

1.UVOD

1. **Računalnike** lahko razdelimo na:

a. **Osebni računalniki (PC)**

- Izdelani za individualno uporabo
- Prenosni računalniki, tablice

b. **Strežniki**

- Boljša zmogljivost, tudi višja cena
- Superračunalniki, zmogljivi desktop računalniki

c. **Vgrajeni računalniki**

- Največja skupina računalnikov
- Mikroprocesorji (mikrokontrolniki) v avtomobilih, mobilnih telefonih, igralnih konzolah...

2. **Računalniška arhitektura** obravnava za programerja vidnih lastnosti računalnika na način, ki je neodvisen od njegove logične in fizične realizacije.

Računalniška organizacija je fizična in logična realizacija računalniške arhitekture. Bliže je aparaturnemu HW) nivoju.

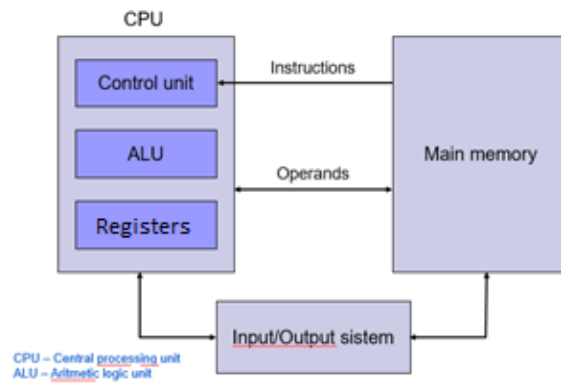
Ena arhitektura se lahko realizira z različnimi vrstami organizacij in obratno.

3. **Strojni jezik** je sestavljen iz ukazov, ki jih lahko računalnik neposredno izvede. Te ukaze imenujemo **strojni ukazi** in jih računalnik vedno razume in zna izvajati. V splošnem imajo računalniki različnih proizvajalcev lahko tudi različne strojne jezike.

4. Kako deluje računalnik?

- Digitalni računalniki** so stroji za reševanje nalog z izvajanjem ukazov, ki jih določijo programerji.
- Zaporedje teh ukazov imenujemo **program**.
- Pred izvedbo se mora vsak program prevesti v omejen nabor strojnih ukazov, razumljivih za elektronsko vezje oz. procesor.
- Različni procesorji imajo lahko različne ukaze v strojnem jeziku.
- Program v vsakem višjem programskem jeziku (Java, Python, C, C++...) se mora prevesti v te osnovne, strojne ukaze.

5. Struktura tipičnega računalnika (računalniški model von Neumanna)

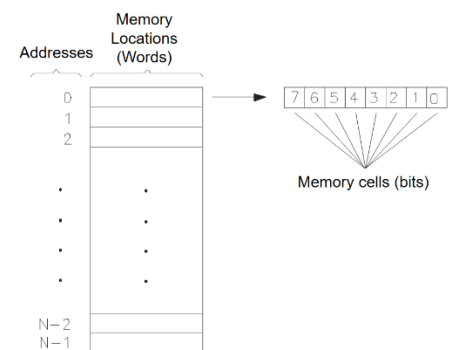


a. **CPE** – Centralna procesna enota

- KE - Krmilna enota (vodi delovanje)
- ALE – Aritmetična logična enota (izvaja operacije)
- Registri (pomnilnik v CPE-ju)

b. **GP- Glavni pomnilnik**

- Sestavljen je iz pomnilniških mest, ki se imenujejo besede, ki vsebujejo pomnilniške celice, imenovane bite, in vsaka beseda ima svoj naslov.



c. **Vhodno/Izhodni (V/I) sistem**

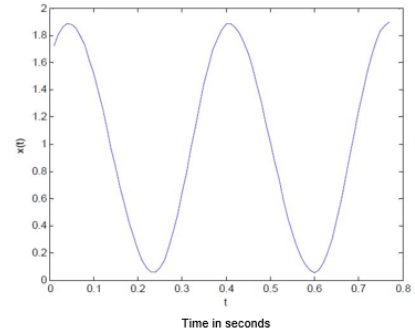
d. **Vodila** (povezujejo komponente sistema, da lahko ukazi, operandi in ostale vsebine potujejo med deli sistema)

6. Signali so lahko:

a. **Analogni** (zvezni)



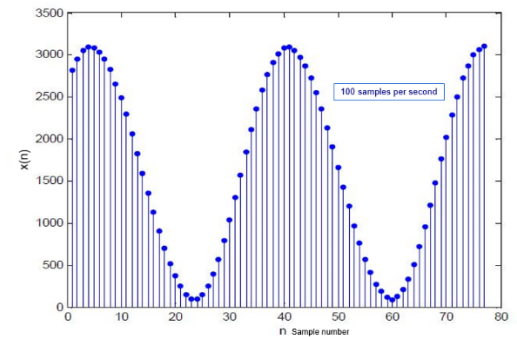
- Vrednost je znana za vsako poljubno točko na x-osi
- Lahko jih predstavimo kot zvezne funkcije
- Običajno nastanejo ob opazovanju fizikalnih veličin (temperatura, zvok, el. napetost)
- Primer: analogna ura: na podlagi urinega kazalca, lahko čas povemo z večjo natančnostjo.



b. **Digitalni** (diskretni)



- Delujejo z vzorčnimi števkami
- Predstavimo jih kot diskretne funkcije (čas, digitalni še diskretna amplituda)
- Pravilna vrednost je približno zajeta s pomočjo vzorcev
- Primer: Digitalna ura: Natančnost je odvisna od parametrov, ki jih ura prikazuje, običajno minute, včasih tudi sekunde.



Digitalno računanje

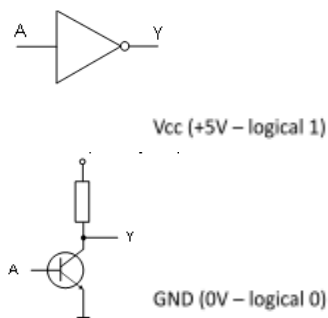
- c. S števki 1 in 0, ki tvorita t.i. **binarni (dvojiški) številski sistem**
- d. Binarna številka se imenuje **bit**, in je predstavljena z 1 ali 0.
- e. Digitalni računalniki temeljijo na dvojiškem številskem sistemu.

7. Pomembne ideje v računalniški arhitekturi (tudi splošneje v računalništvu):

- a. Moorov zakon
 - Število tranzistorjev v integriranem vezju se podvoji vsaki dve leti
- b. Abstrakcija kot poenostavitev
 - Uporabimo pri sestavi aparature, programske opreme, programskih jezikov ...
 - Probleme/module delimo na manjše enote/probleme in rešitve združujemo.
- c. Hitrost
 - Najučinkoviteje je pospešiti pogosto uporabljene postopke
- d. Vzporednost
 - Hkratna izvedba opravil; je posledica trenda v tehnološkem razvoju
- e. Cevovodi
 - Zaporedje stopenj, pri katerih je izhod ene stopnje vhod v naslednjo
 - V vseh stopnjah se hkrati izvaja več ukazov, vsak v svoji stopnji (koraku)
- f. Zmogljivost s predvidevanjem
 - Boljše delati vnaprej v skladu s predvidevanjem, kot pasivno čakati
- g. Hierarhični model pomnilnika
 - Kompromis med hitrostjo pomnilnika in ceno
 - Hierarhija je sestavljena iz nivojev z različnim kompromisom med hitrostjo in ceno
- h. Zanesljivost z redundanco
 - Rezervni (back-up) sistemi; včasih je ceneje imeti podvojen (redundanten) sistem

8. Fizična zgradba računalnika:

- a. Matematični (logični) pregled: logična vrata – ideal
- b. Električna izvedba: elektronsko vezje (transistor kot osnovni gradnik)
 - Slabosti izvedbe:
 1. Zvezna (analogna) napetost
 2. Časovne zakasnitve
 3. Občutljivost na šume (motnje)



9. Stikala

- V računalnikih so informacije predstavljene v dvojiškem sistemu (0,1) z uporabo električnih signalov. vb
- Obstajata dve stanji, ki sta predstavljeni z dvema napetostima:
 - Stanje 0 je nizka napetost - stikalo je sklenjeno
 - Stanje 1 je visoka napetost - stikalo je odprto
- Eno stikalo upravlja 1 bit informacije in je zato lahko v dveh stanjih, 0 ali 1
- Ena osnovna pomnilniška celica je zgrajena s pomočjo teh stikal in lahko shrani 1 bit

10. Stikala v zgodovini:

- Rele** – elektromehansko stikalo – 1939+
- Elektronka** – električno stikalo – 1945-1955
- Tranzistor** – električno stikalo – 1955+

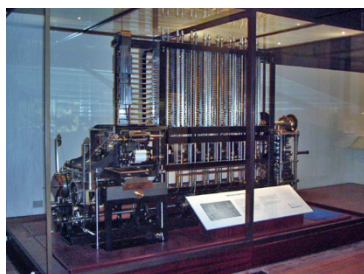


- Integrirano vezje (čip)** – Več tranzistorjev na eni silicijski rezini (plošči) – 1958+

2. RAZVOJ STROJEV ZA RAČUNANJE

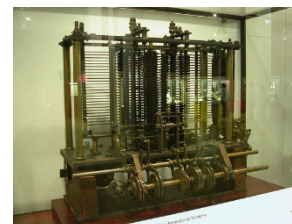
1. Obdobje **mehanike** (1600)

- a. Izdelani so bili prvi kalkulatorji, ki so bili mehanski in ročno upravljani.
- b. **Charles Babbage**
 - Diferenčni stroj



- **Analitični stroj** – prvi pravi predhodnik današnjih računalnikov

- ☐ Delovanje je vodil program
- ☐ Bil je namenjen za reševanje poljubnih problemov
- ☐ Nikoli ni bil dokončan



2. **Elektromehanski stroji** (1939)

- a. Za pogon mehanskih sklopov so bili uporabljeni električni motorji
- b. Prisotnost luknjic na luknjanih karticah se je ugotavljala električno in ne mehansko
- c. Konrad Zuse

- Z1 – Prvi delujoči stroj Babbageve vrste, v celoti mehanski.
- Z2 – Aritmetična enota s telefonskimi releji, nedokončan.

- **Z3**

- ☐ Prvi delujoči **programsko vodeni elektromehanski računalnik za splošne namene**
- ☐ Uporabljal dvojiško aritmetiko
- ☐ Shranjevanje ukazov je bilo na luknjanem traku

- d. Harvard

- **Mark I**

- ☐ Ekvivalenten Babbagevemu analitičnemu stroju
- ☐ Desetiška aritmetika
- ☐ Shranjevanje ukazov je bilo na luknjanem traku

- Mark II, III, IV



*Harvard Mark I in Zuse Z3 sta **podobna stroja**, obadva imata **shranjevanje ukazov na perforiranem traku**. Razlika je v tem, da Mark I uporablja **desetiško aritmetiko**, medtem ko Z3 uporablja **dvojiško aritmetiko**.

3. Prvi elektronski računalniki (1945)

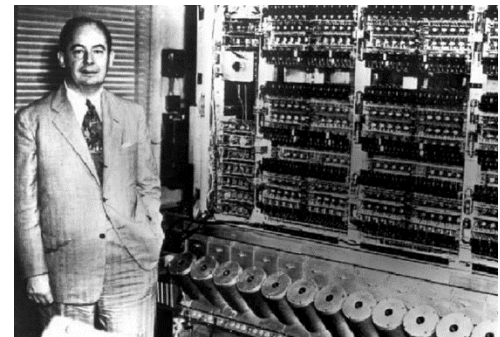
- a. Elektronske cevi so zamenjale releje, kar je povečalo hitrost računalnikov
- b. Uporabljalo se je za dešifriranje sporočil med drugo svetovno vojno v Veliki Britaniji
- c. **ENIAC** – Univerza v Pensilvaniji
 - Electronic Numerical Integrator And Calculator
 - Programiranje s stikali in povezovanjem kablov



4. Elektronski računalniki s shranjenim programom (po letu 1945)

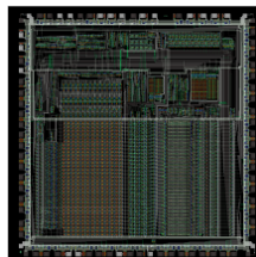
a. John von Neumann

- EDSAC
 - Prvi delujoči računalnik s shranjenim programom
 - Pravilo, ki se še danes upošteva:
“Če navodila ne zahtevajo drugače (navodila JUMP, GOTO), se ukazi preberejo in izvedejo v naraščajočem vrstnem redu.”
- EDVAC
 - Program je bil shranjen v pomnilniku
- IAS
 - RAM – random access memory
 - Programski števec – register, ki vsebuje naslov naslednjega navodila



5. Razvoj po letu 1950

- a. Razvoj je bil bolj tehnološki kot arhitekturni
- b. **Elektronske cevi so zamenjali tranzistorji**, ki so manjši, hitrejši in bolj zanesljivi
- c. Prvi **mikroprocesorji** (1971)
- d. Prvi osebni računalniki IBM PC (1980)
- e. Prvi ARM procesorji (1985)



3. OSNOVNI PRINCIPI DELOVANJA

1. Računalniški model **Von Neumanna**:

- Naprava, ki ima svoje programe shranjene v glavnem pomnilniku.
- **Program** vodi delovanje stroja; **ukazi** povedo napravi, kaj naj naredi.
- Von Neumannov model je sestavljen iz:

a. CPE

- Centralna Procesna Enota
- Bere ukazi iz glavnega pomnilnika, in jih izvrši
- Sestavljena je iz:
 - **Kontrolna enota** – Prevzema ukaze in operande; aktivira operacije, ki so določene z ukazi.
 - **ALE** – Aritmetično Logična Enota – Izvaja aritmetične in logične operacije (seštevanje, in, ali...)
 - **Registri** – Skupine pomnilniških celic, ki služijo za shranjevanje vrednosti. Lahko so:
 - i. Programsko nedostopni – Potrebni za delovanje CPE-ja
 - ii. Programsko dostopni (arhitekturni) – shranjujejo operande, so majhen in hiter pomnilnik v CPE

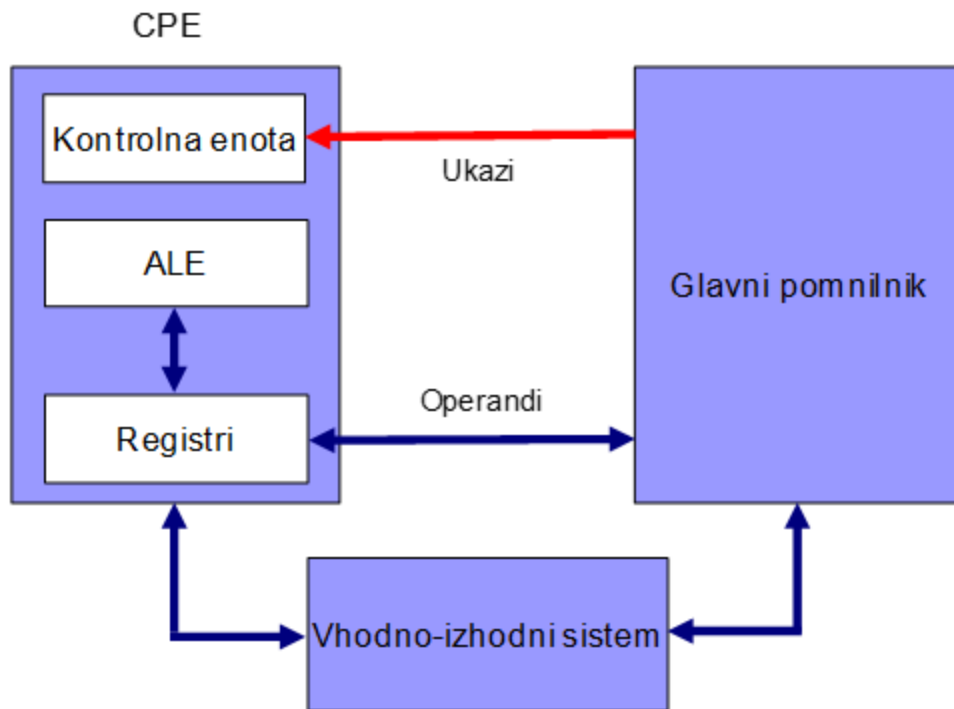
b. Glavni pomnilnik

- Sestavljen je iz pomnilniških besed, kjer ima vsaka beseda svoj naslov
- Ukazi in operandi so shranjeni v glavnem pomnilniku
- Z imenom "glavni" ga ločimo od drugih pomnilniških naprav, kot so predpomnilniki in navidezni pomnilniki.

c. Vhodno-Izhodni (V/I) sistem

- Služi za prenos informacij med napravo in zunanjim svetom
- Informacije iz CPE-ja in glavnega pomnilnika so shranjene v obliki, ki ni dostopna zunanjemu svetu
- Sestavni del V/I sistema so V/I naprave, ki pretvorijo informacije v obliko, ki je primerna za uporabnika, ali služijo kot sekundarni pomnilnik.

Von Neumannov računalniški model

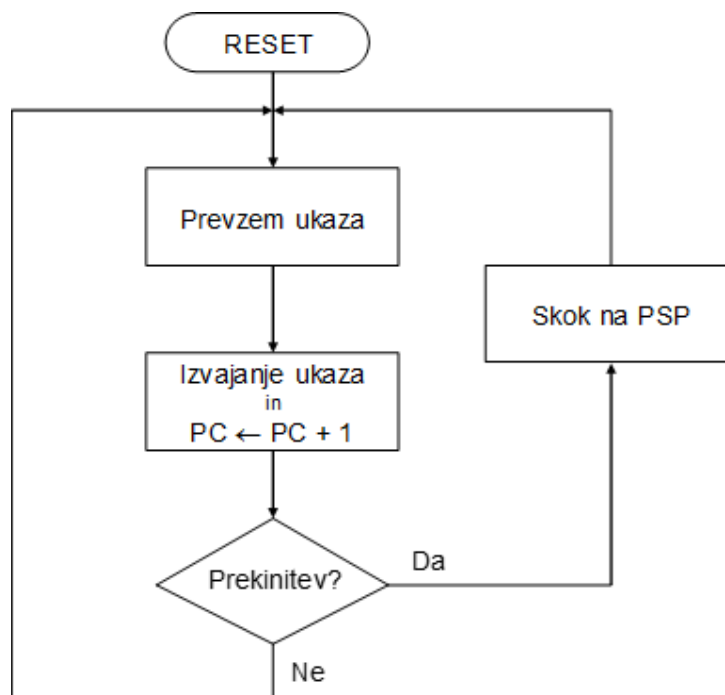


2. Delovanje Von Neumannovega računalnika:

- Njegovo delovanje popolnoma določajo ukazi (strojni ukazi), ki jih CPE jemlje iz glavnega pomnilnika zaporedoma enega za drugim.
- Strojni ukazi so v pomnilniku shranjeni eden za drugim po naraščajočih naslovih.
- Ob zagonu računalnika: prvi ukaz se običajno prebere z določenega naslova v pomnilniku, običajno na najnižjem ali najvišjem naslovu

3. Za vsak ukaz ločimo:

- Vsaki ukaz vsebuje dve vrsti informacij:
 - a. Informacijo o operaciji, ki jo treba izvesti
 - b. Podatke o operandih, nad katerimi se naj operacija izvede
- Koraka za izvedbo ukazov:
 - a. Branje ukaza (FETCH)**
 - Preberi ukaze iz pomnilnika -> instruction fetch cycle (ukazno prevzemni cikel)
 - **Program Counter (Programski Števec) PC** – Poseben register, ki vedno vsebuje pomnilniški naslov na katerem je v pomnilniku shranjen naslednji ukaz
 - Običajno se PC poveča za 1 (ali več), razen v primerih, ko pride do skoka (pogojni ali brezpogojni) ali prekinitve (CPU zažene ustrezen podprogram - Interrupt Service Program (ISP))
 - b. Izvajanje (EXECUTE)**
 - Izvedba ukazov – Execute cycle



4. Flynnova klasifikacija

- Sekvenčno izvajanje navodil iz modela Von Neumann je v splošnem počasno, zato je pospešeno z razširitvami, ki jih opredeli Flynnova klasifikacija
- Klasifikacija temelji na dveh kriterijih:
 - a. Število ukazov ki se izvajajo hkrati
 - b. Število operandov, ki jih obdelata en ukaz
- Obstajajo štiri vrste računalnikov:

a. **SISD** – Single Instruction Single Data

- Klasične Von Neumannove računalnike brez paralelizma

b. **SIMD** – Single Instruction Multiple Data

- Pravi vektorski računalniki
- En ukaz se izvede na več skupinah podatkov

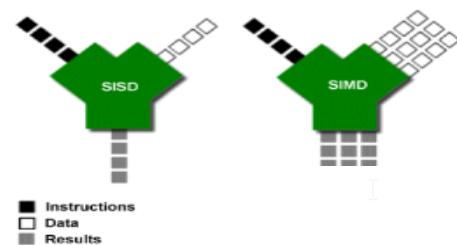
c. **MISD** – Multiple Instructions Single Data

- Nenavadna arhitektura, uporabna za zmanjšanje napak pri računanju
- Več ukazov na enakem operandu

d. **MIMD** – Multiple Instruction Multiple Data

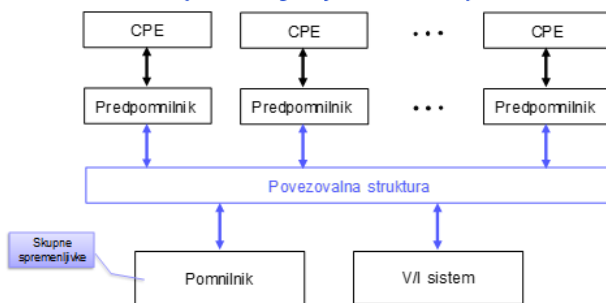
- Večprocesorski računalniki (vzporedni računalniki)
- Več ukazov izvedenih hkrati, vsak na svojih operandih
- Sestavljen iz več medsebojno povezanih CPE'jev

SISD vs SIMD



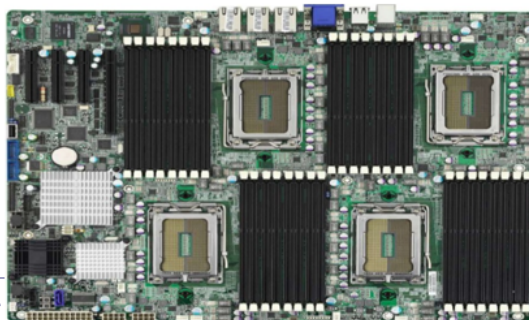
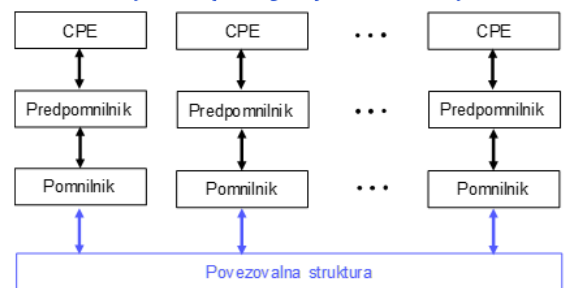
Multiprocessor

(tesneje povezan)



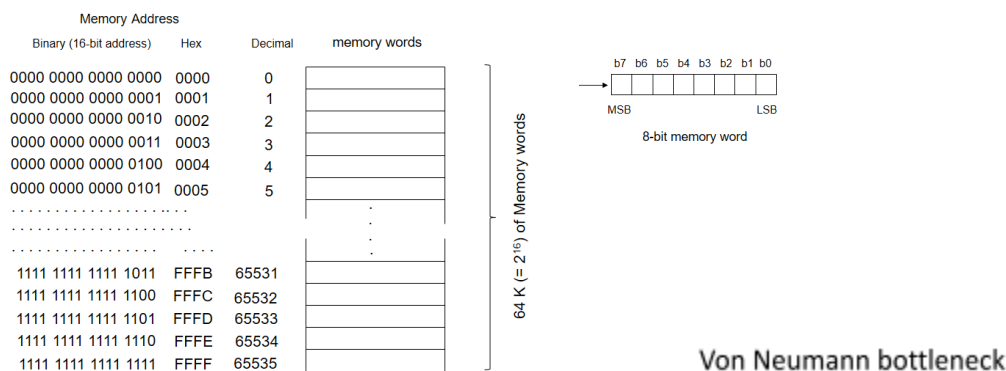
Multiračunalnik

(ohlapneje povezan)



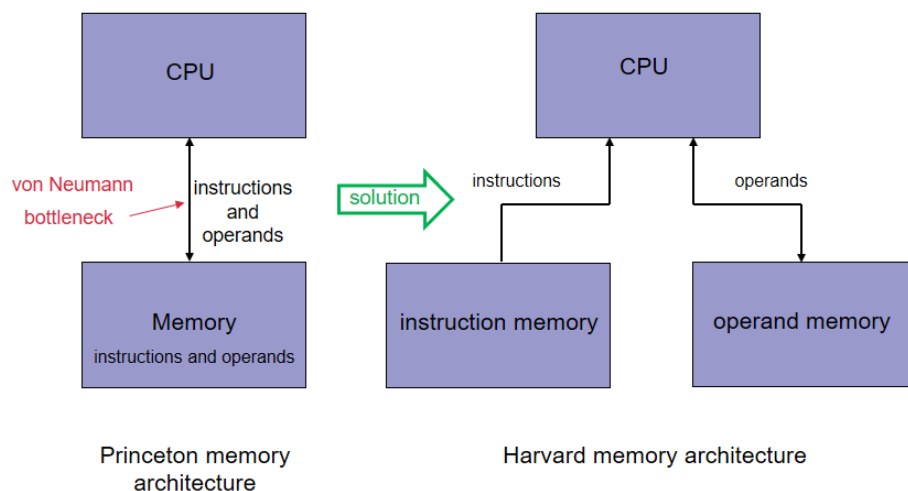
5. Glavni pomnilnik v von Neumannovem računalniku

- Pasivna naprava za shranjevanje ukazov in operandov
- Osnovna celica v pomnilniku je pomnilniška celica, ki lahko hrani 1 bit informacije (1, 0)
- Več pomnilniških celic sestavljajo pomnilniško besedo, ki predstavlja najmanjše število bitov, ki imajo svoj naslov
- Vsebina besed v pomnilniku se lahko spreminja. V n-bitno pomnilniško besedo lahko shranimo 2^n različnih zapisov (vsebin).
- Pomnilnik je enodimenzionalno zaporedje pomnilniških besed.
- Pomnilniški naslov je nespremenljiva, enolična oznaka vsake pomnilniške besede. Število bitov, ki definirajo naslov, je označeno kot dolžina naslova.
- Deli naslovnega prostora so lahko prazni (vsi naslovi niso dejansko izkoriščeni - uporabljeni), glavni pomnilnik je običajno manjši od t.i. naslovnega prostora.
- Prefikse predpomnilnih enot so vedno števila ki so stopenj od dva: $1Ki=2^{10}$, $1Mi=2^{20}$, $1Gi=2^{30}$...



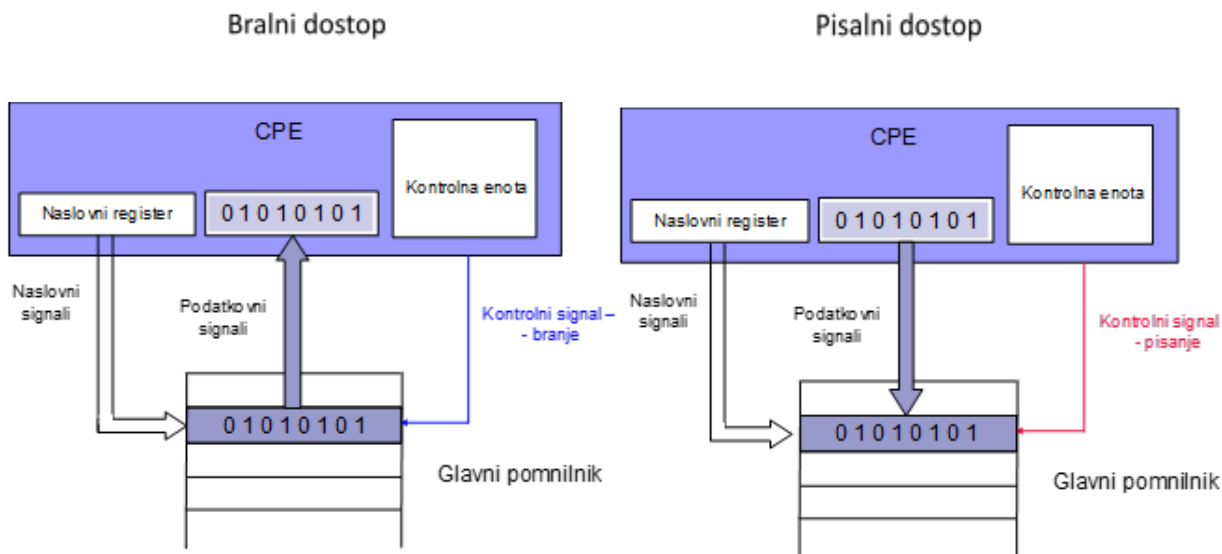
6. Von Neumannovo ozko grlo

- Je povezava med CPE in glavnim pomnilnikom. Iz pomnilnika v CPE se prenašajo vsi ukazi in operandi iz pomnilnika ali v pomnilnik.
- Eden od načinov za razširitev ozkega grla je razdelitev glavnega pomnilnika v dva dela.
- Harvardska pomnilniška arhitektura ima dva pomnilnika, ki delujeta istočasno, enega za ukaze in drugega za operande. Danes se uporablja v predpomnilnikih na najnižjih nivojih.
- Princetonska arhitektura se običajno uporablja v glavnem pomnilniku



7. Dostop do pomnilnika

- CPE dostopa do pomnilniške besede tako, da v pomnilnik pošlje naslov te besede in signal za smer prenosa.
- Ima dve vrsti dostopa do pomnilnika:
 - a. Read access - Branje:** Glavni pomnilnik → CPE branje od pomnilnika, se zgodi v pribl. 80 % vseh dostopov)
 - b. Write access - Pisanje:** CPE → Glavni pomnilnik (Pisanje v pomnilniku, zgodi v pribl. 20 % vseh dostopov)



8. Amdahlov zakon

- Če v računalniku za faktor N (N-krat) pohitrimo delovanje pri vseh operacijah, razen pri f-temu delu od vseh operacij, potem je povečanje hitrosti celotnega računalnika S(N) enako:

$$S(N) = \frac{1}{f + \frac{1-f}{N}} = \frac{N}{1 + (N-1) * f}$$

f je delež operacij, ki niso pohitrene!

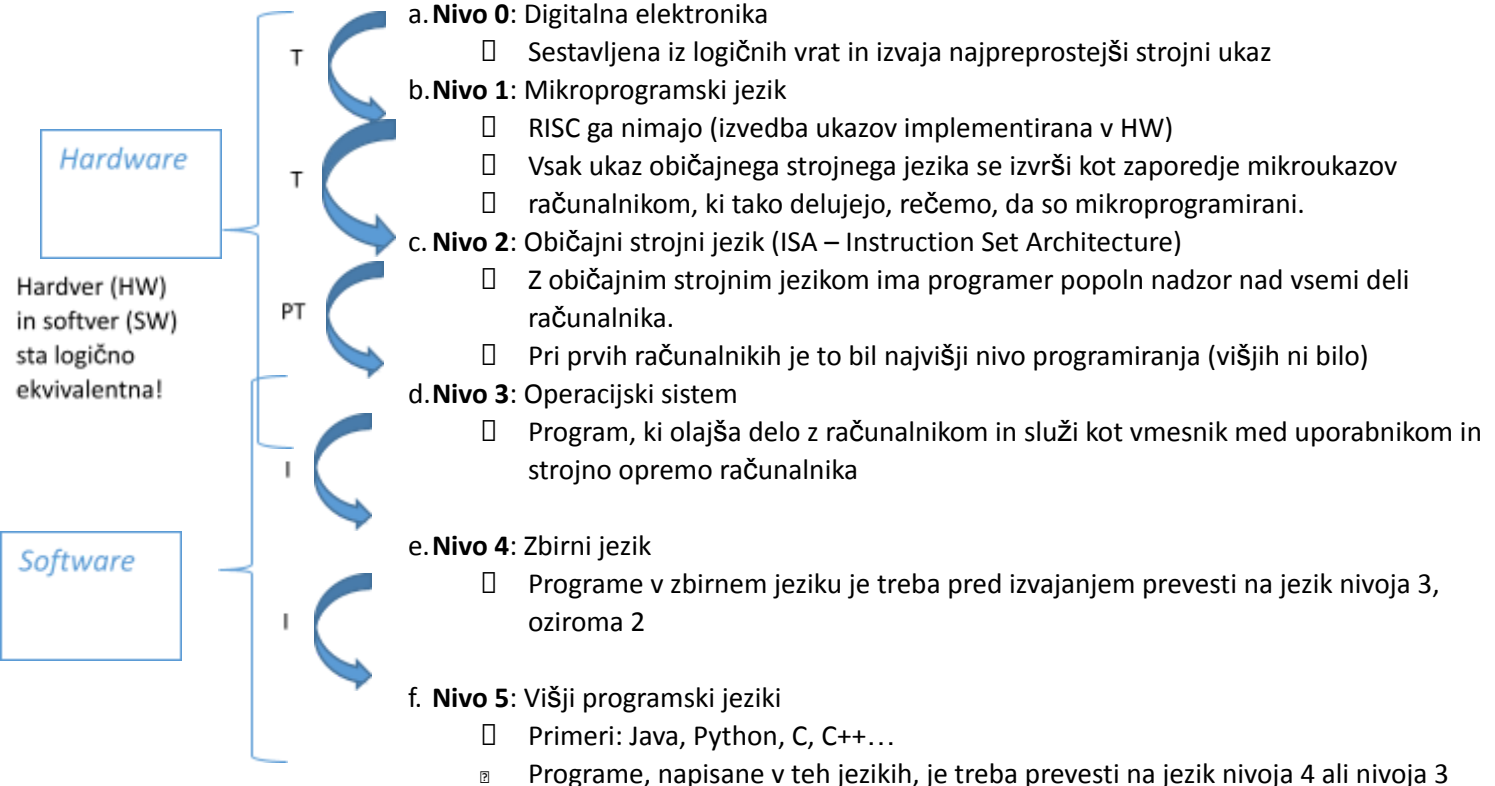
*S(N) = povečanje hitrosti celotnega sistema
 N = faktor povečanja hitrosti (1 - f)-tega dela
 f = delež operacij, ki niso pohitrene
 1 - f = delež operacij, ki so N-krat pohitrene*

- Po Amdahlovem mnenju je računalnik dobro zasnovan (uravnotežen), če izpolnjuje naslednje primere:
 - a. Prvo Amdahlovo pravilo:** velikost glavnega pomnilnika v bajtih mora biti najmanj enaka številu ukazov, ki jih CPE izvede v eni sekundi.
 - b. Drugo Amdahlovo pravilo:** zmogljivost V/I sistema v bitih na sekundo mora biti najmanj enaka številu ukazov, ki jih CPE izvede v eni sekundi.

9. Nivoji računalnika:

- Na vsakem nivoju vidimo računalnik skozi različne programske jezike, ki je predstavljen kot strojni jezik določenega virtualnega stroja.
- Najenostavnejši strojni ukazi na najnižjem nivoju izvajajo elektronske vezje.
- Mikro-programirani računalniki (6 nivojev) vs RISC računalniki (5 nivojev)

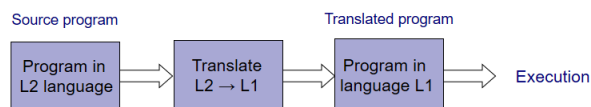
- Nivoji:



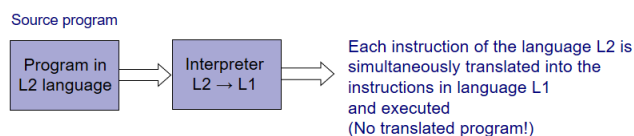
- **Mehanizem prehajanja** iz enega jezika v drugega:

- Prevajanje** – Označeno kot T ("Translate") na zgornji skici
- Interpretiranje** – Označeno kot I ("Interpretation") na zgornji skici
 - Glavna razlika med prevajanjem in interpretiranjem je, da pri interpretiranju ne obstaja prevedeni (ciljni) program.
- Delno prevajanje** – Označeno kot PT ("Partial Interpretation") na zgornji skici
 - Vmesna rešitev med prevajanjem in interpretacijo
 - Hitrejše kot interpretacija, počasnejše kot prevajanje

Translation (compilation)



Interpretation



4. UKAZI

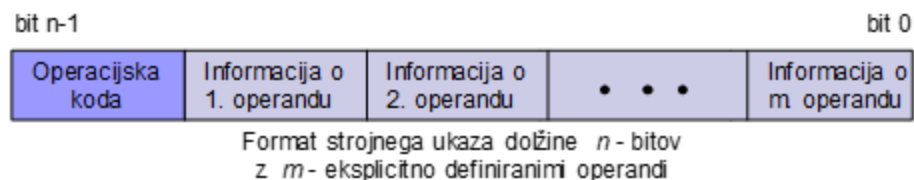
1. Osnovne vrste informacij v računalniku so operandi in ukazi.
2. Različni računalniki imajo različne arhitekture in zaradi tega različne strojne ukaze; zato je arhitektura računalnika v veliki meri določena z naborom strojnih ukazov.
3. Ukaze izvaja CPE in to lahko naredi na dva načina:
 - a) S trdo ožičeno logiko (hitrejša rešitev, vendar manj fleksibilna za spreminjanje in dodajanje ukazov - potrebno je v celoti spremeniti CPE)
 - b) Mikroprogramiranje (počasnejša rešitev, a bolj fleksibilna – lažje spreminjanje, saj se spremeni le mikroprogram)
4. Vsak ukaz vsebuje podatke o:
 - a) Operaciji, ki jo je treba izvesti
 - b) Operandih, na katerih se bo operacija izvedla

Te informacije so določene z biti v posamezni poljih samega ukaza:

- a) Operacijska koda: polje, ki vsebuje podatke o operaciji
- b) Polja, ki vsebujejo informacije o operandih: lahko vsebujejo operand ali njegov naslov

V nekaterih ukazih so tudi podatki o operandih shranjeni v operacijski kodi.

5. Format ukazov določa delitev na polja za informacije, ki se nanašajo na ukaz. Odvisen je od števila operandov, števila registrov v CPU-ju, velikosti pomnilniške besede ...

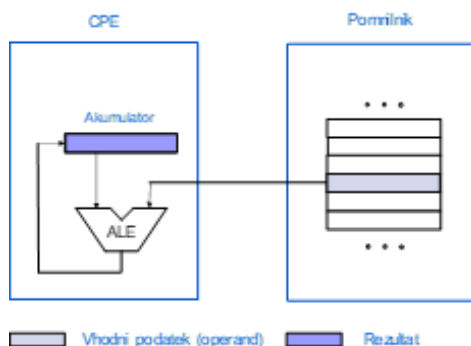


6. Operacija se lahko izvede na različne načine (razl. tipi, dolžine operandov), zaradi česar je število ukazov večje kot število operacij.
7. Osnovne lastnosti ukazov:
 - a) Načini shranjevanja operandov v CPE
 - b) Število eksplicitnih operandov v ukazu
 - c) Lokacija operandov in načini naslavljanja
 - d) Operacije
 - e) Vrsta in dolžina operandov

8. Načini shranjevanja operandov v CPE:

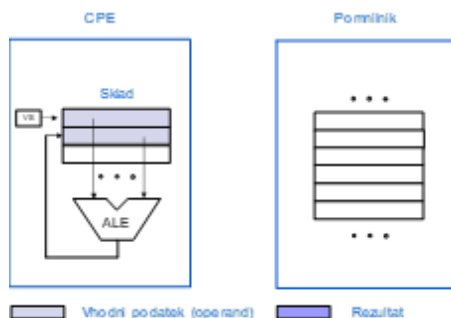
a) Akumulator – en register v CPE-ju

- Shrani samo en operand
- V večini primerov se rezultat shrani v akumulator
- Ukazi: Load, Store (prenos Pomn. – ACC)
- Krajši ukazi
- Veliko prenosov med CPE-jem in glavnim pomnilnikom



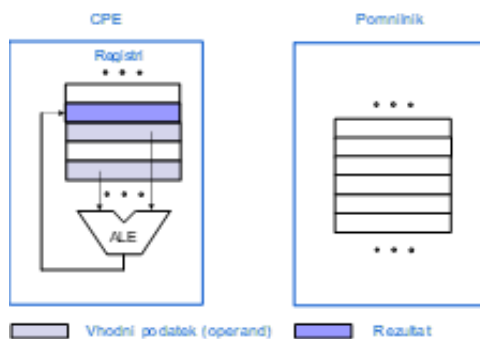
b) Sklad – lahko tudi v CPE-ju kot edini pomnilnik

- Dostopna je samo zgornja lokacija - vrh
- LIFO – Last in first out: Nove informacije shrani na vrhu, in tudi nalaga samo z vrha.
- Ukazi: Push, Pull/Pop



c) Množica registrov – v CPE

- Vsi so dostopni po njihovem enoličnem naslovu
- Večja hitrost
- Krajši ukazi
- Zmanjšan prenos med CPE-jem in glavnim pomnilnikom
- Dve vrsti registrov:
 - Splošno namenski registri – Vsi registri so ekvivalentni
 - Bazni ali indeksni registri – Računanje z naslovi



9. Število eksplicitnih operandov v ukazu:

- a) Manjše število
 - Krajši ukazi = manj prostora v pomnilniku
 - Manj zmogljivi ukazi
- b) Večje število operandov v ukazu
 - Daljši ukazi
 - Zmogljivejši ukazi

10. Večina osnovnih operacij ima do tri operande: 2 vhodna operanda in rezultat. Računalniki, ki uporabljajo ukaze z večinomoma m operandi, se imenujejo m-operandni računalniki (m-naslovni). Lahko so:

- a) 3+1 operandni računalniki
$$\begin{aligned}OP3 &\leftarrow OP2 \oplus OP1 && (\oplus \text{ pomeni poljubno operacijo}) \\PC &\leftarrow OP4\end{aligned}$$
- b) 3-operandni računalniki
$$\begin{aligned}OP3 &\leftarrow OP2 \oplus OP1 \\PC &\leftarrow PC + 1\end{aligned}$$
- c) 2-operandni računalniki
$$\begin{aligned}OP2 &\leftarrow OP2 \oplus OP1 \\PC &\leftarrow PC + 1\end{aligned}$$
- d) 1-operandni računalniki
$$\begin{aligned}AC &\leftarrow AC \oplus OP1 \text{ (AC is abbreviation for accumulator)} \\PC &\leftarrow PC + 1\end{aligned}$$
- e) Brez-operandni (Skladovni) računalniki
$$\begin{aligned}Stack_{TOP} &\leftarrow Stack_{TOP} \oplus Stack_{TOP-1} \\PC &\leftarrow PC + 1\end{aligned}$$

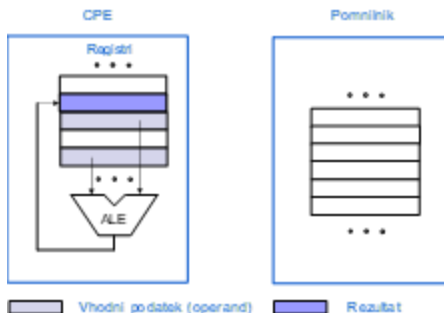
11. Operandi so lahko shranjeni v:

- a) Programsko dostopnih registrih v CPE
 - Registrski operandi
 - Naslov je podan v ukazu
- b) Sosednjih besedah glavnega pomnilnika
 - Pomnilniški operandi
 - Kompleksno naslavljanje
- c) Registrih krmilnika V/I naprave
- d) V samem ukazu
 - Takojšnji operandi

12. Glede na lokacijo operandov so računalniki lahko:

a) Registrsko-registrski računalniki

- Vsi operandi za ALE ukaze so shranjeni v registrih procesorja
- Za dostop do glavnega pomnilnika uporablja LOAD in STORE ukaze
- Čas izvajanja ALE ukazov je vedno enak

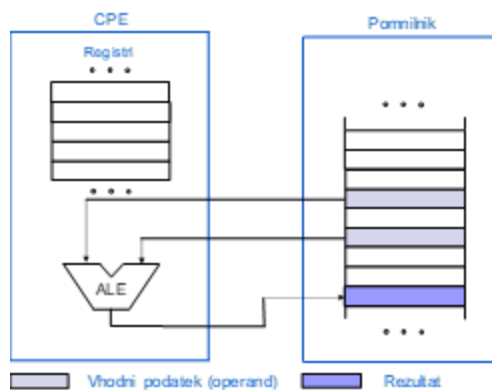


b) Registrsko-pomnilniški računalniki

- En operand je v pomnilniku ali registru, drugi pa v registrih
- Lahko uporabljamo ALE ukaze brez da bi vse operande prenesli v registre z ukazom load
- Daljši ukazi, vendar jih je v programu lahko manj
 - Čas izvajanja je odvisen od lokacije operanda

c) Pomnilniško-pomnilniški računalniki

- Vsak operand je lahko v pomnilniku ali v registrih
- Kompleksni ukazi
- Daljši čas izvajanja



13. Načini naslavljanja – uporabljajo se za naslavljanje operandov:

a) Takojšnje naslavljanje

- Operand je vključen v sam ukaz. Torej se imenuje takojšnji operand.
- MOV r0, #128

b) Neposredno naslavljanje

- Operand je predstavljen s svojim naslovom in se lahko nahaja v:
 - Registru
 - ADD R3, R2, R1
 - Pomnilniški besedi
 - LOAD r1, 20512

c) Posredno naslavljanje

- Operand je predstavljen z nekim drugim – posrednikom
- Posrednik je lahko :
 - Register
 - LDR R3, [R2]
 - Druga pomnilniška beseda
 - LOAD r1, @(20512)
- Uporabno za delo s polji, tabelami oz. nizi podatkov
- Variacije:
 - Bazno naslavljanje
 - Naslov baznega registra Rb, in odmik D
 - Naslov operanda = Rb+D
 - Indeksno naslavljanje
 - Naslov indeksnega registra Rx, odmik D1 enak vsoti vsebine baznega registra in odmika
 - Naslov operanda = Rx+D1 = Rx+Rb+D
 - Avtomatsko indeksiranje
 - Pred-dekrementno naslavljanje
 - Po-inkrementno naslavljanje
 - PC-relativno naslavljanje
 - Uporabljamo PC kot bazni register

14. Število operacij implementiranih z ukazi v računalniku:

- Množica operacij naj bo močna
- Operacije naj bodo podobne že uveljavljenim vrstam operacij

15. Osnovne skupine operacij (ukazov)

a) Aritmetične in logične operacije

- Lahko so skalarne in vektorske
- Aritmetične – osredotočene na cela števila
- Logične operacije
 - Postavljanje, brisanje posameznih bitov
 - Premiki (v bitih) – delitev z 2, množenje z 2...

b) Prenosi podatkov

- Prenos podatkov iz enega dela računalnika v drugega
- Podvajanje operandov
- LOAD, STORE, MOVE, PUSH, POP...

c) Kontrolne operacije

- Običajno: Programski števec $PC = PC + 1$
- 3 vrste izjem za gornje pravilo:
 - Pogojni skoki (if-then-else, zanke)
 - Brezpogojni skoki
 - Klici in vrnitve iz procedur ali podprogramov

d) Operacije v plavajoči vejici

- Enota za operacije v plavajoči vejici (Floating Point Unit – FPU) izvaja ALE operacije na FP zapisih
- Tudi: Kvadratni koren, logaritem, eksponente in trigonometrične funkcije

e) Sistemske operacije

- Privilegirani ukazi
- Spremenijo parametre delovanja računalnika in nadzorujejo njegovo delovanje
- Vplivajo na: Prekinitve, pasti, delovanje predpomnilnika, delovanje navideznega pomnilnika, nivo privilegiranosti, ustavitve delovanja

f) Vhodno/izhodne operacije

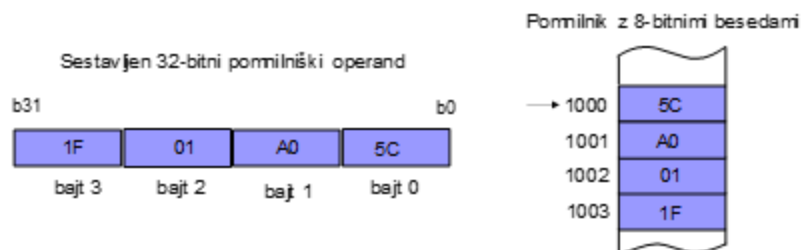
- Ukazi za V/I prenosi in prenosi izmed V/I napravami in CPE ali glavnim pomnilnikom
- Običajno privilegirani

16. Vrste operandov:

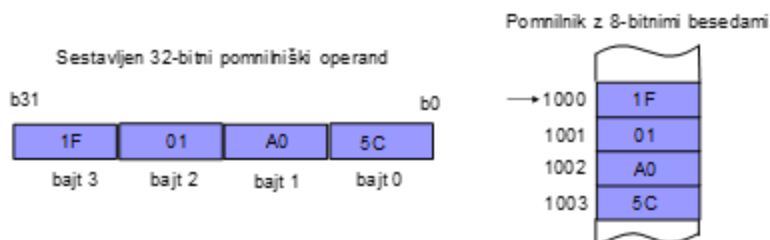
- a) Bit
- b) Znak - Character – 8 ali 16 bitov
- c) Celo število - Integer – 8, 16, 32 ali 64 bitov
- d) Realno število - Real number (FP) - 32, 64 or 128 bitov
- e) Desetiško število - Decimal number – Niz 8 bitnih znakov

17. Operandi, daljši od 8 bitov, zasedajo več pomnilniških besed, in jih imenujemo sestavljeni pomnilniški operandi. Besede morajo biti zaporedne in so lahko organizirane po:

- a) Pravilo tankega konca



- b) Pravilo debelega konca



18. Poravnava (align):

- a) Če so operandi na ustrezno poravnanih naslovih, so lahko dostopi hitrejši
- b) A se imenuje naravni (poravnan) naslov, če velja :
 - $A \bmod s = 0$,
 - kjer je A naslov, s pa število besed v operandu,

19. Dejavniki, ki določajo zgradbo ukazov:

- a) Dolžina pomnilniške besede
- b) Število eksplisitnih operandov v ukazu
- c) Vrsta in število registrov v CPE
- d) Dolžina pomnilniškega naslova

20. Ortogonalni ukazi:

- a) Informacija o operaciji je neodvisna od informacije o operandih
- b) Informacija o enem operandu v ukazu je neodvisna od informacije o drugih operandih

21. Računalniki glede števila ukazov:

- a) RISC – Reduced Instruction Set Computer
- b) CISC – Complex Instruction Set Computer

Zgodovinski razvoj :

- (do leta 1980) Razlogi za povečevanje števila ukazov (CISC računalniki):
 - o Semantični prepad
 - o Mikroprogramiranje
 - o Razmerje med hitrostjo glavnega pomnilnika in CPE
- (od leta 1980 naprej) Razlogi za zmanjšanje števila ukazov (RISC računalniki):
 - o Težave pri uporabi kompleksnih ukazov v prevajalnikih
 - o Spremenjeno razmerje med hitrostjo glavnega pomnilnika in CPE
 - o Lažje uvajanje paralelizma v CPE

Definicija: Računalnik ima RISC arhitekturo, če izpolnjuje naslednjih šest pogojev:

- Večina ukazov se izvrši v eni urini periodi (v enem CPE ciklu)
- Registrsko-registrska zasnova (load/store računalnik)
- Ukazi so realizirani s trdo ožičeno logiko in ne mikroprogramsko
- Malo ukazov in malo načinov naslavljanja
- Vsi ukazi imajo isto dolžino
- Dobri prevajalniki (upoštevajo zgradbo CPE)