Globalna struktura računarskog sistema

Fon Nojmanova mašina

Delovi fon Nojmanovog izveštaja o EDVAC-u:

- .... pošto je takav uredjaj prvenstveno računar, on bi trebao dosta često da izvodi osnovne aritmetičke operacije. Te operacije su sabiranje, oduzimanje, množenje i deljenje. Stoga je razumljivo da treba da sadrži specijalizovane organe koji obavljaju te operacije ... U svakom slučaju, kao poseban deo treba da postoji *centralni aritmetički* (eng. *central arithmetical*) organ, CA računara.

- Logička kontrola uredjaja, tj. redosleda mogućih operacija može da bude izvedena mnogo efikasnije od strane centralnog organa za upravljanje. Kako uredjaj treba da bude prilagodljiv, tj. da bude za opšte namene, mora da postoji razlika izmedju specifičnih instrukcija koje definišu pojedinačni problem i služe za njegovo rešavanje, i opštih, za upravljačke organe koji nadgledaju kako se izvode te specifične instrukcije ... Pod *centralnim upravljanjem* (eng. *central control*), CC podrazumevamo drugu od prethodnih funkcija, a organi koji je sprovode formiraju drugi specifičan deo računara.

- Bilo koji uredjaj koji izvodi dugačak i komplikovan niz operacija (posebno izračunavanja) mora da ima odgovarajuću memoriju ...

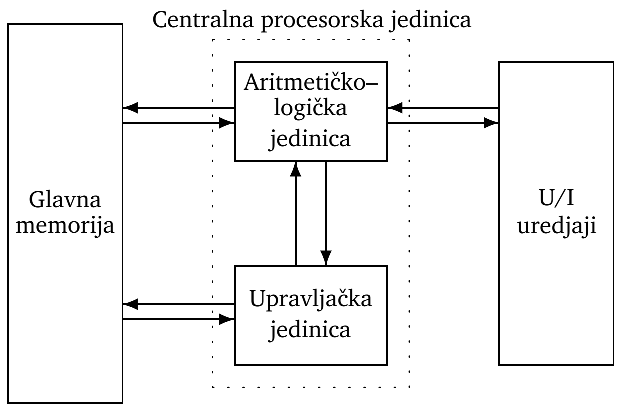
  Instrukcija koja upravlja specifičnim problemom može da sadrži odgovarajući materijal ... Ovaj materijal mora da se pamti ... U svakom slučaju, ukupna *memorija* (eng. *memory*), M čini treći specifični deo računara ...

- Specifični delovi CA, CC (dalje označeni sa C) i M odgovaraju asocijativnim neuronima ljudskog nervnog sistema. Ostaje još da se govori o ekvivalentu senzora i motorike neurona. Taj ekvivalent predstavlja *ulazne* i *izlazne* organe uredjaja ...

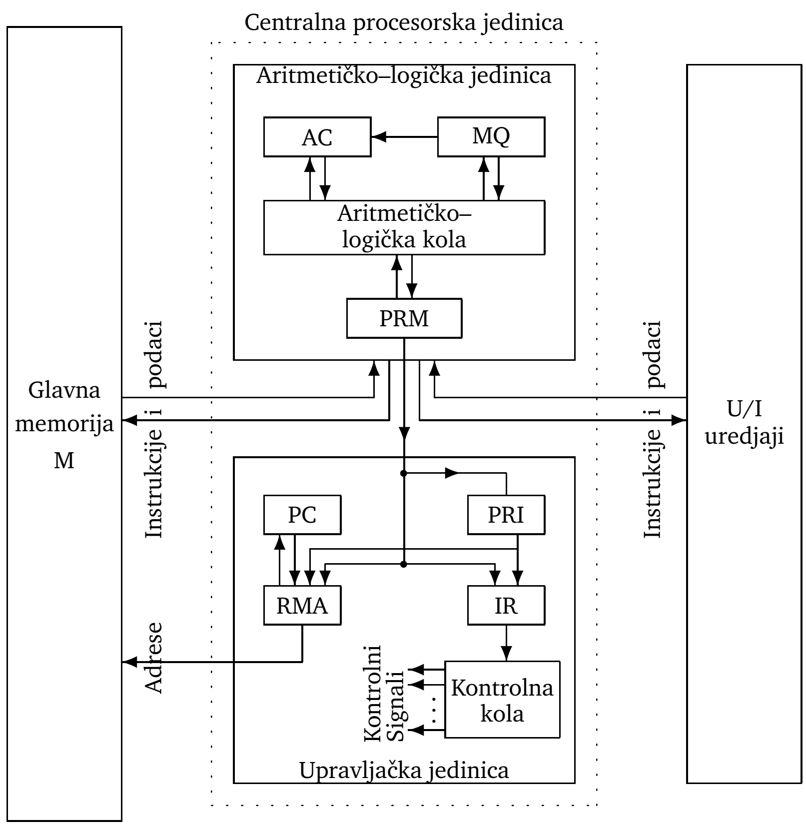
Uredjaj mora da podržava ulaz i izlaz preko kontakta sa nekim specifičnim medijumom tog tipa. Taj medijum koji ćemo nazvati medijum za spoljašnje *snimanje* (eng. *recording*) uredjaja, R ...

- Uredjaj mora da ima organe za prenošenje informacija od R ka specifičnim delovima C i M. Ti organi formiraju četvrti specifični deo računara, *ulaz, I*. Videće se da je najbolje vršiti sve transfere iz R (preko I) u M, a nikada direktno iz C ...
- Uredjaj mora da ima organe za prenošenje ... iz specifičnih delova C i M u R. Ti organi formiraju peti specifični deo računara, *izlaz, O*. Videće se da je opet najbolje vršiti sve transfere iz M (preko O) u R, a nikada direktno iz C ...

Kao što se iz teksta vidi, fon Nojman je pokušao da napravi analogiju izmedju čoveka i predložene strukture svoje mašine.



Slika 1: Struktura IAS računara



Slika 2: Detaljnija struktura IAS računara

Karakteristike IAS računara:

- IAS je imao 1000 lokacija u memoriji koje su nazvane *reči*.
- Reč je sadržavala 40 binarnih cifara.
- Pošto su u memoriji čuvani i instrukcije i podaci, brojevi su bili predstavljeni u binarnom obliku pri čemu je jedan bit označavao znak a ostalih 39 vrednost.
- Instrukcije su bile predstavljene u obliku binarnih kodova; svaka reč je sadržavala dve 20-bitne instrukcije. Instrukcija je posedovala 8-bitni kod operacije i 12-bitnu adresu reči u memoriji (u intervalu 0-999).

Centralni procesor je sadržavao *registre* definisane na sledeći način:

- *Prihvatni registar memorije (PRM)*. Ovaj registar je sadržavao reč koja je trebalo da se sačuva u memoriji, ili se koristio kao prijemno polje pri prenosu reči iz memorije.
- *Registar memorijskih adresa (RMA)*. Sadržaj ovog registra je bio adresa reči u memoriji iz koje je trebalo preneti sadržaj u PRM ili u koju je trebalo preneti sadržaj iz PRM.
- *Instrukcioni registar (IR)*. Ovaj registar je sadržavao 8-bitni kod operacije koja se trenutno izvršava.
- *Prijemni registar instrukcija (PRI)*. U njemu je privremeno čuvana instrukcija iz desne polovine reči.
- *Brojač instrukcija (PC)*. Oznaka PC je akronim od naziva na engleskom: *program counter.* Ovaj registar je čuvao adresu sledećeg para instrukcija koje je trebalo preneti iz memorije.
- *Akumulator (AC)* i *Množilac/Delilac (MQ)* (oznaka MQ je akronim od naziva na engleskom: *multiplicator/quotient*). U njima su se privremeno čuvale operacije i rezultati operacija koje su izvodjene u aritmetičko–logičkoj jedinici (eng. *arithmetic–logic unit, ALU*). Npr. ako je rezultat množenja dva 40-tocifrena broja bio 80-tocifreni broj tada se 40 bitova veće težine čuvalo u AC a 40 bitova manje težine u MQ registru.

IAS je instrukcije izvršavao u *instrukcionim ciklusima*. Svaki instrukcioni ciklus je imao dva podciklusa:

- *Ciklus pripreme* u kome se operacioni kod naredne instrukcije smeštao u IR, dok se adresni deo smeštao u RMA. Instrukcija je uzimana iz PRI ili iz memorije odakle se prvo smeštala u PRM, a zatim njeni delovi prenosili u PRI, IR i RMA.
- *Ciklus izvršavanja* u kome su kontrolna kola u upravljačkoj jedinici interpretirala operacioni kod instrukcije i slala odgovarajuće kontrolne signale koji su označavali da treba preneti podatke ili izvršiti operaciju od strane ALU.

IAS računar je imao ukupno 21 mašinsku instrukciju koje su bile podeljene u pet grupa:

1. *instrukcije za prenos podataka*
2. *instrukcije bezuslovnog grananja*
3. *instrukcije uslovnog grananja*
4. *aritmetičke instrukcije*
5. *instrukcije za modifikaciju adrese*

Pri opisu značenja instrukcija iz naredne tabele korišćena je sledeća notacija:

- `S(L)` označava sadržaj lokacije `L`.
- `X`$\longrightarrow$`Y` označava upisivanje vrednosti `X` u lokaciju `Y`.
- |`X`| označava apsolutnu vrednost `X`.

Pri tome *lokacija* može biti `AC` (akumulator), `MQ` registar ili memorijska lokacija.

Tip instrukcije	Operacioni kod	Instrukcija	Opis
Prenos dataka	00000001	LOAD M(X) po-	Sadržaj memorijske lokacije X se prenosi u akumulator: S(X)⟶AC
	00000010	LOAD -M(X)	-S(X)⟶AC
	00000011	LOAD |M(X)|	|S(X)|⟶AC
	00000100	LOAD -|M(X)|	-|S(X)|⟶AC
	00001001	LOAD MQ,M(X)	S(X)⟶MQ
	00001010	LOAD MQ	S(MQ)⟶AC
	00100001	STOR M(X)	S(AC)⟶X
Bezuslovni skok	00001101	JUMP M(X,0:19)	Izvršava se instrukcija zapisana u levoj polovini sadržaja memorijske lokacije X
	00001110	JUMP M(X,20:39)	Izvršava se instrukcija zapisana u desnoj polovini sadržaja memorijske lokacije X
Uslovni skok	00001111	JUMP+ M(X,0:19)	Ako je sadržaj akumulatora nenegativan izvršavanje se prenosi na instrukciju zapisanu u levoj polovini sadržaja memorijske lokacije X
	00010000	JUMP+ M(X,20:39)	Ako je sadržaj akumulatora nenegativan izvršavanje se prenosi na instrukciju zapisanu u desnoj polovini sadržaja memorijske lokacije X
Aritmetičke instrukcije	00000101	ADD M(X)	S(AC) + S(X)⟶AC
	00000111	ADD |M(X)|	S(AC) + |S(X)|⟶AC
	00000110	SUB M(X)	S(AC) - S(X)⟶AC
	00001000	SUB |M(X)|	S(AC) - |S(X)|⟶AC
	00001011	MUL M(X)	Pomnoži S(MQ) sa S(X). Polovinu rezultata veće težine upiši u AC, a polovinu manje težine u MQ
	00001100	DIV M(X)	Podeli S(MQ) sa S(X). Količnik se upisuje u MQ a ostatak u AC
	00010100	LSH	S(AC)*2⟶AC, tj. pomeri sadržaj akumulatora ulevo za jedno mesto
	00010101	RSH	S(AC)/2⟶AC, tj. pomeri sadržaj akumulatora udesno za jedno mesto
Modifikacija adrese	00010010	STOR M(X,8:19)	Zameni adresno polje u levom delu sadržaja na memorijskoj lokaciji X sa 12 krajnje desnih bitova AC
	00010011	STOR M(X,28:39)	Zameni adresno polje u desnom delu sadržaja na memorijskoj lokaciji X sa 12 krajnje desnih bitova AC

Tabela 1: Instrukcije IAS računara

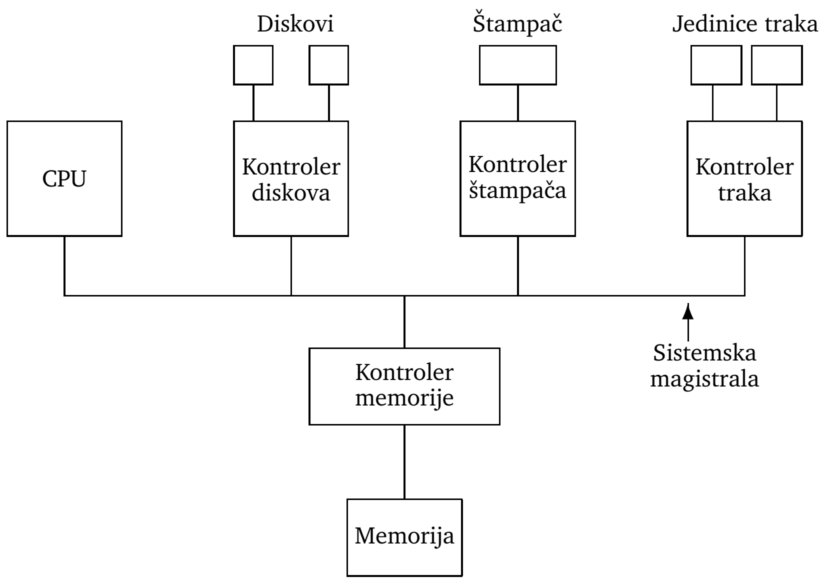
Izgled savremenog računarskog sistema

Najveći broj današnjih računarskih sistema je zasnovan na fon Nojmanovoj arhitekturi računarskog sistema. Osnovni koncepti ove arhitekture su:

1. Računarski sistem poseduje samo jednu memoriju u kojoj se čuvaju i podaci i instrukcije. Izmedju zapisa podataka i instrukcija ne postoje razlike, tj. jedino od načina interpretacije zavisi koji sadržaj će biti shvaćen kao podatak a koji kao instrukcija.

2. Memorija računara je adresibilna po lokacijama kojima se može pristupiti bez obzira kakav im je sadržaj.

3. Izvršavanje instrukcija se izvodi strogo sekvencijalno, sem ukoliko drugačije nije eksplicitno naglašeno (npr. izvršavanjem instrukcije skoka se može modifikovati redosled izvršavanja instrukcija).



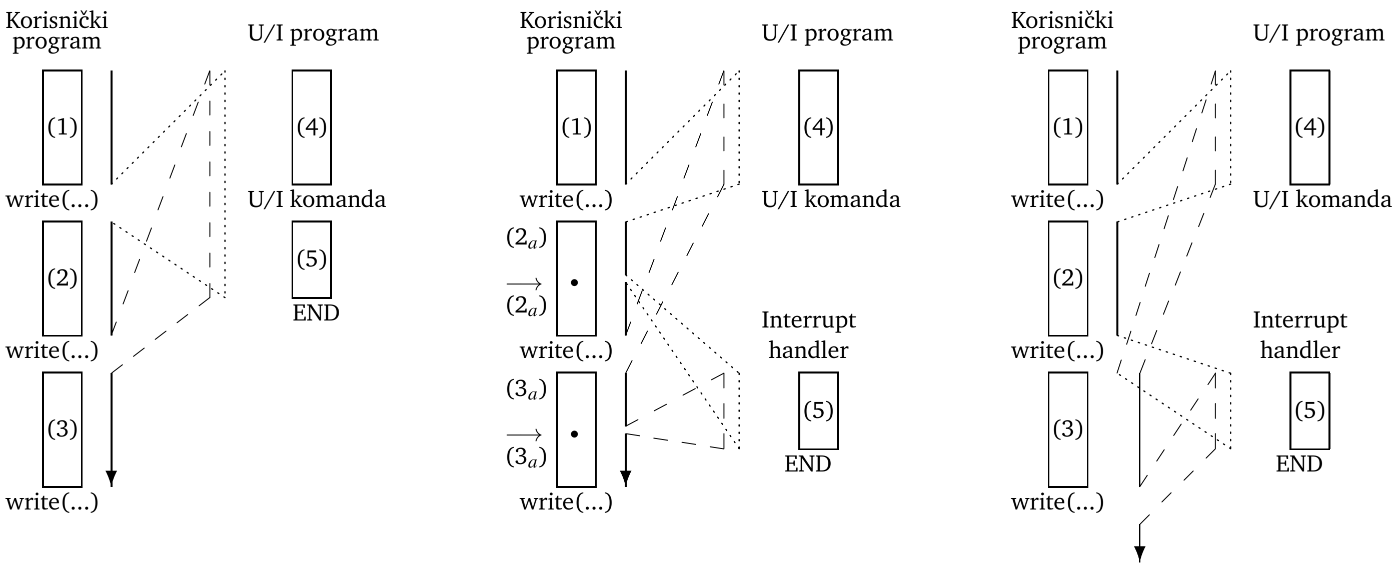
Slika 3: Savremeni računarski sistem

Sistem prekida

- Na slici 4 je prikazan rad aplikacije koja iza svakog koraka obrade štampa dobijene rezultate.
- Segmenti koda označeni sa (1), (2) i (3) predstavljaju niz instrukcija koji ne uključuje ulaz ili izlaz.
- `WRITE` predstavlja poziv U/I programa koji treba da obavi tekuću U/I operaciju.

U/I program čine tri komponente:

- Niz instrukcija označen sa (4) na slici, koji vrši pripremu tekuće U/I operacije. Uključuje prepisivanje podataka koji treba da se odštampaju u posebni bafer i pripremu parametara za izdavanje komande uredjaju.
- Stvarna U/I komanda. U njoj se izdaje komanda uredjaju koji obavlja zahtevanu akciju.
- Niz instrukcija označen sa (5) na slici, koji kompletira zahtevanu operaciju.



(a) Bez sistema prekida

(b) Sa sistemom prekida (kratka U/I operacija – kompletira se pre pojave novog U/I zahteva)

(c) Sa sistemom prekida (duga U/I operacija – ne kompletira se pre pojave novog U/I zahteva)

Slika 4: Tok izvršavanja programa sa i bez sistema prekida

U višekorisničkom sistemu u istom trenutku može da se javi više prekida koje su generisali različiti programi ili uredjaji. Načini obrade višestrukih prekida su:

1. Onemogućivanje prekida, kada se ostali prekidi onemogućuju dok se ne završi obrada tekućeg. Onemogućivanje se vrši tako što procesor ignoriše novonastale prekide koji se odlažu u red. Po završetku obrade tekućeg, procesor proverava stanje u redu i počinje obradu prekida koji je prvi u redu.

2. Definisanje prioriteta prekida, uz davanje dozvole prekidima višeg prioriteta da prekinu izvršavanje (sistemskog) programa koji vrši obradu prekida sa nižim prioritetom.

Brzina obrade podataka

Jedan način merenja brzine računara (odn. procesora) je prema broju mašinskih instrukcija u sekundi koje CPU može da obradi. Jedinica mere se naziva MIPS (eng. *millions of instructions per second*):

- Intel Pentium II mikroprocesor (na 400MHz) ima oko 830 MIPS-a
- Intel Pentium 4 mikroprocesor (na 1.5GHz) ima oko 1700 MIPS-a,
- Intel Pentium 4 mikroprocesor (na 2GHz) ima oko 3900 MIPS-a,
- IBM PowerPC 750CX mikroprocesor (na 400MHz) ima oko 930 MIPS-a
- Motorola PowerPC MPC7450 mikroprocesor (na 733MHz) oko 1300 MIPS-a.
- Motorola PowerPC MPC7457 mikroprocesor (na 1GHz) oko 2310 MIPS-a.
- Intel Core 2 Extreme QX9770 mikroprocesor (na 3.3GHz) oko 177000 MIPS-a.
- Ibm z196 mikroprocesor (na 5.2GHz) oko 1200 MIPS-a po procesoru

Brzina superračunara se meri u **FLOPS**-ima (eng. *floating point operations pre second*) Na primer, prvi superračunar CRAY-1 je imao brzinu od 166MFLOPS-a, dok IBM ASCI White (objavljen juna 2000. godine) ima brzinu od 12.3TFLOPS-a, IBM Roadrunner (objavljen 2008. godine) ima brzinu od 1.105 petaflopsa, dok CRAY Jaguar XT5 (objavljen 2008. godine) ima brzinu od 1.1059 petaflopsa.

Drugi način merenja brzine računara je vreme potrebno za izvršavanje jednog instrukcionog ciklusa.

- milisekunde (10^{-3}) za vrlo spore mašine,
- mikrosekunde (10^{-6}) za uobičajenu klasu mikroračunara,
- nanosekunde (10^{-9}) za mainframe računare i servere,
- pikosekunde (10^{-12}) kod najbržih eksperimentalnih računara.

Propustnost (eng. *throughput*) predstavlja broj programa koji mogu da završe obradu u nekom vremenskom intervalu. Merenje snage na ovaj način je komplikovaniji jer zavisi i od operativnog sistema i od programa sa kojima se merenje vrši. Postoje i drugi načini za merenje snage računarskog sistema kao što je broj transakcija u sekundi, itd.