

Ispit iz Arhitekture i organizacije računara 2

Opis arhitekture i organizacije procesora

Procesor je jednodresni i ima 4 registra opšte namene, R0 do R3, svi su 8-bitni. Postoje i registri PSW i SP sa uobičajenim značenjem, kao i akumulator A. Memorijske adrese su širine 8 bita, širina magistrale podataka je 8 bita, a adresiranje je na nivou bajta. Procesor operiše samo sa 8-bitnim celobrojnim veličinama sa znakom i bez znaka. Vreme odziva memorije je neodređeno, magistrala je asinhrona.

Bitovi 7, 6, 5 i 4 prvog bajta instrukcije imaju vrednost 0000 za instrukcije skoka. Bitovima 3 do 0 prvog bajta instrukcije specificira se kod operacije za instrukcije skoka. Instrukcije uslovnog skoka se realizuju kao relativni skok u odnosu na tekuću vrednost programskog brojača PC, a pomeraj je 8-bitna celobrojna veličina sa znakom data 2. bajtom instrukcije. Instrukcije bezuslovnog skoka se realizuju kao apsolutni skokovi, a adresa skoka je data 2. bajtom instrukcije. Dužina instrukcija je 2 bajta.

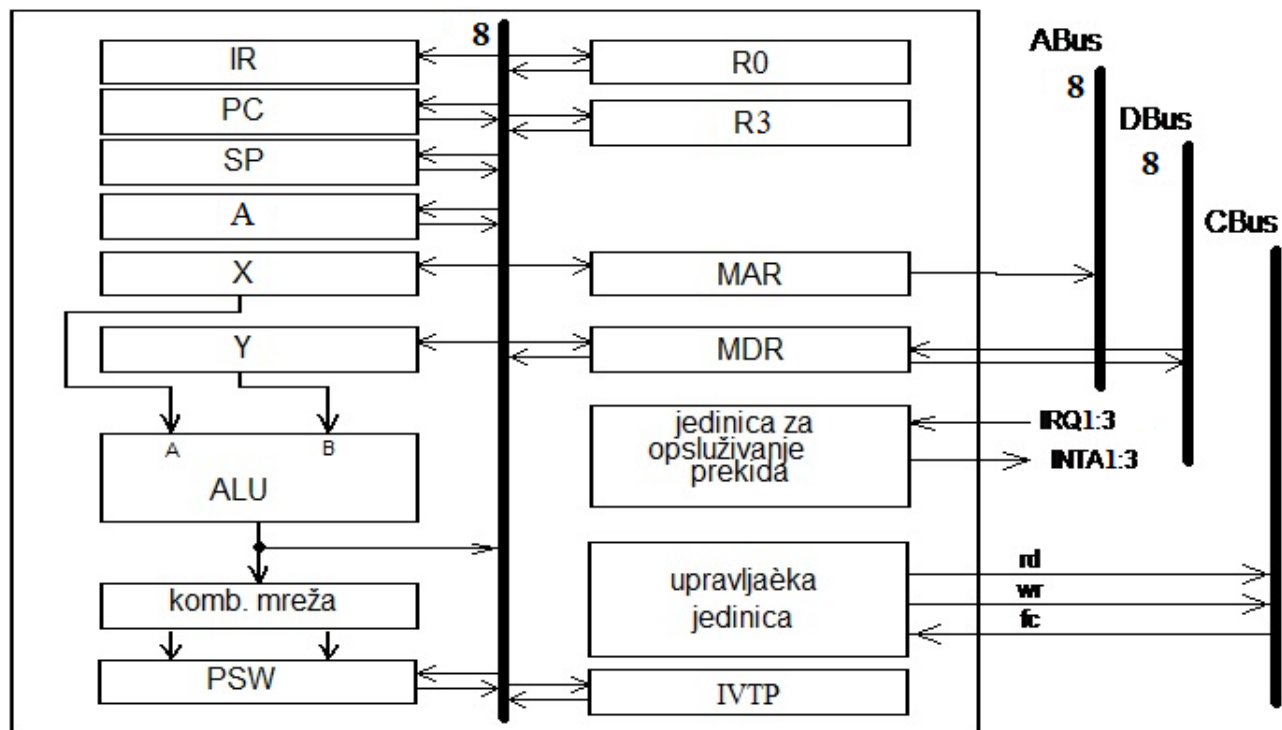
Bitovi 7, 6, 5 i 4 prvog bajta instrukcije imaju vrednost 1111 za bezadresne instrukcije. Bitovima 3 do 0 prvog bajta instrukcije specificira se kod operacije za bezadresne instrukcije. Dužina instrukcija je 1 bajt.

Bitovi 7, 6, 5 i 4 prvog bajta instrukcije u opsegu od 0001 do 1110 specificiraju kod operacije za adresne instrukcije. Dužina instrukcija je 1 ili 2 bajta i zavisi od specificiranog načina adresiranja. Načini adresiranja su specificirani bitovima 3 i 2 prvog bajta instrukcije. Postoje sledeći načini adresiranja: PC-relativno adresiranje, registarsko direktno adresiranje, memorijsko direktno adresiranje i memorijsko indirektno adresiranje. Kod PC-relativnog adresiranja adresa operand određuje se relativno u odnosu na tekuću vrednost programskog brojača PC, pri čemu je 8-bitni pomeraj dat 2. bajtom instrukcije. Bitovi 1 i 0 prvog bajta instrukcije se ne koriste. Dužina instrukcija je 2 bajta. Kod registarskog direktnog adresiranja bitovi 1 i 0 prvog bajta instrukcije se koriste za adresiranje jednog od registara opšte namene. Dužina instrukcija je 1 bajt. Kod memorijskog direktnog adresiranja adresa operanda je data 2. bajtom instrukcije. Bitovi 1 i 0 prvog bajta instrukcije se ne koriste. Dužina instrukcija je 2 bajta. Kod memorijskog indirektnog adresiranja adresa adrese operanda je data 2. bajtom instrukcije. Bitovi 1 i 0 prvog bajta instrukcije se ne koriste. Dužina instrukcija je 2 bajta.

Bezadresne instrukcije su RTS, RTI, INTE, INTD, TRPE, TRPD, ROLC i ASL. Instrukcije skoka su JMP, JSR, JZ i JNZ. Adresne instrukcije su instrukcija prenosa u akumulator (LOAD), instrukcija prenosa iz akumulatora (STORE), aritmetička instrukcija sabiranja (ADD), aritmetička instrukcija oduzimanja (SUB), logička instrukcija I (AND), logička instrukcija ILI (OR) i logička instrukcija ekskluzivno ILI (XOR).

Postoje spoljašnji maskirajući prekidi, za koje zahtevi dolaze po linijama IRQ1 do IRQ3 procesora, pri čemu IRQ3 ima najviši prioritet, a IRQ1 najniži prioritet. Prekidni mehanizam je vektorisan, a periferijama se mogu dodeliti proizvoljni ulazi u vektor tabeli. Vektor tabela počinje od adrese na koju pokazuje registar IVTP. Pretpostaviti da postoji kombinaciona mreža koja na internu magistralu postavlja vrednost ulaza odgovarajućeg prekida kada se generiše upravljački signal IVTEout. Pri prekidu se na steku čuvaju PC i PSW tim redom i maskirajući prekidi se onemogućavaju brisanjem bita I u registru PSW. Stek raste prema nižim adresama, a SP ukazuje na poslednju zauzetu lokaciju.

Organizacija procesora data je na slici 1. ALU ima, pored ostalih, i kontrolne ulaze *incA* i *decA* za inkrementiranje i dekrementiranje vrednosti na A ulazu.



Slika 1. Organizacija procesora

Zadatak:

a) (5p) Nacrtati strukturnu šemu kombinacione mreže koja generiše upravljačke signale za ALU, ali tako da sve naredbe koje koriste ALU za izračunavanje rezultata (aritmetičke i logičke) imaju jedinstven, zajednički mikrokôd, a ne svaka posebni deo mikroprograma.

b) (20p) Napisati mikroprogram za ovaj procesor, sa fazom izvršavanja samo za jednoadresne aritmetičke i logičke instrukcije (ADD, SUB, AND, OR i XOR), a predvideti postojanje ostalih. Mikroprogram napisati sa zajedničkim mikrokôdom za aritmetičke i logičke instrukcije, pretpostavljajući da postoji kombinaciona mreža realizovana u tački pod a). Kôd treba da bude prilagođen mikroprogramskoj upravljačkoj jedinici, pri čemu su mikronaredbe sa upravljačkim signalima i mikronaredbe koje definišu skok u mikroprogramu razdvojene. Ne treba pisati mikroprogram za obradu prekida. Dohvatanje eventualnog drugog bajta instrukcije treba da bude u fazi izvršavanja instrukcije.

c) (5p) Napisati na assembleru ovog procesora program koji umanjuje svaki element niza reči, koji počinje od adrese koja se nalazi u registru R1, za vrednost koja se nalazi na adresi 50h. Sadržaj registra R0 predstavlja dužinu niza. Na raspolaganju za korišćenje je lokacija FFh. Dozvoljeno je menjati vrednost registra R0 u toku izvršavanja programa. Registar R2 ima vrednost 01h.

Napomena: Kolokvijum traje 120 minuta. Nije dozvoljena upotreba literature.