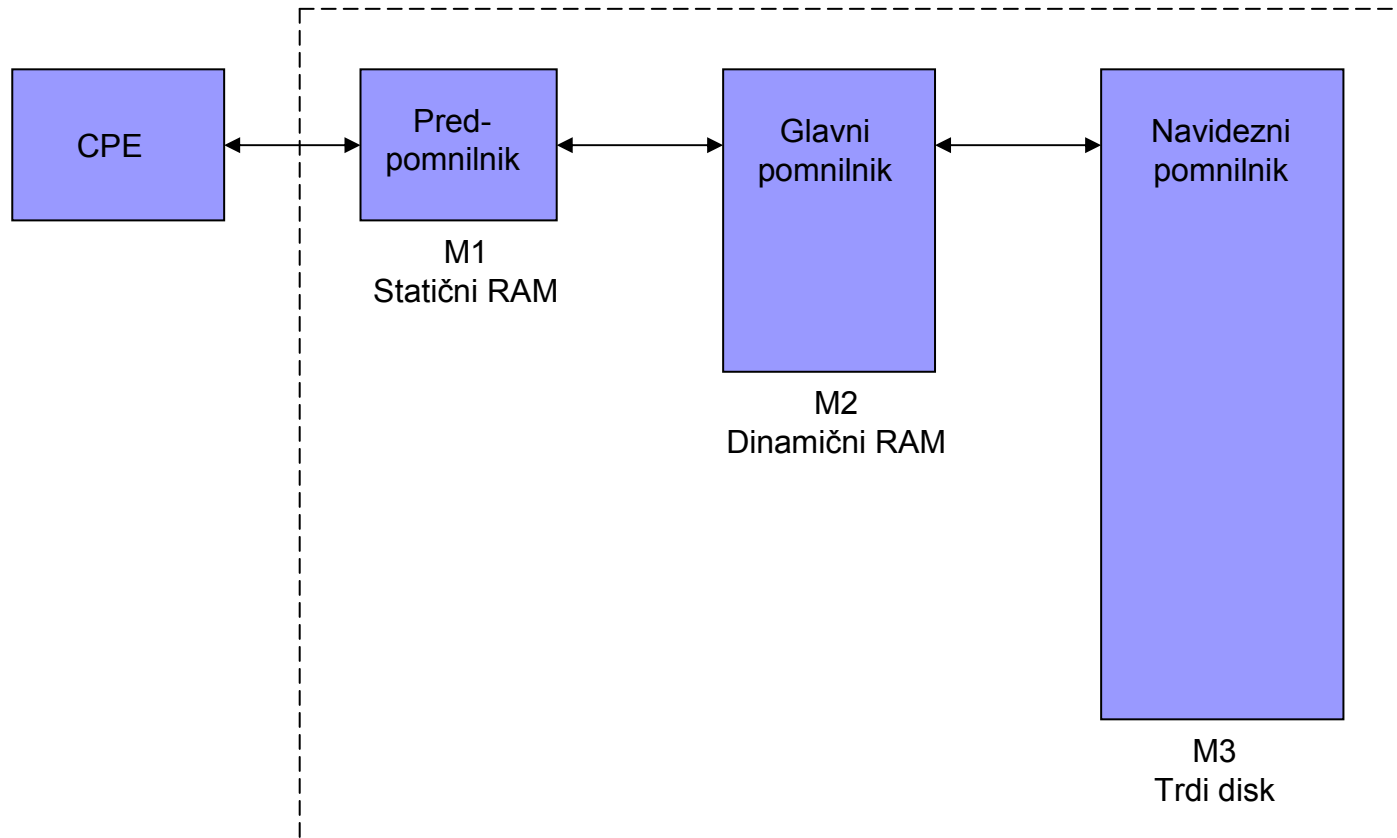
- Princip lokalnosti: eden najpomembnejših pojavov, ki ga opazimo pri delovanju von Neumannovega računalnika.
- Programi pogosto več kot enkrat uporabijo iste ukaze in operande.
- Programi bolj pogosto uporabljajo ukaze in operande, ki so v pomnilniku blizu trenutno uporabljanim.
- Izkustveno pravilo: tipičen program 90% časa uporablja samo 10% ukazov.



# Pomnilniška hierarhija

- Lokalnost pomnilniških dostopov omogoča, da glavni pomnilnik nadomestimo s pomnilniško hierarhijo.
- Pomnilniško hierarhijo sestavlja več ločenih pomnilnikov z različnimi lastnostmi
  - Prvi v hierarhiji pomnilnik M1 (najbližji CPE) je najhitrejši najdražji in najmanjši
  - Zadnji v hierarhiji pomnilnik Mn (najbolj oddaljen od CPE) je najcenejši, največji in najpočasnejši
- Cilj pomnilniške hierarhije je, da je velik, počasen in cenen pomnilnik M3 videti kot hiter in drag pomnilnik M1.

# Trinivojska pomnilniška hierarhija



Glavni pomnilnik kot je definiran v von Neumannovem modelu



## Delovanje pomnilniške hierarhije

- Pravilo delovanja hierarhije je, da je pomnilniški prostor na nivoju  $i$  podmnožica prostora na nivoju  $i+1$ .
- Če informacije do katere želi CPE ni v M1, se mora prenesti iz M2 v M1. Če je ni tudi v M2, se najprej prenese iz M3 v M2 in nato iz M2 v M1.




- Prenášanje se izvaja avtomatsko, brez sodelovanja programerja.
- Za CPE je pomnilniška hierarhija videti kot glavni pomnilnik velikosti M3, s hitrostjo ki je blizu hitrosti M1.
- Pomnilniška hierarhija je brez lokalnosti neuporabna.



# Lokalnost pomnilniških dostopov

- Lokalnost je posledica načina delovanja von Neumannovega računalnika
  - Ukazi se iz pomnilnika jemljejo eden za drugim po naraščajočih naslovih (če ni skoka)
- in načina pisanja programov
  - Zanke v programih
  - V programih pogosto nastopajo podatkovne strukture, ki se uporabljajo po naraščajočih naslovih (polja,...)



- 
- Prostorska lokalnost: po pojavu pomnilniškega naslova  $A(i)$  je verjetno, da bodo naslednji naslovi npr.  $A(i+n)$ , blizu naslova  $A(i)$ 
    - Če ni skokov, se ukazi berejo iz pomnilnika po naraščajočih naslovih
    - Podatkovne strukture, procedure
  - Časovna lokalnost: program ob nekem času  $t$  pogosto tvori naslove, ki jih je tvoril malo pred  $t$  in jih bo tvoril tudi nekoliko po  $t$ .
    - Zaradi zank v programih se isto zaporedje ukazov in s tem pomnilniških naslovov večkrat ponovi

## 3.5 Amdahlov zakon (1967)

- G.M. Amdahl eden od arhitektov slavne serije računalnikov IBM 370
- Če v računalniku za faktor  $N$  ( $N$ -krat) pohitrimo delovanje pri vseh operacijah razen pri  $f$  – temu delu od vseh operacij, potem je povečanje hitrosti celotnega računalnika  $S(N)$  enako:

$$S(N) = \frac{1}{f + (1 - f) / N} = \frac{N}{1 + (N - 1) f}$$

$f$  = delež operacij, ki niso pohitrene

$(1 - f)$  = delež operacij, ki so  $N$ -krat pohitrene



## Case/Amdahlovi pravili

- Nastali pri razvoju IBM 370. Računalnik je dobro zasnovan, če sta izpolnjeni zahtevi dveh pravil:
- Prvo Case/Amdahlovo pravilo: velikost glavnega pomnilnika v bajtih mora biti najmanj enaka številu ukazov, ki jih CPE izvede v eni sekundi
- Drugo Case/Amdahlovo pravilo: zmogljivost V/I sistema v bitih na sekundo mora biti najmanj enaka številu ukazov, ki jih CPE izvede v eni sekundi



## 3.6 Prenosne poti v von Neumannovem računalniku

- Tvorijo povezavo med CPE, glavnim pomnilnikom in V/I sistemom
- Hiter računalnik zahteva povezave, ki omogočajo prenos čim več podatkov v časovni enoti

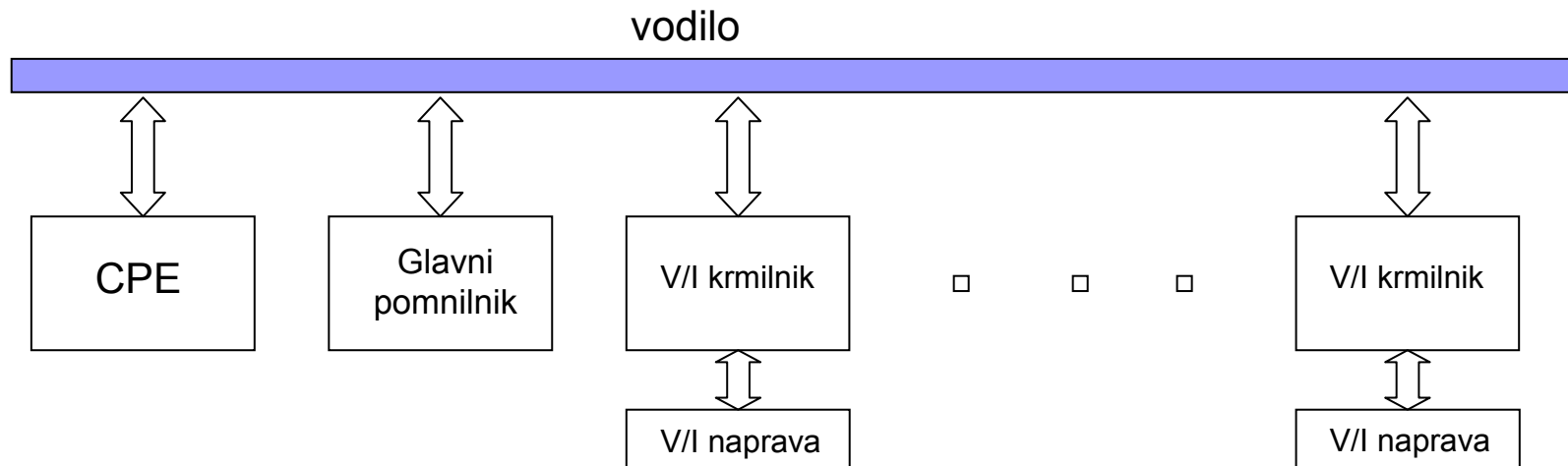


- Količino informacije, ki jo lahko prenesemo po določeni poti, lahko povečamo:
  - S krajšanjem časa, ki je potreben za en prenos
  - Z večanjem števila bitov, ki se prenesejo v enem prenosu
  
- Ena prenosna pot omogoča, da se v nekem trenutku izvaja samo en prenos
  
- Na hitrost vpliva zato tudi število prenosnih poti – govorimo o povezovalnih strukturah

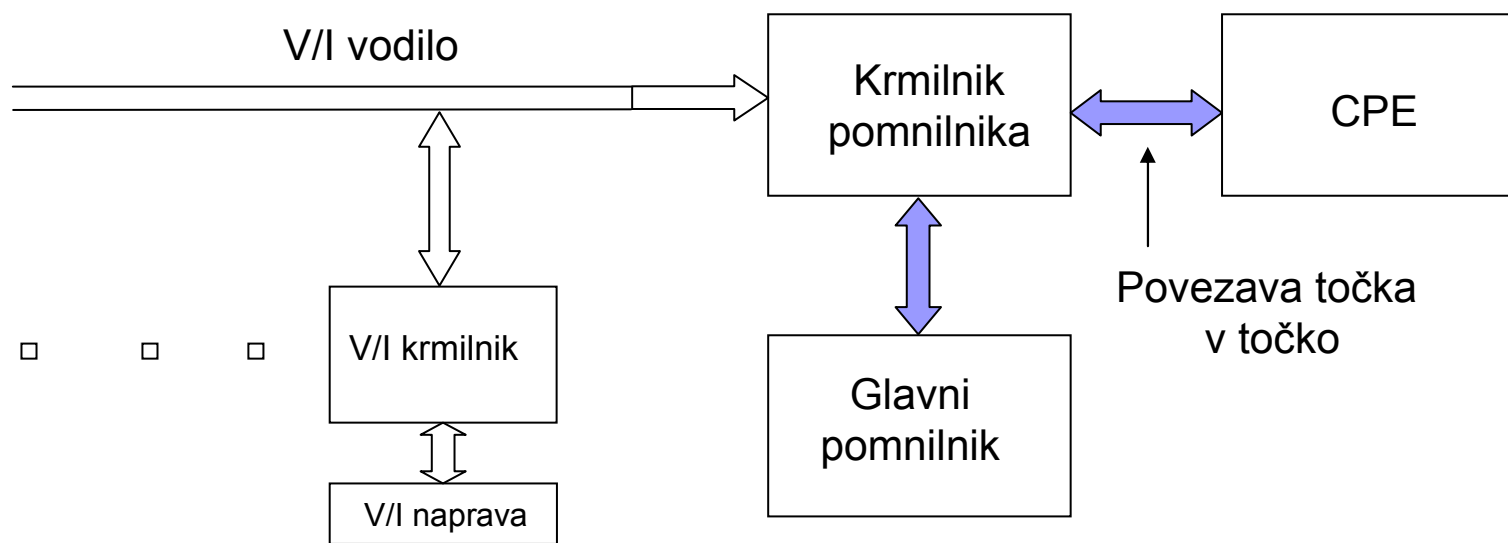
# Vrste prenosnih poti

## ■ Vodilo (angl. bus)

- Vodilo si delijo vse enote, ki so nanj priključene. Fizično je vodilo množica paralelnih linij (žic) po katerih potujejo električni signali. Linije imajo odcepe, na katere so priključene enote



- Povezava točka v točko (angl. point to point)
  - Prenosna pot, ki povezuje samo dve napravi





## Signali, ki se prenašajo po prenosnih poteh

- Prenosno pot sestavlja določeno število povezovalnih linij, po katerih se prenašajo signali.
- Podatkovni signali
  - Število podatkovnih signalov (linij) je enako številu bitov, ki se naenkrat prenašajo – širina prenosne poti
  - Pri širini 64 bitov so možni 8, 16, 32 in 64-bitni prenosi





## ■ Naslovni signali

- Določajo naslov pomnilniške besede ali V/I naprave (registra v krmilniku) na katero se nanaša prenos – število naslovnih signalov (bitov) določa velikost naslovnega prostora

## ■ Kontrolni signali

- Določajo smer prenosa (branje ali pisanje), število prenešenih bitov in časovno zaporedje dogodkov pri prenosu



## Vloge naprav pri prenosu

- Prenos vedno vodi gospodar (angl. master), ostale naprave so sužnji (angl. slave). Vsak prenos poteka samo med dvema napravama
- Pri vodilih je gospodarjev lahko več, pri povezavi točka v točko samo eden
- Pri več gospodarjih je potreben mehanizem za dogovor (arbitraža), kdo od gospodarjev bo imel pri prenosu nadzor nad vodilom



## Zaporedje dogodkov pri prenosu

- Gospodar pošlje na prenosno pot naslov do katerega želi dostop (naslovne signale postavi v stanje, ki ustreza ciljnemu naslovu)
- Gospodar s kontrolnimi signali poda smer prenosa in število bitov, ki naj se prenesejo (če so možni prenosi različnih širin)
- Kadar je smer prenosa pisanje, gospodar na prenosno pot pošlje tudi podatkovne signale



- Gospodar s kontrolnimi signali tudi določi začetek in konec prenosa
- Ko se prenos začne vse enote priključene na prenosno pot (sužnji) svoje naslove z naslovom na naslovnih signalih. Enota (suženj), ki ugotovi enakost opravi prenos, ki ga gospodar zahteva
- ! Prenos: enkratni paralelni prenos največ toliko bitov, kot je širina podatkovne prenosne poti

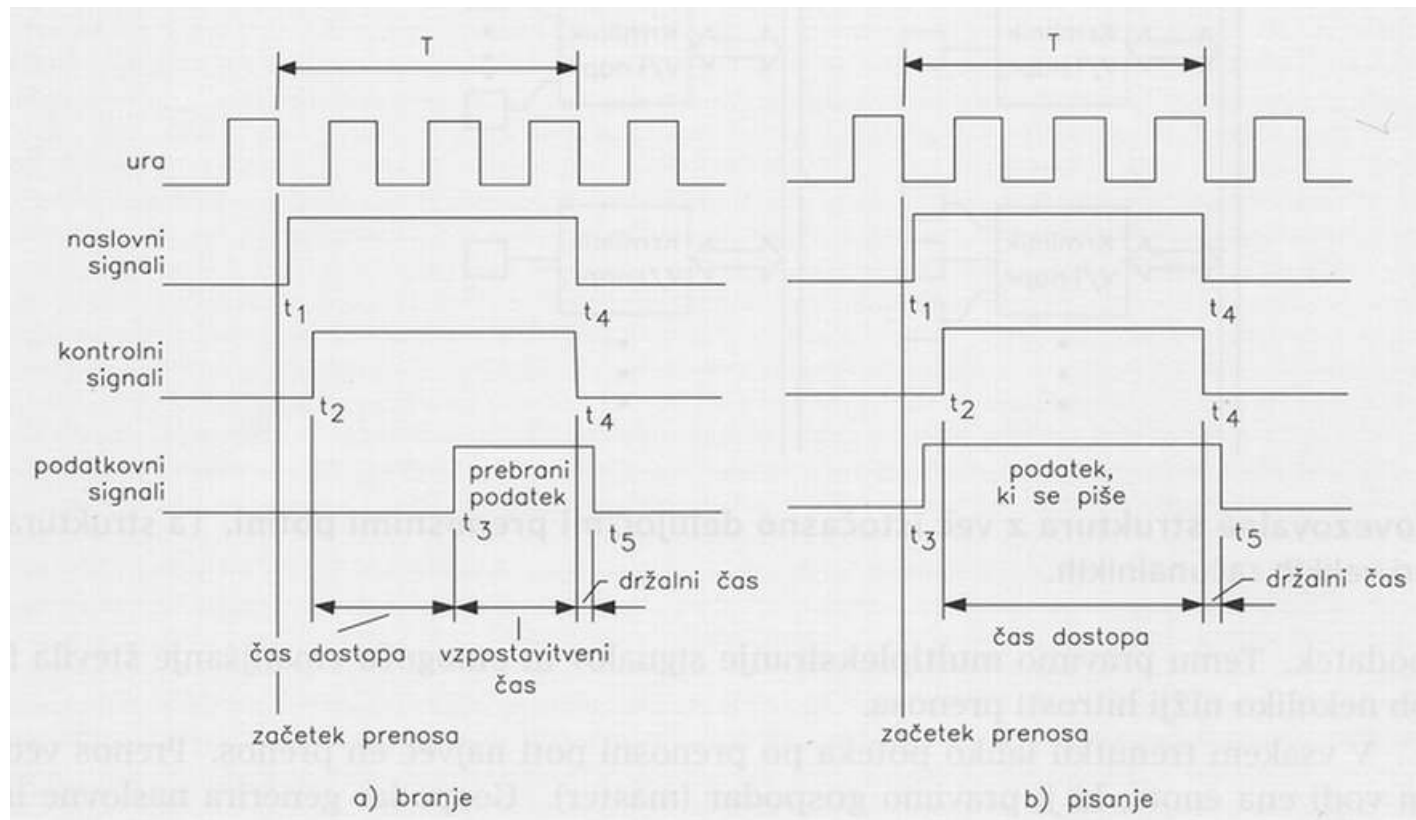


# Dve vrsti prenosa glede na določanje začetka in konca prenosa

## ■ Sinhronski prenos

- Čas prenosa  $T$  je vedno enak in vnaprej določen (celo število urinih period gospodarja)
- Če so na prenosno pot priključene različno hitre naprave (sužnji), mora biti čas  $T$  izbran glede na najpočasnejšo enoto
- Po preteku časa  $T$  gospodar prenos konča

# Sinhronski prenos



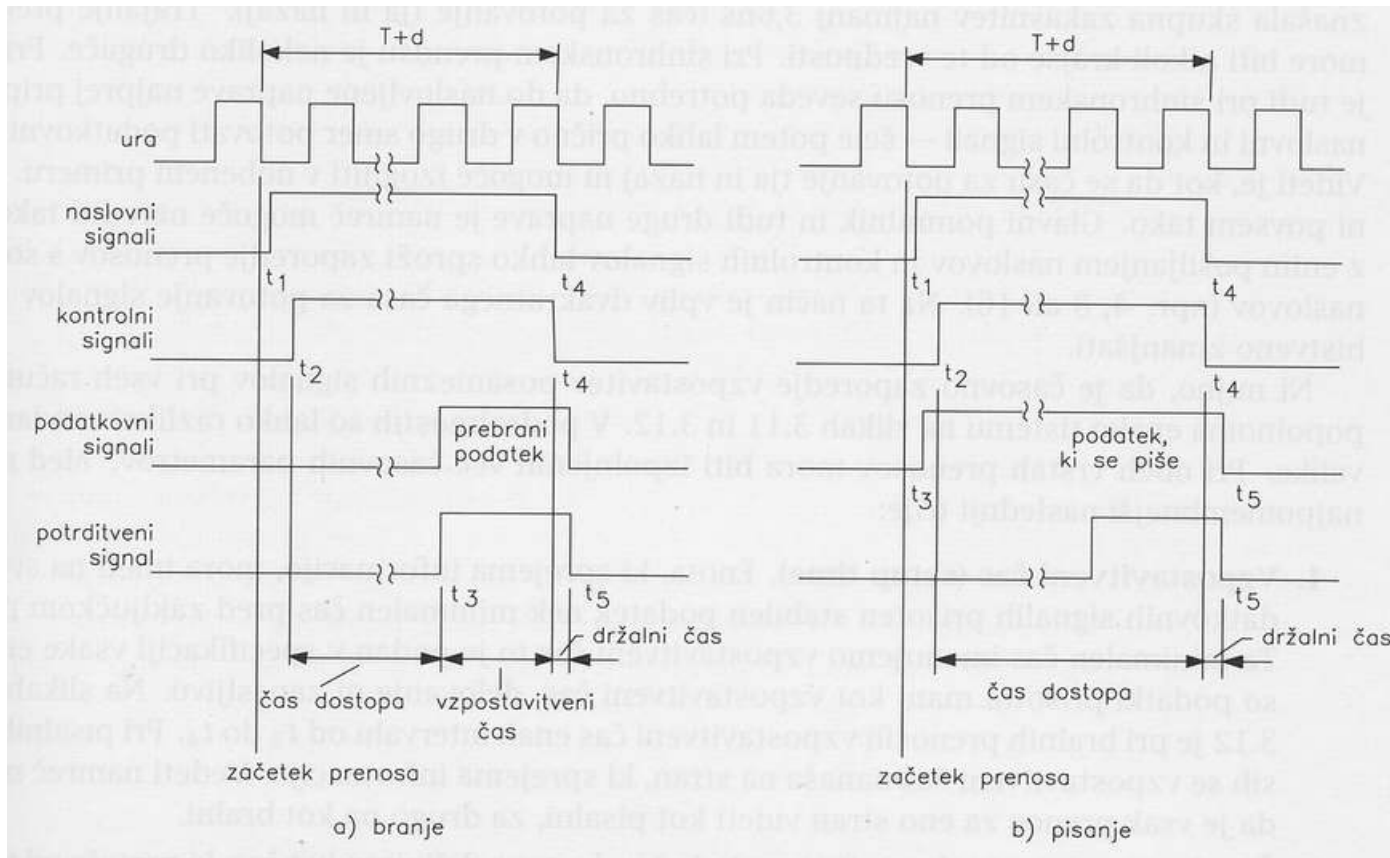
D.Kodek: Arhitektura računalniških sistemov, str.66



## ■ Asinhronski prenos

- Čas prenosa  $T$  ni vnaprej določen (vendar tudi tu celo število urinih period)
- Začetek prenosa je enak kot pri sinhronskem prenosu
- Gospodar prenosa zaključi, ko od naslovljene naprave (sužnja) dobi potrditveni signal
- Trajanje prenosa je torej odvisno od naslovljene naprave
- Hitre enote aktivirajo potrditveni signal hitreje, počasnejše kasneje
- Zaščita da ob napaki prenos ne bi trajal neskončno dolgo

# Asinhronski prenos



D.Kodek: Arhitektura računalniških sistemov, str.67





# Časovni parametri pri prenosu

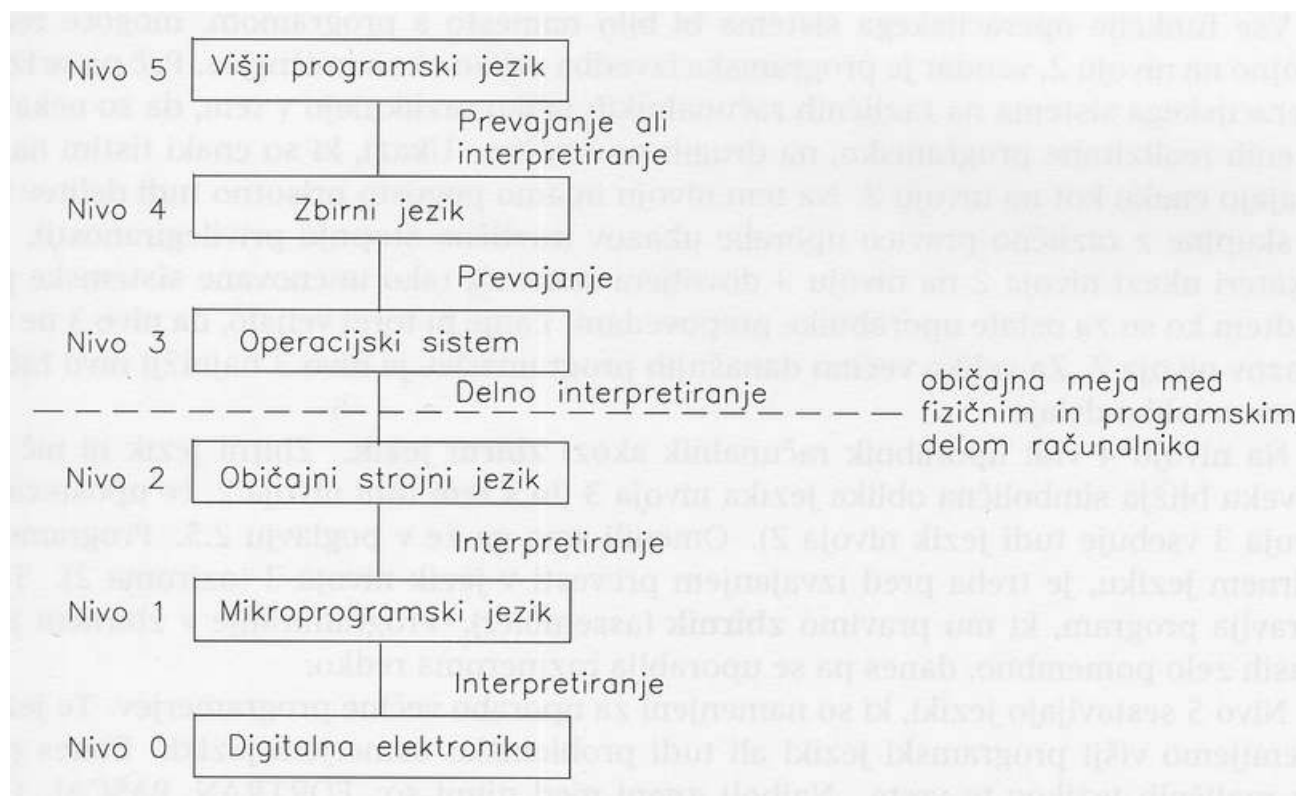
- Vzpostavitevni čas (angl. Setup time)
- Čas dostopa (angl. access time)
- Držalni čas (angl. hold time)



# Kapaciteta prenosne poti

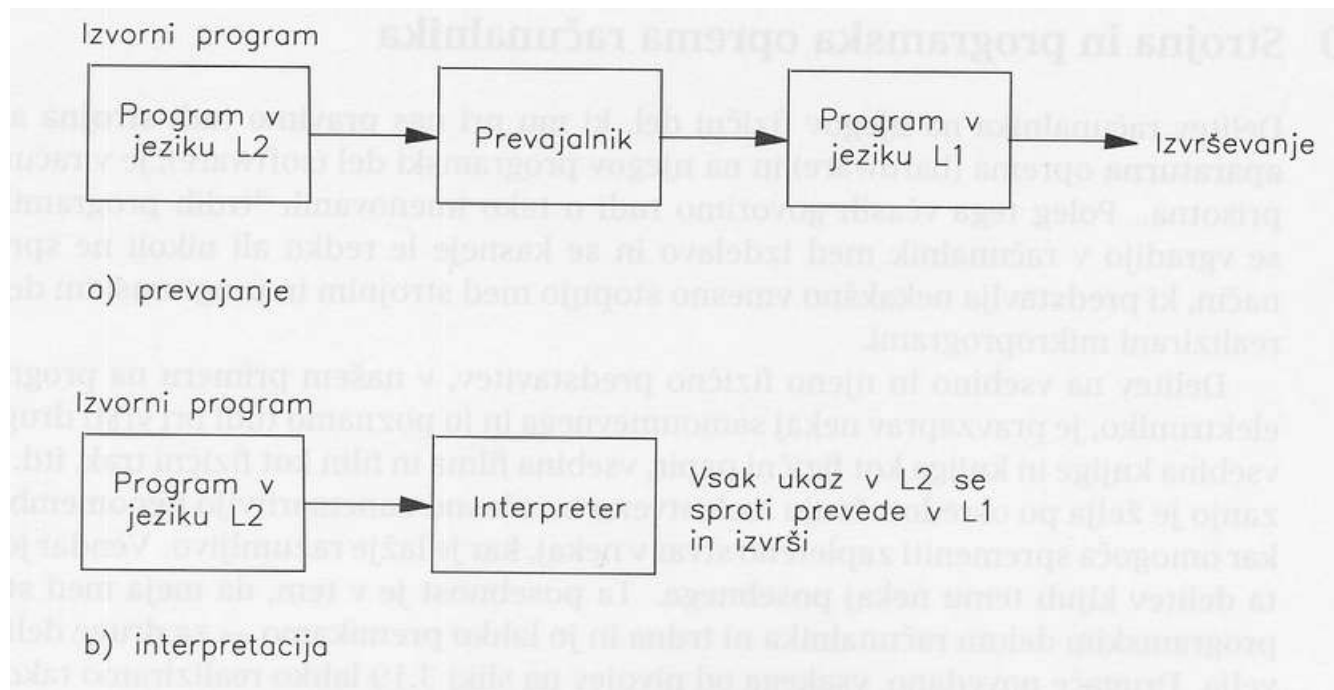
- Kapaciteta prenosne poti  $B$  je največje možno število prenešenih besed (bitov, bajtov) v sekundi
- Primer: Kapaciteta PCI vodila
  - Širina prenosne poti = 32 bitov (=4 bajte)
  - Frekvenca urinega signala na vodilu = 33 MHz
  - En prenos traja eno urino periodo  $\Rightarrow$
  - V eni sekundi se opravi  $33 \cdot 10^6$  prenosov
  - Kapaciteta vodila  $B = 32\text{b} \cdot 33 \cdot 10^6 \text{ 1/s} = 1056 \cdot 10^6 \text{ b/s} = 1056 \text{ Mb/s} = 132 \text{ MB/s}$

## 3.7 Računalnik kot zaporedje navideznih računalnikov



D.Kodek: Arhitektura računalniških sistemov, str.75

# Prehajanje iz jezika L2 v jezik L1



D.Kodek: Arhitektura računalniških sistemov, str.77



## 3.8 Strojna in programska oprema računalnika

- Meja med strojnim in programskim delom računalnika ni trdna – lahko jo premikamo
- Vsakega od nivojev lahko realiziramo tako strojno kot tudi programsko
- Nivo 0 je npr. lahko realiziran s programom, ki teče na drugem računalniku

Strojna in programska oprema sta ekvivalentni