

Racunarske funkcije I medjusobno povezivanje

Racunarske komponente

Racunarske funkcije I medjusobno povezivanje

- Skoro sve savremene racunarske arhitekture zasnovane su na konceptu John von Neumann-a I takva konstrukcija je poznata kao von Neumannova arhitektura I zasniva se na tri koncepta:

Racunarske komponente, pogled odozgo

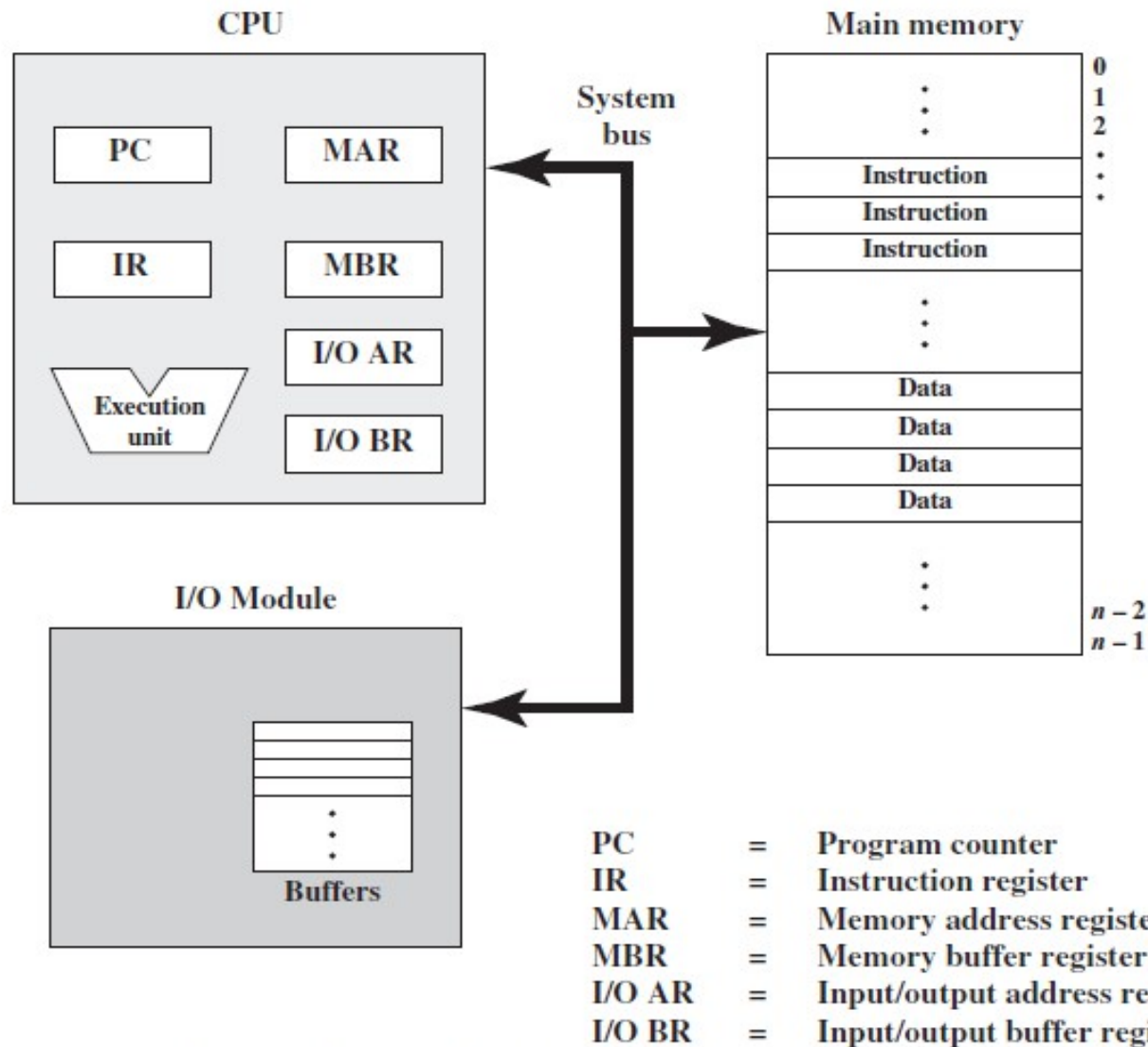


Figure 3.2 Computer Components:Top-Level View

Racunarske funkcije I medjusobno povezivanje

- Podaci I instrukcije se skladište u jednoj memoriji za citanje I upisivanje
- Sadržaj te memorije može da se adresira po lokaciji, bez obzira na vrstu podataka koji se tamo nalaze
- Izvršenje se desava na sekvencijalni način, izuzev ako se to eksplicitno ne promeni, od jedne do sledeće instrukcije

Racunarska funkcija

- Osnovna funkcija koju izvodi racunar je izvršenje programa koji se sastoji od niza instrukcija u memoriji.
- Obrada instrukcija u najjednostavnijem obliku sastoji se od dva koraka, citanje instrukcije iz memorije I izvorsavanje instrukcije.
- Program se izvrsava tako sto se ponavlja proces citanja I izvorsavanja instrukcija.

Racunarska funkcija

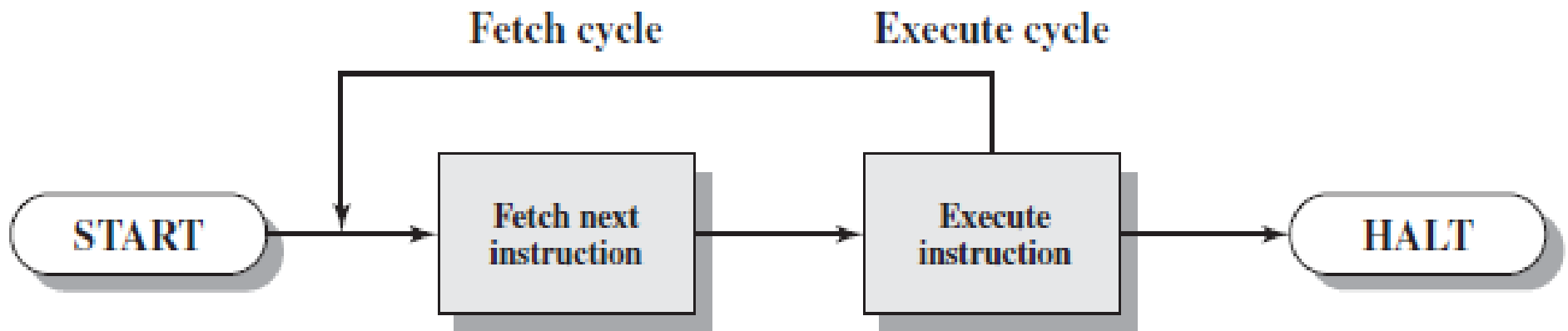


Figure 3.3 Basic Instruction Cycle

Ciklus donosenja I ciklus izvršenja,

Donosenje I izvršenje instrukcije

- PC, programski brojac cuva adresu instrukcije koja treba da se donese.
- Procesor uvek povecava sadrzaj PC posle svakog donosenja instrukcije, ukoliko mu se drugacije ne “kaze”.
- Donesena instrukcija se ucitava u registar u procesoru koji se naziva instrukcijski registar IR.
- Procesor interpretira instrukciju I izvodi zahtevanu akciju koja spada u jednu od cetiri kategorije:

Donosenje I izvršenje instrukcije

- Procesor-memorija: podaci mogu da se prenose od procesora do memorije ili obrnuto
- Procesor UI: podaci mogu da se prenose do periferije ili obratno, procesor UI modul
- Obrada podataka: procesor izvodi aritmetičke ili logičke operacije nad podacima
- Instrukcija može da odredi da se promeni sekvenca izvršenja

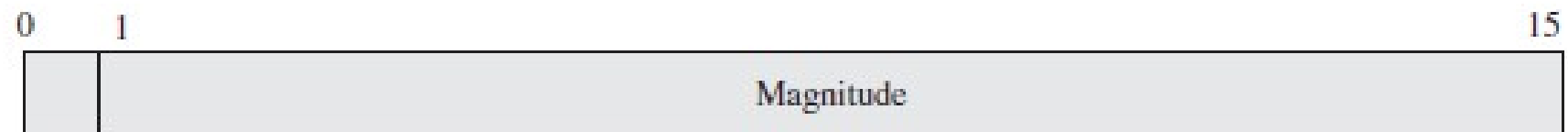
Donosenje I izvršenje instrukcije

- Posmatrajmo hipoteticku masinu na slici:

Donosenje I izvršenje instrukcije



(a) Instruction format



(b) Integer format

Program counter (PC) = Address of instruction
Instruction register (IR) = Instruction being executed
Accumulator (AC) = Temporary storage

(c) Internal CPU registers

0001 = Load AC from memory
0010 = Store AC to memory
0101 = Add to AC from memory

(d) Partial list of opcodes

Figure 3.4 Characteristics of a Hypothetical Machine

Donosenje I izvršenje instrukcije

- Procesor sadrzi jedan registar za podatke, akumulator, AC
- Instrukcije I podaci su duzine 16 bitova
- Memoriju treba organizovati da koristi 16-bitne reci.
- U formatu instrukcije 4 bita su kod operacije, pa moze da ima 16 razlicitih kodova operacije I 4096 ili 4K memorije koja moze direktno da se adresira.

Donosenje I izvršenje instrukcije

- Primer programa koji dodaje sadržaj memorijske reci na adresi 940 sadržaju memorijske reci na adresi 941 I skladišti rezultat na poslednju lokaciju.

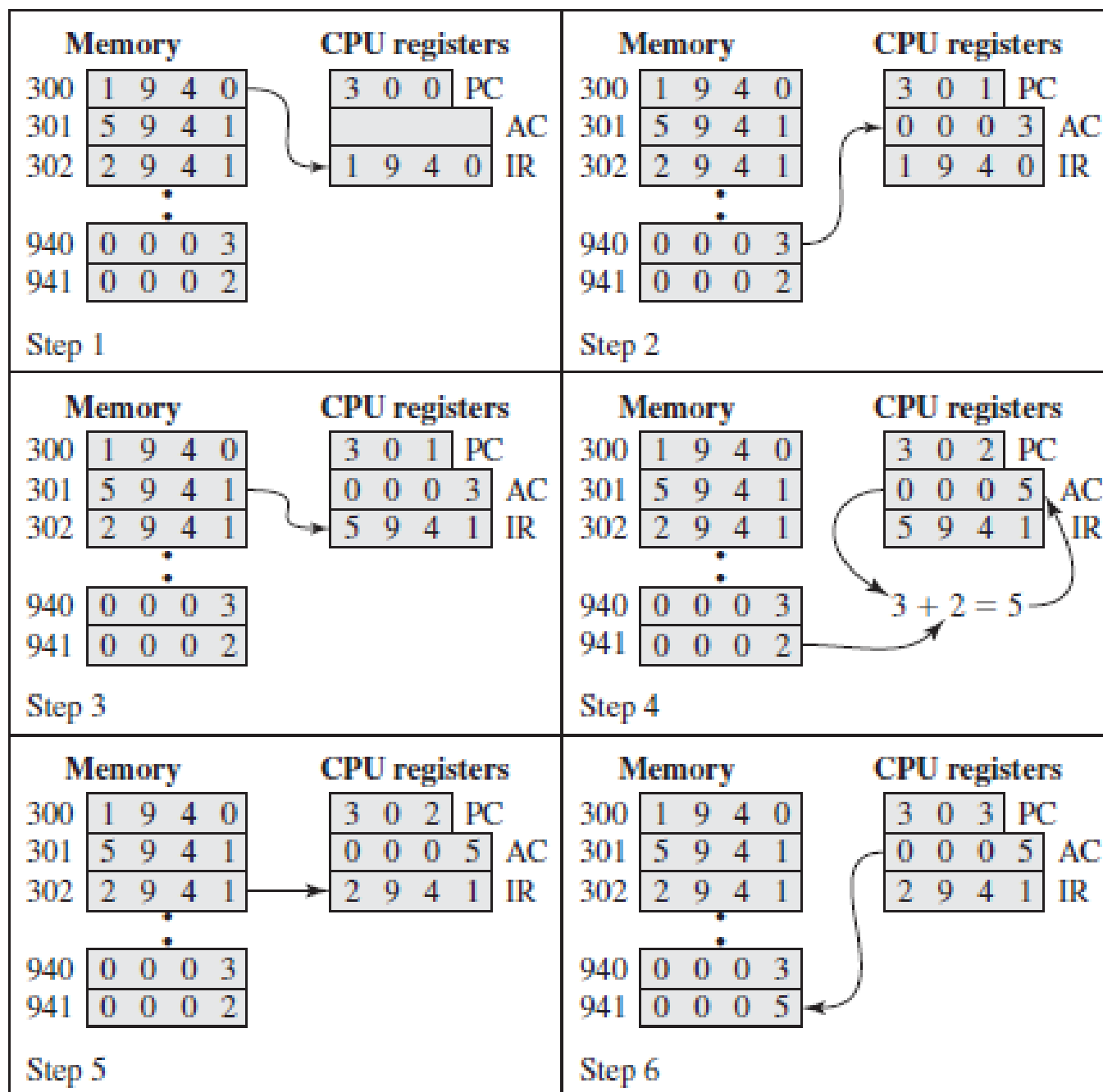


Figure 3.5 Example of Program Execution (contents of memory and registers in hexadecimal)

Dijagram stanja instrukcijskih ciklusa

- Instrukcijski ciklus za posebnu instrukciju može da obuhvati više od jednog pozivanja na memoriju.
- Umesto pozivanja na memoriju, instrukcija može da poziva UI modul.
- Detaljan opis instrukcijskog ciklusa prikazan je na sledećoj slici:

Dijagram stanja instrukcijskog ciklusa

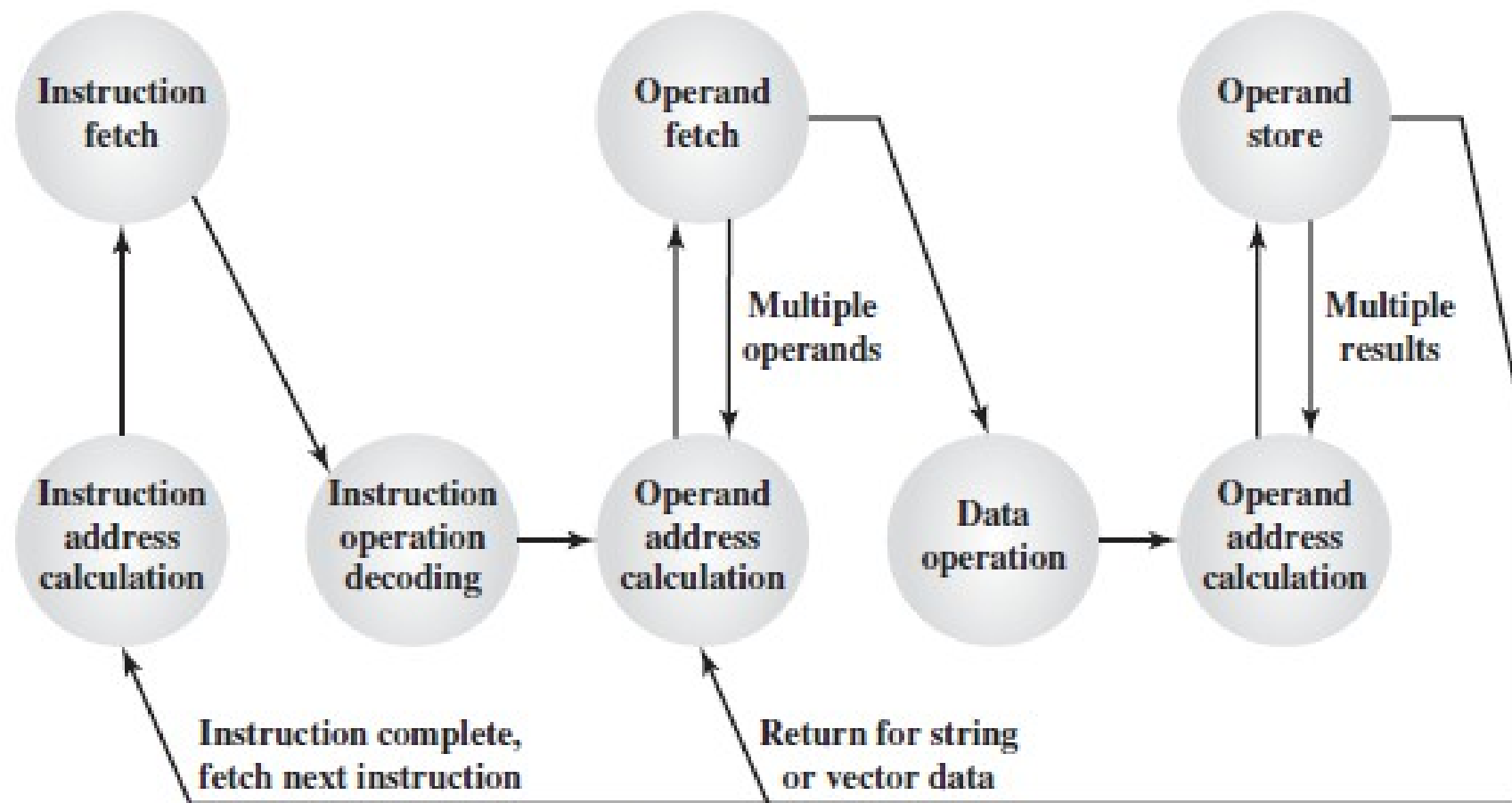


Figure 3.6 Instruction Cycle State Diagram

Dijagram stanja instrukcijskog ciklusa

- Proracun adrese instrukcije(iac) odredjuje se adresa sledece instrukcije za izvršenje. To znaci da se dodaje fiksni broj na adresu prethodne instrukcije. Ako je instrukcija 16 bitna, a memorija organizovana po 16 bitnim recima, onda se dodaje, 1, ako je memorija organizovana po 8 bitnim recima, onda se dodaje 2, a memoriji se pristupa pojedinačno adresibilnim bajtovima.

Dijagram stanja instrukcijskog ciklusa

- Donosenje instrukcije(if): instrukcija se učitava iz njene memorijske lokacije u procesor.
- Dekodiranje operacije instrukcije(iod): određuje se vrsta operacije za izvršenje.
- Proracun adrese operanda(oac): određivanje adrese operanda, memorija ili UI
- Donosenje operanda(of):operand se donosi iz memorije ili UI
- Operacija nad podacima(do): izvodi se operacija iz instrukcije
- Skladistenje operanda(os):upisuje se rezultat u memoriju ili u UI

Dijagram stanja instrukcijskog ciklusa

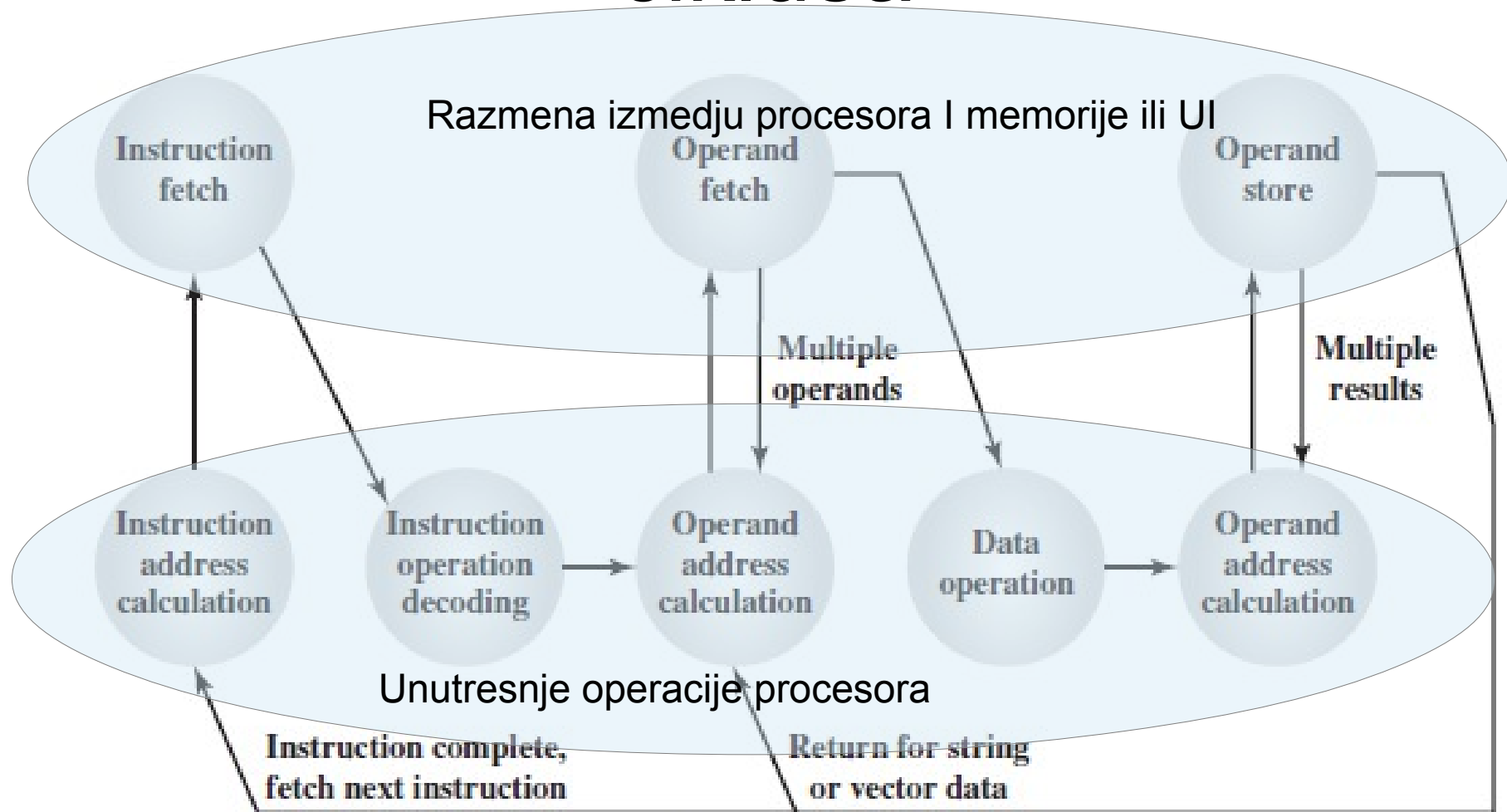


Figure 3.6 Instruction Cycle State Diagram