

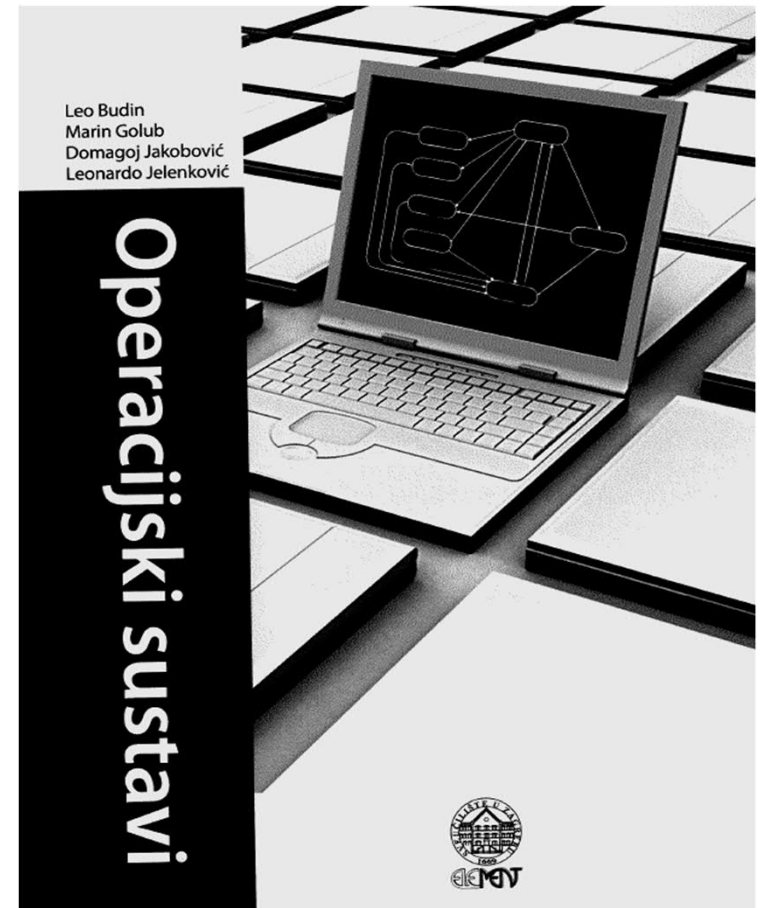
Operacijski sustavi

Operativni sustavi

Sveučilište Jurja Dobrile u Puli

Studijski programi:

- Informatika,
- Poslovna informatika



Operacijski sustavi

Operativni sustavi

Literatura:

- Budin, Golub, Jakobović, Jelenković: **Operacijski sustavi**, Element
- Slideovi kolegija
- Ostala dokumentacija sa e-učenja

Organizacija kolegija:

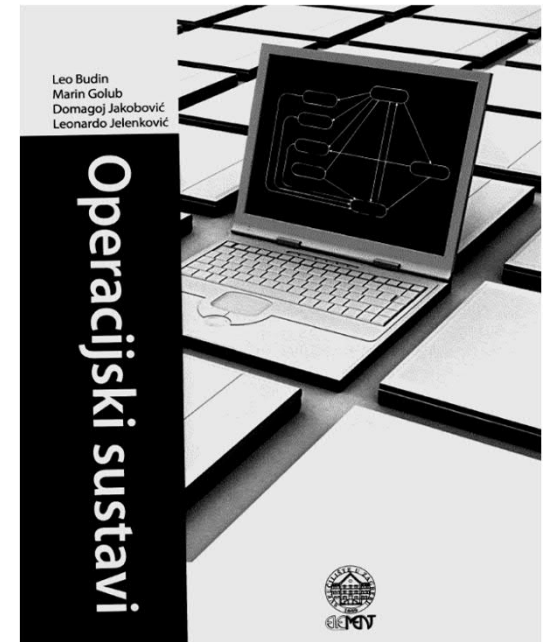
- Predavanje
- Obavezne vježbe u laboratoriju
- Neobavezne vježbe u laboratoriju

Provjera znanja:

- aktivnost & izrada zadatka
- pismeni ispit (e-učenje)
- usmeni ipit

Komunikacija:

- e-učenje – obavijesti i teme
- E-mail: wstemb@unipu.hr

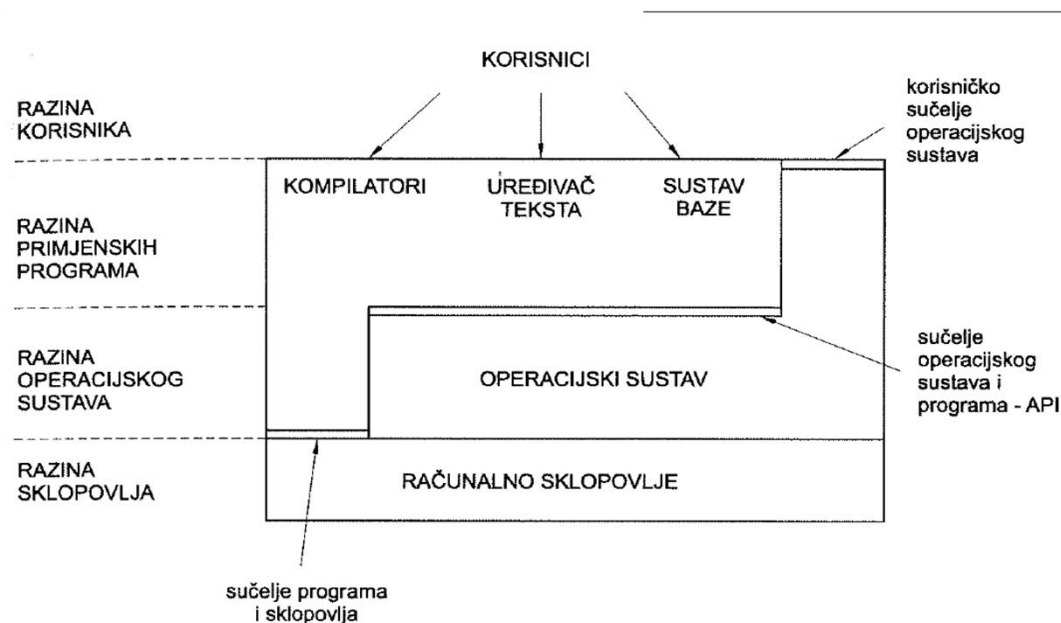


Operacijski sustavi: Uvod

Odgovor na pitanja o operacijskim sustavima:

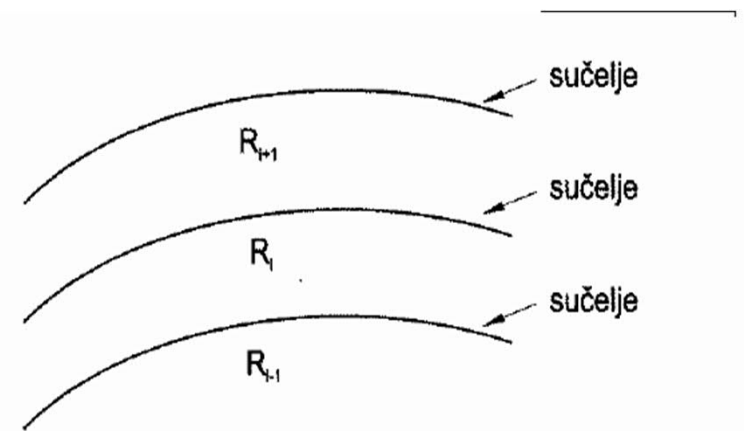
- Što su?
- Čemu služe?
- Kako rade?

Generička obrada pojedinih funkcionalnosti, kolegij se izvan primjera i zadataka ne bavi implementacijom u konkretne OS



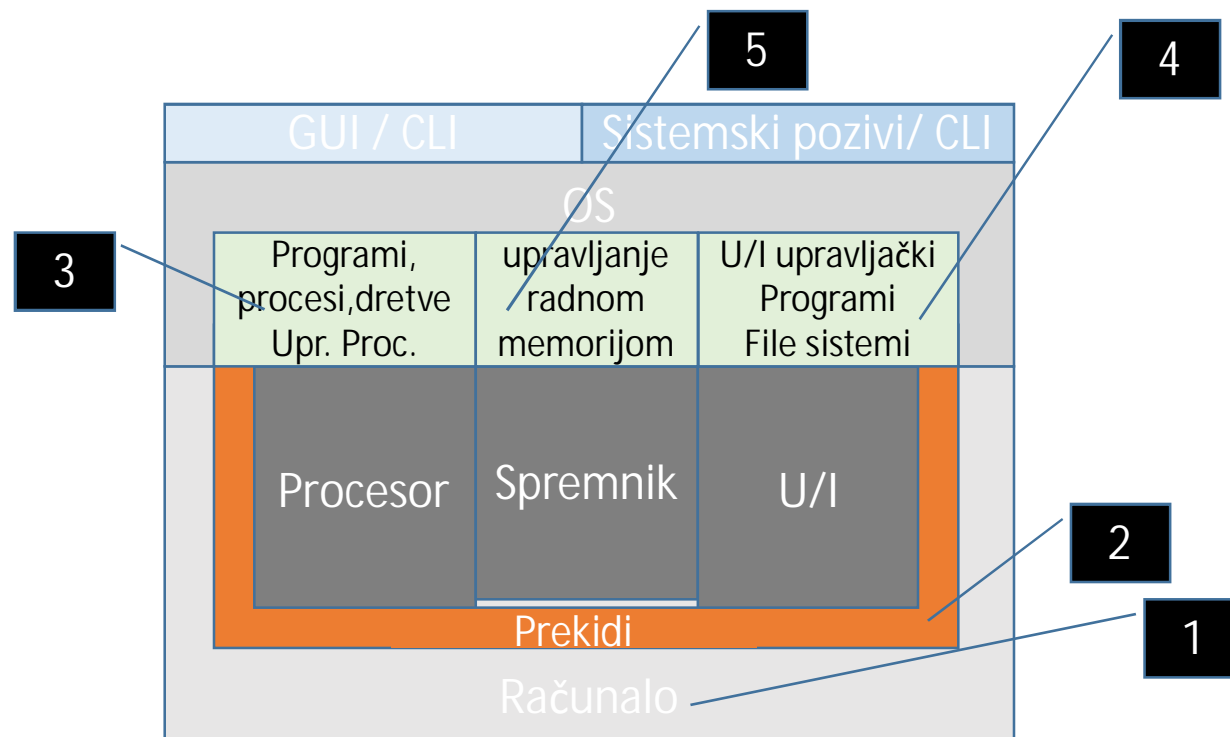
Operacijski sustavi: Uvod

- Potpora korisniku i aplikativnim programima
- Omogućuje provođenje radnih zahvata na računalu
- Jednostavno korištenje računala
- Efikasno korištenje računala
- Strategija „Podijeli, pa vladaj” -> slojeviti pristup:
 - sustav se izgrađuje po razinama
 - svaka se razina sastoji od objekata i operacija nad tim objektima
 - objekti i operacije neke razine izgrađuju se samo pomoću objekata i operacija prve neposredne niže razine
 - detalji ostvarenja objekata i operacija pojedine razine su skriveni



Operacijski sustavi: Uvod

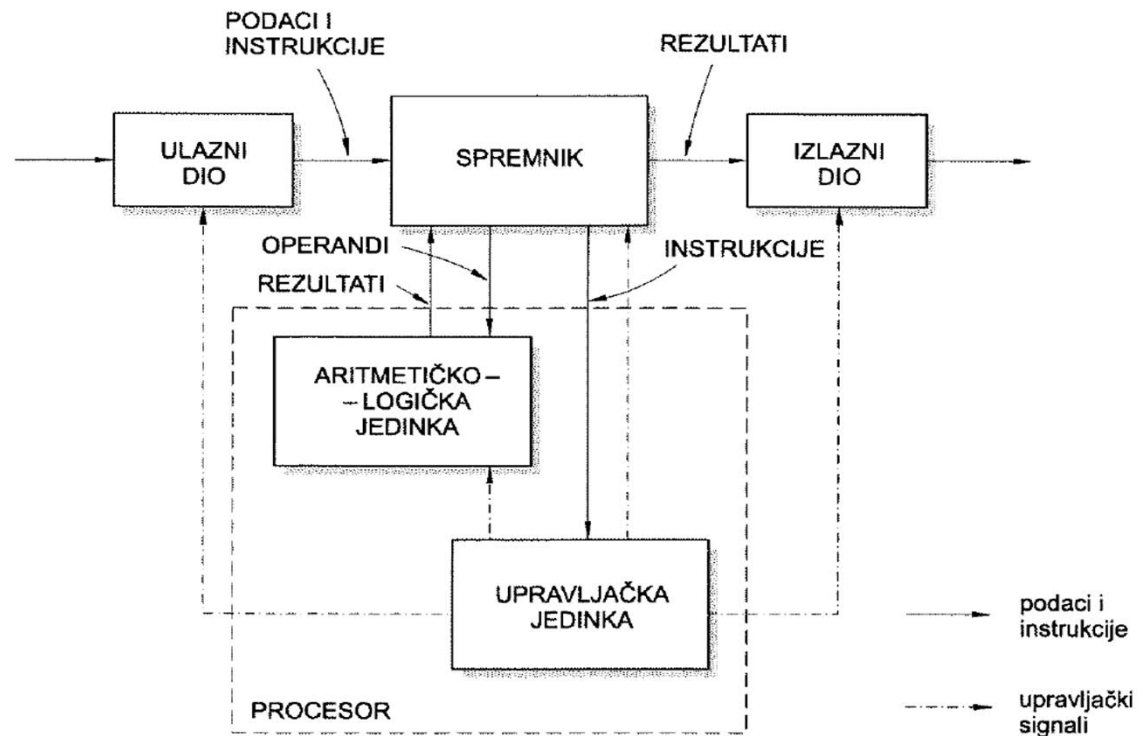
- Jednostavno računalo – izvođenje strojnog jezika
- Prekidni mehanizmi
- Programi, procesi , dretve, jezgra OS-a
- Upravljanje spremnikom
- file sistemi



Operacijski sustavi: Računalo

Von Neumannov model (1945):

- Ulazni dio
- Izlazni dio
- Spremnik
- ALJ
- UJ



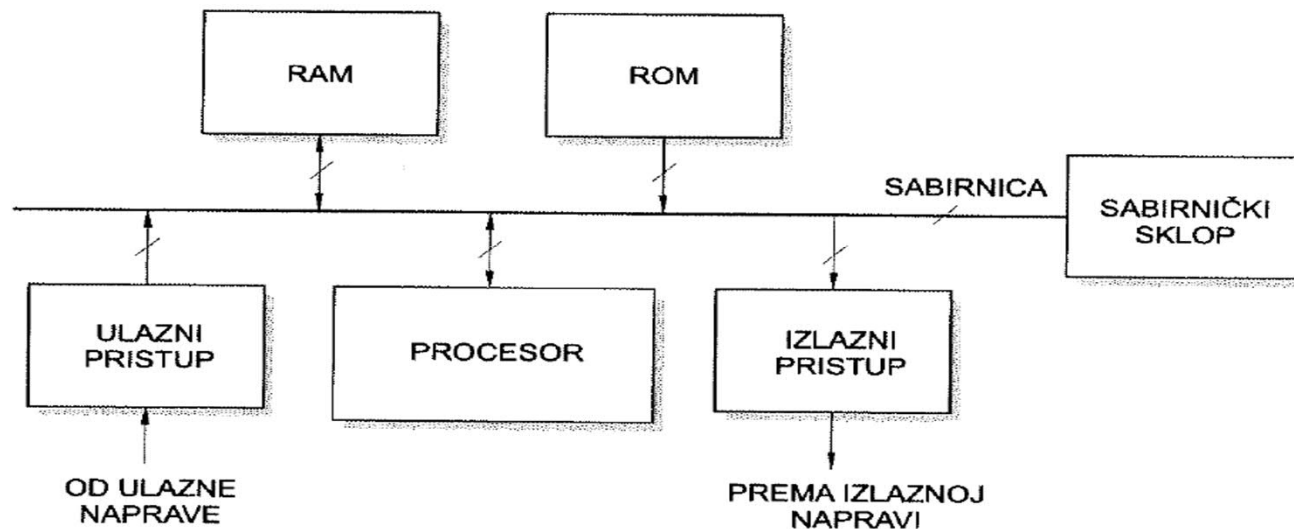
Karakteristike :

- programi i podaci koriste jedinstvenu glavnu memoriju
- glavnoj se memoriji pristupa kao jednodimenzionalnom nizu
- značenje (semantika) ili način primjene podataka nije spremljeno s podacima

Operacijski sustavi: Računalo

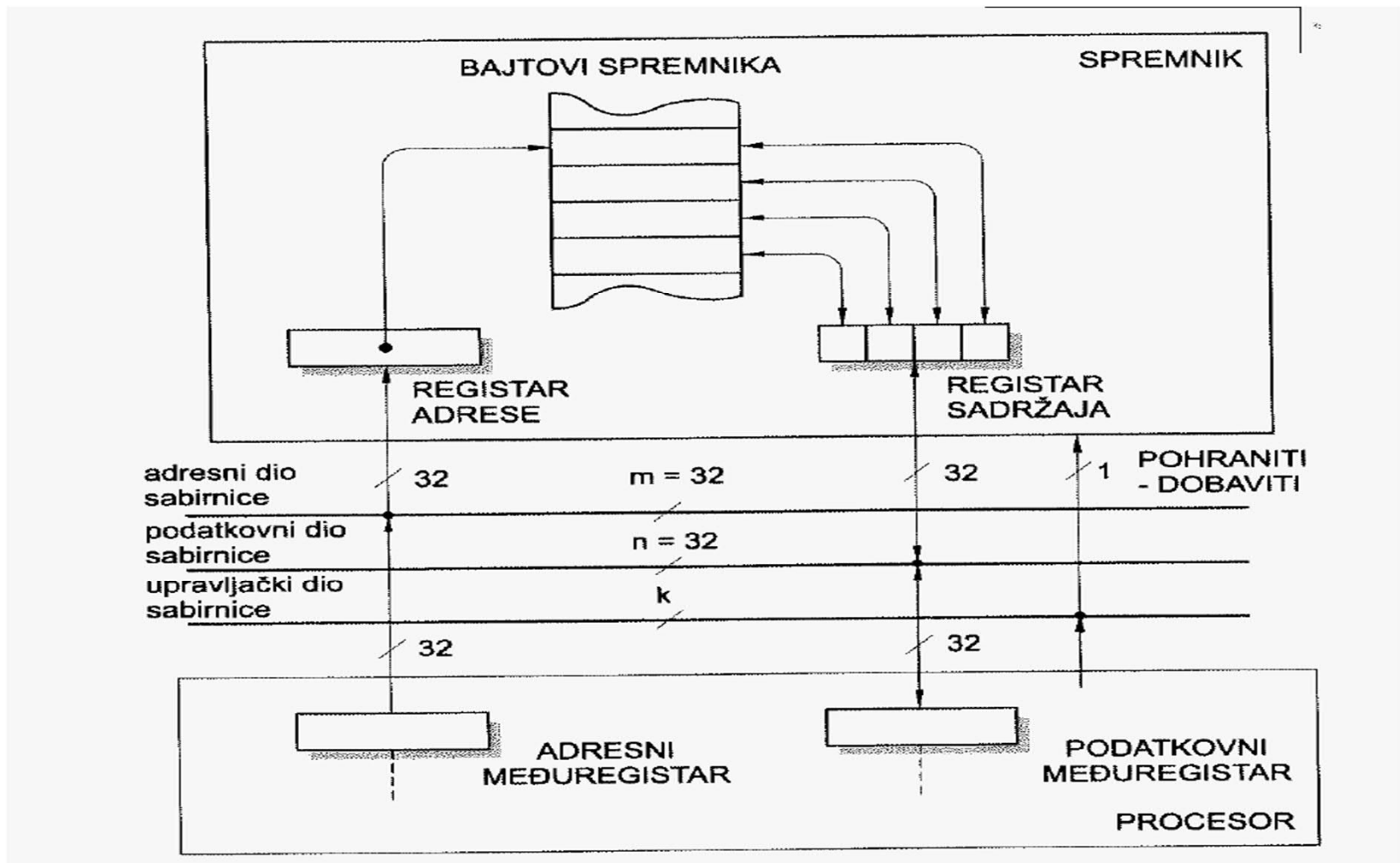
Sabirnička građa računala

- Za prijenos svakog bita potreban je jedan vodič (paralelni spoj)
- Sabirnica: zajednički snop vodiča na koji su spojeni svi dijelovi računala
- time-share – dijeljenje vremena za uspostavu veza među dijelovima računala



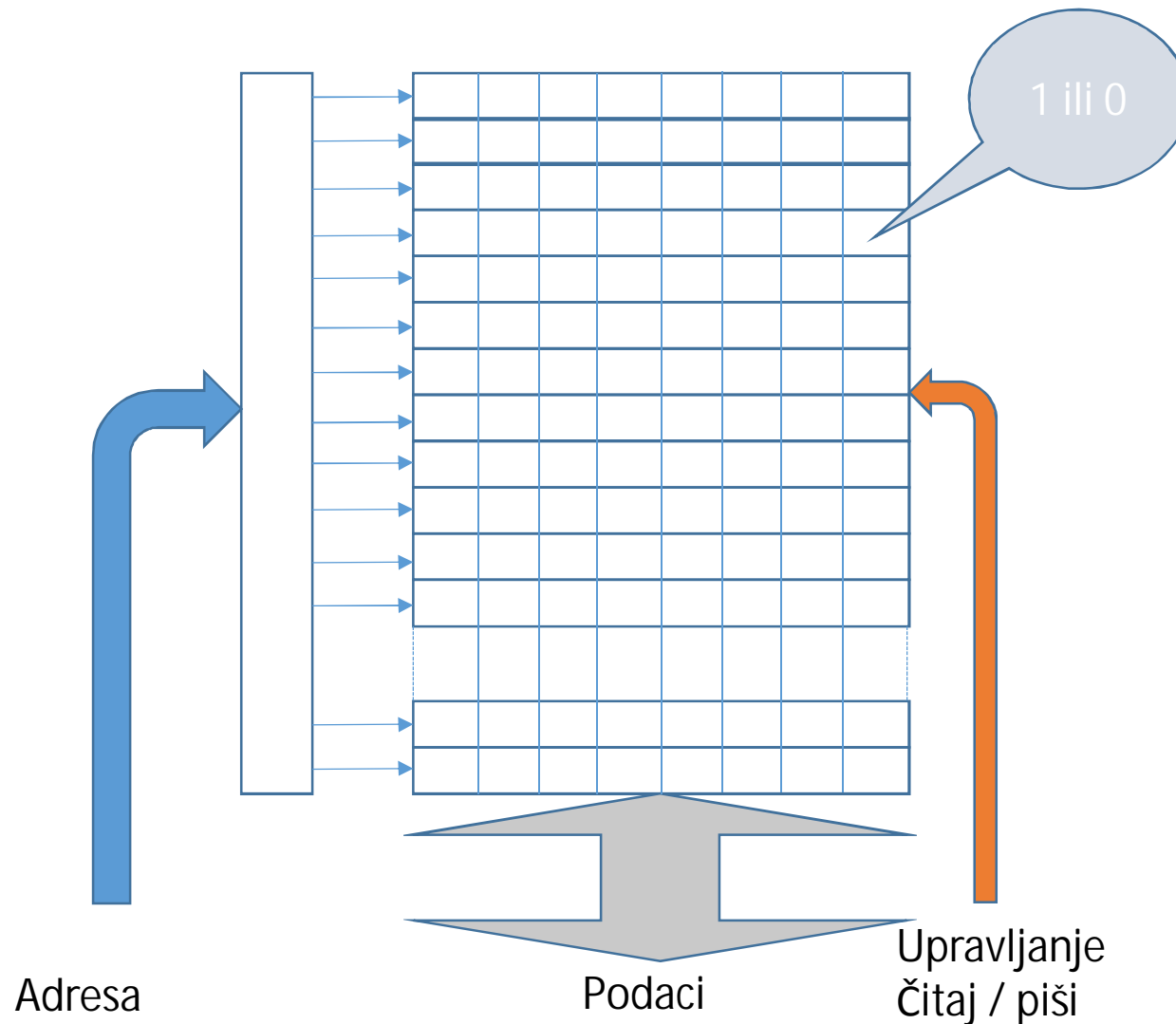
Operacijski sustavi: Računalo

Način povezivanja procesora i memorije na sabirnicu



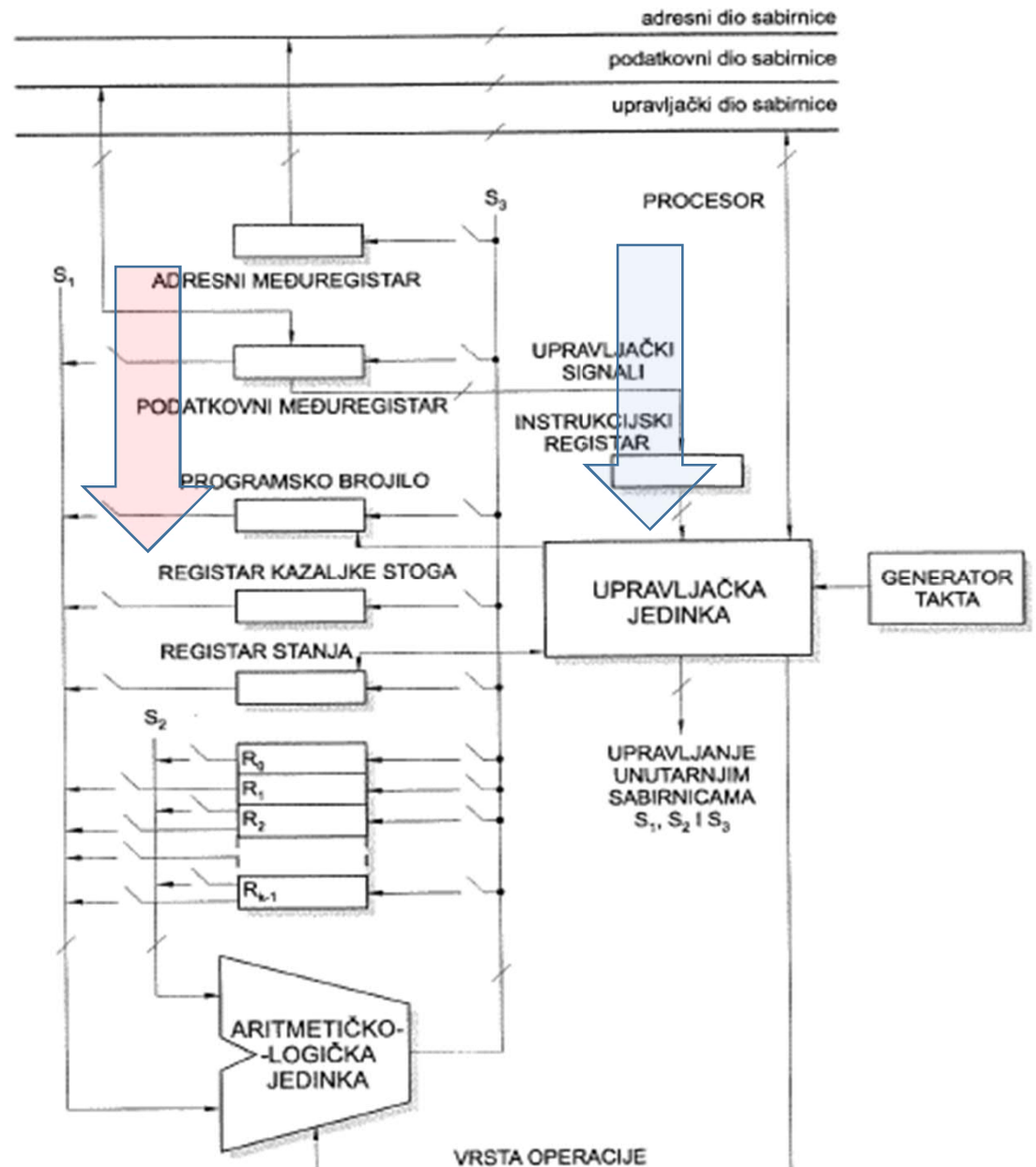
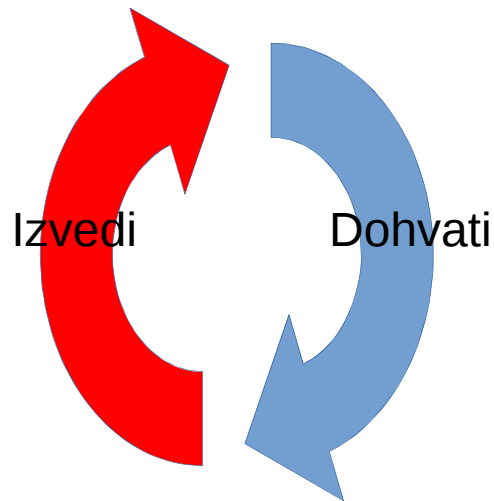
Operacijski sustavi: Računalo

Spremnik:




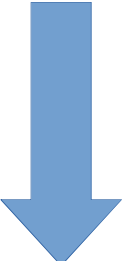
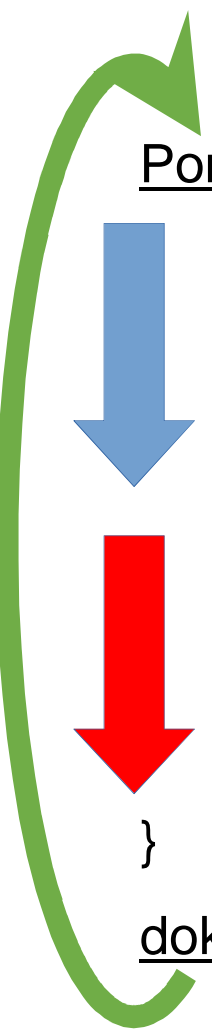
Operacijski sustavi: Računalo

- Struktura procesora



Operacijski sustavi: Računalo

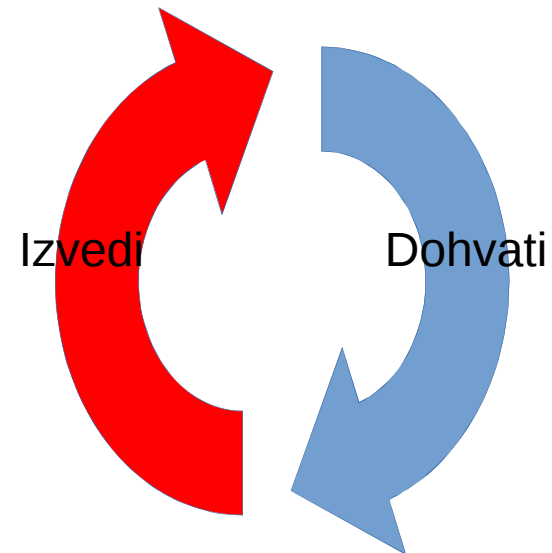
Ponavljati {



dohvatiti iz spremnika instrukciju na koju pokazuje programsko brojilo;
dekodirati instrukciju, odrediti operaciju koju treba izvesti;
povećati sadržaj programskog brojila tako da pokazuje na slijedeću instrukciju;
odrediti odakle dohvatiti operande i gdje pohraniti rezultat;
dovesti operande na ALJ, izvesti zadanu operaciju;
pohraniti rezultat u odredište;

}

dok je (procesor uključen);



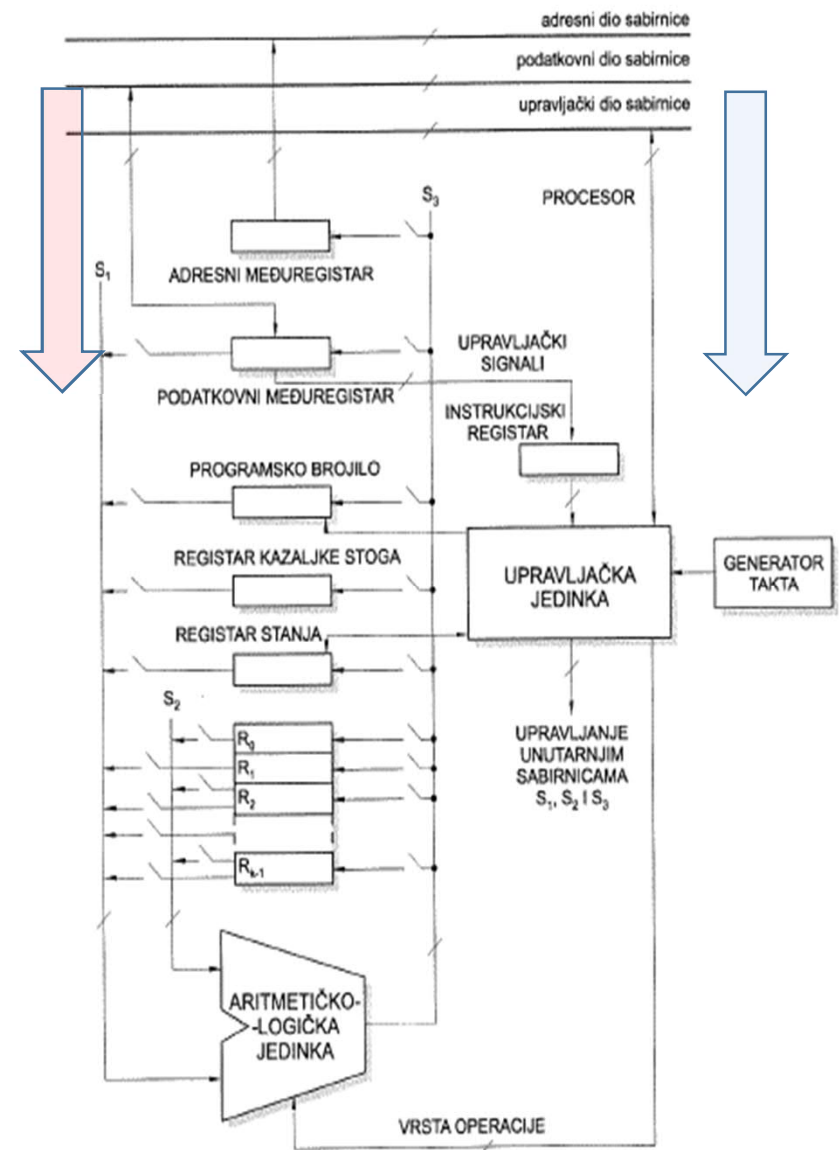
Operacijski sustavi: Računalo

Ponavljati {

dohvatiti iz spremnika instrukciju na koju pokazuje PB;
dekodirati instrukciju, odrediti operaciju koju treba izvesti;
povećati sadržaj PB tako da pokazuje na slijedeću instrukciju;
odrediti odakle dohvatiti operande i gdje pohraniti rezultat;
dovesti operande na ALJ, izvesti zadanu operaciju;
pohraniti rezultat u odredište;

}

dok je (procesor uključen);



Operacijski sustavi: Računalo

Instrukcijski skup procesora (Instruction Set)

- RISC – Reduced Instructions Set Processor
- CISC – Complex Instructions Set Processor
- instrukcije za premještanje sadržaja između memorije i registara (move / load / store)
- instrukcije za obavljanje A / L operacija
- instrukcije za programske skokove ili grananja
- instrukcije za posebna upravljačka djelovanja

Operacijski kod

Operand(i)

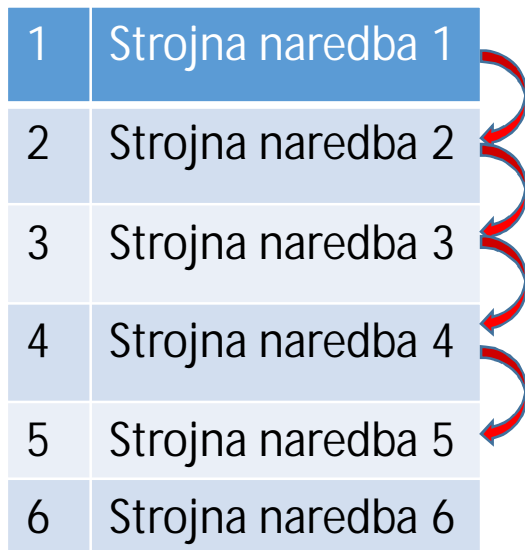
Što?

Nad čime?

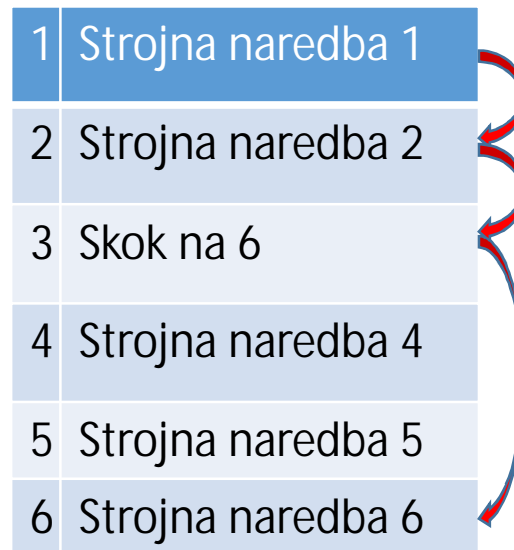
Operacijski sustavi: Računalo

Instrukcijski skup procesora (Instruction Set)

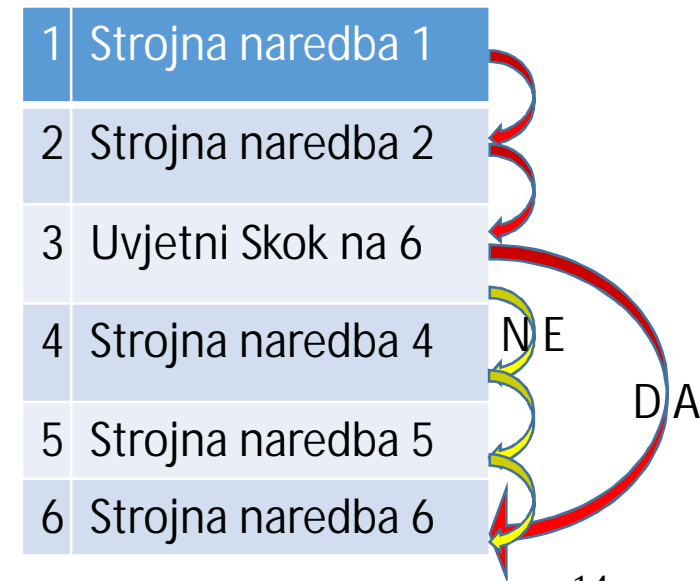
- instrukcije za premještanje sadržaja između memorije i registara (move / load / store)
- instrukcije za obavljanje A / L operacija
- instrukcije za programske skokove ili grananja
- instrukcije za posebna upravljačka djelovanja



Sekvenca



Skok



Uvjetni skok

Operacijski sustavi: Računalo

Instrukcija bezuvjetnog skoka



Ponavljati {



dohvatiti iz spremnika instrukciju na koju pokazuje programsko brojilo;

dekodirati instrukciju, odrediti operaciju koju treba izvesti;

povećati sadržaj programskog brojila da pokazuje na slijedeću instrukciju;

ako je (dekodirana instrukcija skoka)

iz pomaknuća zapisanog u instrukciji izračunati adresu i pohraniti tu adresu u PB;

inače



obaviti instrukciju na način određenim dekodiranim operacijskim kodom;

}

dok je (procesor uključen);



Operacijski sustavi: Računalo

Instrukcija uvjetnog skoka



Ponavljati {



dohvatiti iz spremnika instrukciju na koju pokazuje programsko brojilo;

dekodirati instrukciju, odrediti operaciju koju treba izvesti;

povećati sadržaj programskog brojila da pokazuje na slijedeću instrukciju;

ako je (dekodirana instrukcija skoka I uvjet ispunjen)

iz pomaknuća zapisanog u instrukciji izračunati adresu i pohraniti tu adresu u PB;

inače



obaviti instrukciju na način određenim dekodiranim operacijskim kodom;

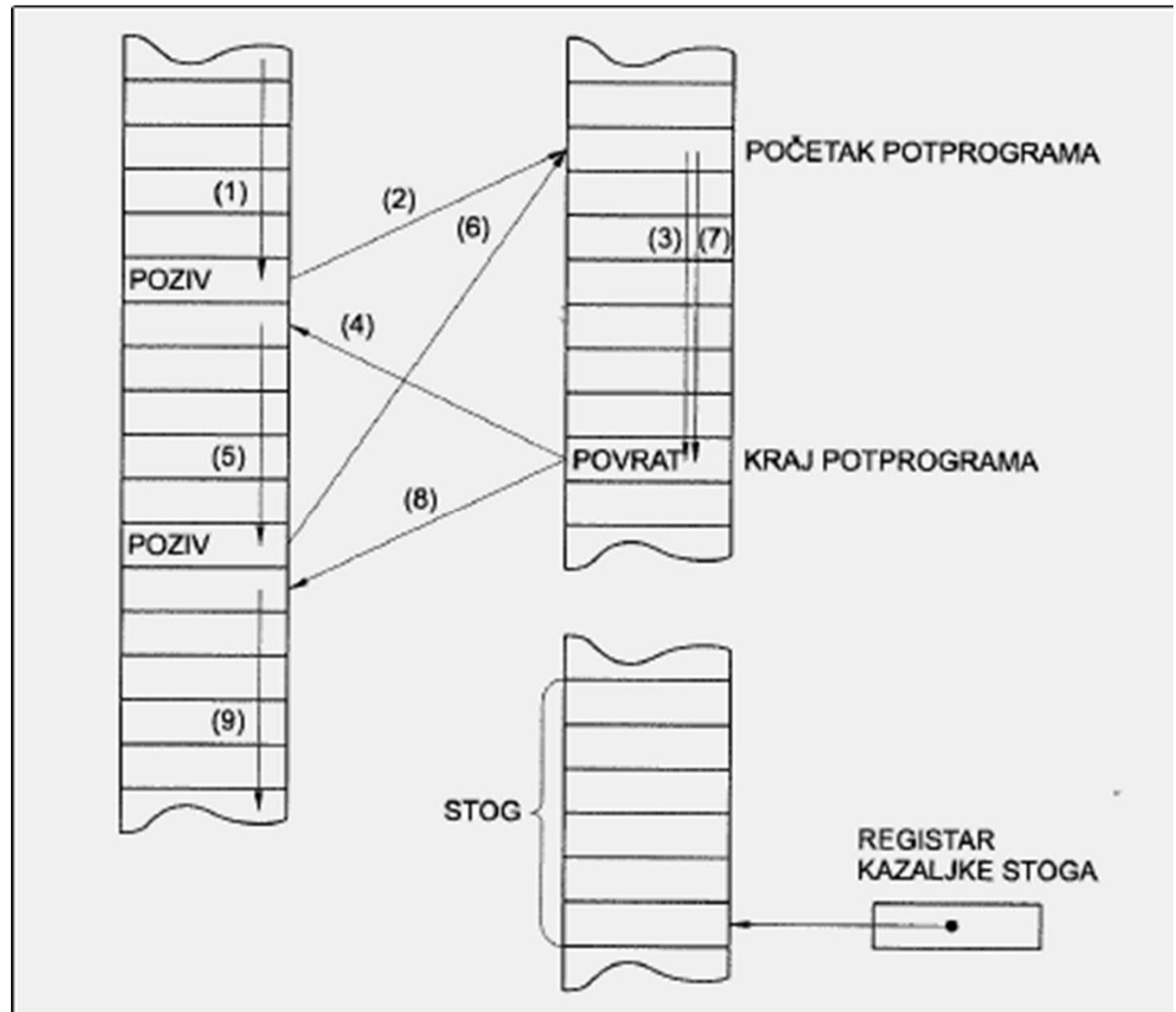
}

dok je (procesor uključen);



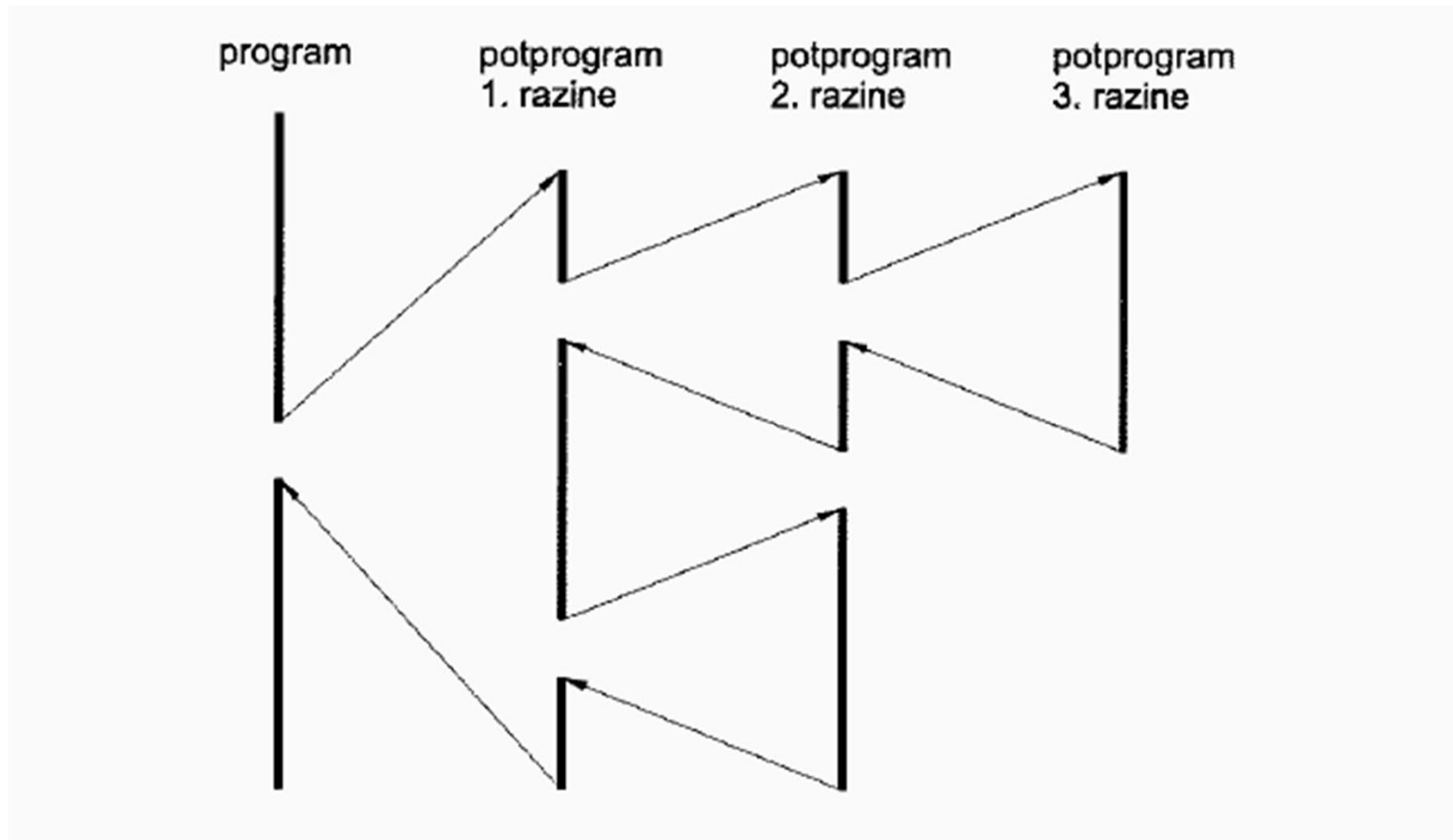
Operacijski sustavi: Računalo

Poziv potprograma



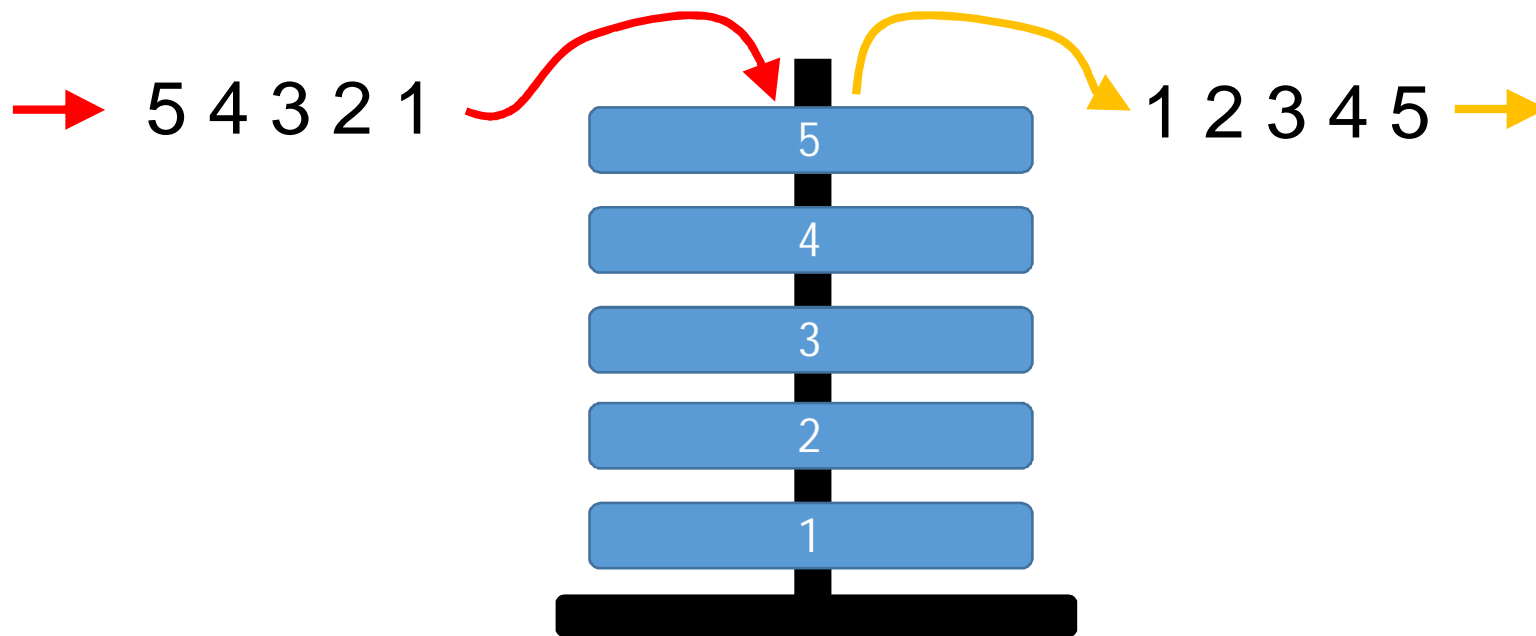
Operacijski sustavi: Računalo

Poziv potprograma iz drugog potprograma



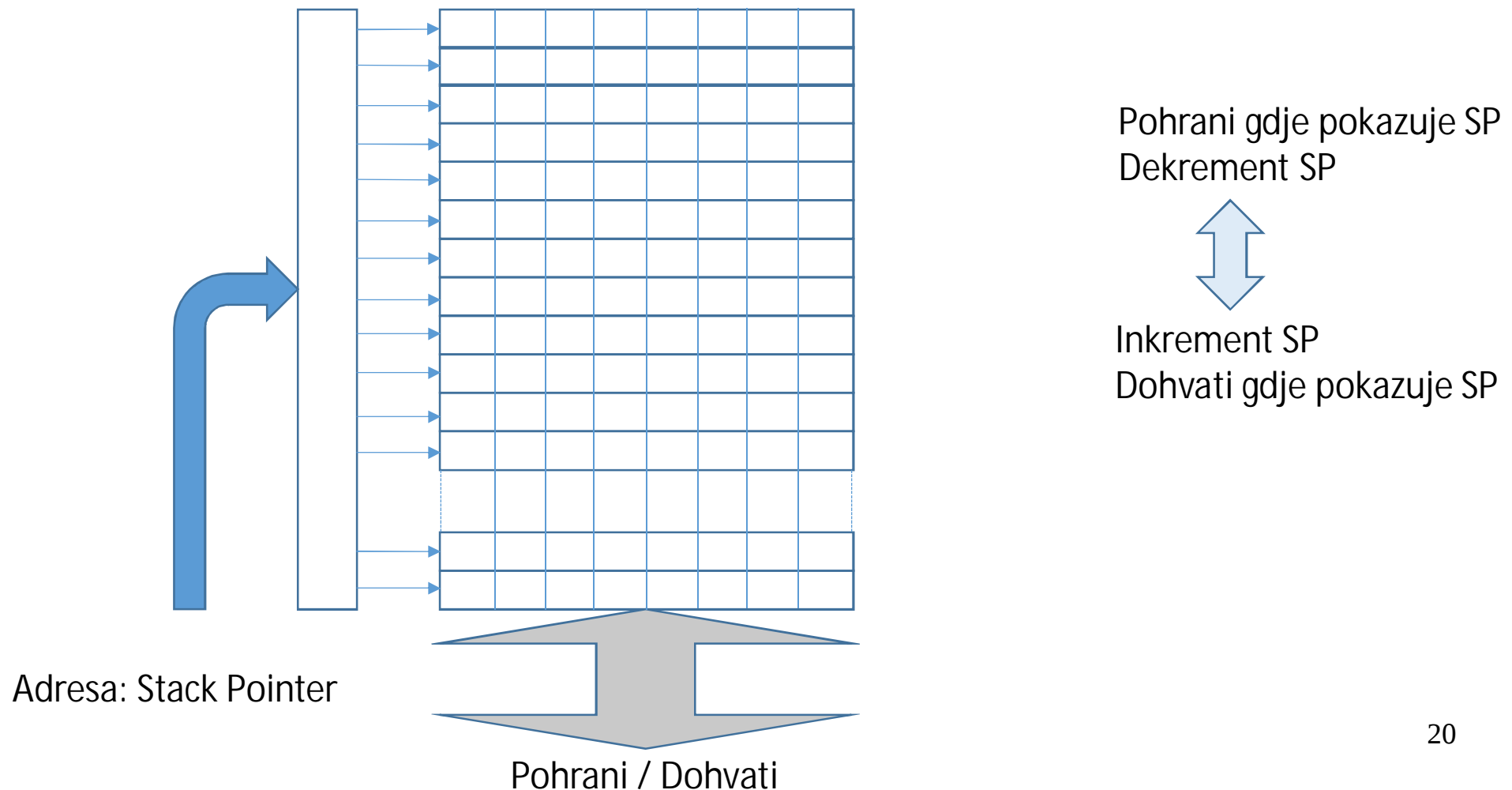
Operacijski sustavi: Računalo

Struktura stoga (Stack) – LIFO struktura



Operacijski sustavi: Računalo

Struktura stoga (Stack) – LIFO struktura





Operacijski sustavi: Računalo

Poziv potprograma



Ponavljati {



dohvatiti iz spremnika instrukciju na koju pokazuje programsko brojilo;
dekodirati instrukciju, odrediti operaciju koju treba izvesti;
povećati sadržaj programskog brojila da pokazuje na slijedeću instrukciju;
ako je (dekodirana instrukcija poziva potprograma)
 pohraniti sadržaj programskog brojila na stog;
 smanjiti sadržaj pokazivača stoga tako da pokazuje na slijedeće prazno mjesto;
 iz adresnog dijela instrukcije odrediti adresu početka potprograma
 pohraniti tu adresu u programsko brojilo;

inače

 obaviti instrukciju na način određenim dekodiranim operacijskim kodom;

dok je (procesor uključen);



Operacijski sustavi: Računalo

Povratak iz potprograma



Ponavljati {



dohvatiti iz spremnika instrukciju na koju pokazuje programsko brojilo;

dekodirati instrukciju, odrediti operaciju koju treba izvesti;

povećati sadržaj programskog brojila da pokazuje na slijedeću instrukciju;

ako je (dekodirana instrukcija povratka iz potprograma)

povećati sadržaj pokazivača stoga SP;

Premjestiti sadržaj sa vrha stoga na koji pokazuje SP u programsko brojilo;

inače

obaviti instrukciju na način određenim dekodiranim operacijskim kodom;



dok je (procesor uključen);



Operacijski sustavi: Računalo

Prenos podataka u potprogram:

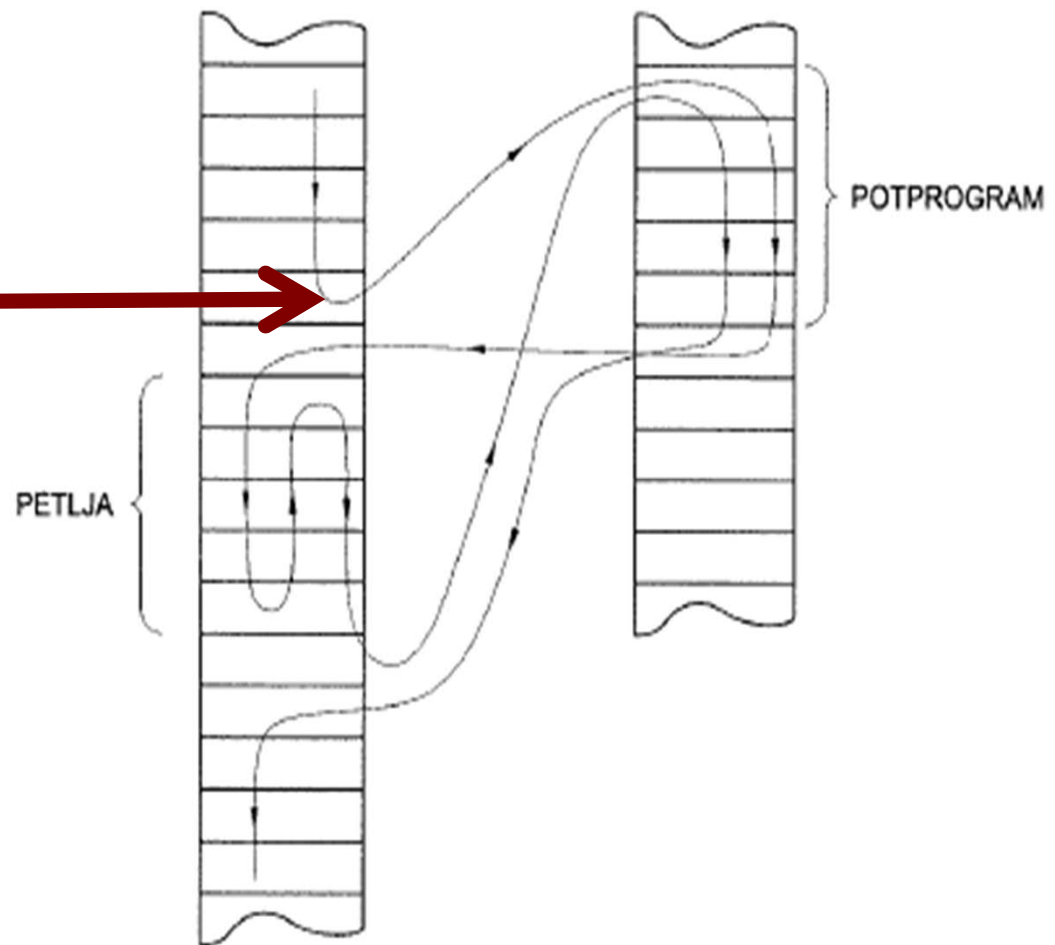
- prenošenjem vrijednosti parametara (call by value)
 - registri procesora
 - na stogu
- prenošenje adrese parametara (call by reference)

Operacijski sustavi: Računalo

Instruction thread (instrukcijska dretva)

Slijed instrukcija
koji procesor izvodi

kroz vrijeme



Operacijski sustavi: Računalo

VisuSim:

- Java applet sa simulatorom jednostavnog računala po Von Neumannovoj arhitekturi:

Programsko brojilo

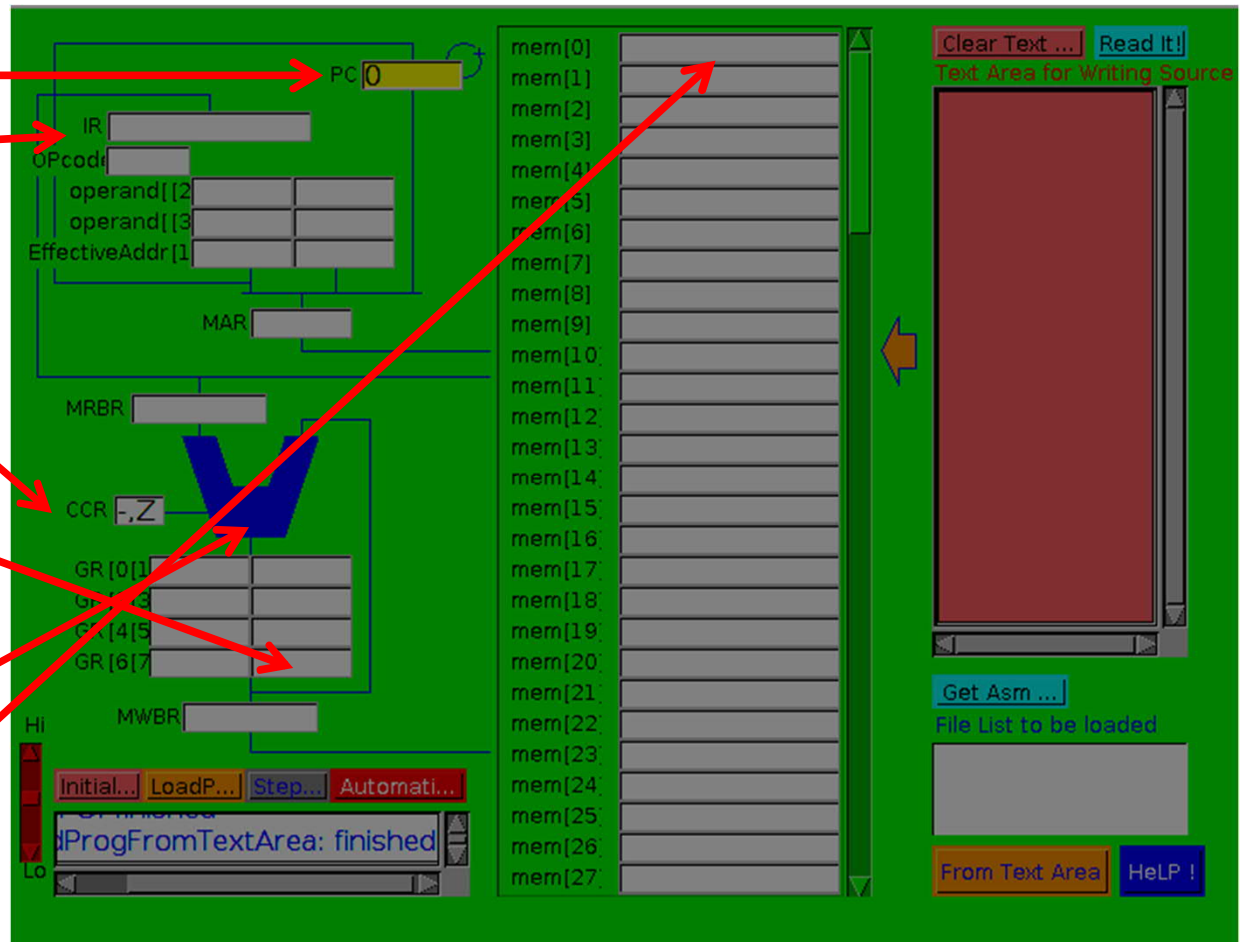
Instrukcijski registar

Registar stanja

Kazaljka stoga (GR7)

ALJ

Memorija



Operacijski sustavi: Računalo

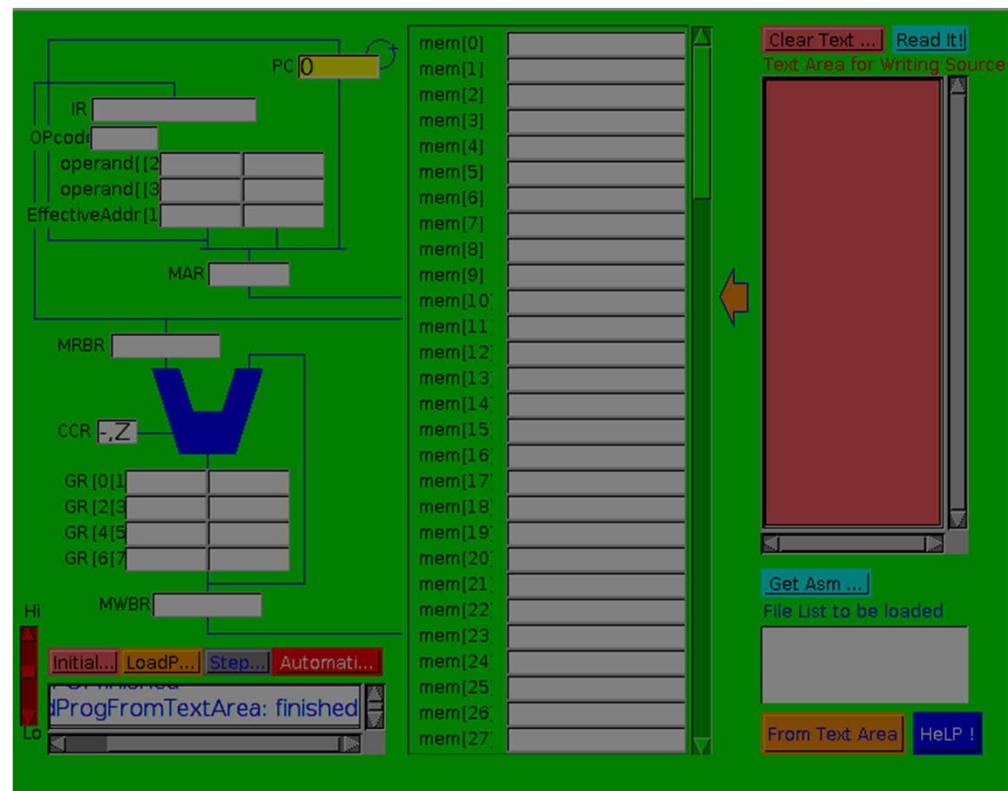
VisuSim: simulacija jednostavnog procesora

- 7 općih registara R0-R6, R7 je kazaljka stoga
- Dvoadresna instrukcija
- Mogući načini adresiranja operanda:

- Direktno adresiranje
- ID registra (R0-R7)
- Indirektno adresiranje
 - kao pomak na sadržaj registra
- (konstanta)

- CCR (Registar stanja):

- Z – zero flag
- N – negative flag



Operacijski sustavi: Računalo

Table 1: Instruction Set of Computer Simulator

	OPcode	1st Operand		OPcode	1st Operand	2nd Operand	
Control Instruction	halt noop	None	Binary Operation Instruction	move	Direct Addr RegId IndirAddr(Reg)	RegId	
Jump Instruction	jump	Direct Addr* RegId** IndirAddr(Reg)***			RegId	Direct Addr RegId IndirAddr(Reg)	
	jpgt jpge jplt jple jpeq jpne			add sub		Direct Addr RegId IndirAddr(Reg) Immed Val****	RegId
	call				mul div	RegId Immed Val	RegId
	Unary Operation Instruction	neg		Direct Addr RegId IndirAddr(Reg)	and or xor	Direct Addr RegId IndirAddr(Reg) Immed Val	RegId
push pop							
*direct addressing, ** register identification, ***indirect addressing with register modification ****immediate value addressing							

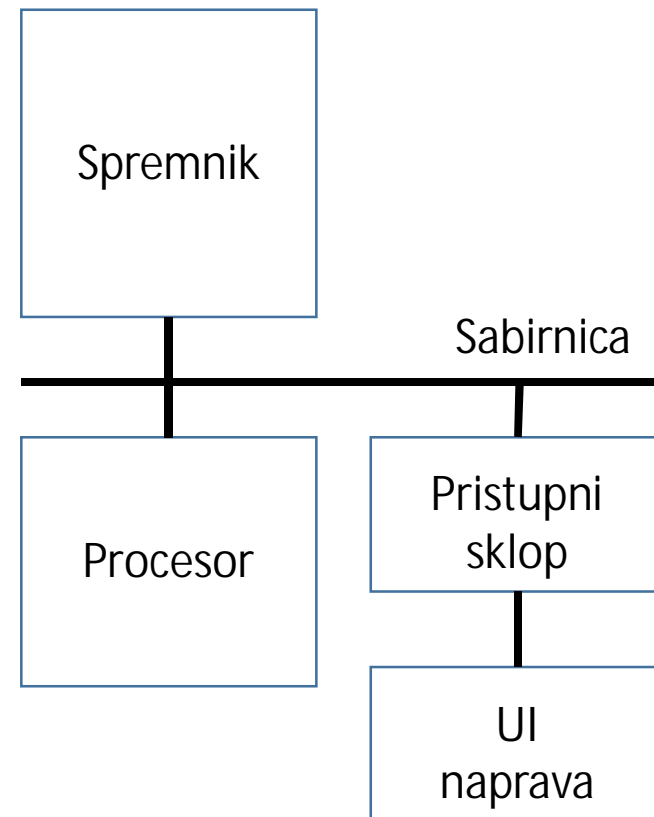
Operacijski sustavi: U/I

Ulazno-izlazni uređaji:

- tipkovnica, miš
- monitor, pisač
- vanjski memorijski uređaji
- ...

vrste pristupnih sklopova:

- prijenos pojedinačnih znakova
- prijenos blokova



Operacijski sustavi: U/I

Svaki I/O uređaj zauzima određene adrese iz I/O adresnog seta za svoje registre

Memory Mapped I/O (1) vs. Port mapped I/O (2)

(1) Adresno područje je dijeljeno

(2) Arhitektura predviđa preklapanje I/O i memorijskog adresnog područja:

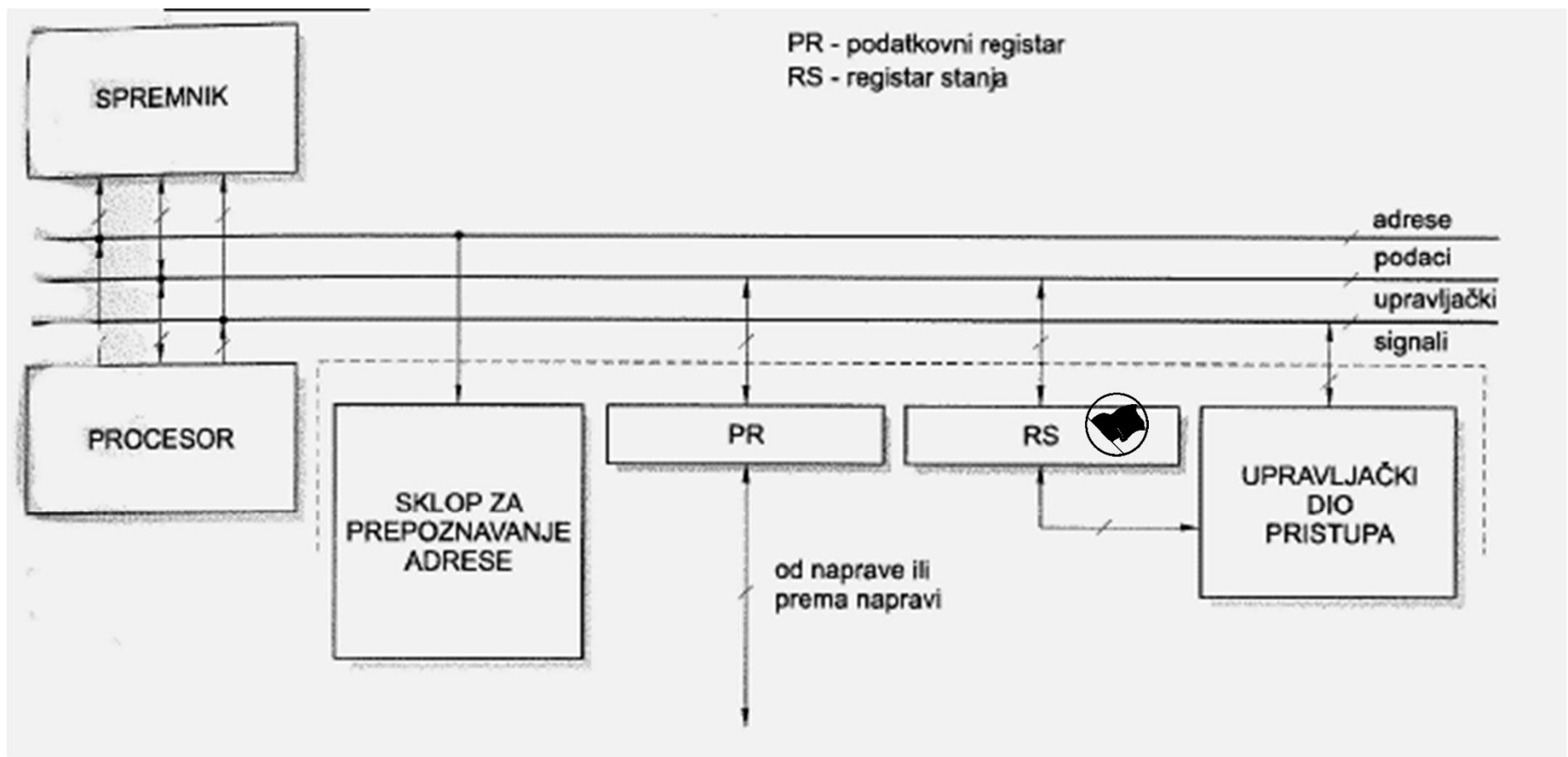
„Nove” instrukcije u instrukcijskom setu:

Ulazno izlazne instrukcije (IN, OUT)

https://en.wikipedia.org/wiki/Memory-mapped_I/O

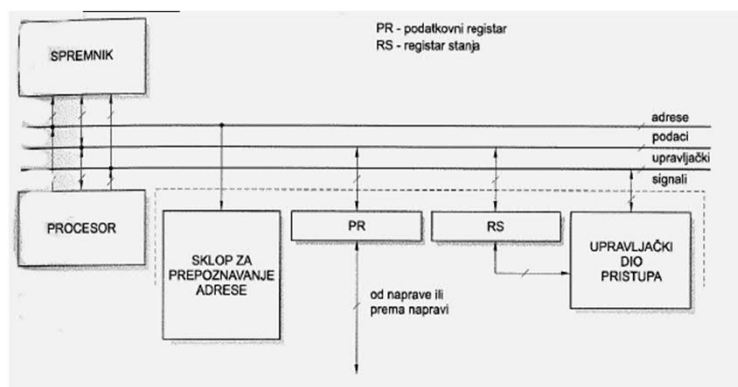
Operacijski sustavi: U/I

Pristupni sklop prijenosa pojedinačnog znaka



Operacijski sustavi: U/I

Pristupni sklop prijenosa pojedinačnog znaka



0: Početno stanje – zastavica spuštena

1: Petlja čekalica na zastavicu

2: Znak stigao iz okoline

3: UI upr. jedinica upisuje znak u PR

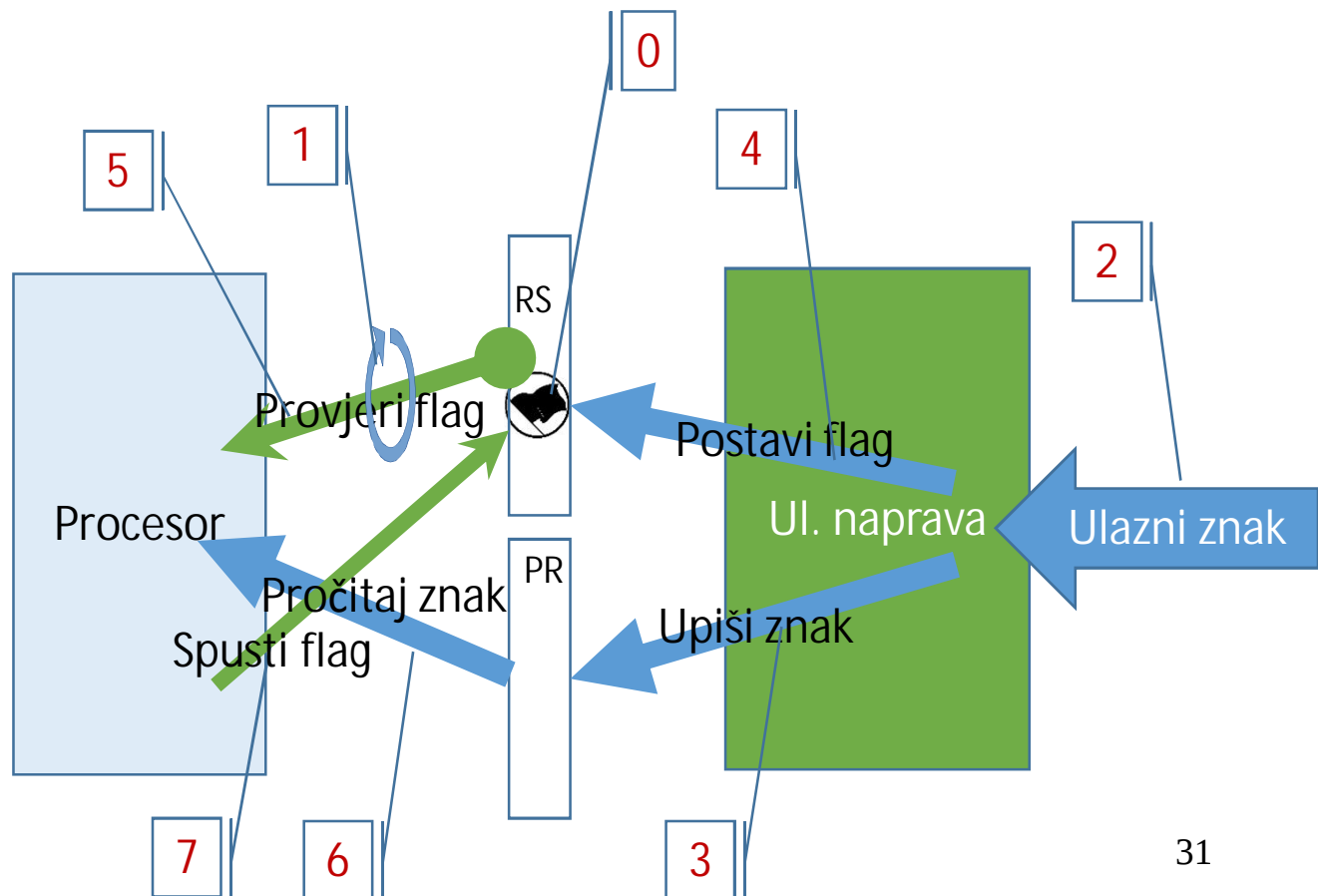
4: postavlja se zastavica

5: Iskače se iz unutarnje petlje čekalice

6: Spušta se zastavicu

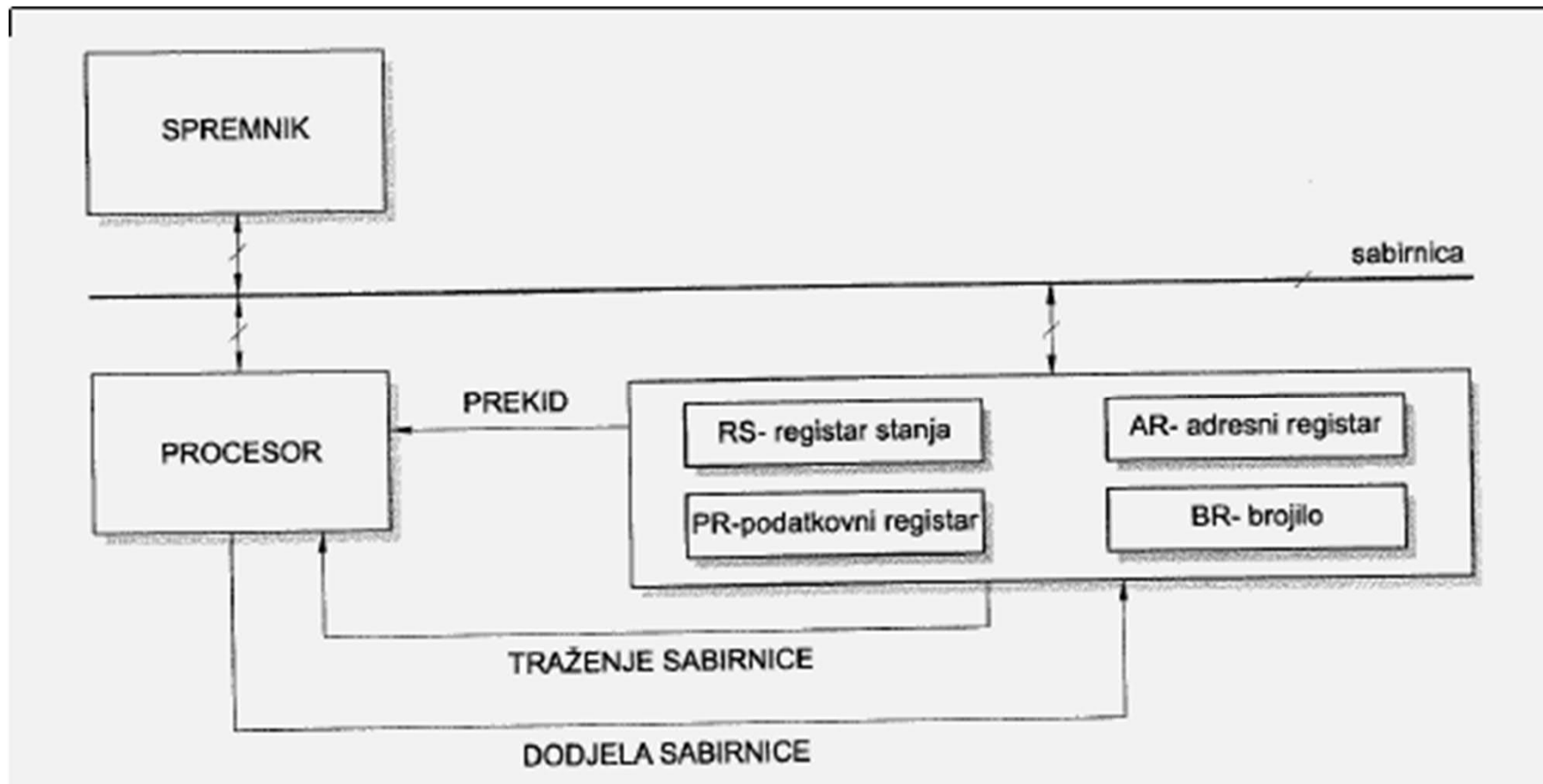
7: Učitava se znak u procesor

Vrati se na 1 ili nastavi



Operacijski sustavi: U/I

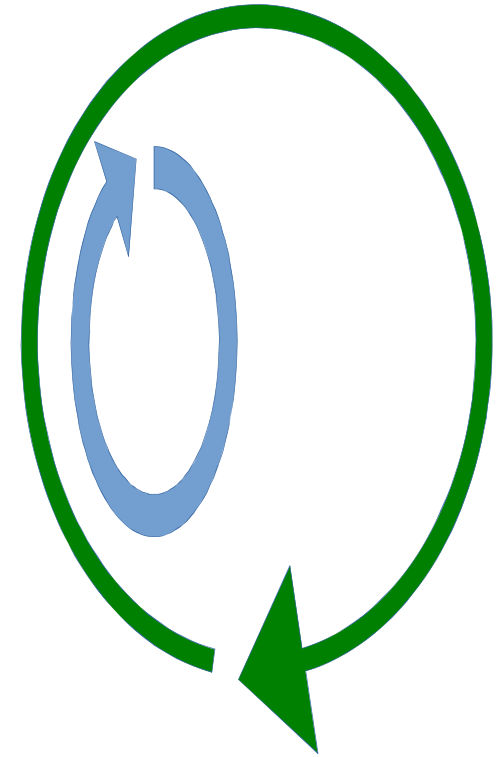
Model sklopa za neposredni pristup spremniku



Operacijski sustavi: U/I

Čekanje znaka u petlji:

```
ponavlja {  
    ponavlja {  
        pročitaj zastavicu stanja u RS  
    }  
    dok zastavica stanja u RS nije postavljena  
    pročitaj znak iz PR  
    poništi zastavicu stanja u RS  
    pohrani znak  
}  
dok znak nije "oznaka kraja"
```



Procesor većinu vremena provjerava da li je znak stigao: Petlja čekalica!

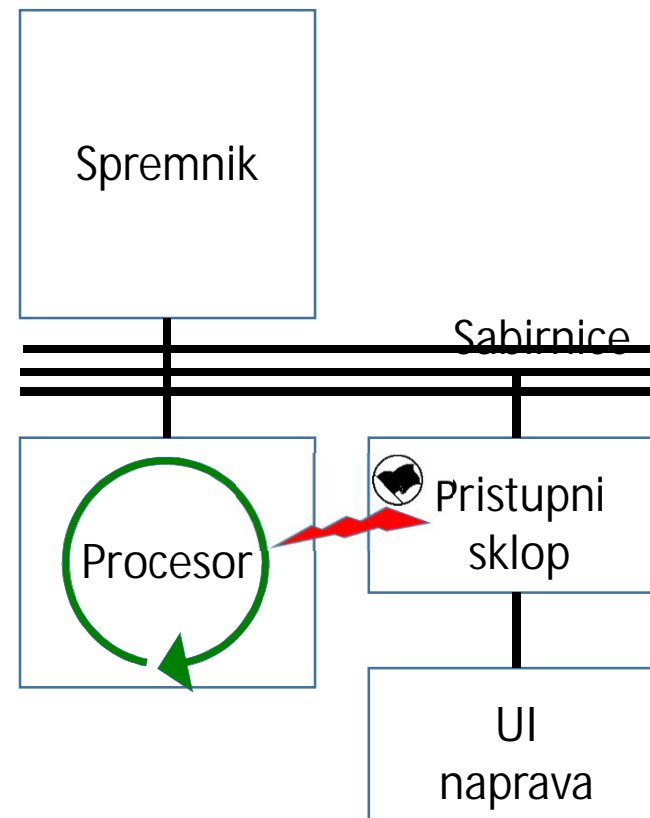
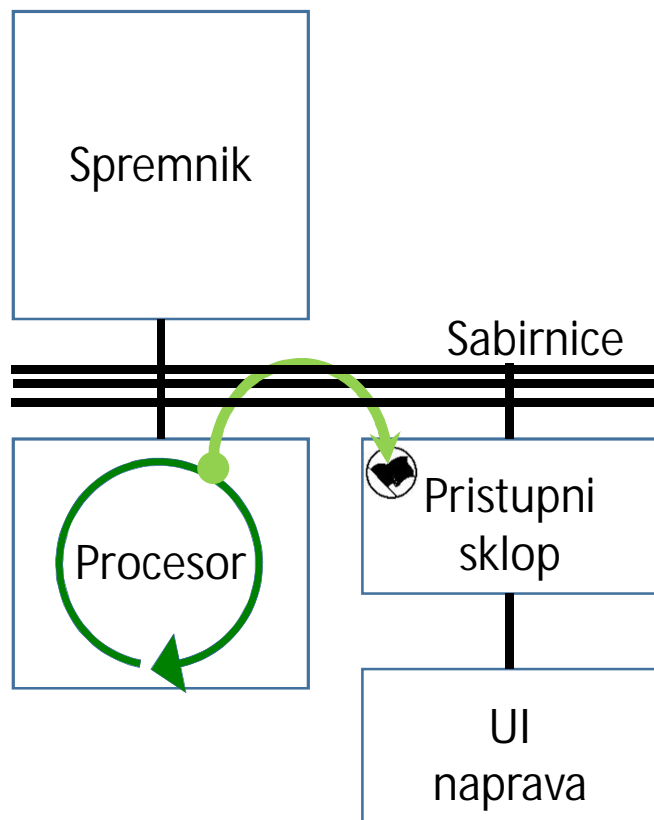
Rad procesora izgubljen na čekanje, ne na koristan rad

Operacijski sustavi: U/I

Prekidni način rada procesora

čekanje na asinkroni događaj dolaska znaka – iracionalno!

Prekidni mehanizam (interrupt mechanism)



Operacijski sustavi: U/I


Dva načina rada:

- Korisnički (user) –
 - ograničen pristup sistemskim resursima
 - Podskup instrukcijskog seta procesora
 - Sistemski
- .Sistemski način rada -> jezgreni način rada**


Operacijski sustavi: U/I



Ponavljati {



dohvatiti iz spremnika instrukciju na koju pokazuje programsko brojilo;
dekodirati instrukciju, odrediti operaciju koju treba izvesti;
povećati sadržaj programskog brojila da pokazuje na slijedeću instrukciju;
odrediti odakle dohvatiti operande i gdje pohraniti rezultat;
dovesti operande na ALJ, izvesti zadanu operaciju;
pohraniti rezultat u odredište;



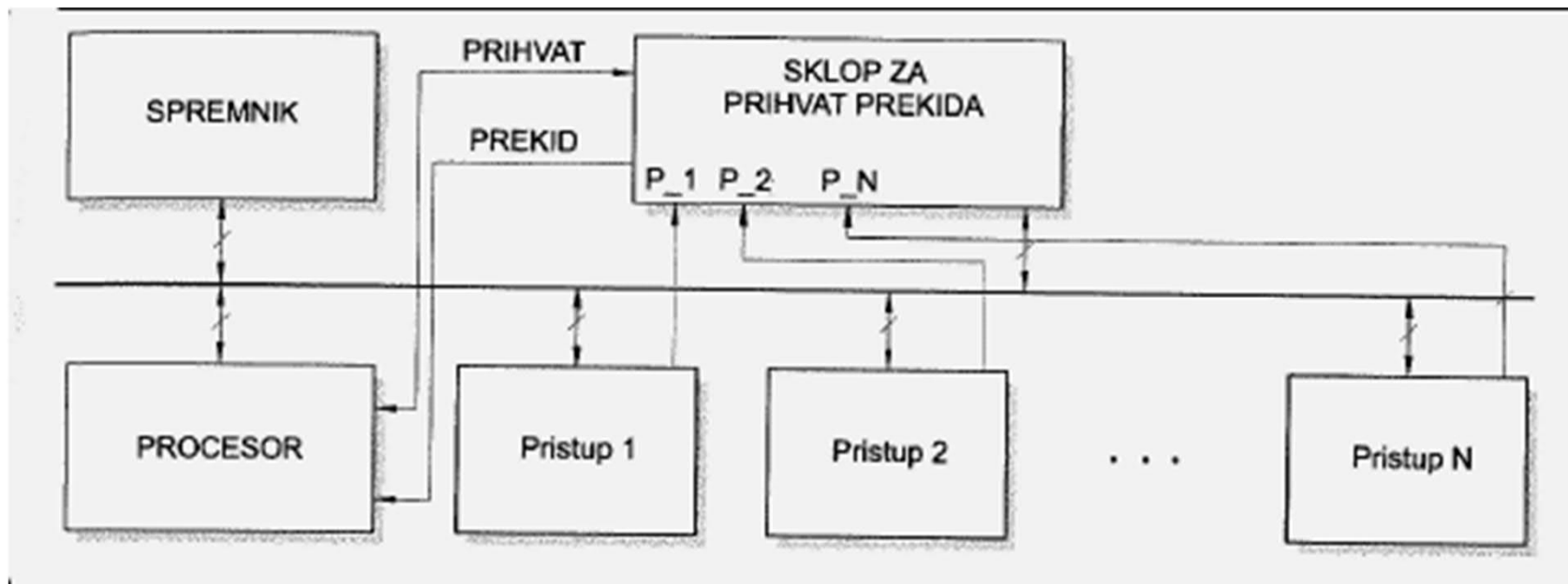
ako je (prekidni signal postavljen) i (omogućeno je prekidanje) {
onemogućiti prekidanje;
prebaci adresiranje u sistemski adresni prostor i aktiviraj sistemsku kazaljku stoga;
pohrani programsko brojilo na sistemski stog;
stavi u programsko brojilo adresu potprograma za obradu prekida;



}
}
dok je (procesor uključen);

Operacijski sustavi: U/I

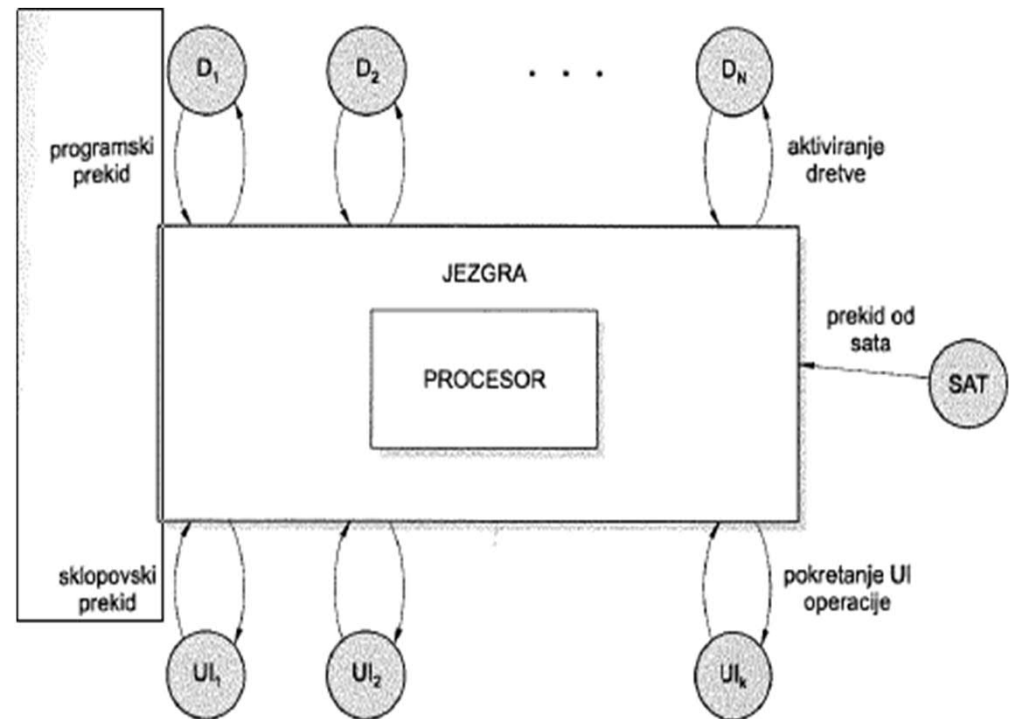
Sustav sa sklopom za prihvatanje prekida



Operacijski sustavi: jezgra

Jezgra: prvi sloj OS iznad procesora

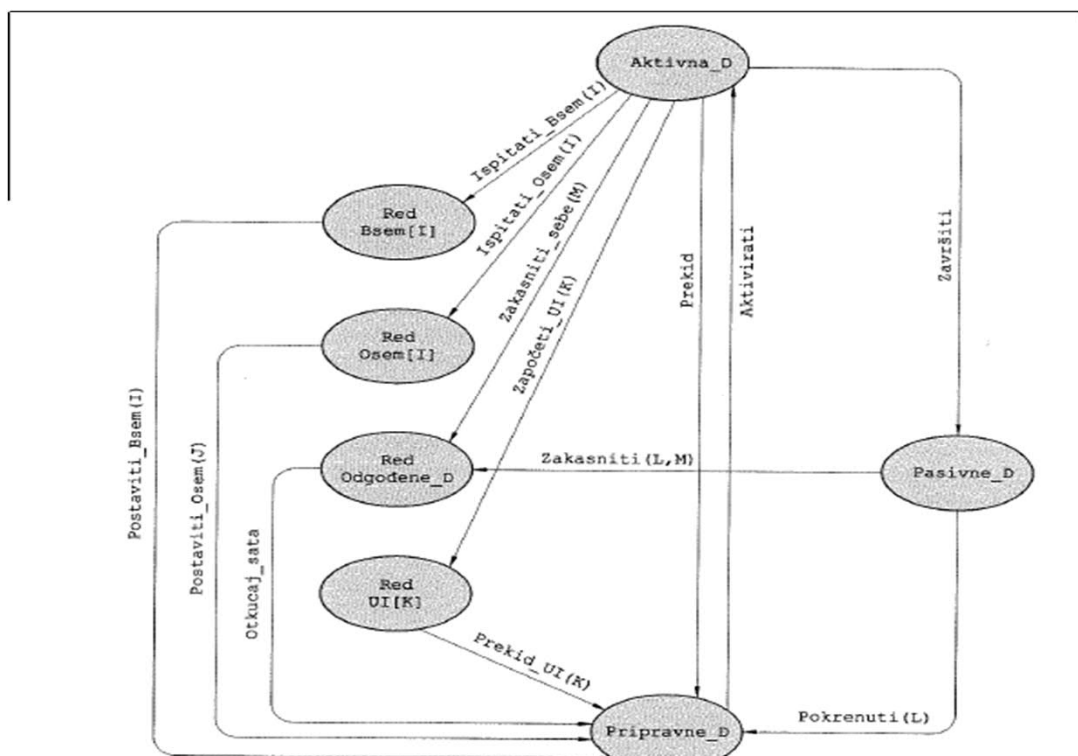
- Upravljanje dretvama i sistemskim resursima
- Skupina funkcija koje se pozivaju sklopovskim ili softverskim prekidima:
 - prekidi U/I naprava (HW)
 - prekidi sata (HW)
 - prekidi dretvi (SW)



Operacijski sustavi: jezgra

Struktura podataka jednostavnog modela jezgre:

- pohranjuju se sve inf. o dretvama potrebne za donošenje odluka o njihovom izvođenju
- opisnik dretve (thread descriptor)
- zaglavlja listi



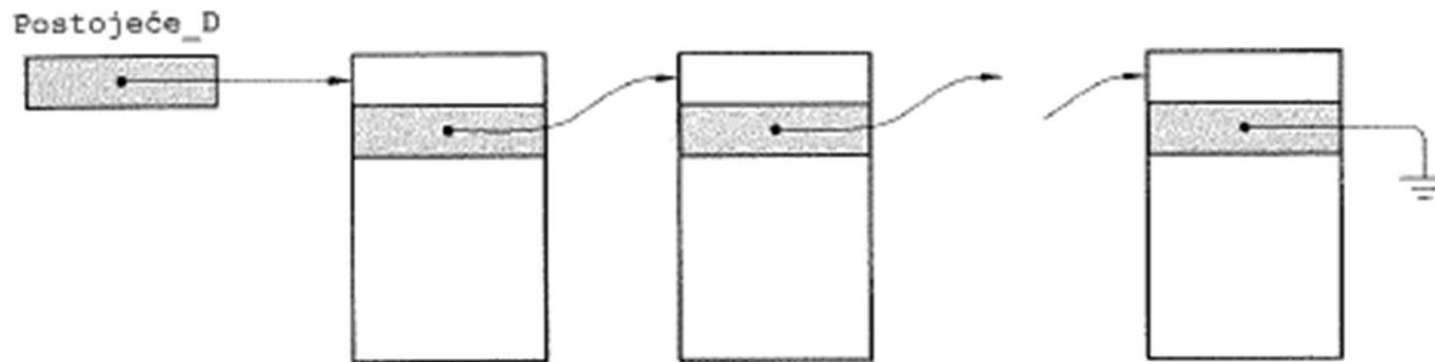
kazaljke	←→
kazaljke	←→
identifikator procesa	
identifikator dretve	
stanje dretve	
prioritet dretve	
početna adresa DAP	
veličina DAP	
entry point dretve	
zadano kašnjenje	
prostor za smještaj konteksta dretve	
programsko brojilo	

DAP – Dretveni adresni prostor

Operacijski sustavi: jezgra

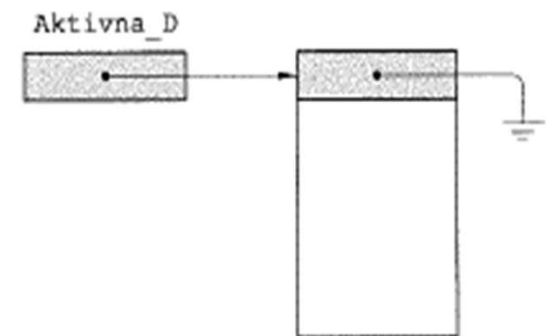
Lista postojećih dretvi

- omogućava pregled svih deskriptora, bez obzira na kojoj se listi trenutno nalazi
- dretva čiji je deskriptor samo na listi postojećih dretvi nalazi se u *pasivnom* stanju



Aktivno stanje dretve

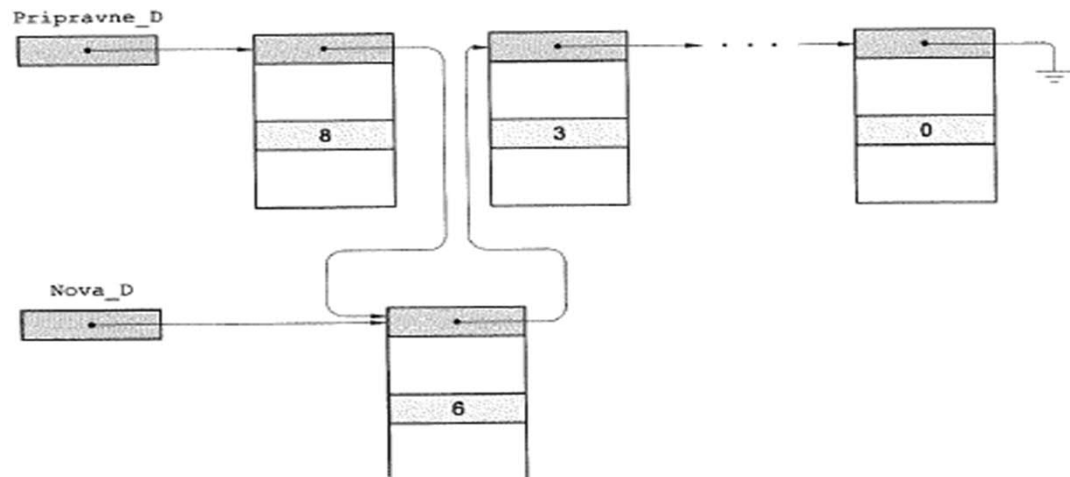
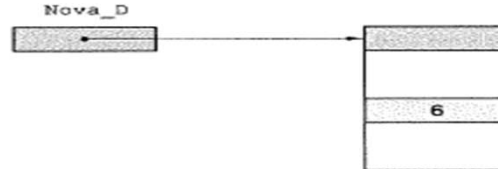
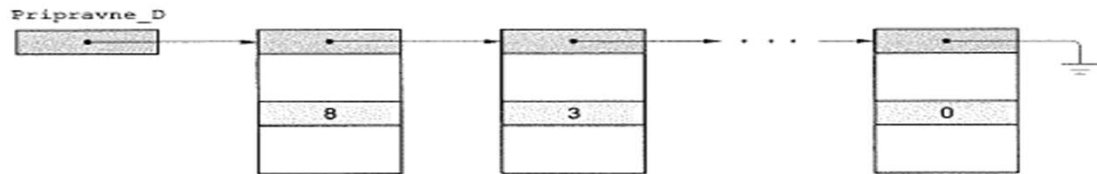
- u jednoprocesorskom sustavu samo jedna dretva može biti aktivna: ona čije se instrukcije upravo izvode u procesoru



Operacijski sustavi: jezgra

Pripravno stanje dretve

- sve one dretve koje bi se mogle izvoditi, ali moraju čekati na dodjelu procesora
- dodjeljivanje dretvi:
 - po redoslijedu dolaska?
 - po prioritetu?

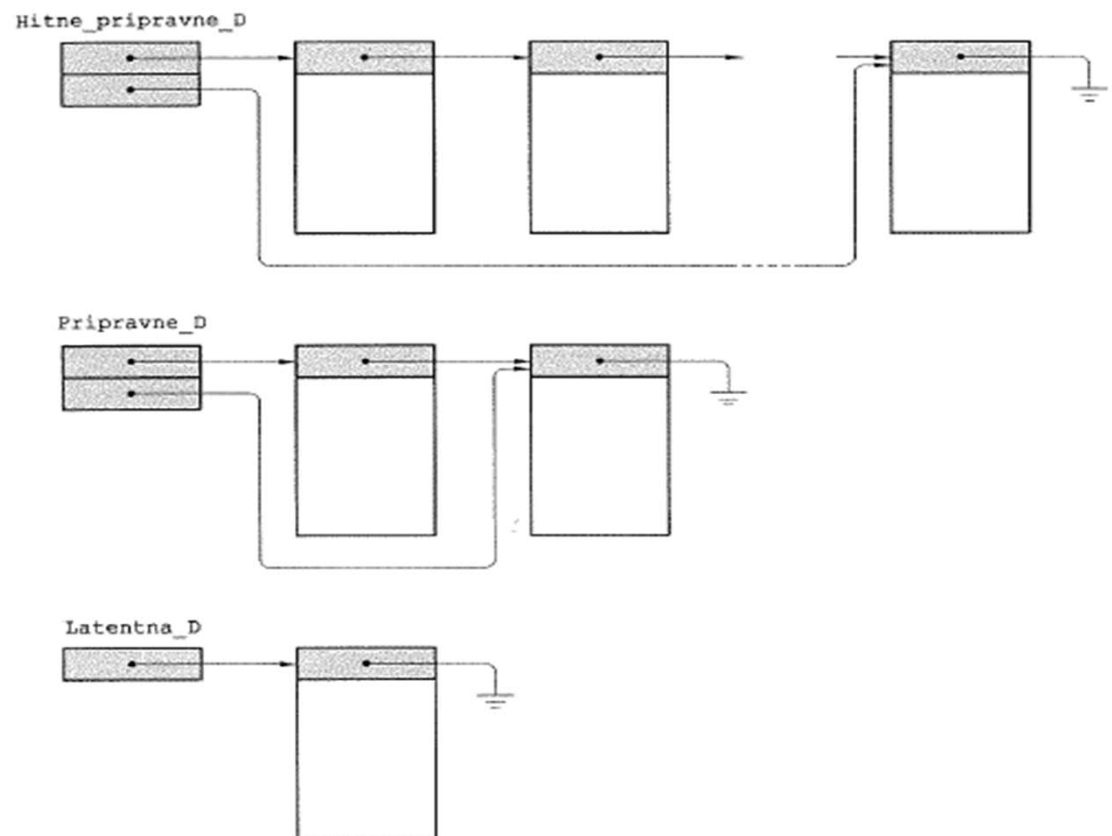


Operacijski sustavi: jezgra

Pretraživanje i insertiranje deskriptora dretvi po prioritetu može biti kompleksan posao, pa je moguće formirati više listi pripravnih, po diskretnim prioritetima

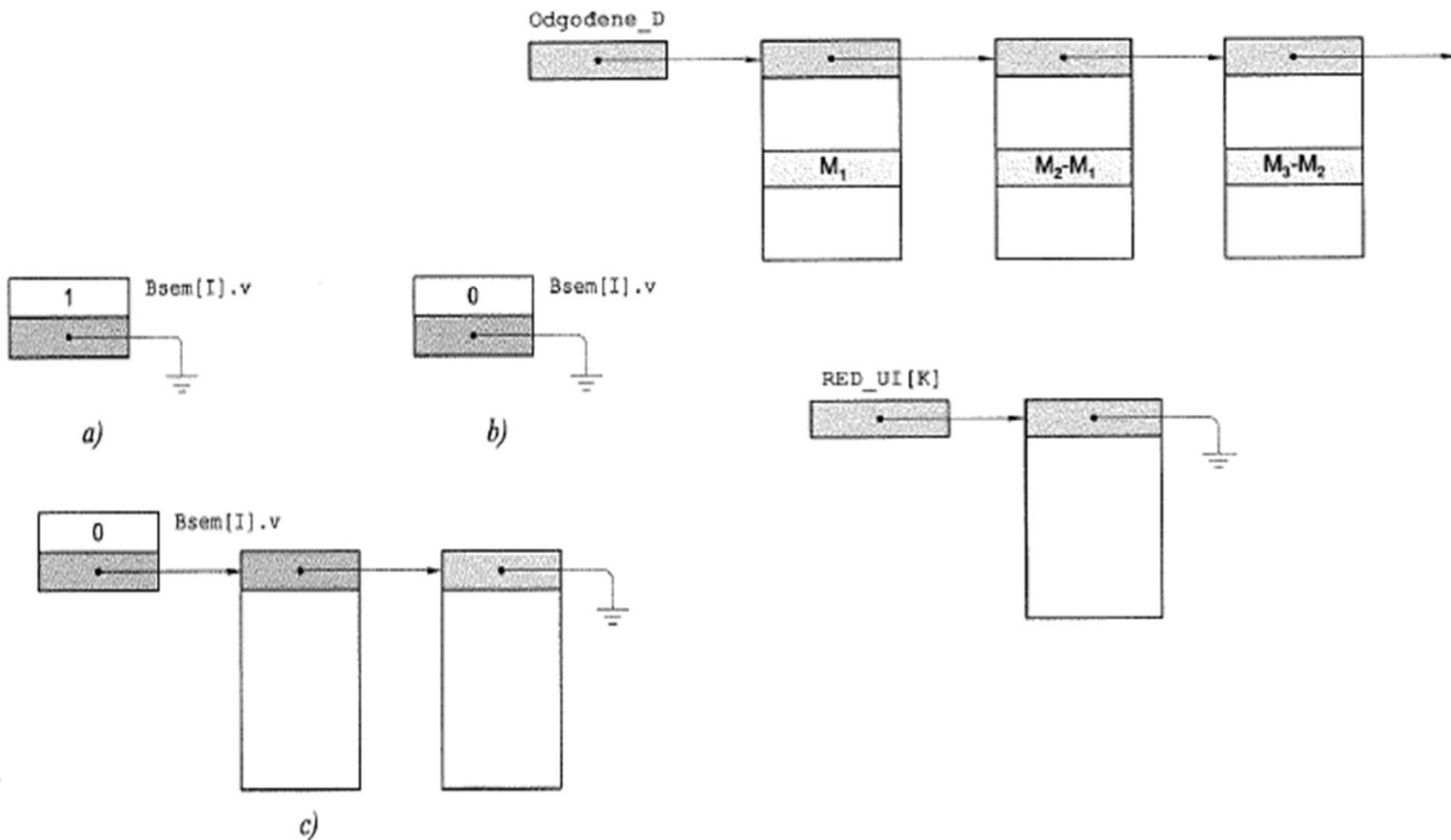
Latentna dretva:

- ona koja se izvodi kada sistem nema što raditi



Operacijski sustavi: jezgra

•Primjeri blokiranih listi



Operacijski sustavi: jezgra

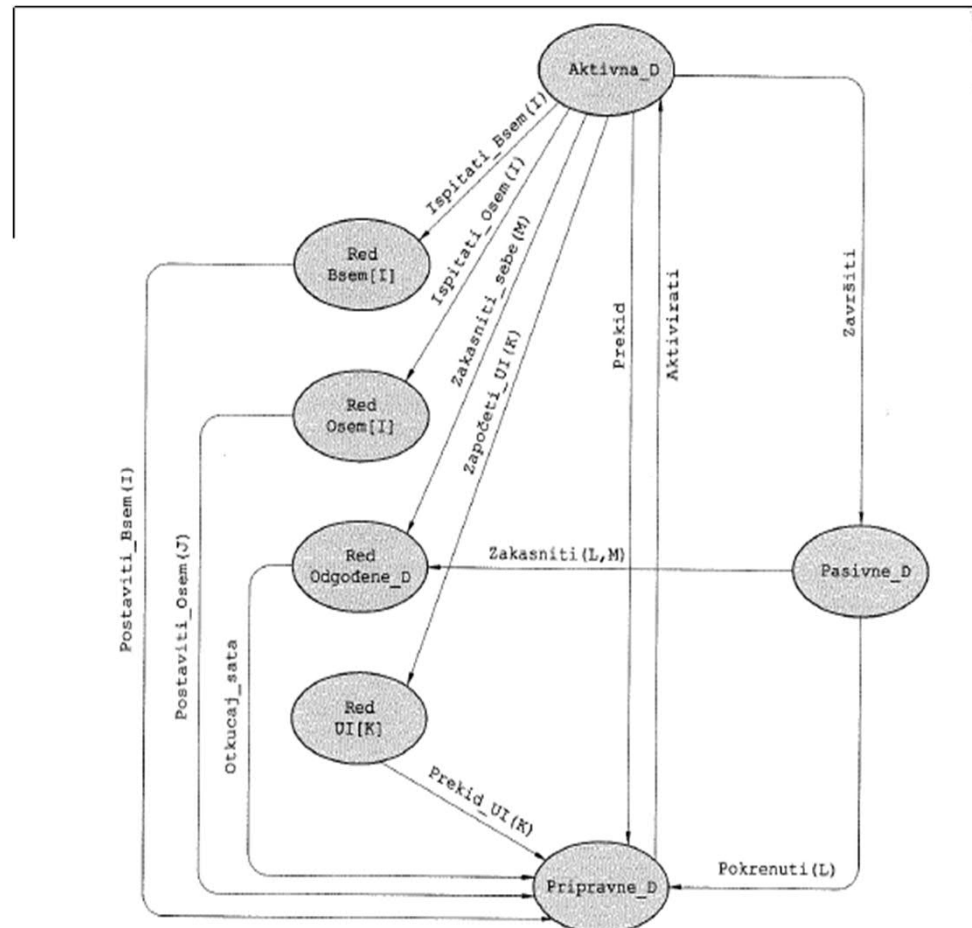
Graf mogućih stanja dretvi

- Dretva se istovremeno može nalaziti samo u jednom od stanja:

- lista u kojoj se nalazi i/ili
- varijabla deskriptora

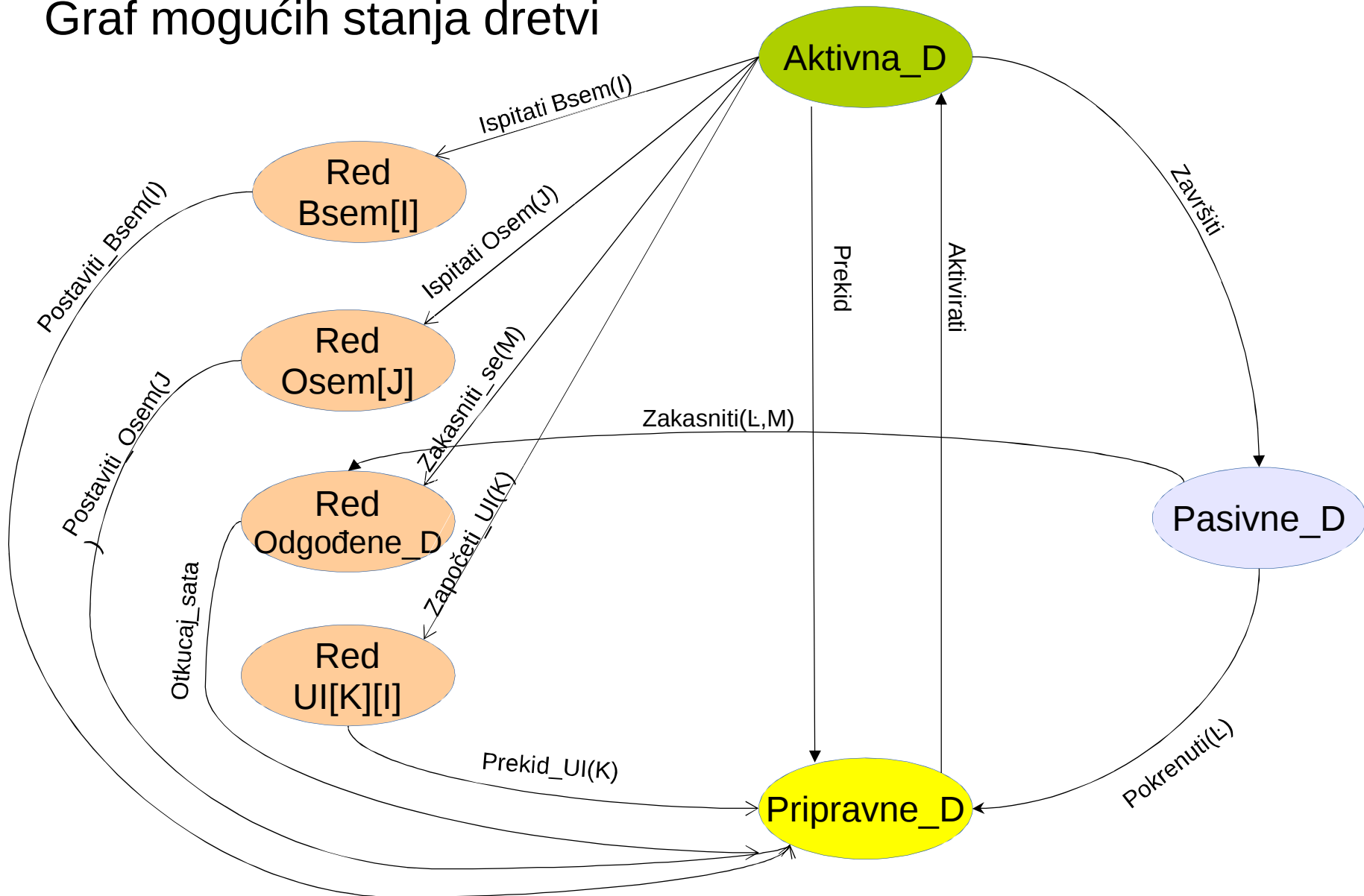
- Prijelazi

- pozivom jezgrinih funkcija,
- prekidi



Operacijski sustavi: jezgra

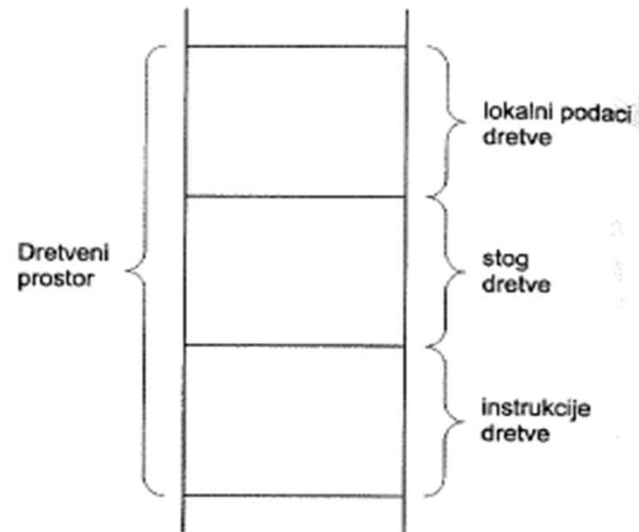
Graf mogućih stanja dretvi



Operacijski sustavi

Međusobno isključivanje

- Višedretvenost (multithreading):
 - svaki se proces sastoji od barem jedne dretve
 - dretve unutar jednog procesa dijele sva sredstva koje je OS stavio na raspolaganje tom procesu

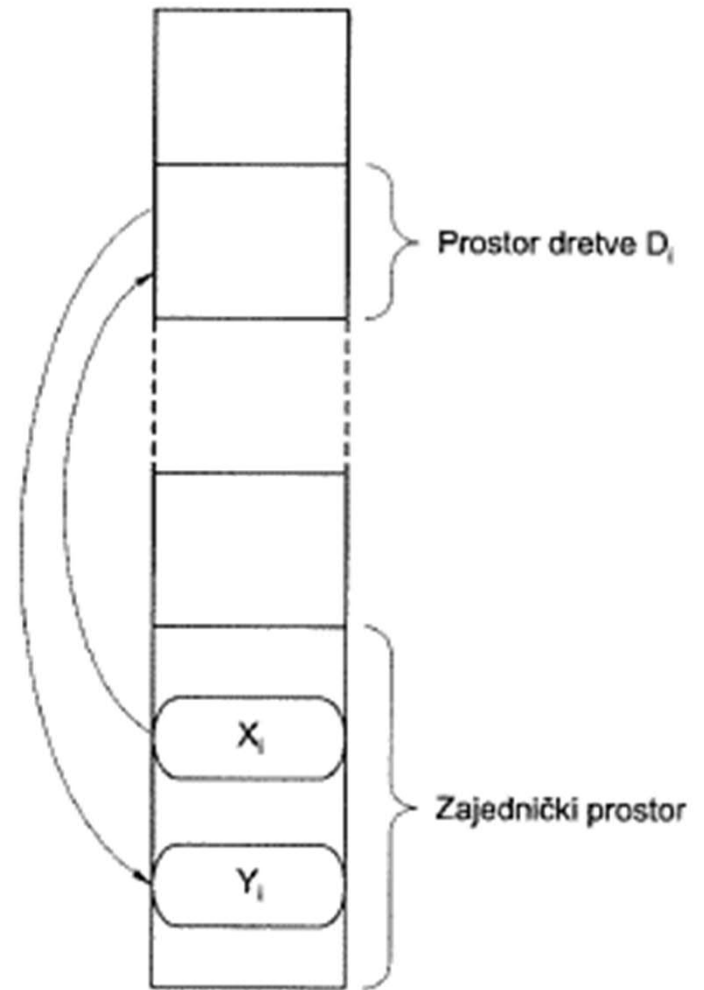


- dretve se mogu izvoditi istodobno (paralelno) na više procesora ili prividno istodobno (prividno paralelno) ako se izvode na istom procesoru (dretve se izvode naizmjenično)

Operacijski sustavi

Međusobno isključivanje

- Višedretvenost:
 - Pretpostavke (pojednostavljenja):
 - D_i čita podatke iz X_i (domena)
 - D_i piše podatke u Y_i (kodomena)
 - $f_i : X_i \rightarrow Y_i$
 - $Y_i = f_i (X_i)$



Operacijski sustavi

Međusobno isključivanje

- Međusobno nezavisni zadaci:
 - nemaju dodirnih točaka
 - ako pojedini zadaci samo čitaju sa pojedinih lokacija svoje ulazne podatke

$$(X_i \cap Y_j) \cup (X_j \cap Y_i) \cup (Y_i \cap Y_j) = 0$$

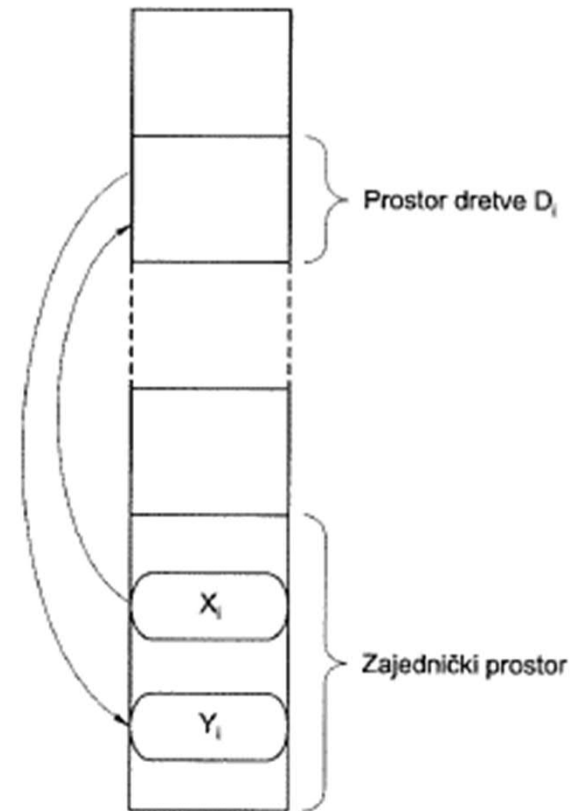


piše gdje drugi čita



pišu na isto mjesto

$(X_i \cap X)$ - smije se čitati sa iste lokacije

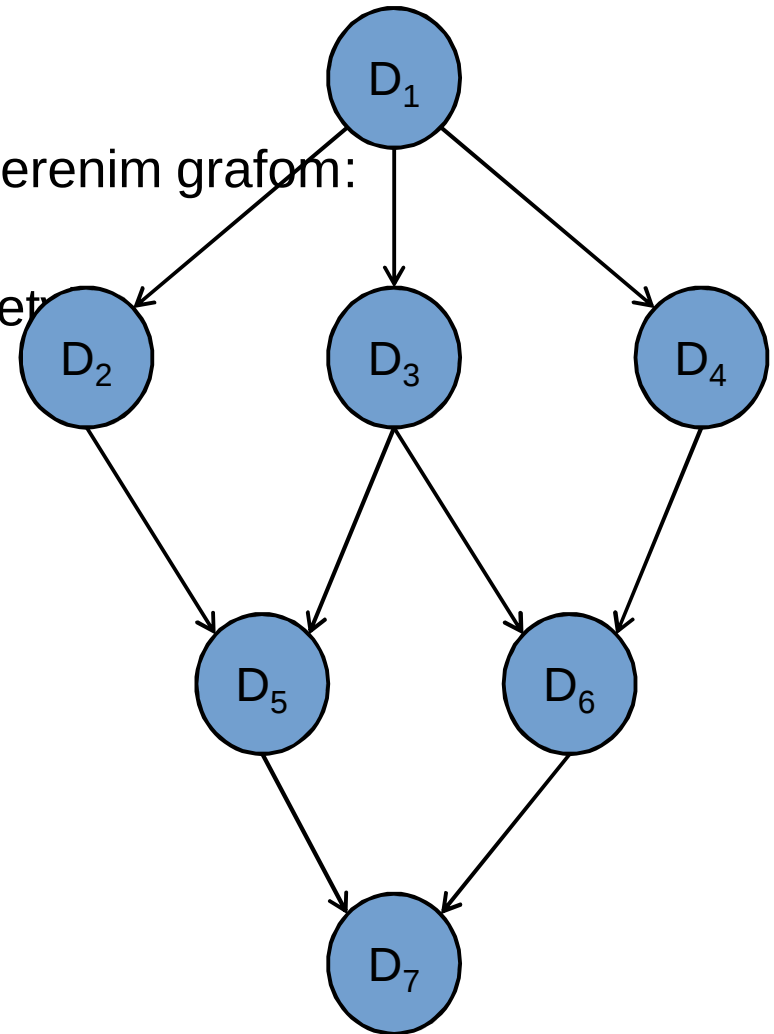


Operacijski sustavi

Međusobno isključivanje

- Višedretvenost:

- $\underline{D} = \{ D_1, D_2, D_3, \dots, D_N \}$
- Sustav dretvi je najlakše predložiti usmjerenim grafom:
- čvorovi: pojedine dretve
- strelice (grane): redoslijed izvođenja dretvi
- $<$ - treba se izvesti prije
- $D_1 < D_2, D_1 < D_3, D_1 < D_4$
- $D_2 < D_5 \dots$
- $(D_i < D_j) \text{ i } (D_j < D_k) \text{ onda } (D_i < D_k)$

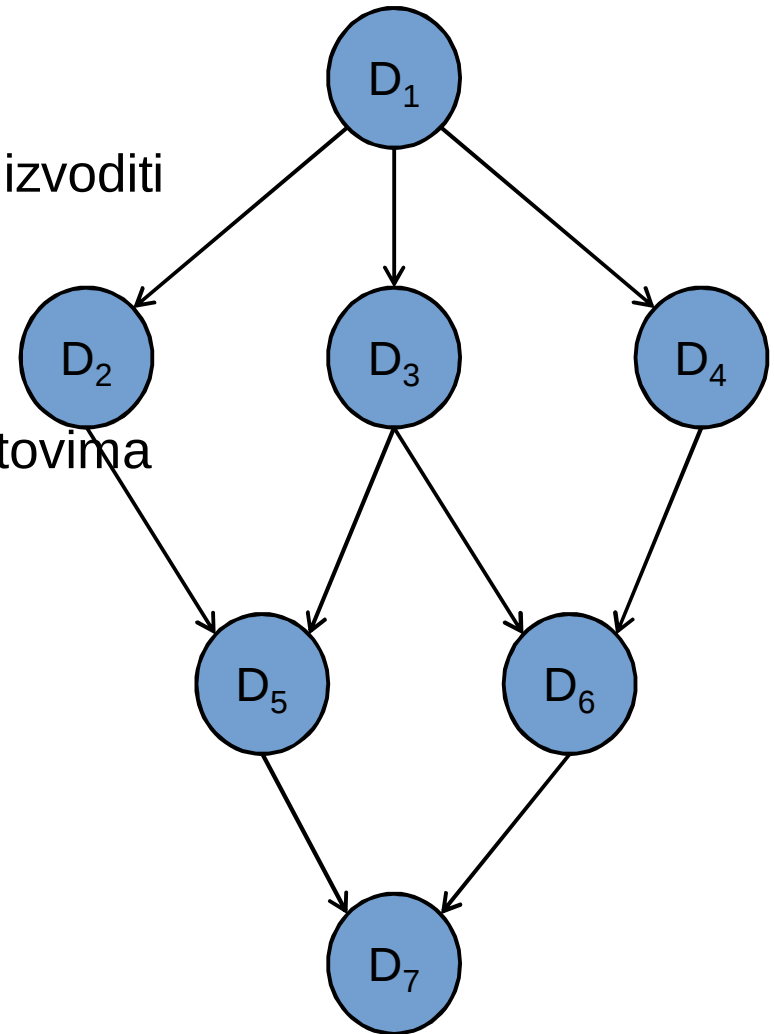


Operacijski sustavi

Međusobno isključivanje

- Višedretvenost:

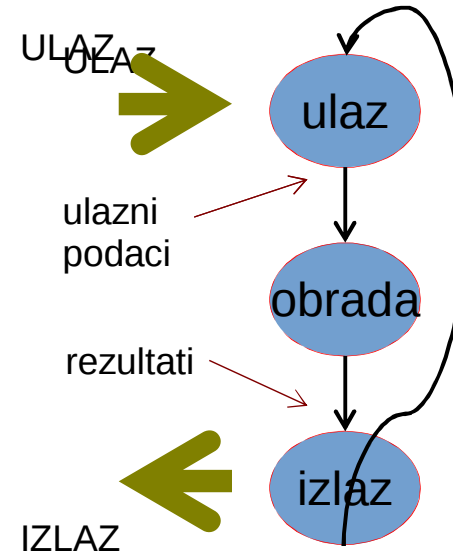
- Međusobno nezavisne dretve mogu se izvoditi proizvoljnim redoslijedom, i istodobno
- Dretve koje se ne nalaze na istom putu moraju biti međusobno nezavisne
- Dretve koje se nalaze na paralelnim putovima mogu se izvoditi paralelno:
 - istodobno (više fizičkih procesora)
 - prividno istodobno (isti procesor)



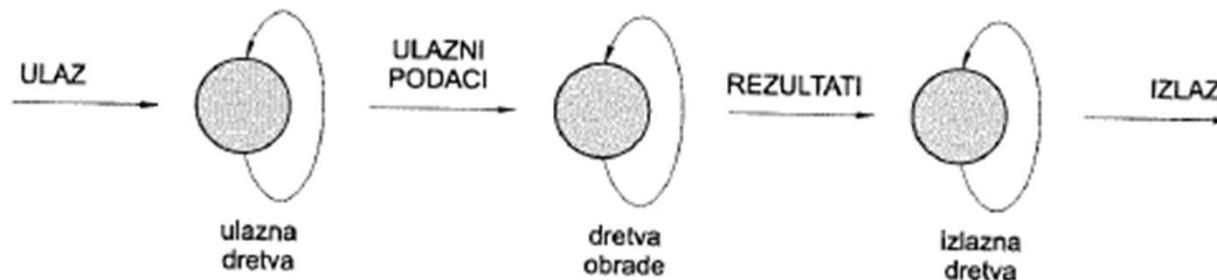
Operacijski sustavi

Međusobno isključivanje

- Višedretvenost:



- cikličke dretve – trajno se ponavljaju
- cjevovodna obrada



Operacijski sustavi

Međusobno isključivanje

- kritični odsječak:
 - dijelovi dretvi koji koriste neko zajedničko sredstvo
 - kroz kritični odsječak smiju prolaziti samo pojedinačno

dok je (1) {

K.O. // Kritični odsječak (critical section)

NK.O. // Ne-Kritični odsječak

Operacijski sustavi

Međusobno isključivanje

- Algoritam međusobnog isključivanja dretvi mora zadovoljiti:
 - Samo jedna dretva u nekom trenutku može biti u K.O.
 - Algoritam međusobnog isključivanja mora djelovati i onda kada su brzine izvođenja dretvi različite.
 - Ako neka dretva zastane u N.K.O. to ne smije spriječiti drugu dretvu da uđe u K.O.
 - Izbor koja dretva treba ući u K.O. mora se obaviti u konačnom vremenu.

Operacijski sustavi

Sinkronizacija dretvi

- Ograničeni spremnik

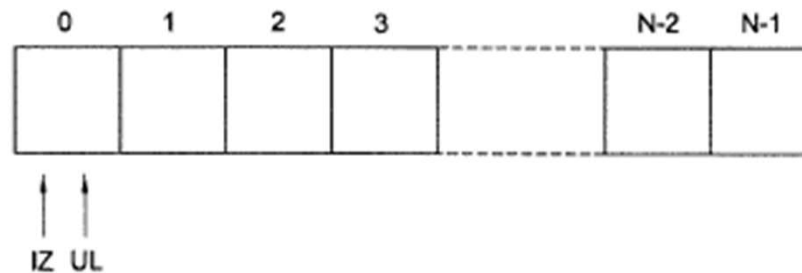
Pomicanje kazaljki:

$$UL = (UL + 1) \bmod N$$

$$IZ = (IZ + 1) \bmod N$$

N – veličina spremnika

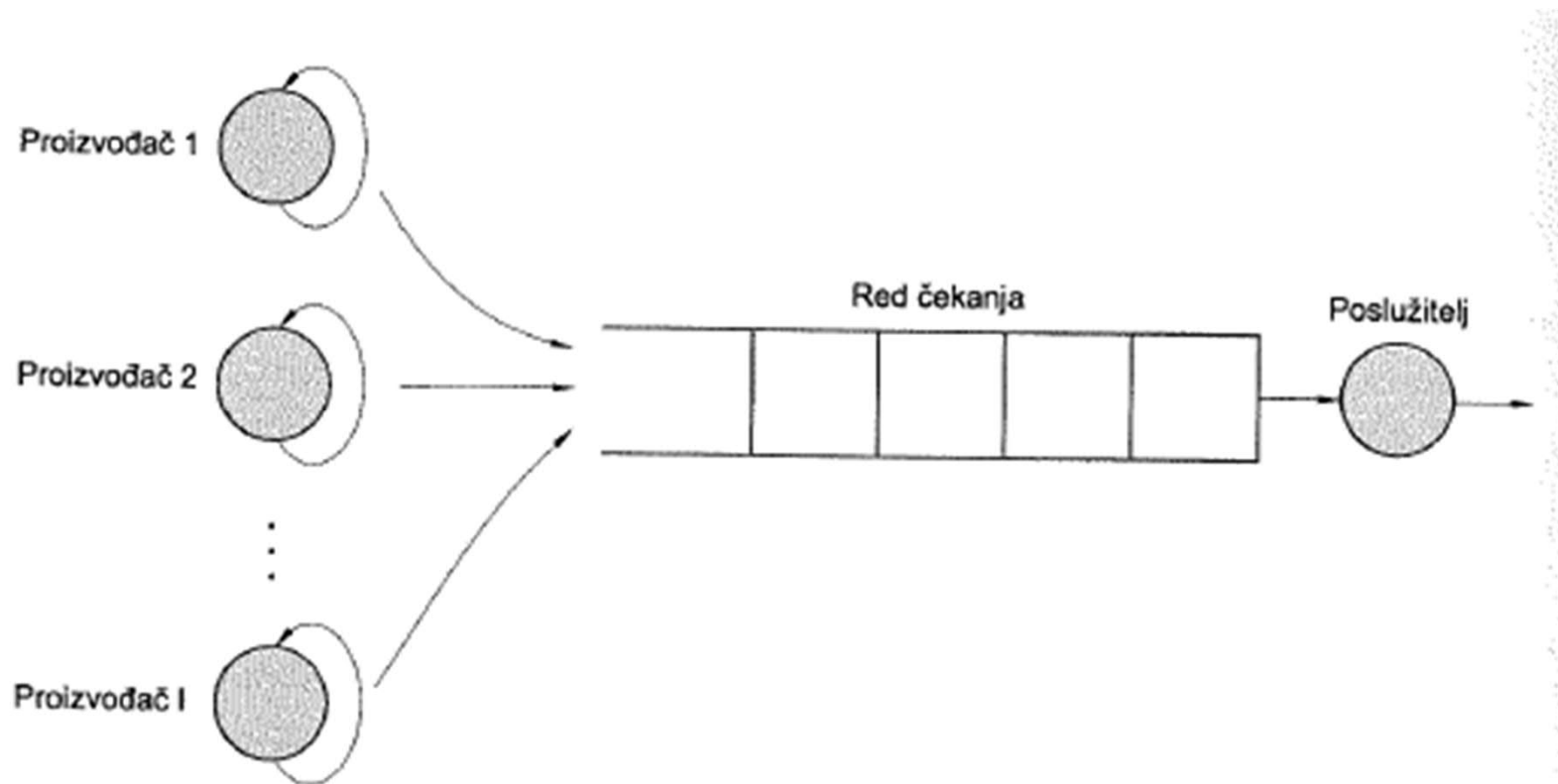
Uz postojeći "brojač poruka", potrebno uvesti novi brojač (opći semafor) praznih pretinaca, kako ne bi došlo do prepunjavanja!



Operacijski sustavi

Sinkronizacija dretvi

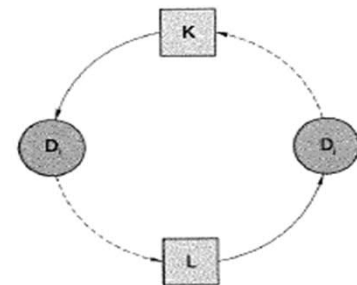
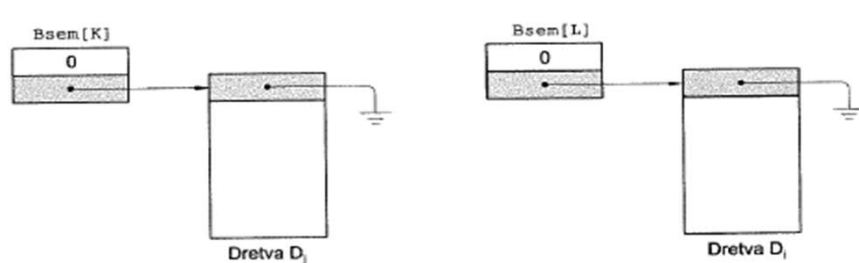
- Sustav sa više proizvođača
 - Pristup do kazaljke UL u ovakvom slučaju postaje kritični odsječak za sve proizvođače i moguće je riješiti binarnim semaforom.



Operacijski sustavi

Sinkronizacija dretvi

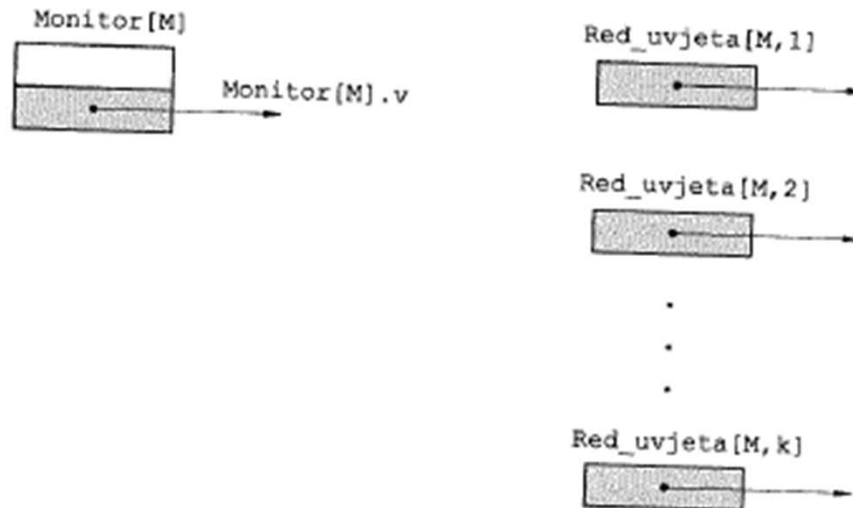
- Potpuni zastoј (Deadlock)
 - Pojedinačnim ispitivanjem jednostavnih uvjeta (i nepažljivim korištenjem istih moguć je slučaj potpunog zastoja:
 - najmanje dvije dretve se moraju takmičiti za najmanje dva sredstva koja se moraju upotrebljavati međusobno isključeno
 - Primjer "filozofa" iz knjige!
- Nužni uvjeti:
 - neko sredstvo u istom trenutku može upotrebljavati samo jedna od dretvi
 - dretvi se sredstvo ne može oduzeti
 - dretva drži dodijeljeno sredstvo dok čeka na dodjelu dodatnog sredstva



Operacijski sustavi

Sinkronizacija dretvi

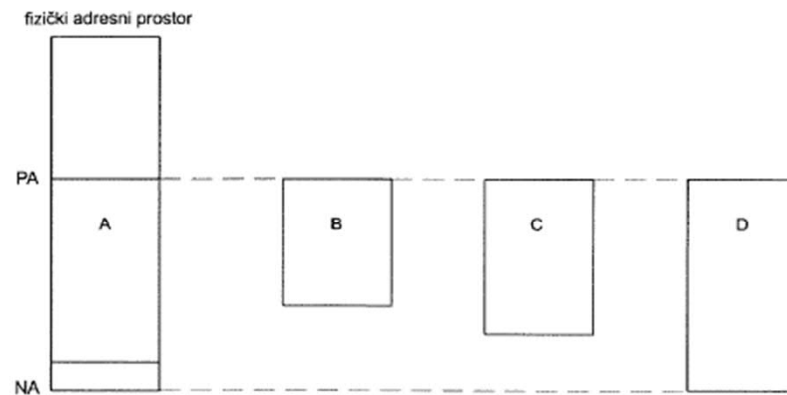
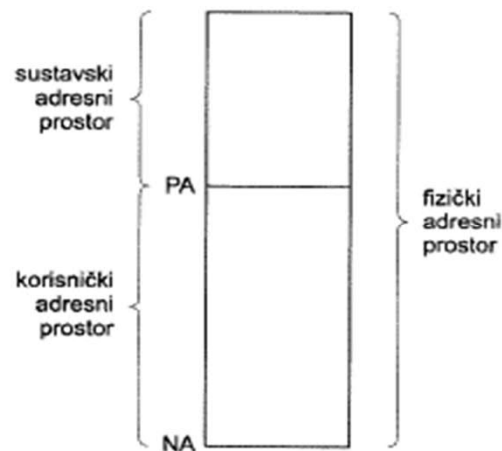
- Monitor
 - Mehanizmi suradnje dretvi objedinjeni u prikladne skupine funkcija za razrješavanje cjelovitih problema.
 - Objedinjeni nadzorni program koji nadzire ovu skupinu nazivamo *monitor*
 - *Monitor* se sastoji od svoje strukture podataka i funkcija koje djeluju nad tom strukturom.
 - Izvođenje monitorskih funkcija mora biti pojedinačno
 - pozivanje funkcija je kritični odsječak



Operacijski sustavi

Spremnički prostor

- Adresni prostor:
- PA - početna adresa korisničkog adresnog prostora
- NA - najveća adresa korisničkog adresnog prostora



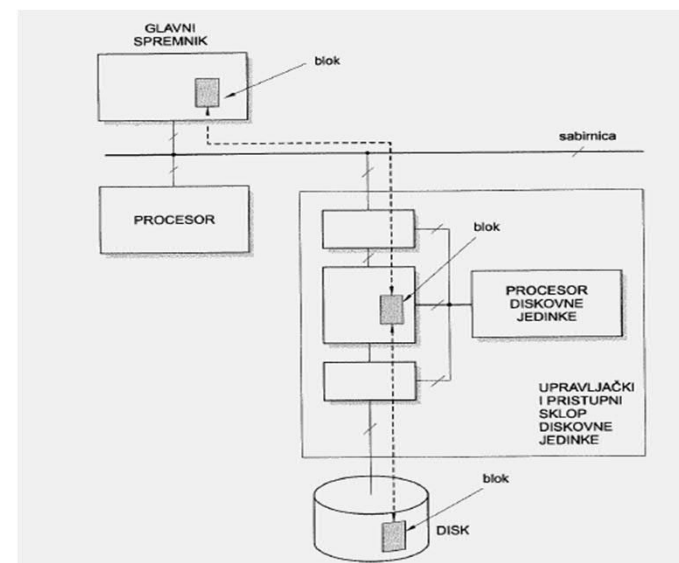
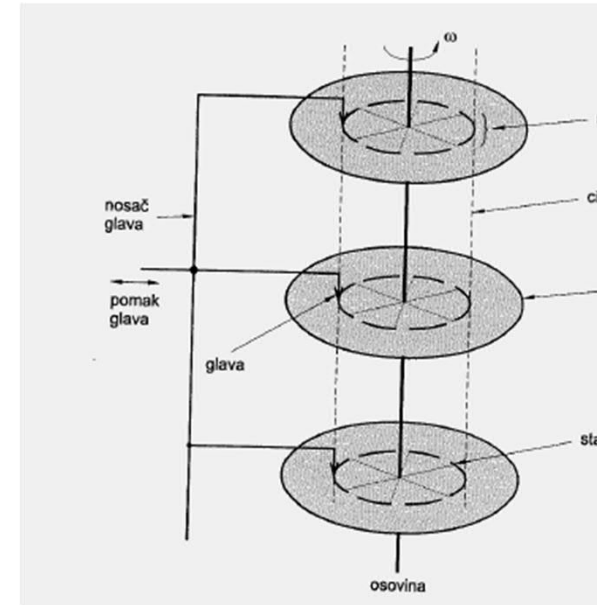
Slika 8.2. Programi pripremljeni za izvođenje

Operacijski sustavi

Spremnički prostor

•Disk

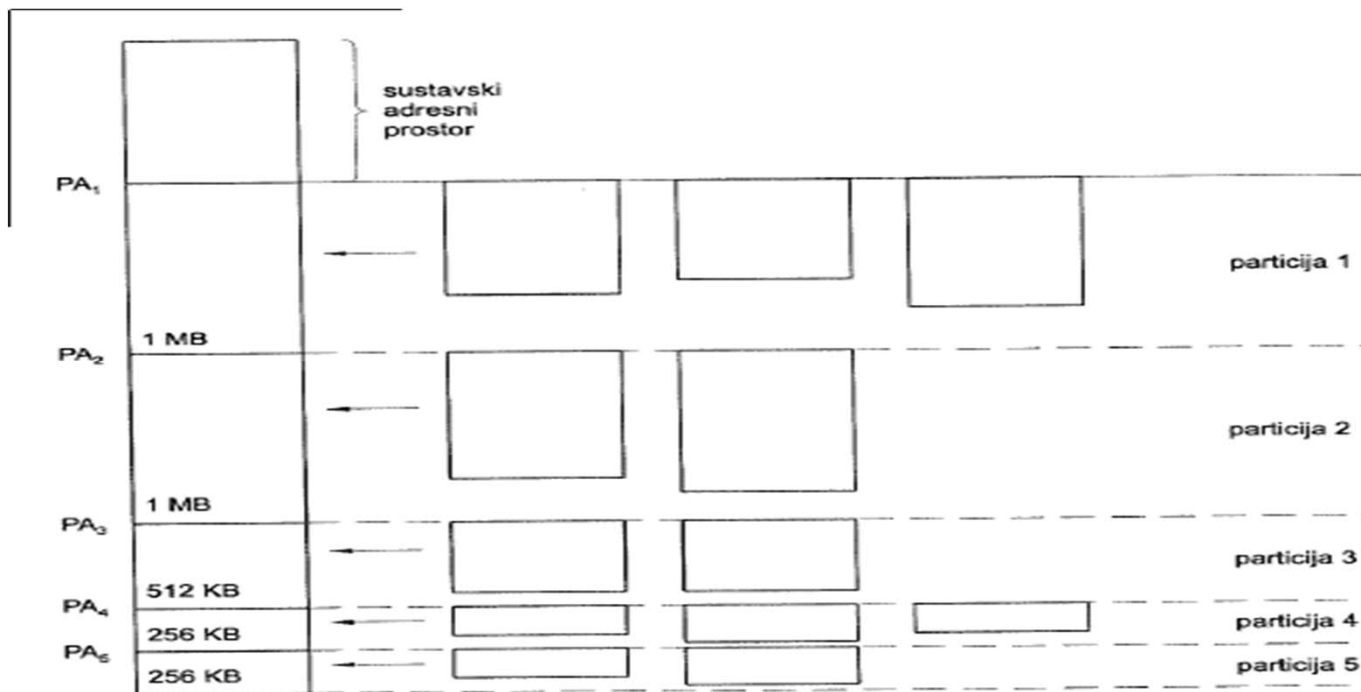
- trajanje postavljanje glave (*head positioning time*)
- trajanje traženje staze (*seek time*)
- rotacijsko kašnjenje (*rotational latency*)
- trajanje prijenos podataka (*data transfer time*)
- Disk se može koristiti kao dopunski spremnik radnom spremniku:
- swap* – zamjena programa



Operacijski sustavi

Spremnički prostor

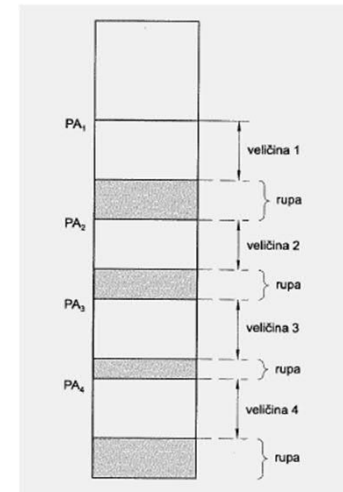
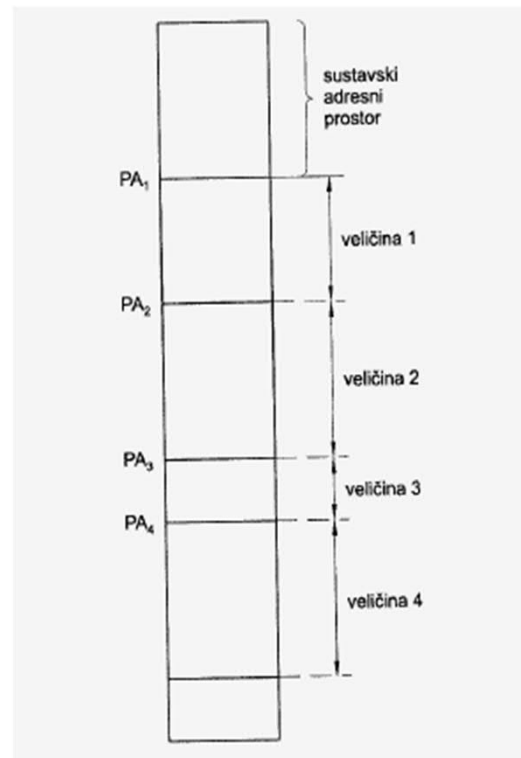
- Statičko dodjeljivanje radnog spremnika
- *fixed partitions*



Operacijski sustavi

Spremnički prostor

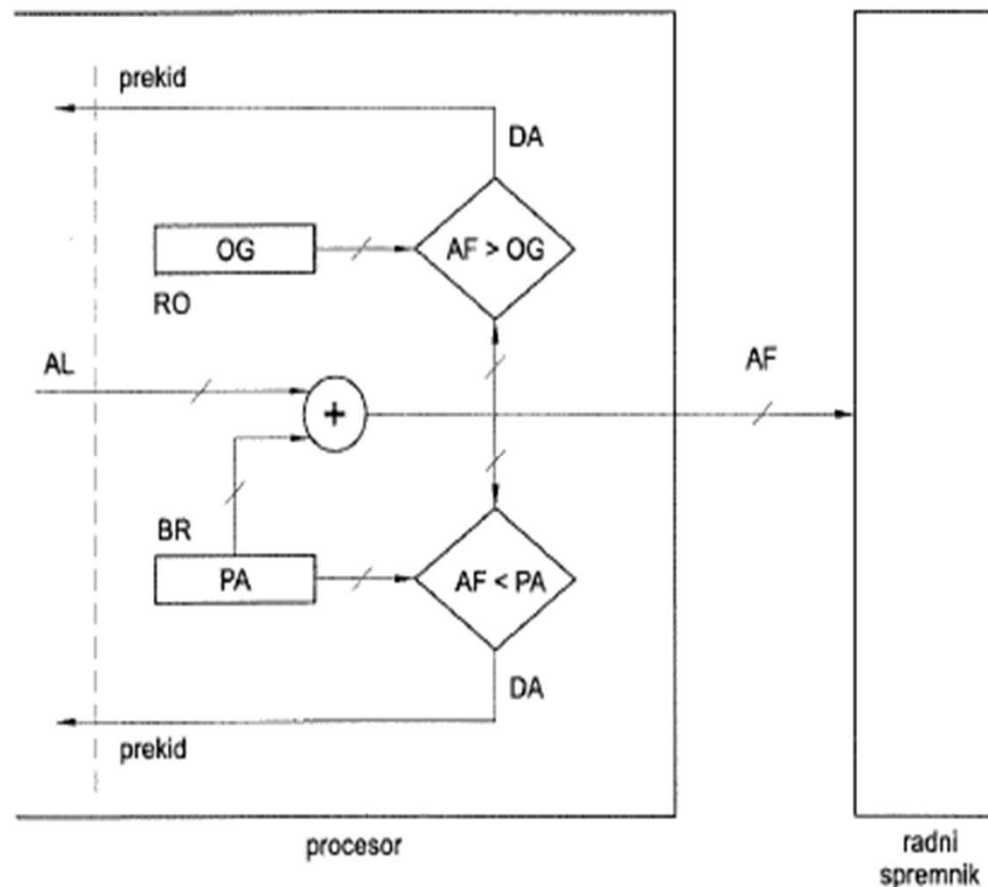
- *Dinamičko dodjeljivanje spremničkog prostora*
- Ograda – sprečava izlazak iz particije u "tuđi" adresni prostor
- Fragmentacija



Operacijski sustavi

Spremnički prostor

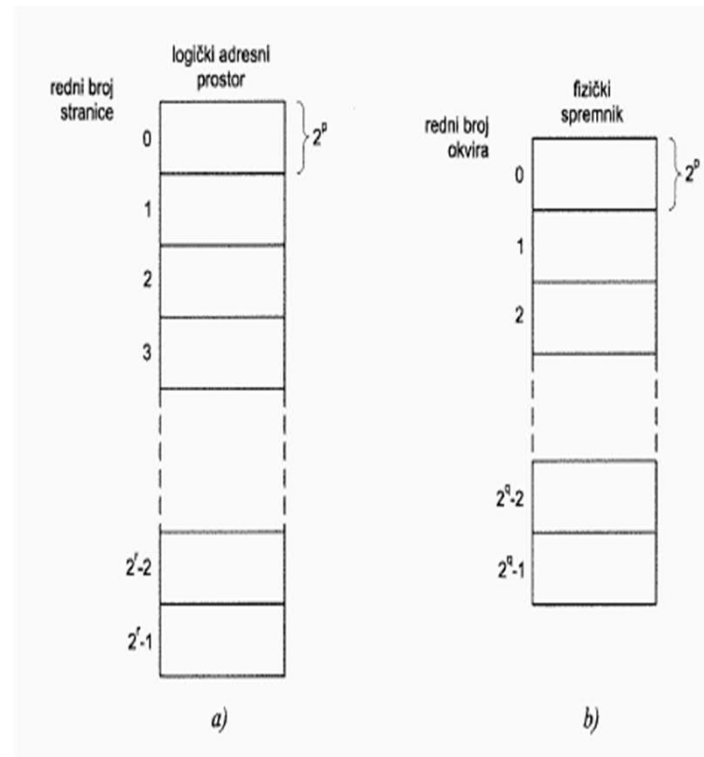
- *Dinamičko dodjeljivanje spremničkog prostora*
- Ograda – sprečava izlazak iz particije u "tuđi" adresni prostor
- Sklopovski realizirana pretvorba
- logičke adrese



Operacijski sustavi

Spremnički prostor

- *Straničenje / paging*
- Logički adresni prostor je podijeljen na *stranice*
- Fizički je spremnik podijeljen na *okvire*
- U *okvir* možemo smjestiti jednu *stranicu*

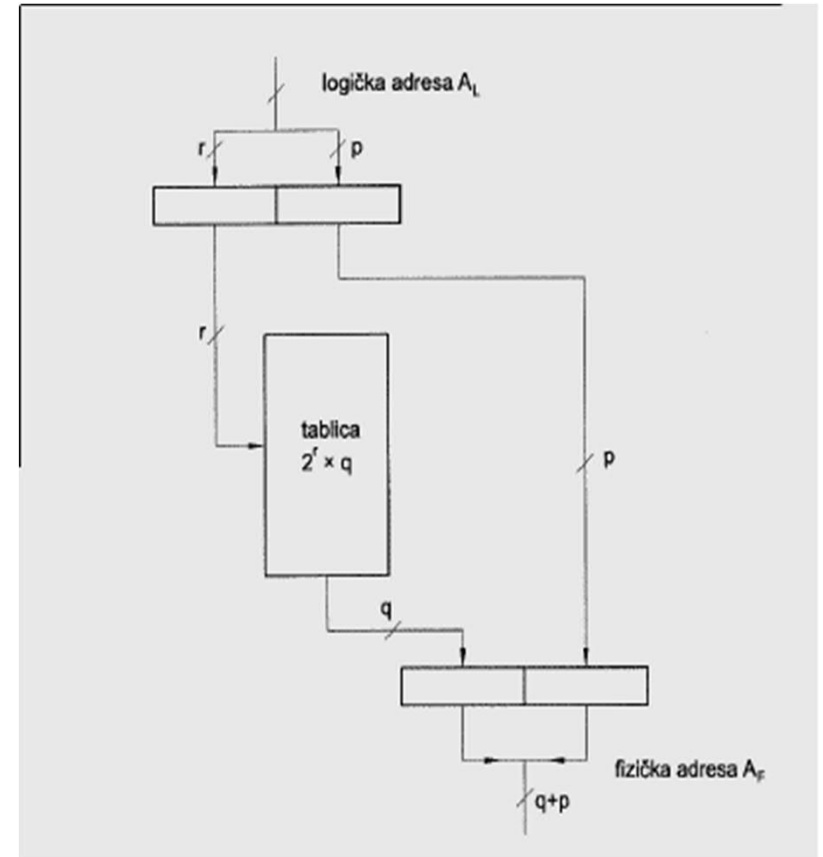


Operacijski sustavi

Spremnički prostor

• *Straničenje / paging*

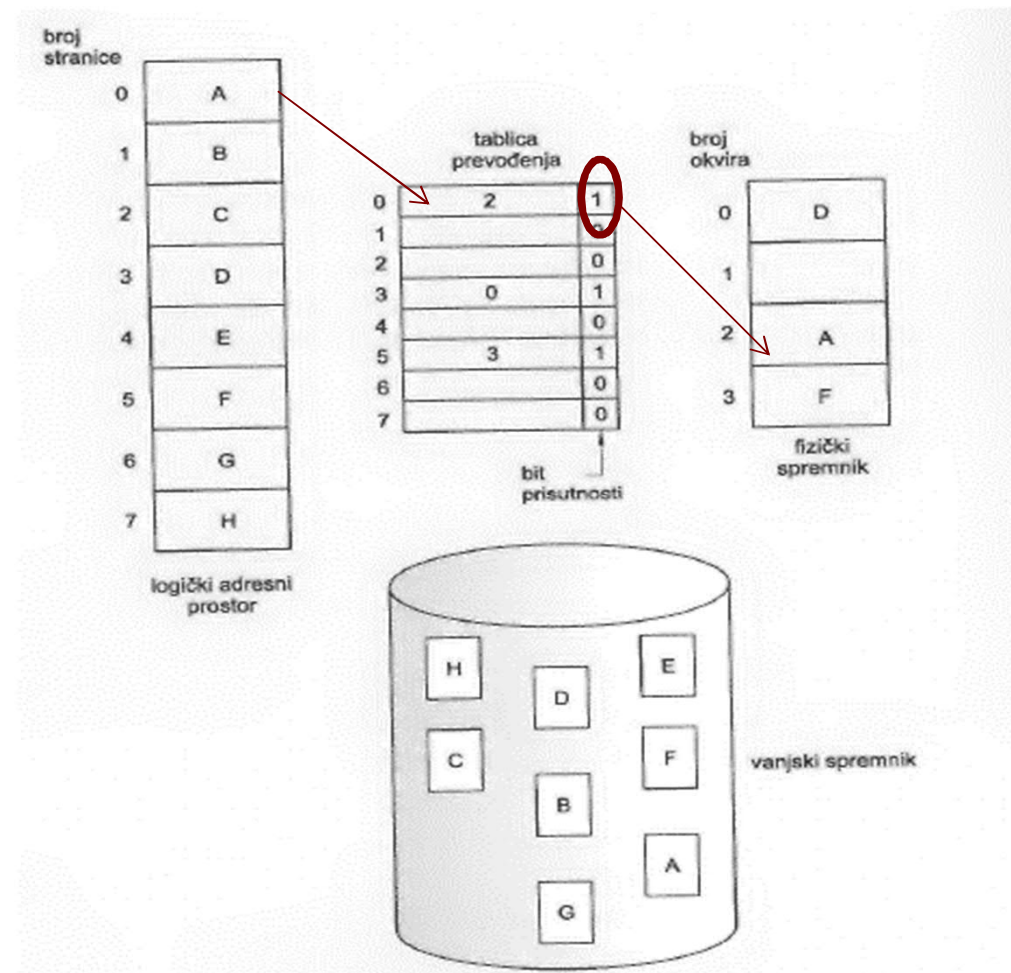
- logička adresa se dijeli na $r+p$ bitova
- p – adresira unutar pojedine stranice
- veličina stranice je 2^p
- r – adresira stanicu u logikom adresnom prostoru
- q – adresira okvir u fizičkom adresnom prostoru
- *bit prisutnosti* – označava da je stranica
- u okviru fizičkog spremnika



Operacijski sustavi

Spremnički prostor

.Straničenje / paging

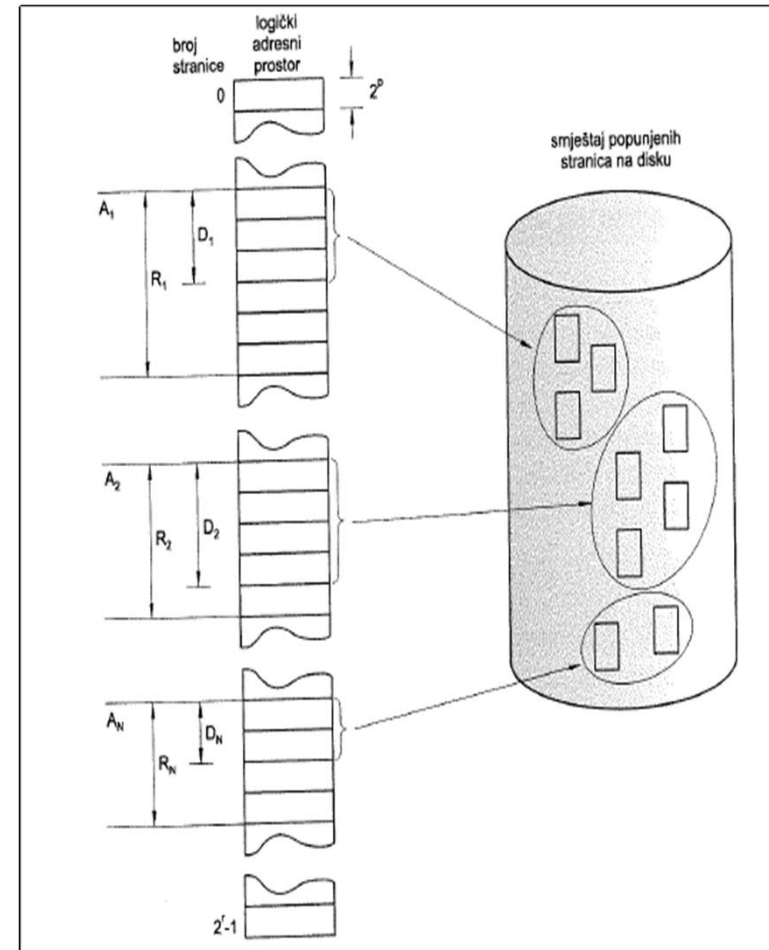


Operacijski sustavi

Spremnički prostor

•Straničenje / paging

- Opisnik virtualnog adresnog prostora:
- sadrži tablicu koja opisuje rezervirani adresni prostor:
- granice pojedinih područja rezerviranih adresa
- za popunjeni logički adresni prostor ima adresu na disku gdje je stranica spremljena
- sve stranice programa pripremljenog za izvođenje moraju biti smještene na disku, opisnik VAP sadrži za svaku stranicu jedan redak

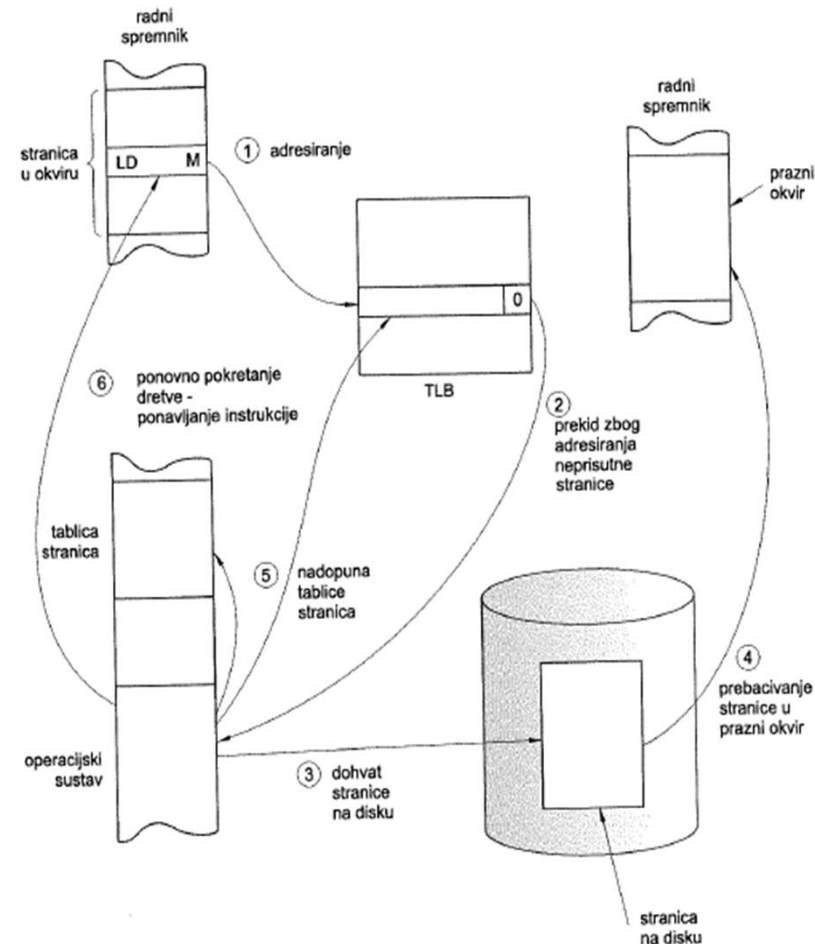


Operacijski sustavi

Spremnički prostor

• Straničenje / paging

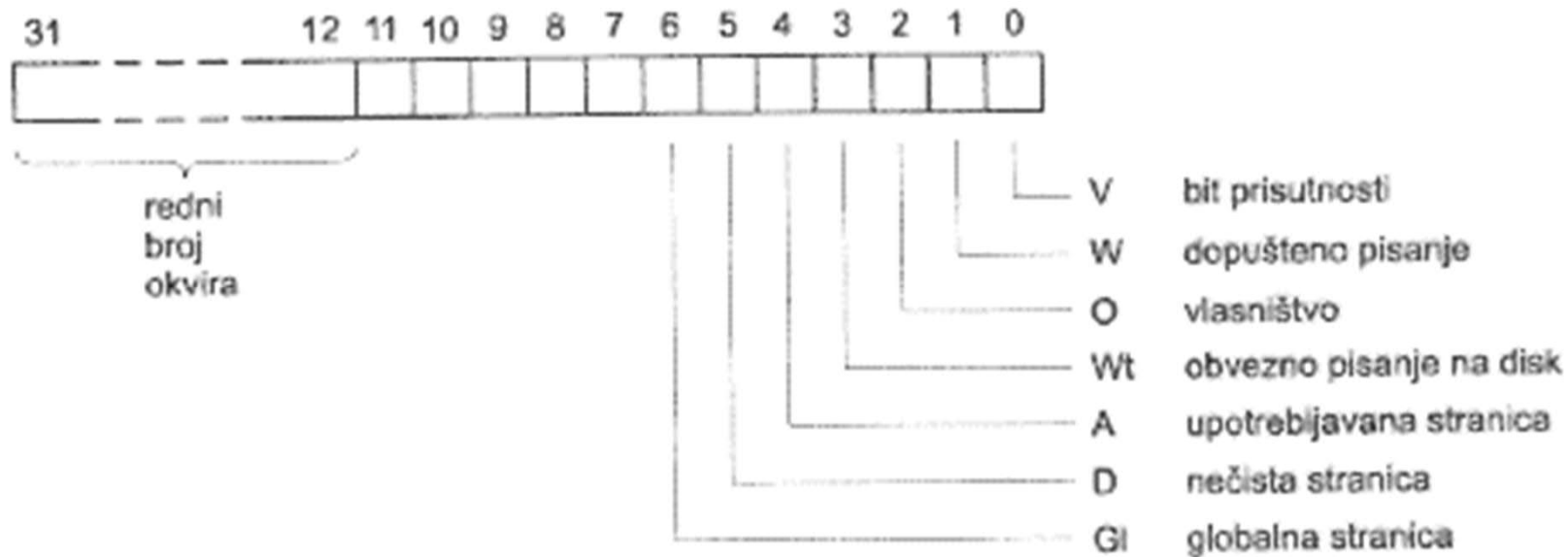
- na zahtjev!
- Prilikom pokretanja programa (->proces) uspostavlja se opisnik virtualnog adresnog prostora i rezervira se odgovarajući prostor
- U radnom je spremniku barem jedna stranica, ona na kojoj se nalaze početne instrukcije strojnog programa
- Ostale se stranice dobavljaju na zahtjev (*demand paging*)
- prekid zbog adresiranja nepostojeće stranice



Operacijski sustavi

Spremnički prostor

- *Straničenje / paging*
- čiste i nečiste stranice
- dodaju se bitovi u tablicu stranica



Operacijski sustavi

Spremnički prostor

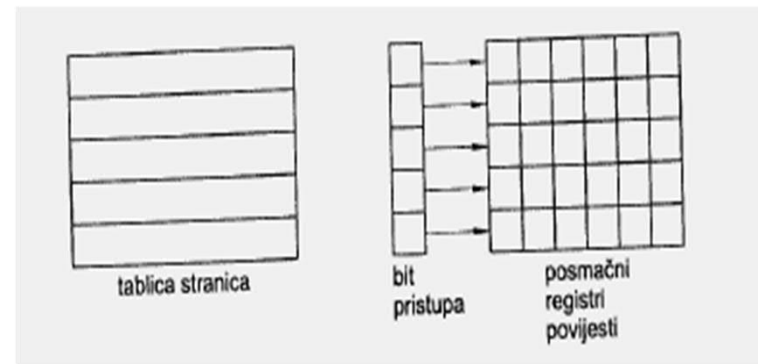
.Straničenje / *paging*

- .Teorijski algoritmi odabira stranice koja će biti izbačena:
- .FIFO strategija – ona koja je najdulje u radnom spremniku
- .LRU strategija (*least recently used*) – koja se najdulje nije upotrebljavala
- .OPT – optimalna strategija – ona koja se u najdaljoj budućnosti neće upotrebljavati
- .kvaliteta strategije -> mjerenje *page fault*
- .teoretski je najbolja OPT – praktički je neizvediva (teoretska mjera gornje granice)
- .I FIFO i LRU se mogu samo djelomično praktično primjenjivati

Operacijski sustavi

Spremnički prostor

- *Straničenje / paging*
- Praktične aproksimacije:
- Satni algoritam:



Tip stranice	Bit pristupa <i>A</i>	Bit čistoće <i>D</i>	Napomena
0	0	0	u prethodnoj periodi nije bilo pristupa i stranica je čista
1	0	1	u prethodnoj periodi nije bilo pristupa i stranica je nečista
2	1	0	u prethodnoj periodi bilo je pristupa i stranica je čista
3	1	1	u prethodnoj periodi bilo je pristupa i stranica je nečista

Operacijski sustavi

Spremnički prostor

• *Straničenje / paging*

• stanja okvira:

• aktivno,

• slobodno,

• slobodno sa obrisanim sadržajem

• radni skup / working set

