

Arhitektura računara. Tipovi instrukcija. Makro naredbe. Memorije.

ARHITEKTURA RAČUNARA

Današnji procesori uglavnom izvršavaju instrukcije po **fon Nojmanovoj arhitekturi**. Ovaj način podrazumeva pripremu i izvršavanje instrukcija programa koji je upućen u operativnu memoriju. Skupovi instrukcija sadrže više desetina ili više stotina instrukcija i to različitih. Osnovna karakteristika većine instrukcija je izvršavanje operacija nad operandima od kojih se jedan poziva iz operativne memorije a drugi se nalazi u akumulatoru. Računari sa ovako realizovanim skupom instrukcija nazivaju se **CISC računari** (računari sa kompleksnim skupom instrukcija). Pored CISC računara, u novije vreme se realizuje i novija arhitektura, tj. **RISC računari** (računari sa redukovanim skupom instrukcija).

Osnovni razlog zašto je došlo do redukovanja skupa instrukcija je jedno pravilo koje je uočeno u toku rada računara: *“Preko 80% vremena izvršenja bilo kog programa troši se na izvršenje instrukcija čiji broj ne prelazi 20% ukupnog broja instrukcija procesora”*. Rezultat ovog pravila je RISC arhitektura koja se karakteriše sledećim svojstvima:

1. izvršavanje najvećeg broja instrukcija za samo jedan takt
2. većina instrukcija obrađuje podatke iz registra procesora i smešta ili vraća ih nazad u isti taj registar
3. obraćanje operativnoj memoriji vrši se po instrukcijama LOAD i STORE
4. postoji više različitih adresiranja
5. format instrukcija je relativno prost i čist
6. postoji mogućnost pročitane obrade instrukcija
7. postoji veliki broj registara u procesoru

TIPOVI INSTRUKCIJA

Broj i vrsta instrukcija realizovanih u nekom računaru zavisi od namene i veličine procesora. Kod procesora iz familije I8086 imamo sledeće instrukcije:

1. **instrukcije za prenos podataka**
2. **aritmetičko-logičke instrukcije** (aritmetičke instrukcije, logičke instrukcije, instrukcije pomeranja i kruženja)
3. **instrukcije zaupravljanje tokom izvršenja** (instrukcije granjanja, instrukcije za preskoke, instrukcije za potprograme, instrukcije za programske prekide)
4. **ulazno-izlazne instrukcije** i
5. **ostale instrukcije.**

Skupom mašinskih instrukcija u procesoru realizuju se određene elementarne operacije koje se klasifikuju prema sledećim parametrima: **prema broju binarnih pozicija operanda** i **prema broju operanada** nad kojima se primenjuje operacija.

Prema broju razreda u operandima operacije mogu biti: **jednorazredne** i **višerazredne**. Jednorazredne operacije su proste i primenjuju se na dva bita istog razreda operand. Pri višerazrednim operacijama istovremeno se izvršava operacija nad svim razredima operanada.

Instrukcije za prenos podataka:

Instrukcijama za prenos podataka vrši se: **pozivanje sadržaja memorije u akumulator ili u neki drugi registar procesora, slanje sadržaja akumulatora ili nekog registra u memoriju, slanje iz jednog u drugi registar procesora, slanje iz jedne memorijske lokacije u drugu, i slično...**

Aritmetičke instrukcije:

Služe za izvršenje osnovnih aritmetičkih operacija: **sabiranje, oduzimanje, množenje i deljenje**, zatim upoređenje dva sadržaja, komplementiranje, povećanje ili smanjenje sadržaja za neku zadatu vrednost.

Logičke instrukcije:

Instrukcije za logičke operacije realizuju prekidačke operacije (bullove algebre): AND, OR, XOR, NOT.

Instrukcije pomeranja i kruženja:

Instrukcijama za pomeranje vrši se pomeranje binarnog sadržaja nekog registra procesora za određeni broj pozicija levo ili desno. Pomeranje može biti kružno ili ne.

Instrukcije za upravljanje tokom izvršenja:

Instrukcije za upravljanje tokom izvršenja programa omogućavaju izvršavanje nekog određenog dela programa u zavisnosti od dobijenog uslova u toku rada.

Instrukcije za rad sa podprograma:

Podprogram je deo programa koji se može samostalno izvršavati, a organizovan je tako da se njemu može preneti upravljanje iz nekog drugog programa i da se po završetku upravljanja vrati nazad u isti taj program. Za rad sa podprogramima se najčešće zadavaju dve instrukcije i to: jedna za pozivanje podprograma a druga za vraćanje iz podprograma.

Ulazno-izlazne instrukcije:

Služe za prenos podataka iz procesora na neki periferni uređaj ili za čitanje sa perifernog uređaja na procesor, takođe i za ispitivanje, brisanje ili postavljanje indikatora stanja na periferni uređaj.

Ostale instrukcije:

U grupu ostale instrukcije su uključene instrukcije koje se ne svrstati ni u jednu od opisanih grupa. Tu spadaju, na primer: instrukcije bez dejstva, zabrana ili dozvola programskog prekida.... i slično.

INSTRUKCIJE PREKIDA

Instrukcije prekida su karakteristične za mašinske jezike. Izvršavanjem ove instrukcije odvijaju se sledeći procesi:

- sadržaj registra uslova prepisuje se na stek
- kontrolni bitovi registra uslova (IF i TF) se postavljaju na nulu da bi se sprečilo prihvatanje zahteva nekog drugog prekida.
- sadržaj registra CS se prepisuje na stek
- na osnovu broja prekida formira se adresa vektora prekida
- sadržaj brojača instrukcije prepisuje se na stek
- prva reč vektora prekida upisuje se u brojač i instrukcije
- prelazi se na izvršenje

MAKRO NAREDBE

Makro naredba predstavlja simboličko ime izvornog jezika kojom je označen unapred definisan niz mašinskih instrukcija. **Makro naredba predstavlja otvoreni podprogram i pri njenom pozivu u osnovni program umeće se deo niza mašinskih instrukcija.**

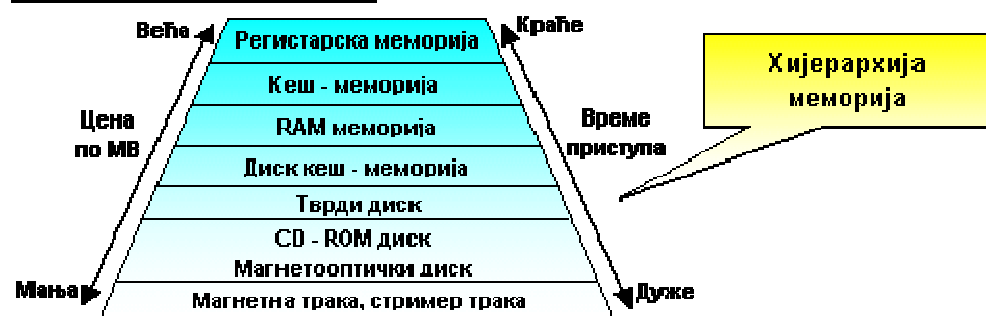
Delovi programa koji se često koriste definišu se samo jednom u obliku mikronaredbe koja ima svoje ime i parametre. Osnovna namena mikronaredbi je skraćenje izvornog programa.

MEMORIJE

Memoriju čine uređaji koji obezbeđuju zapisivanje binarnih podataka. Memorija se koristi za upis, pamćenje i čitanje podataka i programa. Operacija upisa i operacija čitanja nazivaju se *pristup memoriji*. Za opis karakteristike memorije koriste se više parametara, od kojih su osnovni sledeći:

1. **Kapacitet memorije** (broj bajtova ili bitova koji se mogu zapamtiti u memoriju)
2. **Vreme pristupa ili kašnjenje** (vremenski interval koji protekne od dovođenja signala za definisanje pristupa do završetka upisa ili čitanja)
3. **Ciklus pristupa li memorijski ciklus** (minimalni vremenski interval između dva uzastopna pristupa memorije)
4. **Jedinica prenosa** (osnovna jedinica prenosa podataka je memorijska reč, a ako se istovremeno prenosi više memorijskih reči onda one čine blok, kod hard diskova jedinica prenosa sektor)
5. **Brzina prenosa podataka** (broj bitova, bajtova ili memorijskih reči koje uređaj može prenositi u jednoj sekundi)
6. **Cena jednog bita memorija** (odnosno kvalitet i cena memorije).

HIJERARHIJA MEMORIJA



U računaru se koriste sledeće memorije:

- registarska memorija
- keš memorija
- operativna memorija
- keš-memorija diska
- spoljna memorija direktnim pristupom
- spoljna memorija sa sekvencijalnim pristupom

Ovakav redosled memorija odgovara u smanjenju nivoa u hijerarhiju (manje brzine i cene po jednom bitu a povećanje kapaciteta). Deo operativnog sistema koja upravlja tokovima podataka između memorijskih uređaja naziva se *sistem za upravljanje memorijom*.

REGISTARSKA MEMORIJA

Ona predstavlja skup registara u procesoru koji se nazivaju *registri opšte namene*. Registri opšte namene

se koriste u različite svrhe, a najčešće za privremeno pamćenje operanada, međurezultata, komponenata adresa... Registarska memorija se realizuje poluprovodničkim memorijskim elementima (bistabilnim kolima ili flip-flopovima) i ima brzinu istu kao i centralni procesor.

U radu registarska memorija koristi adrese na sličan način kao i operativna memorija. Za te svrhe se koristi registarsko adresiranje.

KEŠ MEMORIJA

Zbog razlike brzine procesora i operativne memorije procesoru se dodaje keš memorija.

Keš je memorija mala, ultrabrza popluprovodnička memorija sa neposrednim pristupom. U keš memoriji se čuvaju podaci i instrukcije iz operativne memorije koje procesor trenutno koristi.

Keš memorija i operativna memorija su podeljene na jedinice koje se zovu blokovi, pa se prenos između operativne i keš memorije vrši pomoću blokova. Veličina bloka obično iznosi između 4 i 128 bajtova. Jedinica prenosa podataka između keš memorije i procesora je fizička reč.

Kada centralni procesor generiše adresu memorijske lokacije formira se upravljački signal za pristup keš memorije.

Ukoliko se podatak sa traženom adresom nalazi u keš memoriji on se prenosi u procesor radi obrade ili se realizuje novom vrednošću iz procesora koja predstavlja rezultat obrade.

Ukoliko u keš memoriji nema bloka sa traženom adresom, tada se aktivira procedura kojom se iz keš memorije šalje jedan blok u operativnu memoriju, a na njihovo mesto se iz operativne memorije poziva traženi blok.

Keš memorije funkcionišu na bazi lokalnosti ponašanja kojom se karakteriše većina programa. Postoje tri principa koji uključuju lokalnost: prostorna lokalnost, vremenska lokalnost i uzastopnost.

Problemi kod korišćenja keš memorije

Pri korišćenju keš memorije treba rešiti sledeća 4 osnovna problema:

1. **Smeštanje blokova**
2. **Identifikacija bloka**
3. **Zamena blokova**
4. **Strategija upisa**

Najbolji način za smeštanje blokova je tzv **asocijativno preslikavanje**, ali u tom slučaju zbog identifikacije blokova, kao keš memorija treba koristiti asocijativnu memoriju. Za zamenu blokova postoji više strategija ali se najčešće zamenjuje najranije korišćeni blok ili blok koji je najduže prisutan u keš memoriji.

Kada se blok u keš memoriji modifikuje upisom novog podatka iz procesora, postoje dva načina koji se koriste za ažuriranje operativne memorije:

- 1) **neposredni upis** – istovremeno se podatak upisuje u keš-memoriju i u operativnu memoriju
- 2) **posredni pristup** – podatak se upisuje samo u blok keš-memorije, a modifikovani samo u blok keš memorije, a modifikovani blok se vraća u operativnu memoriju tek kada se zamenjuje drugim blokom.

Faktor podataka ‘h’ je parametar kojim se meri efikasnost keš memorije. To je verovatnoća da se sadržaj memorijske lokacije koju adresira procesor nalazi u keš memoriji. Veličina ‘1-h’ je faktor promašaja.

OPERATIVNA MEMORIJA

Operativna memorija je adresivna memorija koja služi za čuvanje podataka koji se koriste u procesu

izvršenja mašinskih instrukcija u procesoru. Karakteristike operativne memorije direktno utiču na brzinu računarskog sistema.

Operativna memorija se realizuje bistabilnim kolima i sastoji se od memorijskih lokacija. Svaka memorijska lokacija ima svoju adresu. Podaci se u njoj čuvaju u obliku grupe bitova koje nazivamo memorijska reč.

Postoje *statičke i dinamičke* poluprovodničke memorije sa direktnim pristupom.

Statička memorija kao memorijske elemente koristi poluprovodnička bistabilna kola u MOS tehnologiji. Dinamička memorija kao memorijske elemente koristi spregu kondezatora i tranzistora realizovanih na integrisanom kolu u MOS tehnologiji kao celina. Dinamičke memorije jedan bit podataka pamte kao veću ili manju količinu naelektrisanja na PN spoju tranzistora. Ovo naelektrisanje ima ograničeno vreme zadržavanja pa se zato vrši periodično osvežavanje (otprilike svakih 2ms).

KEŠ-MEMORIJA DISKA

Keš memorija diska (keš-bafer diska) ima sličnu ulogu kao keš memorija u sistemu keš memorija – glavna memorija. To je memorijski bafer koji pamti poslednje korišćene delove adresnog prostora diska, smanjujući tako zahtevani broj pristupa i efektivno vreme pristupa disku, odnosno povećavajući efektivnu brzinu diska. Procesor uvek pristupa podacima predhodno smešteni u keš-memoriju diska kad god mu zatrebaju.

Keš memorija diska je veličine od jednog do više megabajta. Jedinica prenosa između diska i keš-memorije diska obično sadrži jednu ili više staza diska. Keš memorija diska može se pridružiti ulazno-izlaznom uređaju, kontroleru tog uređaja ili operativnoj memoriji procesora. Ispitivanja su pokazala da je efikasnost keš memorije diska najbolja kada je ona uključena kao deo operativne memorije.

STEK MEMORIJA

Stek memorija ili magacinska memorija **jeste bezadresna registarska memorija sa sekvencijalnim pristupom**. Kod ove memorije registri formiraju jednodimenzionalni niz u kome su susedni registri povezani kolima za paralelni prenos binarnih reči tako da se u njima upis i čitanje vrši po principu „poslednji upisan – prvi pročitani” (eng. LIFO – Last In – First Out).

Upis u stek i čitanje iz steka vrši se samo na jednom mestu koji se naziva vrh steka. Operacija upisa naziva se PUSH, dok se operacija čitanja naziva POP. Stek se najčešće koristi za izračunavanje vrednosti aritmetičkih izraza.

Memorija tipa reda jeste registarska memorija slična steku, samo se podaci u nju upisuju i iz nje čitaju po principu „prvi upisan – prvi pročitani” (eng. FIFO – First In – First Out).

ČITAČKE MEMORIJE (ROM)

To su memorije sa neposrednim pristupom koje omogućavaju da se jednom uposani podaci mogu samo čitati, dok upis kod njih nije moguć. Čitačke memorije su najčešće poluprovodničke memorije ali njihov sadržaj ne zavisi od napajanja pa ne gube svoj sadržaj. Čitačke memorije se koriste za programe i podatke koji se ne menjaju često.

Najpoznatija čitačka memorija kod računara je BIOS.

Postoji 4 tipa čitačkih memorija:

1. Prvu grupu čine **najprostice čitačke memorije** kod kojih se podaci upisuju još u procesu proizvodnje poluprovodničkih integrisanih kola. Što znači da se ove memorije mogu programirati samo jednom – i to

u fabrici.

2. Drugu grupu čine *programabilne čitačke memorije* koje omogućavaju programiranje samo jednom, i to radni korisnik, gde se jednom upisani podaci više ne mogu menjati.

3. Treću grupu čine *reprogramabilne memorije* koje omogućavaju da posebnim postupkom korisnik može da programira i može da vrši izmenu tih podataka nakon programiranja.

4. I četvrtu grupu čine *električno-reprogramabilne čitačke memorije* koje programiranje i izmena sadržaja mogu vršiti električnim putem. Za izmenu sadržaja ovih memorija nisu potrebni nikakvi dodatni uređaji.

ASOCIJATIVNE MEMORIJE

Vrlo često je potrebno pronaći podatke u operativnoj memoriji u toku neke obrade. Procedura traženja se sastoji od poređenja zapamćenih podataka sa zadatom vrednošću.

Broj pristupa operativnoj memoriji zavisi od mesta traženja i algoritama po kome se vrši istraživanje.

Potrebno vreme za pronalaženje i broj pristupa operativnoj memoriji se mogu znatno skratiti ako zapamćeni podaci mogu prelaziti ne samo na osnovu adrese već i na osnovu njihovog sadržaja. **Memorija koja omogućava pristup na osnovu sadržaja naziva se asocijativna memorija.**

Asocijativna memorija je memorija sa neposrednim pristupom kojoj su dodatno ugrađena kola koja omogućavaju istovremeno poređenje u samo jednom memorijskom ciklusu svih lokacija sa zadatim sadržajem-ključem. Ta dodatna kola nazivaju se **asocijativni memorijski elementi (AME)**.