

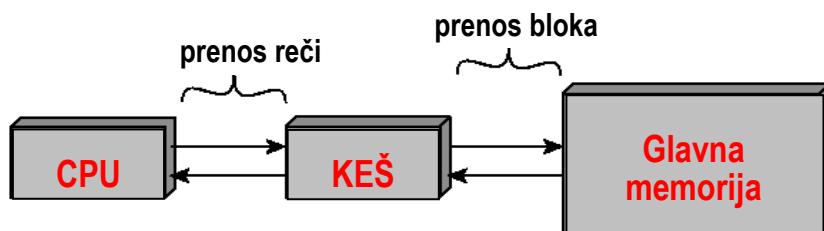
ARHITEKTURA I ORGANIZACIJA RAČUNARA

MEMORIJSKI SISTEM (3)

KEŠ MEMORIJA

KEŠ MEMORIJA

- Brze memorije malog kapaciteta
- Nalazi se između glavne memorije i CPU
- Može da bude u CPU čipu ili modulu
- Upravljane su tako da bi u svakom trenutku trebalo da sadrže aktuelne podatke pri radu procesora.



KEŠ MEMORIJA

* Procesor održava svoju brzinu zasnovanu na generatora takta samo kada se memorjske stavke koje on zahteva drže u keš memoriji.

* Ukupne performanse sistema strogo zavisi od proporcije pristupa memoriji koji se mogu zadovoljiti pomoću keša.

OBRAĆANJE KEŠU

* U nastojanju da pribavi potreban podatak, procesor se obraća keš memoriji. Ako je traženi podatak prisutan u keš memoriji, on se iz nje čita i dostavlja procesoru, koji nastavlja sa radom bez zastoja.

* Ako traženi podatak nije prisutan u keš memoriji, obraćanje procesora prosleđuje se glavnoj memoriji. Traženi podatak iz glavne memorije prenosi se u keš memoriju, a odatle u procesor.

PRESLIKAVANJE BLOKOVA PODATAKA

- * Između glavne memorije i keš memorije ne preslikavaju se pojedinačne reči već blokovi podataka.
- * Glavna memorija i keš memorija podeljene su na okvire blokova, dužina jednakih dužinama blokova podataka.
- * Blok podataka i uvek se nalazi u i-tom okviru bloka u glavnoj memoriji.

KEŠ BLOK

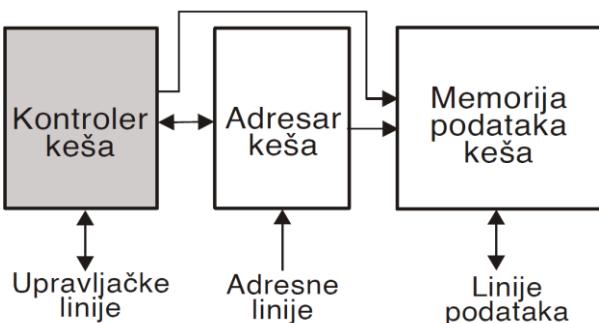
- * Veza između blokova podataka i okvira blokova u keš memoriji je složenija. Npr. okvir bloka keš memorije može sadržati jedan od većeg broja blokova podataka.
- * Npr, blok podataka i može se naći u okviru bloka j ili okviru bloka k keš memorije, ili i ne biti prisutan u njoj.
- * Okvir bloka – okvir bloka u keš memoriji ili keš blok.

STRUKTURA KEŠ MEMORIJE

- Neophodno je u keš memoriji imati informacije o tome koji su blokovi podataka prisutni u njoj.
- Ovo upućuje da se keš memorija mora sastojati od memorije podatka i adresara (direktorijuma).
- U memoriji podatka keša čuvaju se blokovi podataka prisutni u keš memoriji.
- Adresar keša ima po jednu stavku za svaki keš blok. Stavka adresara sadrži informacije o bloku podataka prisutnom u keš bloku kome je ona pridružena.

STRUKTURA KEŠ MEMORIJE

- Stavke adresara najčešće sačinjavaju:
 - Etiketa – viši deo adrese bloka podataka prisutnog u keš bloku
 - V – indikator važeći blok podatka prisutan u keš bloku
 - M – indikator modifikovan blok podatka u keš bloku.



ISKAZIVANjE VELIČINA U BLOKOVIMA

* Blokovi podataka, kao i okviri blokova odnosno keš blokovi, su dužine $B = 2^b$ [B]

* Glavna memorija (GM) je kapaciteta $G = 2^g$ [B]. Iskazan blokovima, njen kapacitet je $G_B = G/B = 2^{g-b}$ [blokova]

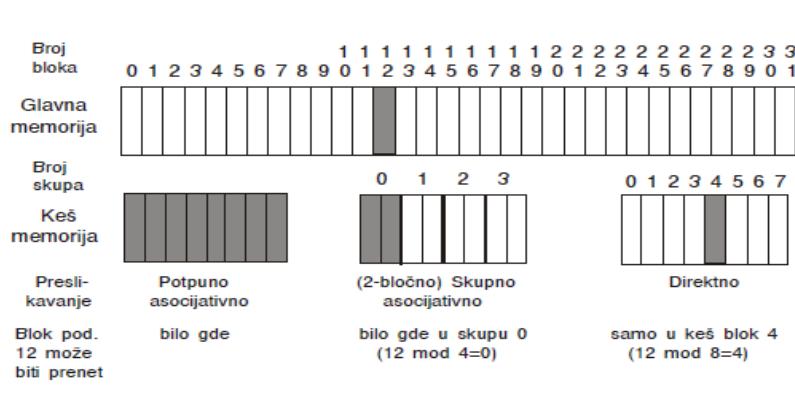
* Keš memorija KM je kapaciteta $C = 2^c$ [B] Iskazan blokovima, njen kapacitet je $C_B = C/B = 2^{c-b}$ [blokova]

SMJEŠTANjE I TRAŽENjE PODATAKA U KEŠ MEMORIJI

* Prema broju keš blokova u koje se može preneti blok podataka i iz glavne memorije, $i \in \{0, 1, \dots, G_B - 1\}$, keš memorija može biti sa:

- potpunim asocijativnim preslikavanjem,
- skupno-asocijativnim preslikavanjem, i
- direktnim preslikavanjem.

Preslikavanje blokova podataka iz glavne memorije u sva tri tipa keš memorija



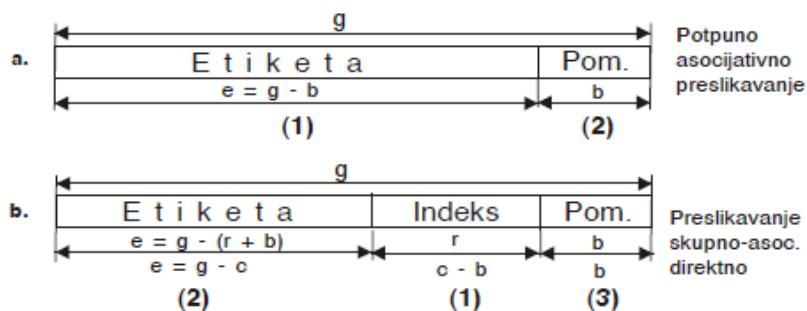
Krajnosti u preslikavanju

- * Kod keš memorije sa **potpunim asocijativnim preslikavanjem**, blok podataka i može biti prenesen u **bilo koji keš blok k, $k \in \{0, 1, \dots, C_B - 1\}$**
- * Kod keš memorije sa **direktnim preslikavanjem**, blok podataka i može biti prenesen samo u keš blok k određen izrazom $k = i \bmod C_B$.

Skupno-asocijativna keš memorija

- Kod keš memorije sa skupno-asocijativnim preslikavanjem, po A keš blokova obrazuje skup keš blokova S_j , $j = 0, 1, \dots, s-1$. $s = 2^r = C_B$ /A je broj skupova keš blokova u keš memoriji.
- Blok podataka i može biti prenesen u bilo koji od A keš blokova u skupu keš blokova S_k , $k = i \bmod s$.
- Broj keš blokova A u skupu naziva se asocijativnost skupa. Za keš memoriju sa skupno asocijativnim preslikavanjem, sa A keš blokova po skupu, kaže se da je A-bločna skupno-asocijativna keš memorija

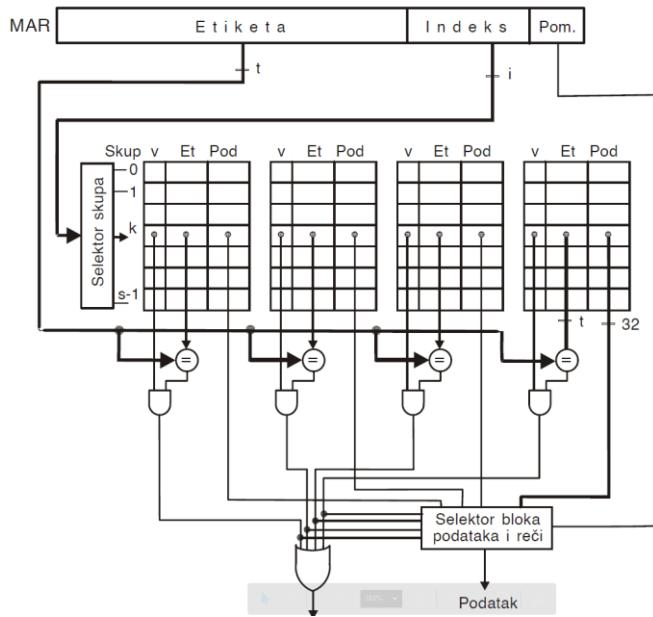
Adresna polja keš memorija



Opšti oblik keša

- **Način rada keš memorije sa skupno-asocijativnim preslikavanjem reprezentativan je i za ostala dva oblika keš memorija.**
- **Skupno-asocijativno preslikavanje prelazi u direktno preslikavanje kada skup spadne na samo jedan keš blok ($A=1, s=C_B$).**
- **Skupno-asocijativno preslikavanje prelazi u potpuno asocijativno preslikavanje kada jedan skup obuhvati sve blokove keš memorije ($A=C_B, s= 1$).**

Adresna polja keš memorija



VARIJANTE KEŠ MEMORIJA

- Glavna memorija računara sadrži programe i podatke (operande), koji se mogu prenosi u keš memoriju i čuvati u njoj.
- Objedinjeni (jedinstveni) keš je onaj u kome se zajedno čuvaju programi i podaci (Princeton-ska arhitektura računara).
- Keš instrukcija i keš podataka odvojeno čuvaju programe i podatke (Harvard arhitektura).
- Odvojeni keševi za instrukcije i podatke široko su zastupljeni u savremenim računarima, jer omogućuju istovremeni pristup i jednom i drugom kešu, čime udvajaju širinu opsega keš memorija.

OBJEDINJENA I ODVOJENA KEŠ MEMORIJA

- **Prednosti objedinjenih keš memorija :**
 - one su sposobne da bolje uravnoteže opterećenje između donošenja instrukcija i podataka, zavisno od izvršenja programa;
 - jeftinija konstrukcija i implementacija.
- **Предности одвојених кеш меморија (Харвард архитектуре):**
 - nadmetanje za keš između jedinica za obradu i izvršenje instrukcija je eliminisano ⇒ donošenje instrukcije može da se obavlja paralelno sa pristupanjem izvršne jedinice memoriji.

VIŠE NIVOA KEŠA

- * Primarna keš memorija radi pri taktnoj učestanosti procesora.
- * Sekundarna keš memorija znatno većeg kapaciteta (nekoliko stotina KB do nekoliko MB), i nekoliko puta sporija od primarne keš memorije, smešta se u čipu procesora ili van njega između primarne keš memorije i glavne memorije.
- * Tako, npr, procesori Intel P4 i AMD Athlon HP imaju sekundarne keš memorije kapaciteta 512 KB u čipu procesora.

ZAMENA BLOKOVA PODATAKA U KEŠ MEMORIJI

- * Pri promašaju podatka sa adresom a u keš memoriji, on se u nju mora preneti zajedno sa blokom podataka i kome pripada.
- * Kod keš memorije sa direktnim preslikavanjem, blok podataka i može se preneti samo u keš blok $k = i \ mod \ C_B$.
- * Kod keš memorija sa skupno-asocijativnim i potpunim asocijativnim preslikavanjem, blok podataka i može se prenijeti u jedan od više keš blokova. Neka su to keš blokovi j_1, j_2, \dots, j_h .

Izbor bloka za zamenu

- * Izbor bloka za zamenu može uticati na buduće promašaje. Zato se kao cilj postavlja izbor keš bloka j_z , $1 \leq z \leq h$, u kome će zamena bloka podataka dati najmanji broj budućih promašaja.
- * Pravilo po kome se bira blok podataka za zamenu novim blokom, odnosno keš blok koji ga sadrži, zvaćemo strategija (politika) zamene bloka.

Algoritmi zamene

- Kada novi blok treba da se smesti u keš memoriju, blok koji je uskladišten u jednom od redova keša treba da se zameni.
- Kod direktnog preslikavanja ovde nema nikakvog izbora.
- Kod potpuno asocijativnog, ili asocijativnog preslikavanja skupova, potreban je algoritam za zamenu, da bi se odredilo koji blok da se zameni (i, implicitno, u koji red keša da se stavi blok).
 - Kod asocijativnog preslikavanja skupova, redovi kandidati su oni u izabranom skupu;
 - kod potpuno asocijativnog preslikavanja, svi redovi u kešu su mogući kandidati.

Algoritmi zamene

- Keš bloka jz u kome će blok podatka biti zamjenjen novim, bira se prema jednoj od sledećih strategija zamene
- Slučajna zamena (slučajni izbor - RAND):
 - Jedan od redova kandidata se bira na slučaj.
 - Sve druge politike zasnivaju se na informacijama koje se odnose na istoriju upotabe blokova u kešu.
- Najmanje skoro upotrebljen (LRU):
 - Bira se onaj keš blok koji sadrži najmanje skoro korišćeni blok podataka.
- Prvi-unutra-prvi-napolje (FIFO):
 - Bira se onaj keš blok koji je najduže bio u kešu tj. prvi napunjen.
- Najređe korišćen (LFU):
 - Bira se onaj red kandidat u kome je blok koji je bio najređe referenciran.

Optimalna strategija zamene

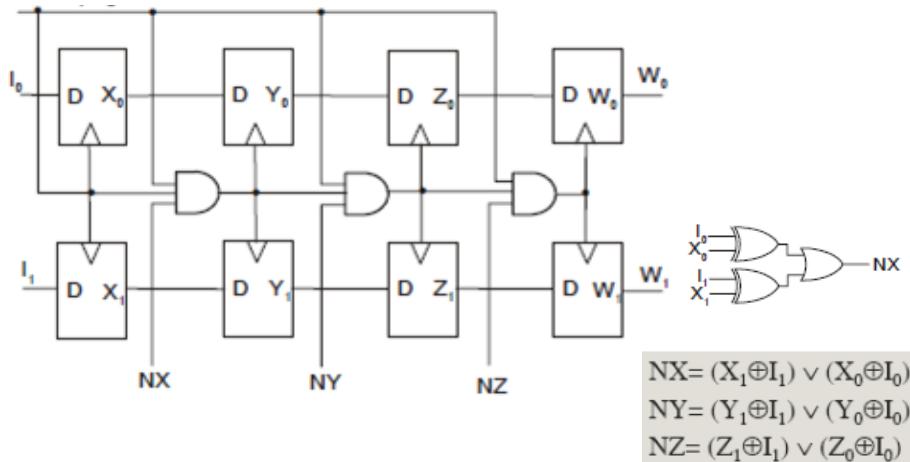
- Strategija zamjene blokova podataka u keš memoriji koja bi dala najbolje rezultate zahteva poznavanje programa u budućnosti.
- Ovakvu optimalnu strategiju zamene (OPT) predložio je L. Belady 1966. godine u radu koji se odnosio na virtuelne memorije.
- Primjenjena na keš memorije, može se formulisati ovako: za zamenu se bira blok podataka koji će od svih kandidata za zamjenu biti korišćen u najdaljoj budućnosti.

LRU stek

- LRU algoritam zamene može se implementirati sledećim hardverskim rešenjem, koje je deo kontrolera keša.
Nazovimo ovaj deo kontrolera keša blok izbora žrtvovanog bloka podataka.
- Po jedan LRU stek pridružuje se svakom skupu keš blokova. LRU stek sadrži po jedan stepen za svaki keš blok u skupu keš blokova, dakle A stepena.
- Svaki stepen sadrži k flip-flopova tipa D, gdje je k određeno izrazom $k = \lceil \log_2 A \rceil$, tako da svojim stanjem adresira jedan od A keš blokova u svom skupu keš blokova.

LRU stek

Такт погодака



Promena sadržaja LRU steka

- * Pri pogotku aktivira se takt pogodaka, i na ulaze I_1 i I_0 vodi se oznaka keš bloka kome se u slučaju pogotka obraća CP.
- * Prostiranjem takta pogodaka do stepena Y, Z i W upravlja se preko ulaza NX, NY i NZ respektivno.

$$\begin{aligned}NX &= (X_1 \oplus I_1) \vee (X_0 \oplus I_0) \\NY &= (Y_1 \oplus I_1) \vee (Y_0 \oplus I_0) \\NZ &= (Z_1 \oplus I_1) \vee (Z_0 \oplus I_0)\end{aligned}$$

Promena sadržaja LRU steka

- Signali na ovim ulazima formiraju se tako da obezbede da se u slučaju pogotka, recimo u keš blok adresiran stanjem stepena Z, zatečena stanja stepena X i Y prenose u stepene Y i Z respektivno, a stanje stepena Z prenosi se u stepen X.

**if ($I_1I_0 = Z_1Z_0$) then ($Z_1Z_0 \leftarrow Y_1Y_0$),
 $(Y_1Y_0 \leftarrow X_1X_0)$, $(X_1X_0 \leftarrow I_1I_0)$.**

-/**stanje stepena W ostaje nepromenjeno/**