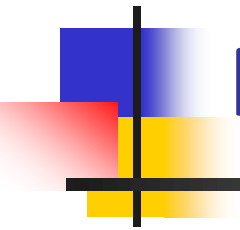
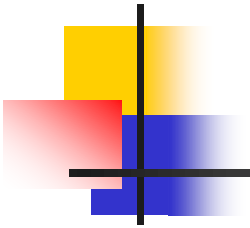


TEHNIKE UBRZANJA OBRADE U RAČUNARSKOM SISTEMU

NRS Predavanje br. 5

Stalan zahtev:
Što veća propusnost i brzina obrade

- **Tehnološki** - usavršavanjem izrade IC
- **Organizaciono** (tema ovog kursa)
  - ubrzanje i/ili paralelizacija procesora
  - paralelizacija računara – multiprocesorski sistemi
- **Programski**
  - razvoj složenijih i efikasnijih kompajlera i OS, koji smanjuju trošenje sistemskih resursa
  - reorganizacija programskog koda radi boljeg iskorišćenja paralelnih resursa procesora, ili multiprocesora)



Tehnike ubrzanja organizacione prirode

- Ubrzanje procesora (izvršenja instrukcija)
  - Vektorski procesor
- Paralelizacija procesora
  - Protočna obrada (organizacija)
  - Umnožavanje funkcionalnih jedinica
- Multiprocesori
  - Više procesora koji dele opterećenje

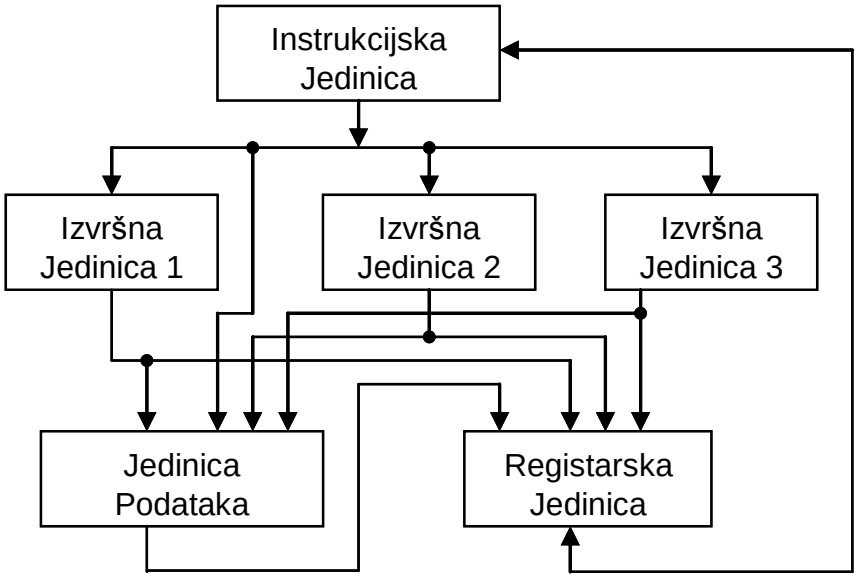
Vektorski procesor

- Vektorski procesor je optimizovan za sekvencijalno (uzastopno) izvršenje iste operacije nad nizom operanada
- Vektorska instrukcija
  - C[i] = A[i] ⊗ B[i], i = 1,...,n
  - Operacija, inicijalna adresa operanada, dužina vektora
  - Samo jednom se učitava i dekodira!
- Tipične vektorske instrukcije su:
  - Aritmetičke operacije u pokretnom zarezu;
  - Logičke operacije, upoređivanje i test;
  - Matrične operacije, pretrage tipa min/max i sl.

```
for( i=0; i<50; i++ )      // petlja x 50
    c[i] = a[i]+ b[i];     // (2 x čitaj + 1 x piši) x 50
```

Umnožavanje funkcionalnih jedinica

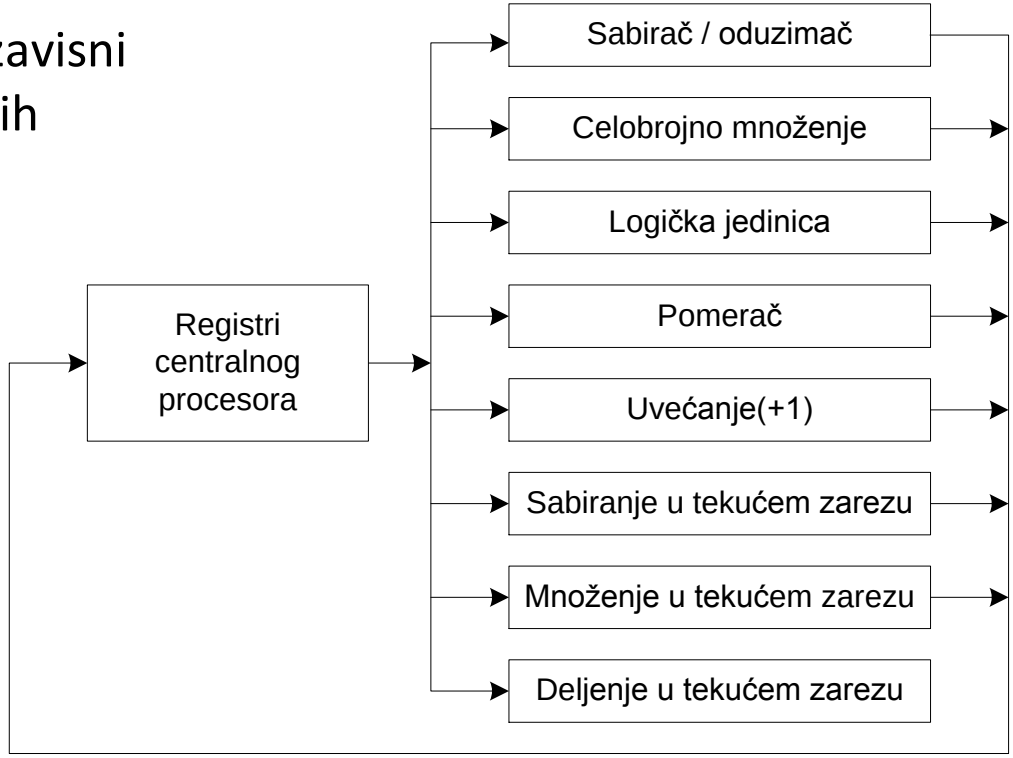
- Paralelna obrada se ostvaruje distribucijom podataka (instrukcija) između paralelnih izvršnih jedinica
- Mogućnost rizika između instrukcija u izvršenju (registri, izv.jedinice, pristup memoriji, raspoloživost rezultata)
- Propusnost CP je ograničena trajanjem najduže od instrukcija u izvršenju



Broj izvršnih jedinica	Trajanje obrade	Faktor ubrzanja
1	17 + 8 + 2 = 27 taktova	-
3	max(17, 1+8, 2+2) = 17 taktova	27 / 17 = 1,58 puta

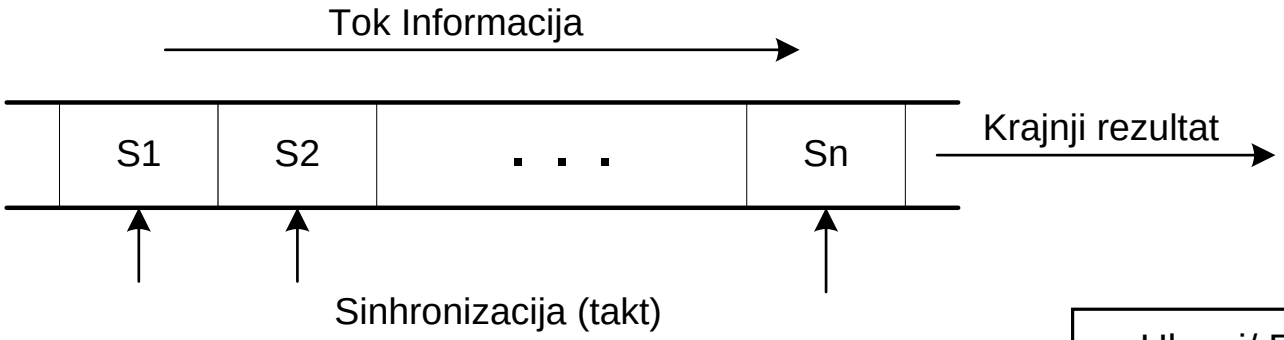
Varijanta sa usmeravanjem instrukcije ka jednoj od specijalizovanih izvršnih jedinica

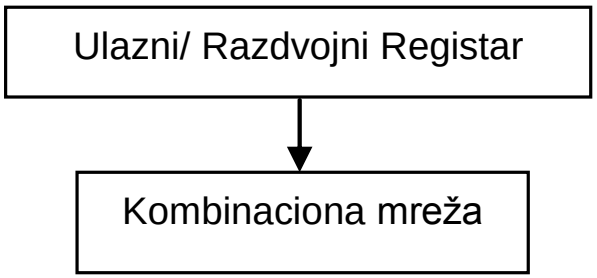
- Superskalarni procesori
- Svi blokovi su međusobno nezavisni
- Postoji mogućnost međusobnih rizika
- On-line analiza i distribucija instrukcija
- Prekoredno izvršenje
- VLIW procesori – pomoć kompajlera



Procesori sa protočnom obradom

- **Protočna obrada** (*pipeline processing*)
  - sekvencijalni proces se izvršava u fragmentima, pri čemu se svaki fragment izvršava u posebnom segmentu konkurentno sa svim ostalim



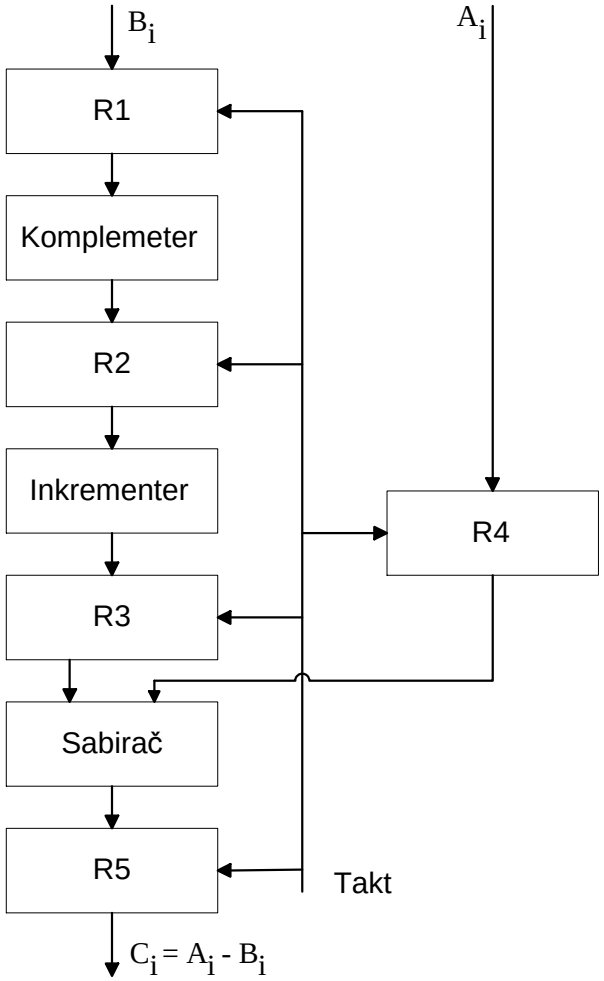


$\overline{Bi}$

Primer:

Ci = Ai - Bi, i = 1,2,..,n

Korak	Operacija	Opis
1	R1 ← Bi	Ulaz Bi
2	R2 ← $\overline{R1}$	Komplementiranje
3	R3 ← R2 + 1, R4 ← Ai	Uvećanje $\overline{Bi}$; Unos Ai
4	R5 ← R3 + R4	R5 ← Ai + $\overline{Bi}$ + 1



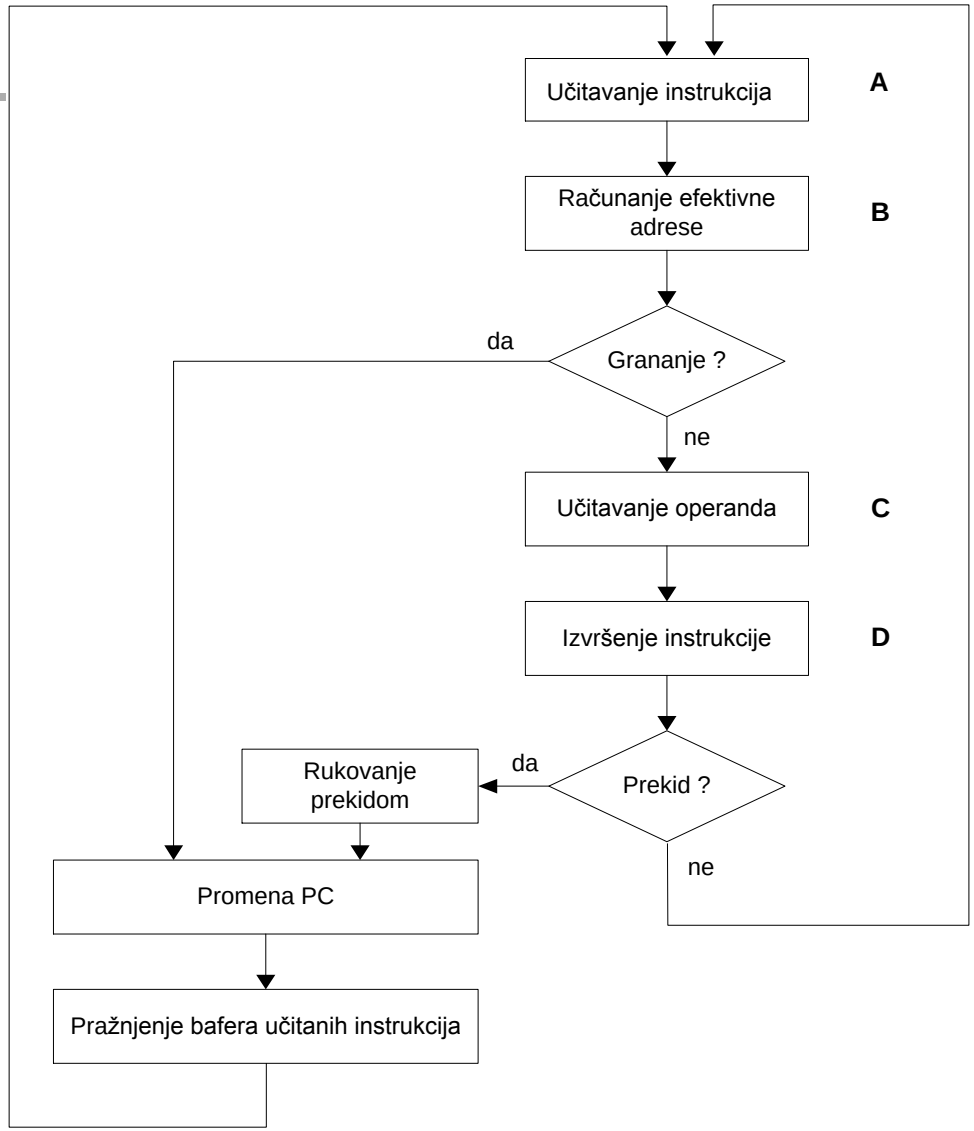
t_i	R1	R2	R3	R4	R5
1	B_1	-	-	-	-
2	B_2	$\overline{B}_1$	-	-	-
3	B_3	$\overline{B}_2$	$\overline{B}_1+1$	A_1	-
4	B_4	$\overline{B}_3$	$\overline{B}_2+1$	A_2	C_1
5	B_5	$\overline{B}_4$	$\overline{B}_3+1$	A_3	C_2
6	B_6	$\overline{B}_5$	$\overline{B}_4+1$	A_4	C_3
7	B_7	$\overline{B}_6$	$\overline{B}_5+1$	A_5	C_4
8	-	$\overline{B}_7$	$\overline{B}_6+1$	A_6	C_5
9	-	-	$\overline{B}_7+1$	A_7	C_6
10	-	-	-		C_7

Efikasnost protočne strukture

- Zavisi od njene popunjenosti
  - Pun “pipeline” – ubrzanje jednako broju segmenata
  - Punjenje, pražnjenje smanjuju efikasnost
  - Grananja su najčešći uzrok pražnjenja
- Problemi koji ograničavaju ubrzanje u odnosu na maksimalno su pre svega:
  - Različito trajanje obrade u raznim segmentima;
  - Neki segmenti se preskaču za deo instrukcija;
  - Dva segmenta mogu istovremeno zahtevati pristup memoriji, tj. jedan mora čekati.
- RISC procesori su pogodniji za protočnu organizaciju

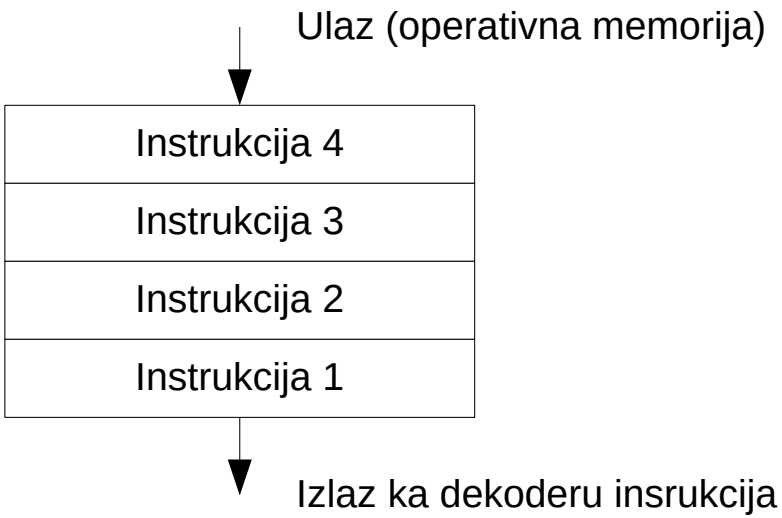
Principijelna organizacija protočnog procesora

- Četiri segmenta
  - Učitavanje instrukcija
  - Obračun efektivne adrese
  - Učitavanje operanda
  - Izvršenje instrukcije



Protočno učitavanje instrukcija

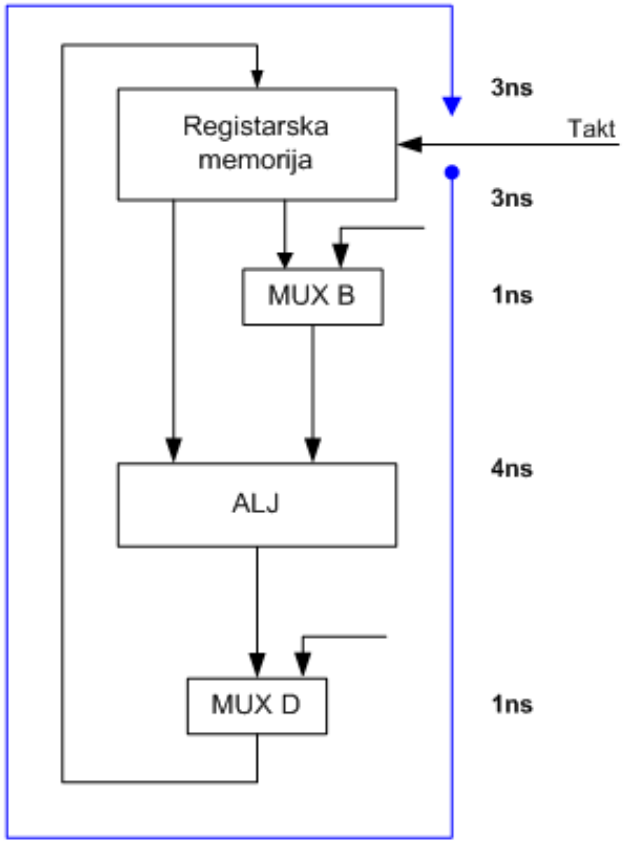
- Pogodnost paralelizacije faze učitavanja
- Instrukcijski bafer (*Instruction Pipeline* - IP)
- Bafer unapred učitanih instrukcija
- Remetilac su instrukcije grananja



Segmentacija internih prenosnih puteva

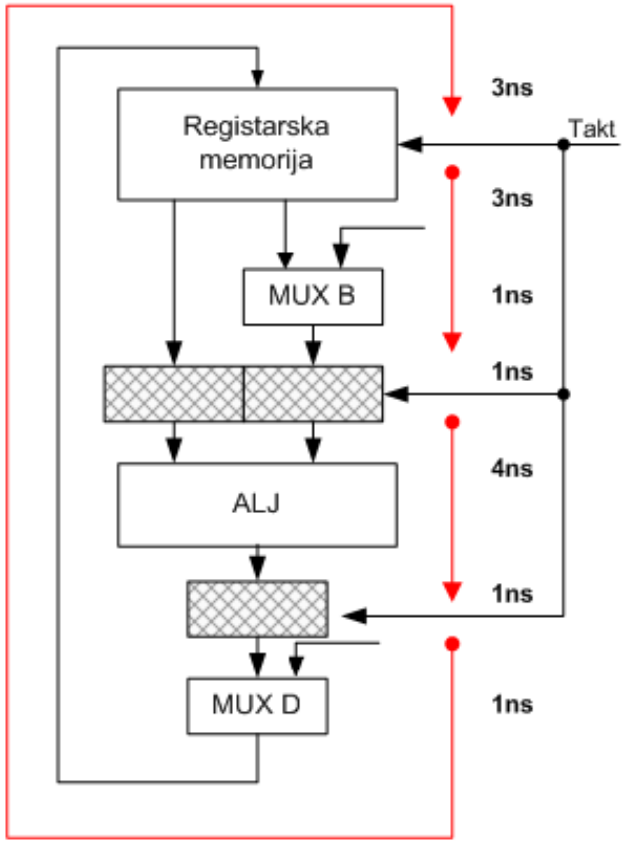
Izvršenje instrukcije u tri segmenta:

- Pribavljanje operanda (DOF)
- Izvršenje instrukcije (EX)
- Zapis rezultata u registarsku memoriju (WB)



Konvencionalna organizacija

Σ d = 12 ns, f_{op} = 83,3 MHz



Protočna organizacija

d_{max} = 5 ns, f_{op} = 200 MHz

Izvršenje instrukcija u protočnom procesoru

Instrukcija↓ Takt→	1	2	3	4	5	6
R1 ← R2 – R3	DOF	EX	WB			
R4 ← shl R4		DOF	EX	WB		
R7 ← R7 + 1			DOF	EX	WB	
R1 ← R0 – R2				DOF	EX	WB

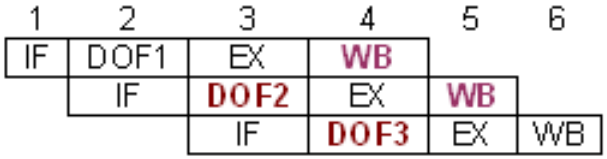
- Istovremeno se mogu izvršavati 3 µ-instrukcije, pa i 3 instrukcije
- Maksimalno (teorijsko) ubrzanje iznosi 12 / 5 = 2,4 puta
- Realno ubrzanje zavisi od pojave instrukcija grananja

Broj instrukcija bez grananja	Konvencionalni procesor			Protočni procesor			Ubrzanje
	T_{cp}	N taktova	T_{ukupno}	T_{cp}	N taktova	T_{ukupno}	
1	12 ns	1	12 ns	5	3	15 ns	0,8
4	12 ns	4	48 ns	5 ns	6	30 ns	1,6
7	12 ns	7	84 ns	5 ns	9	45 ns	1,9
100	12 ns	100	1200 ns	5 ns	102	510 ns	2,35

Rizik podataka (*data hazard*)

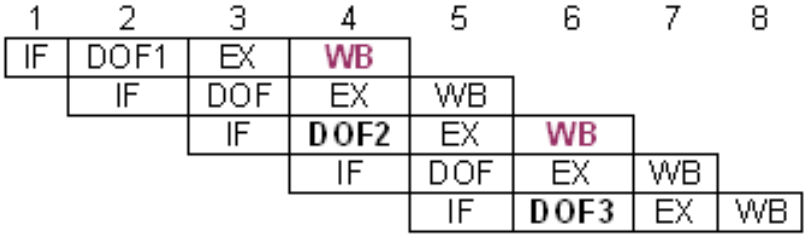
- Nastaje kada sledeća instrukcija pokuša da koristi rezultat prethodne, pre nego što je on raspoloživ

```
MOV R1, R5        R1 ← R5
ADD R2, R1, R6    R2 ← R1 + R6
ADD R3, R1, R2    R3 ← R1 + R2
```

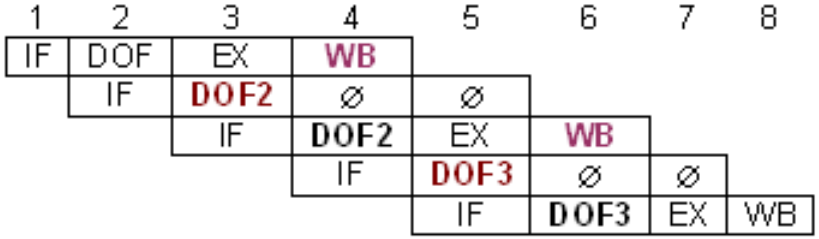


- Lek: umetanje NOP, stall, prosleđivanje podataka (sa MUXD)

```
MOV R1, R5        R1 ← R5
NOP
ADD R2, R1, R6    R2 ← R1 + R6
NOP
ADD R3, R1, R2    R3 ← R1 + R2
```




```
MOV R1, R5        R1 ← R5
(ADD R2, R1, R6)  stall
ADD R2, R1, R6    R2 ← R1 + R6
(ADD R3, R1, R2 ) stall
ADD R3, R1, R2    R3 ← R1 + R2
```



Rizik grananja (*control hazard*)

```
BZ    R1, L1    if(R1=0) PC ← adr(L1)
MOV   R2, R3    R2 ← R3
MOV   R1, R2    R1 ← R2
L1 MOV R5, R6   R5 ← R6
```

	1	2	3	4	5	6	7
	IF	DOF	**EX1**	**WB**			
		IF	DOF	**EX**	**WB**		
			IF	DOF	**EX**	**WB**	
				IF	DOF	EX	WB

- Iza uslovnog grananja se izvršavaju instrukcije i onda kada to nije potrebno
- Lek 1: umetanje NOP/stall (vreme se gubi u svakom slučaju)

```
BZ    R1, L1    if(R1=0) PC ← adr(L1)
NOP
NOP
L1 MOV R5, R6   R5 ? R6
```

	1	2	3	4	5	6	7
	IF	DOF	**EX1**	**WB**			
		IF	DOF	EX	WB		
			IF	DOF	EX	WB	
				IF	DOF	EX	WB

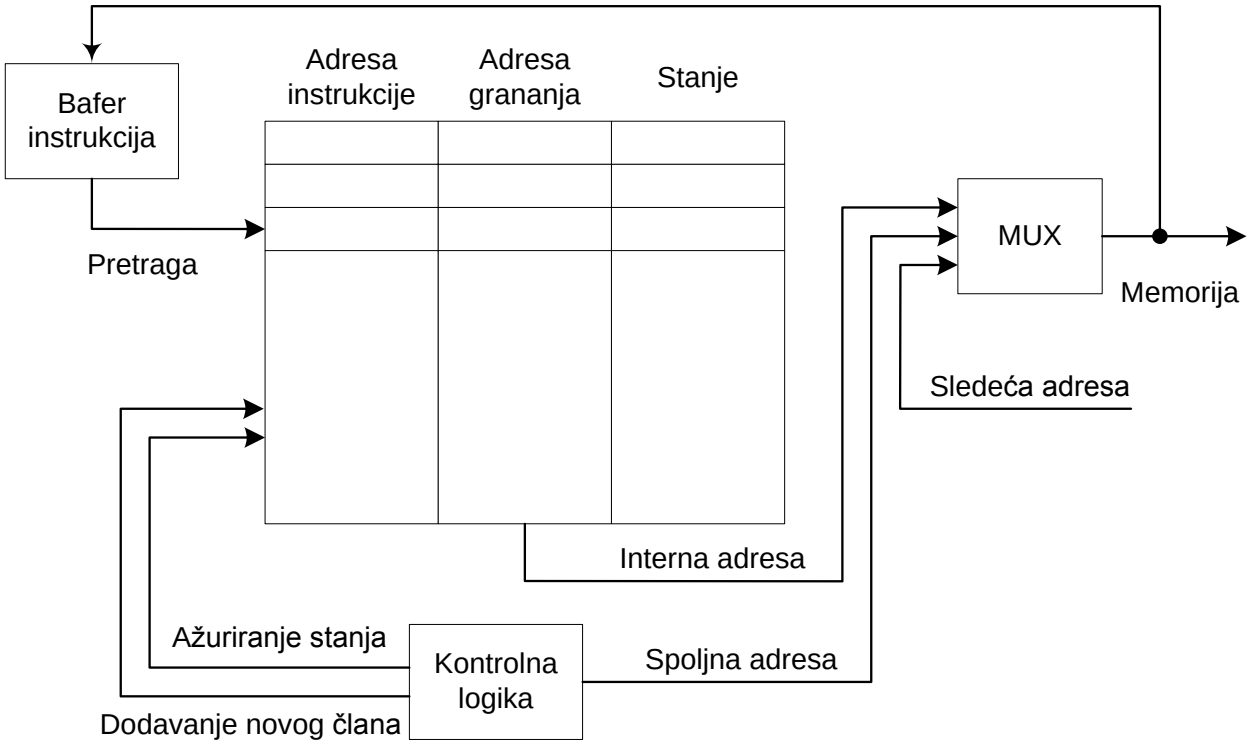
- Lek 2: **predviđanje grananja**: grananja neće biti, UJ zaustavlja izvršenje ako se grananje ipak desi (samo se tada gubi vreme)

```
BZ    R1, L1    if(R1=0) PC ← adr(L1)
MOV   R2, R3    R2 ← R3
MOV   R1, R2    R1 ← R2
L1 MOV R5, R6   R5 ← R6
```

	1	2	3	4	5	6	7
	IF	DOF	**EX1**	**WB**			
		IF	DOF	∅	∅		
			IF	∅	∅	∅	
				IF	DOF	EX	WB

Dinamičke metode predviđanja skoka

- Uspešnije pogađanje na osnovu prethodnog izvršenja grananja
  - **Indikator grananja** (*taken/not taken switch*)
  - **Tabela istorijata grananja (***branch history table*)

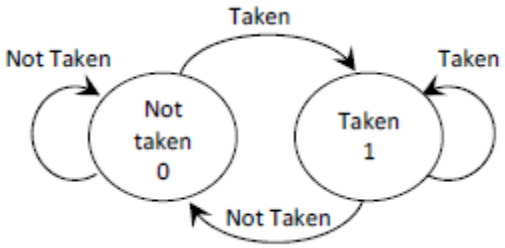


Indikator grananja

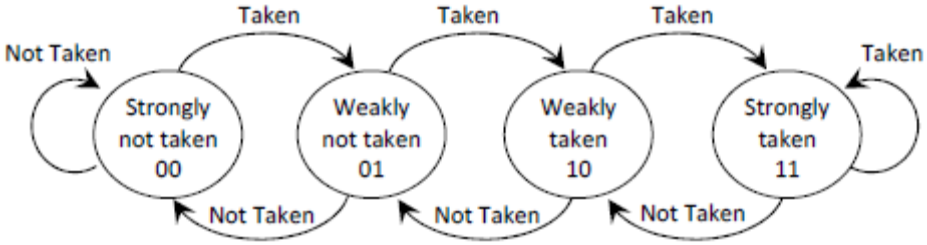
- *Taken/Not taken switch*
  - 1 ili 2 bita
- Prebroj promašaje u oba slučaja

```
int i, j, c=0;

for (i=0; i<500; i++)
{
   for (j=0; j<4; j++)
   {
      c++;
   }
}
```



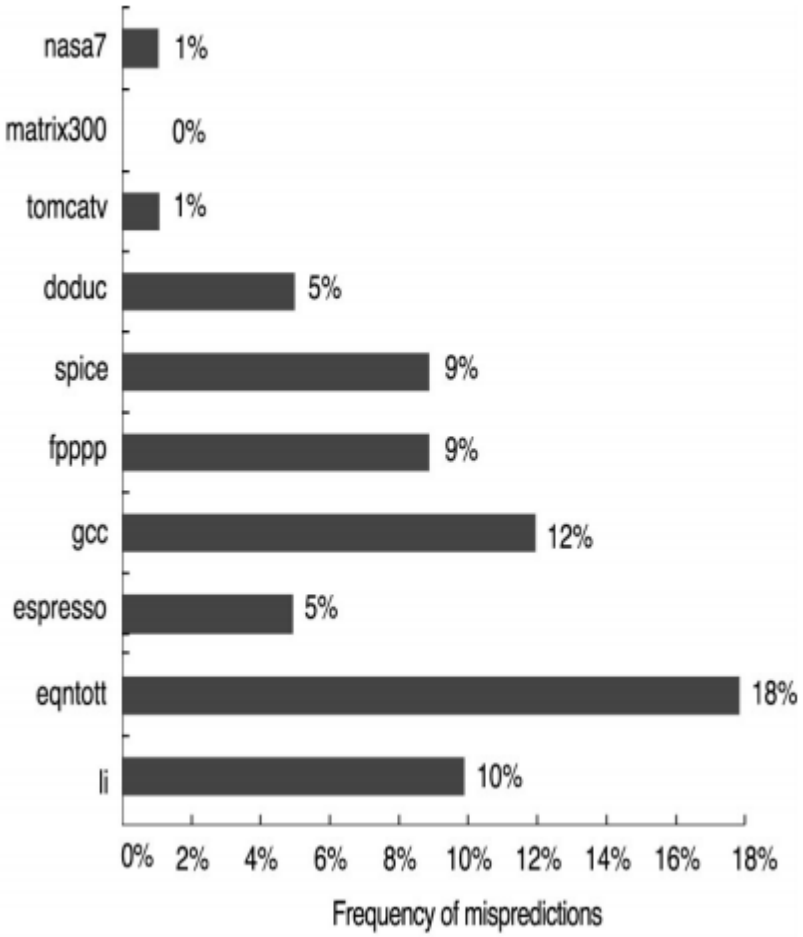
1-bit	P	B	
0	0	1	x
1	1	1	✓
.	.	.	✓
1	1	0	x

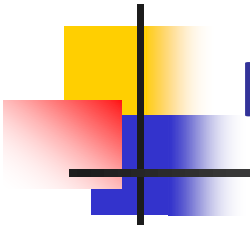


2-bit	P	B	
00	0	1	x
01	0	1	x
10	1	1	✓
11	1	1	✓
.	.	.	✓
11	1	0	x
10	1	1	✓
11	1	1	✓
.	.	.	✓

Efikasnost BHT predviđanja

- Oko 20% instrukcija su uslovna grananja
- Cena promašaja je sve skuplja zbog dubine pipeline-a
- Zato je važno što pouzdanije predviđanje skoka
- Efikasnost - iznad 80%!



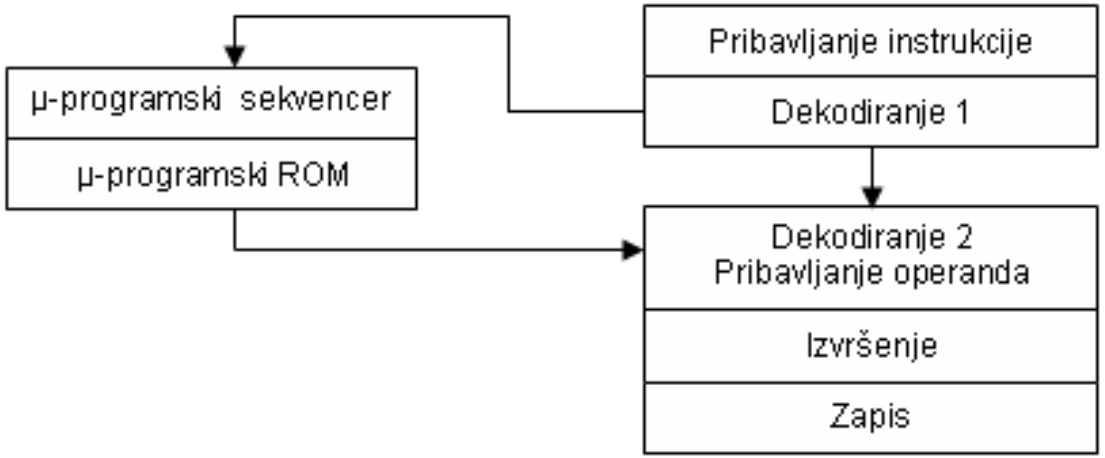


Koncept procesora visokih performansi

- Savremene arhitekture
  - Kombinovana CISC-RISC arhitektura,
  - Super-protočni procesori (*super-pipelined*)
  - Višestepeni procesori (*superscalar*)
  - Procesori sa vrlo dugom instrukcijskom reči (*VLIW*)
- Dodatne tehnike ubrzanja procesora
  - izvršavaju deo operacija koje nisu potrebne, tj. koje se odbacuju, uz utrošak resursa, ali bez gubitka vremena
  - Uslovno izvršenje (*predication*)
  - Sumnjivo-spekulativno punjenje podataka (*speculative loading*)
  - Nagađanje podatka (*data speculation*)

Kombinovana CISC-RISC arhitektura

- Jednostavne instrukcije direktno se usmeravaju na izvršenje
- Kompleksnije se izvršavaju pomoću mikroprogramske UJ
- U stonim računarima od procesora intel x486



Super-protočni procesori

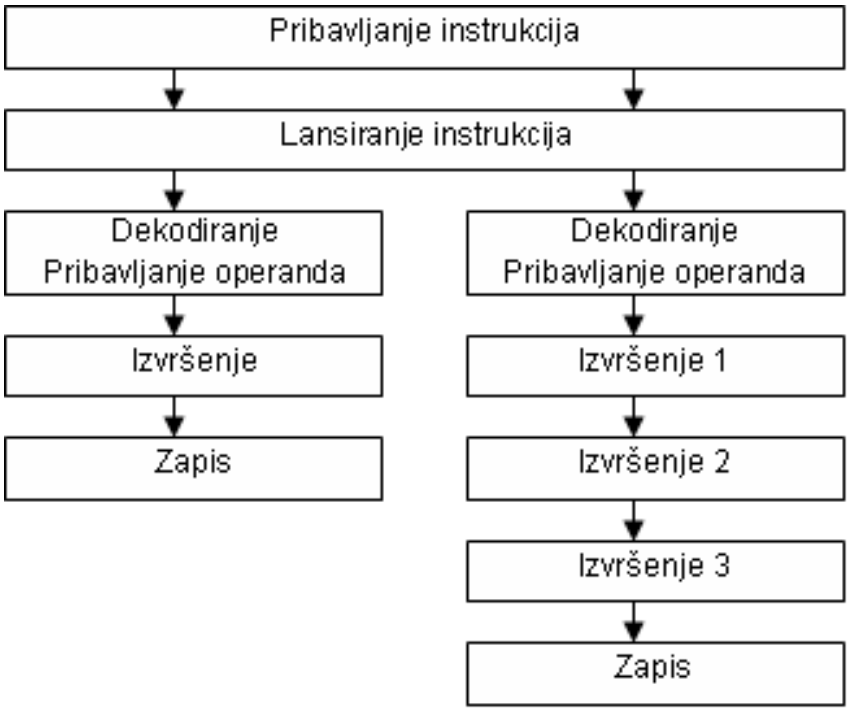
Super-pipelined

- Ubrzanje takta povećanjem broja protočnih segmenata
- Broj segmenata > 5 (arbitrarno)
- Rukovanje rizicima je kritično

Intel procesori	BrS
P5 (Pentium)	5
P6 (Pentium 3)	10
P6 (Pentium Pro)	14
P68 (NetBurst)	20-31
Core II (i3,i5,i7)	14

ARM procesori	BrS
ARM <7	3
ARM 8-9	5
ARM 11	8
Cortex A7	8-10
Cortex A8	13
Cortex A15	15-25

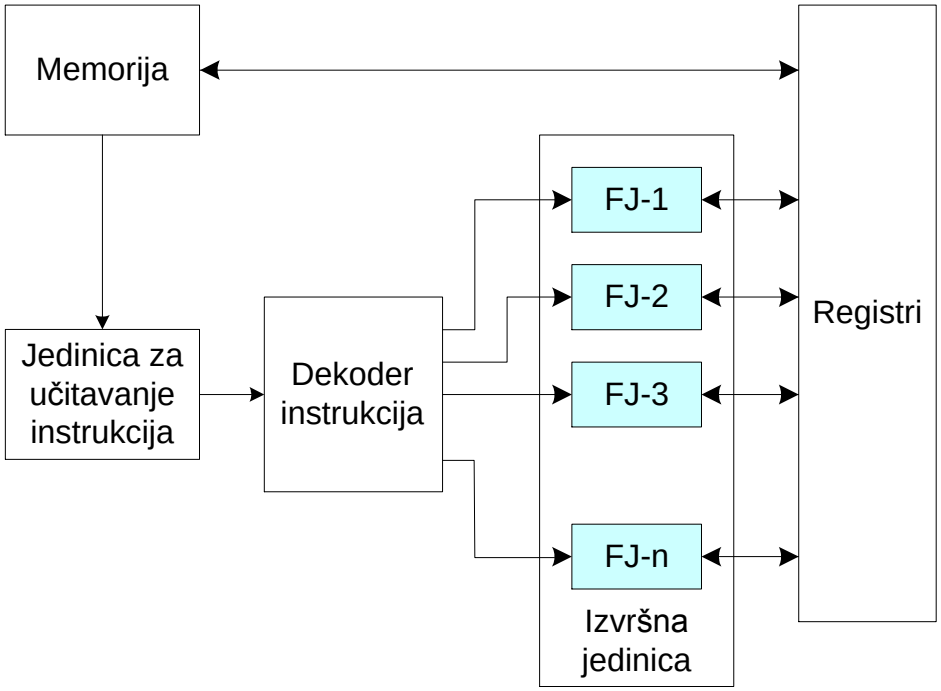
Višestepeni (superskalarni) procesori

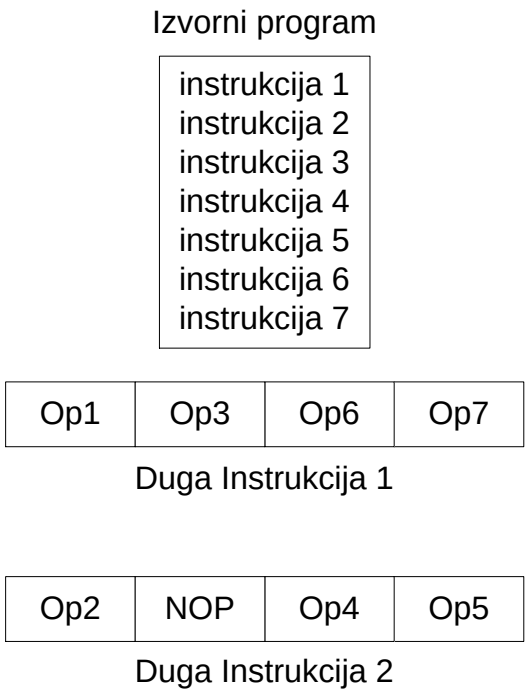


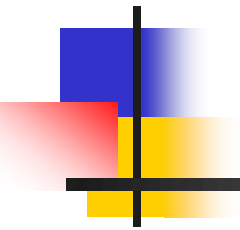
- Pokretanje više instrukcija u jednom taktu istovremeno
- Umnožene izvršne jedinice
- Kompleksne tehnike dinamičkog raspoređivanja instrukcija
- Prekoredno izvršenje
- Prozor izvršenja – broj instrukcija koji se u realnom vremenu analizira tražeći potencijalni paralelizam između njih

Procesori sa vrlo dugom instrukcijskom reči

- *VLIW - Very Long Instruction Word*
- Kompajler ispituje paralelizam između susednih instrukcija i formira jednu "veliku instrukciju" koja se paralelno izvršava
- Intel I860, Itanium, DSP procesori danas





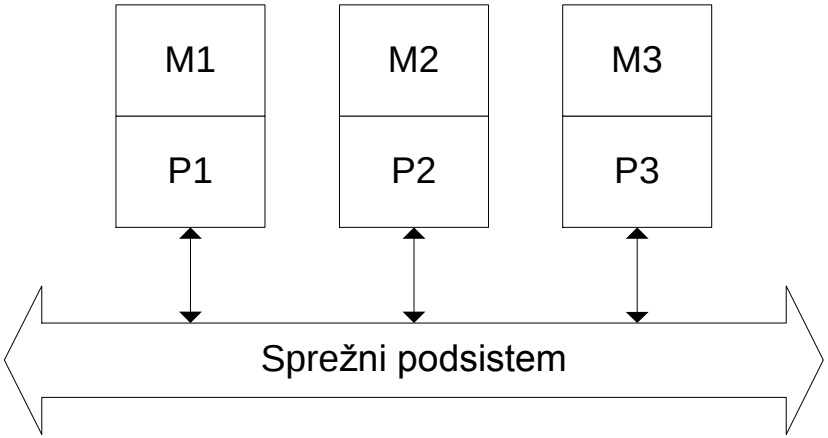


MULTIPROCESORSKI SISTEMI

Multiprocesorski sistemi (MPS)

- Umnožavanje i međusobno povezivanje raspoloživih računarskih komponenti, radi zajedničkog izvršenja posla
- Dva su osnovna cilja uvođenja multiprocesorskih sistema:
  - **Poboljšavanje sistemskih performansi**, odnosno ubrzanje obrade. Operativni sistem obezbeđuje pokretanje i sinhronizaciju rada MPS.
  - **Podizanje pouzdanosti i raspoloživosti** celog sistema, što opet ima dva aspekta:
    - *Rekonfiguracija* – postupak prilagođavanja sistema
    - *Udvajanje* - paralelan rad dva ili više računara nad istim podacima (*fault - tolerant* konfiguracije).

Kategorije višeprocesorskih sistema



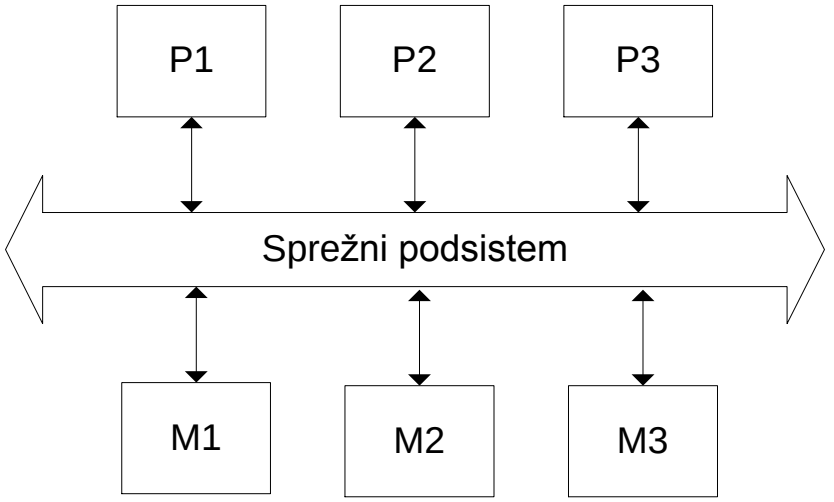
Slabo povezani MPS

PE = procesor + memorija + kom.procesor

Decentralizovan OS

razmena poruka, nema deljene memorije

DISTRIBUIRANI RS



Čvrsto povezani MPS

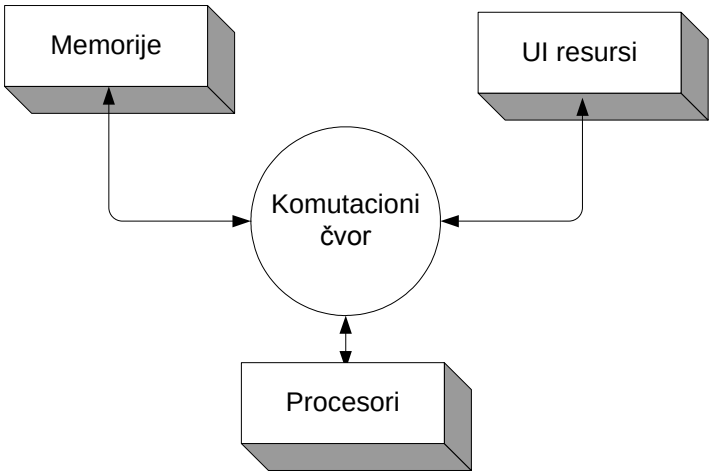
PE = procesor, memorija ili kom.procesor

Centralizovan OS

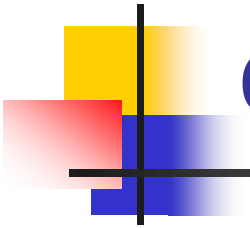
memorija i drugi resursi se dele

MULTIPROCESORSKI RS

Definicija P.H. Enslow-a, 1978.



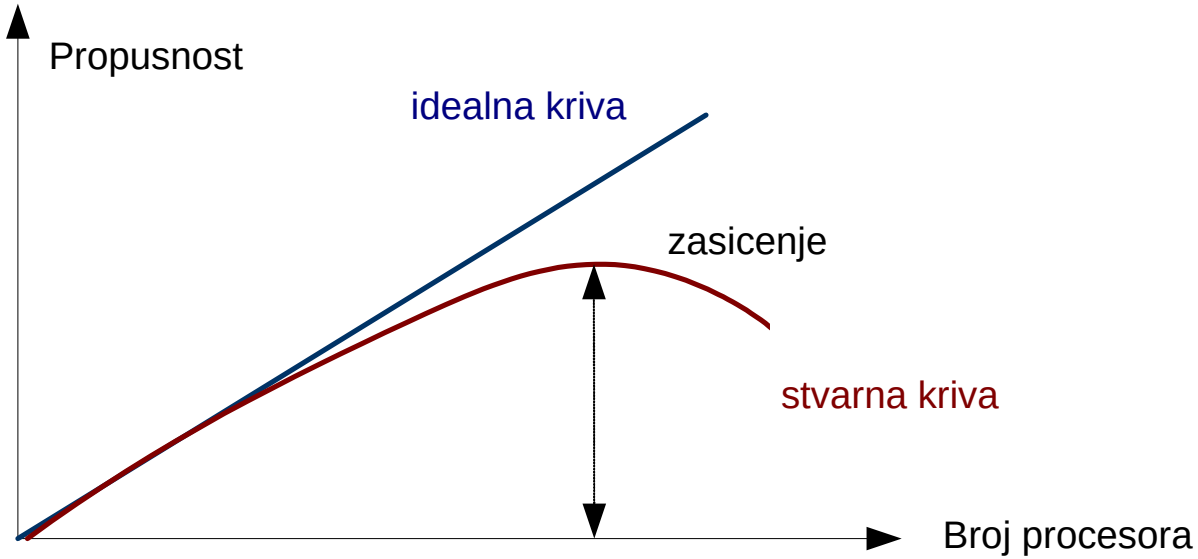
- Multiprocesorski sistem ima sledeće karakteristike:
  - MPS ima 2 ili više procesora,
  - Svi procesori imaju pristup zajedničkoj memoriji,
  - Svi procesori imaju pristup U/I resursima,
  - Celim sistemom upravlja jedan OS,
  - Postoji saradnja između procesora na nivou fizičke arhitekture i programske podrške.



Osnovni zahtevi MPS

- Povećanje propusne moći (sledeći slajd)
- Povećanje pouzdanosti i raspoloživosti
  - *Pouzdanost* - verovatnoća da će se traženi program izvršiti u predviđenom vremenskom intervalu.
  - *Raspoloživost* - vreme u kojem je MPS raspoloživ korisniku (upotrebljiv)
- Pogodnost promene konfiguracije
- Ekonomičnost realizacije
  - Deljenjem memorijskih i U/I modula povećava se njihovo iskorišćenje.
  - U cilju povećanja performansi, pogodnije je koristiti više jednostavnih. (i jeftinijih) procesora, uz uslov da sprežni podsistem, svojom cenom, ne poništi efekte.

Zavisnost propusnosti multiprocesora od broja čvorova



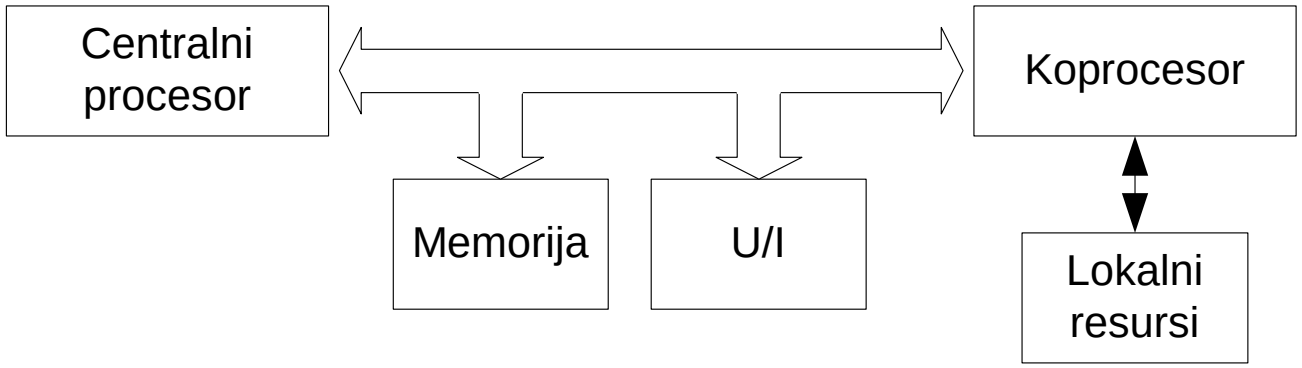
- Zasićenje nastaje zbog sve veće međuprocesorske komunikacije i sve većeg broja režijskih zadataka (sinhronizacija)

Najčešće organizacije multiprocesorskih sistema

- Koprocesori
- Multiprocesorski sistemi zasnovani na zajedničkoj magistrali na bazi vremenskog multipleksa
- Multiprocesorski sistemi sa "krosbar" spregom između procesora
- Multiprocesorski sistemi bazirani na memoriji sa više pristupa

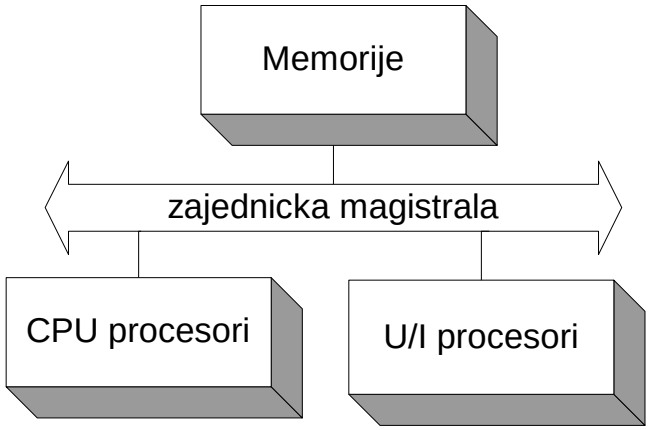
Koprocesori

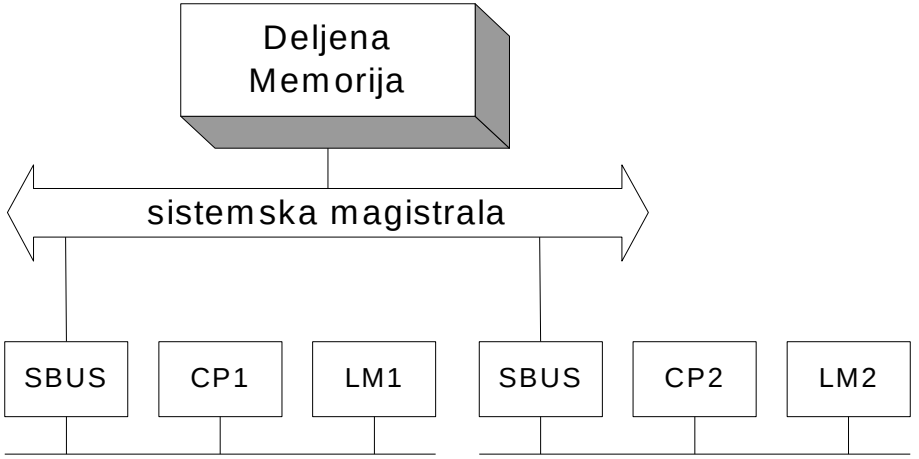
- Specijalizovani procesor, rasterećenje vodećeg
- Deli magistralu i zajedničke resurse
- Autonoman i polu-autonoman režim rada
  - sa i bez korišćenja lokalnih resursa



MPS sa zajedničkom magistralom

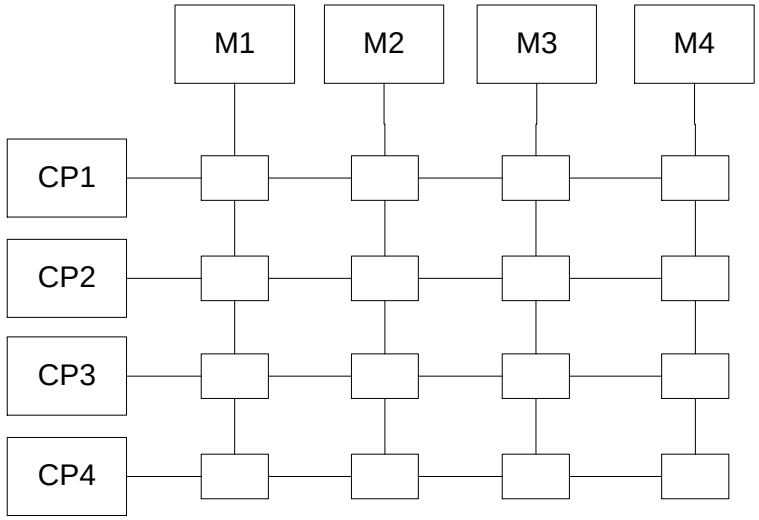
- Zajednička magistrala, na bazi vremenskog multipleksa
- Jednostavno, fleksibilno, relativno neefikasno rešenje
- Magistrala je usko grlo, kašnjenje zbog jednog prenosnog puta
- Kašnjenje se pokušava redukovati udvajanjem magistrale



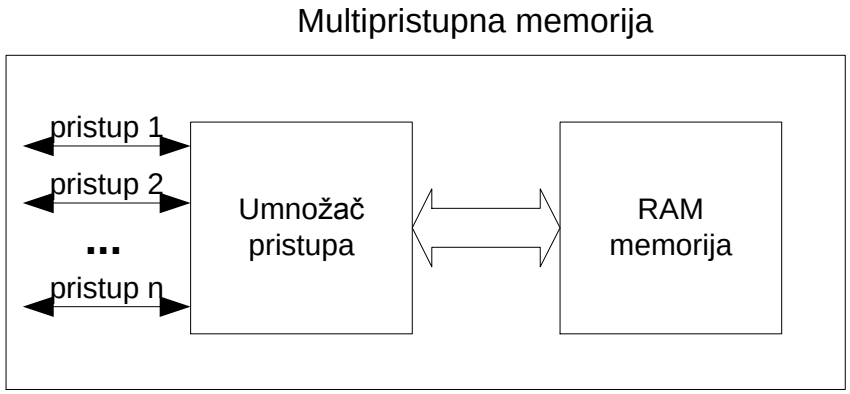


MPS sa krosbar spregom

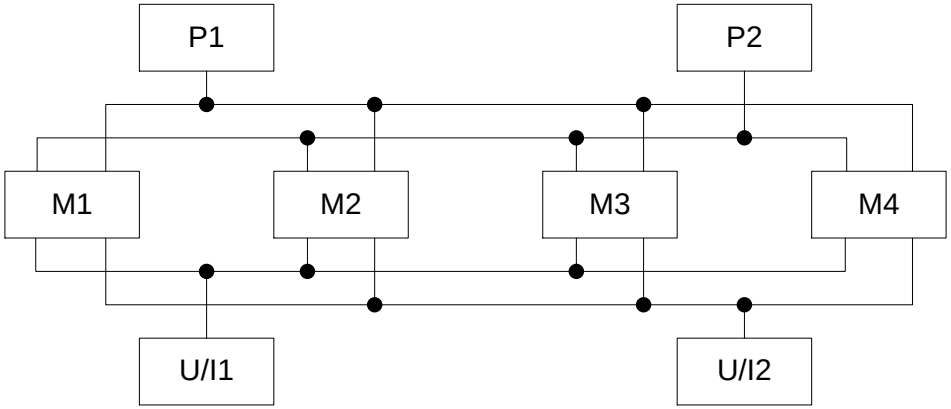
- Krosbar matrica - organizacija sa više magistrala
- Čvorni sprežni element, sa 3 stanja
  - 0: prenos poruke sa horizontalne na vertikalnu magistralu
  - 1: prenos na magistralu istog tipa (propuštanje)
  - 2: nema prenosa kroz sprežni element (otvoreno)
- Prednost – simultane veze
- Mana - kompleksnost rešenja

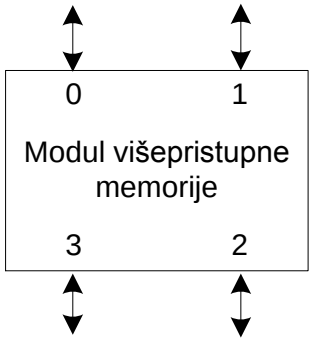


MPS sa višepristupnom memorijom

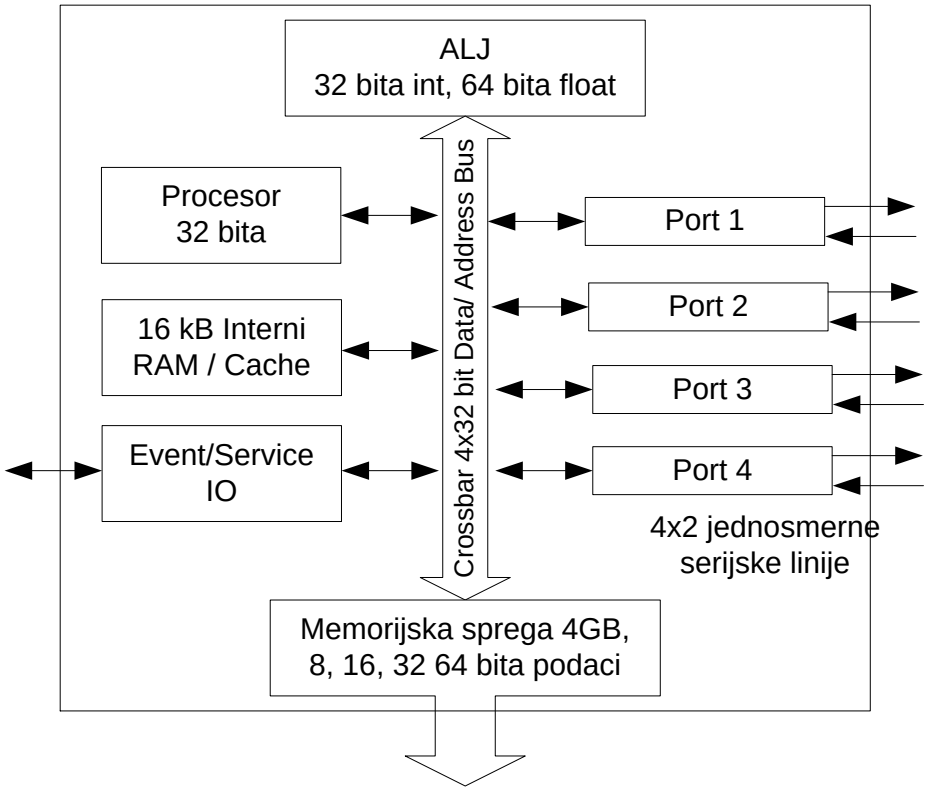


- Umnožač pristupa rešava konflikte
- Brzina prenosa je ograničena trajanjem memorijskog ciklusa
- Dvopristupne memorije (DPM)



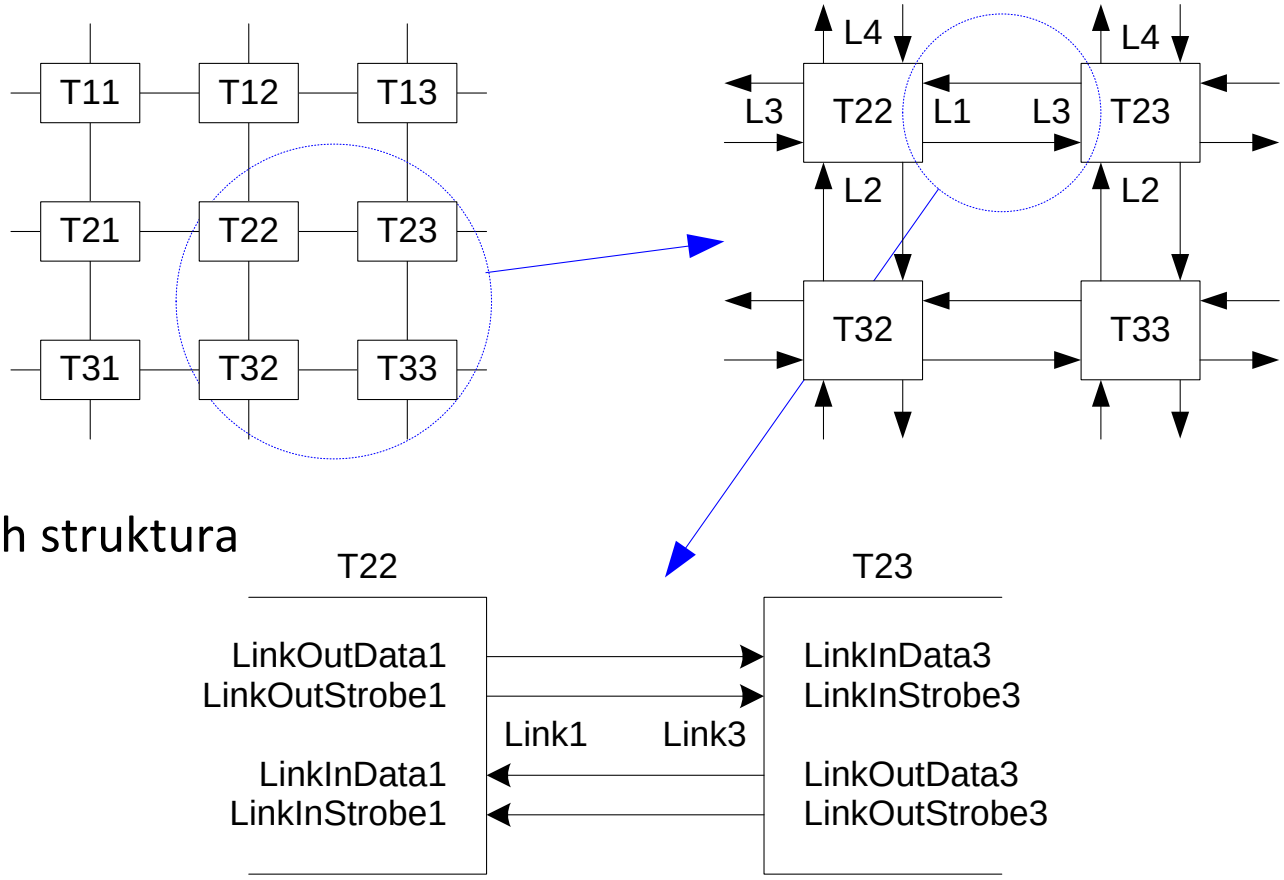


Transpjuteri



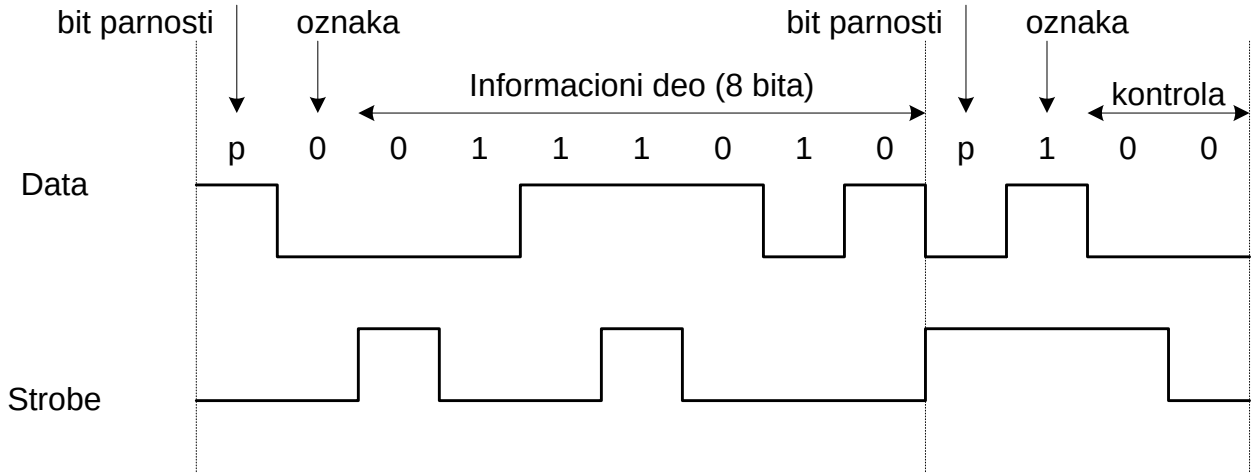
- Kraj 1980-tih, sjajan koncept, propao zbog razvoja tehnologije i jeftinih protočnih procesora
- Specijalizovana komponenta
- Superskalarna, protočna (7-seg)
- 20MHz, do 120MIPS i 70FLOPS
- Lokalna memorija 16kB (*cache*)
- Spoljna memorija 4GB, 64 bita
- 4 specijalizovana prolaza (porta), brzine do 100 Mbit/s
- OCCAM – operativni sistem

Međusobno povezivanje transpjutera



- Formiranje mrežnih struktura
- Data + Strobe

Razmena podataka između transpjutera



- Podaci 10 bita, kontrola 4 bita
- Bit pariteta (p), tip (0/1-*Data*/*Control*), informacioni deo
- 4 tipa upravljačkih znakova: kontrola prenosa, kraj paketa, kraj poruke i prekid sekvence
- Poruka se prenosi se kao niz znakova, čiji se prijem potvrđuje
- Komunikacioni takt - *Strobe* signal se promeni svaki put kada se signal podataka (*Data*) ne menja