

## Ispit iz Arhitekture i organizacije računara 2

### Opis arhitekture i organizacije procesora

Procesor je troadresni i ima 16 registara opšte namene, R0 do R15, svi su 32-bitni. Postoje i registri PSW i SP sa uobičajenim značenjem Memorijske adrese su širine 32 bita, širina magistrale podataka je 32 bita, a adresiranje je na nivou 32-bitnih reči. Procesor operiše samo sa 32-bitnim celobrojnim veličinama (u daljem tekstu *reč* označava 32-bitnu veličinu). Ulazno/izlazni i memorijski adresni prostori su memorijski preslikani. Vreme odziva memorije je neodređeno, magistrala je asinhrona.

Prekidni mehanizam je vektorisan. Vektor tabela počinje od adrese na koju pokazuje registar IVTP. Postoje spoljašnji maskirajući prekidi, za koje zahtevi dolaze po linijama IRQ0 do IRQ2 procesora, pri čemu IRQ0 ima najviši prioritet, a IRQ2 najniži prioritet. Periferije PER0 do PER2 vezane su na linije IRQ0 do IRQ2, respektivno, i dodeljeni su im ulazi 1, 2 i 3 u vektor tabeli. 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 višim adresama, a SP ukazuje na prvu slobodnu lokaciju na steku.

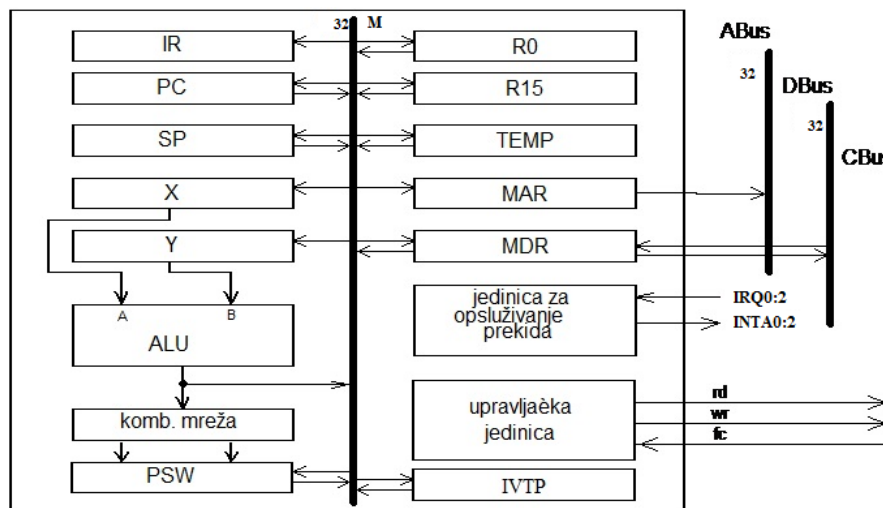
Postoje sledeće grupe instrukcija: troadresne instrukcije (aritmetičke, logičke itd.), dvoadresne instrukcije za prenos podataka (LOAD i STORE), jednoadresne instrukcije (CLR, INC, DEC, PUSH, POP itd.), instrukcije uslovnog i bezuslovnog skoka i ostale (manipulacije indikatorima, poziv potprograma, povratak iz potprograma ili prekida itd.). Instrukcije su dužine jedne ili dve reči. U prvoj reči su uvek kôd operacije i informacije o načinu adresiranja operanada.

Način adresiranja	Primer u assembleru
Neposredno	LOAD R0, #00001234h
Memorijsko direktno	STORE R2, 00000100h
Registarsko direktno	STORE R1, R3
Registarsko indirektno	STORE R2, (R7)
Registarsko indirektno sa pomerajem	LOAD R1, Pom(R1)

Slika 1. Načini adresiranja

Kod svih troadresnih i jednoadresnih instrukcija operandi i odredište su isključivo u registrima R0 do R15. Kod dvoadresnih instrukcija za prenos podataka (LOAD i STORE), odredište za LOAD i izvorište za STORE su isključivo registri R0 do R15 (polje REG1 u prvoj reči instrukcije). Ovaj operand biće nazivan prvim operandom. Drugi operand je neposredni podatak u drugoj reči instrukcije (samo za LOAD), u nekom od registara R0 do R15 (polje REG2 u prvoj reči instrukcije), ili u memoriji. Samo ove instrukcije operišu podacima u memoriji. Postoji pet načina adresiranja drugog operanda, kao što je prikazano na slici 1.

Organizacija procesora data je na slici 2. ALU ima, pored ostalih, i kontrolne ulaze *inca* i *add*.



Slika 2. Principijelna šema organizacije procesora

**Zadatak (30p):**

a) **(5p)** Nacrtati deo strukturne šeme mreže za obradu prekida koji na internu magistralu postavlja broj ulaza u IV tabeli prilikom zahteva za prekidom, kada je aktivan signal IVTEout. Pretpostaviti da postoje signali IRM0 do IRM2, koji odgovaraju linijama IRQ0 do IRQ2, respektivno, i koji predstavljaju aktuelno stanje zahteva za prekid prilikom obrade prekida i koje treba koristiti za određivanje broja ulaza u vektor tabeli prilikom obrade prekida. Ostatak šeme za obradu prekida ne treba crtati.

b) **(20p)** Napisati mikroprogram za ovaj procesor, sa fazom izvršavanja samo za instrukciju LOAD (za sve načine adresiranja), a predvideti postojanje ostalih. Kôd treba da bude prilagođen mikroprogramskoj upravljačkoj jedinici, pri čemu se u jednoj mikronaredbi nalaze i polje sa upravljačkim signalima i polja koja definišu uslovni skok u mikroprogramu. Treba pisati mikroprogram za obradu prekida. Dohvatanje eventualne druge reči instrukcije treba da bude u fazi izvršavanja instrukcije.

Pretpostaviti da postoji kombinaciona mreža koja povezuje izlaze registra  $R_i$  ( $i = 0 \dots 15$ ) sa internom magistralom M kada je aktivan upravljački signal REGout i jedan od signala regsel1, regsel2 i regsel3, koji služe za selekciju registra pomoću polja REG1, REG2 i REG3 instrukcijske reči, respektivno. Pretpostaviti da postoji kombinaciona mreža koja povezuje internu magistralu M sa ulazima registra  $R_i$  ( $i = 0 \dots 15$ ) kada je aktivan upravljački signal REGin i jedan od signala regsel1, regsel2 i regsel3, koji služe za selekciju registra pomoću polja REG1, REG2 i REG3 instrukcijske reči, respektivno.

c) **(5p)** Napisati na assembleru ovog procesora program koji uvećava svaki neparan element niza reči, koji počinje od adrese 100h, za vrednost koja se nalazi u registru R4, odnosno umanjuje svaki paran element niza reči za vrednost koja se nalazi u registru R4. Niz je dugačak onoliko koliko pokazuje sadržaj registra R3.

Smatrati da je prvi element niza reči (na adresi 100h) neparan, sledeći (101h) je paran i tako naizmenično. Na raspolaganju za korišćenje su još registri R0, R1 i R2. Sadržaj registra R3 ne mora da ostane nepromenjen nakon izvršavanja. U ovoj tački smatrati da postoje načini adresiranja sa slike 1 izuzev registarskog indirektnog adresiranja.

*Napomena: Ispit traje 120 minuta. Nije dozvoljena upotreba literature.*