

Ispit iz Arhitekture i organizacije računara 2

Procesor je troadresni i ima 16 registara opšte namene, R0 do R15, svi su 32-bitni. Postoji i registar PSW sa statusnim indikatorima, ali bez uobičajenih indikatora N, Z, C i V. Registar R15 služi kao pokazivač steka. 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 adresni prostor je memorijski mapiran. Vreme odziva memorije je neodređeno, magistrala je asinhrona.

Postoje sledeće grupe instrukcija: troadresne instrukcije (aritmetičke, logičke itd.), dvoadresne instrukcije za prenos podataka (LOAD, STORE), jednoadresne instrukcije (CLR, PUSH, POP itd.), instrukcija uslovnog skoka i ostale (poziv potprograma, povratak iz potprograma ili prekida itd.). Troadresne instrukcije operišu isključivo nad registrima opšte namene. Instrukcije su dužine jedne ili dve reči. U prvoj reči su uvek kôd operacije i informacije o načinu adresiranja operanada. Prvi operand je odredišni.

Instrukcija LOAD ima dva operanda, pri čemu je prvi (odredišni) jedan od registara R0..R15, a drugi može biti u svim dozvoljenim načinima adresiranja. Za instrukciju STORE važi analogno, samo što je drugi operand izvorišni, jedan od R0..R15. Ostale instrukcije za operande imaju samo R0..R15. Načini adresiranja su sledeći:

Način adresiranja	Primer u assembleru
Neposredno	LOAD R0, #1234h
Registarsko direktno	STORE R1, R3
Registarsko indirektno	STORE (R2), R7
Registarsko indirektno sa pomerajem	LOAD R1, Pom(R1)

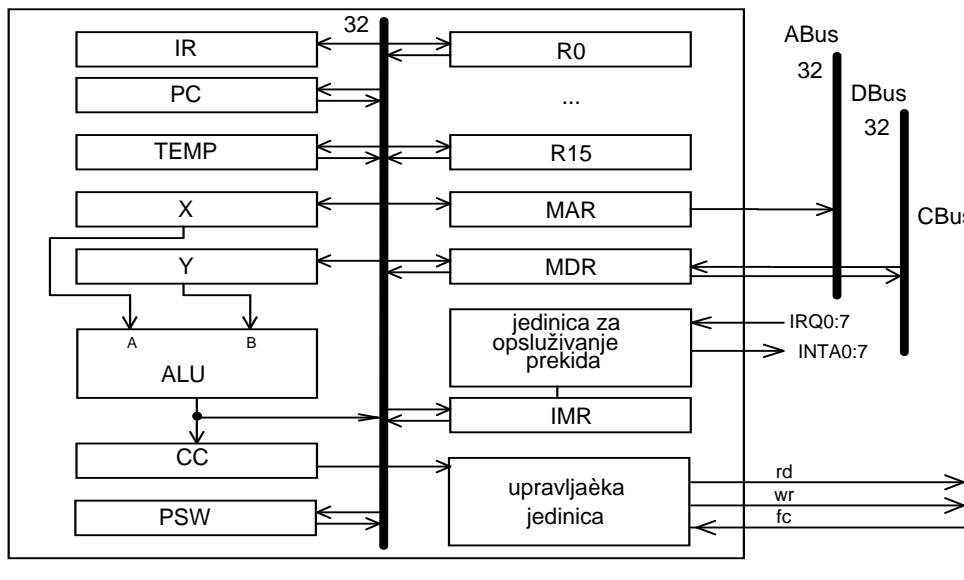
Umesto uobičajenih indikatora N, Z, C i V u PSW, kao podršku uslovnom izvršavanju procesor poseduje poseban skup od 16 tzv. uslovnih jednobitnih registara CC0..CC15. Na ove registre utiču uslovne instrukcije iz date tabele na sledeći način: ukoliko je relacija između operanada zadovoljena, uslovni registar specifikovan instrukcijom postavlja se na 1, inače se postavlja na 0. Registar CC0 je ožičen na 1, tako da se njegova vrednost ne može menjati. Operandi mogu biti samo registri R0..R15. Format uslovnih instrukcija dat je na slici.

Asemblerska oznaka	Code	Značenje
EQ CCi, Rj, Rk	0001	CCi := (Rj=Rk)
NE CCi, Rj, Rk	0000	CCi := (Rj≠Rk)
LT CCi, Rj, Rk	0010	CCi := (Rj<Rk)
GT CCi, Rj, Rk	0101	CCi := (Rj>Rk)
LE CCi, Rj, Rk	0100	CCi := (Rj≤Rk)
GE CCi, Rj, Rk	0011	CCi := (Rj≥Rk)

31	15	11	7	3	0
OpCode=FF00h	Code	CCi	REGj	REGk	

Svaka aritmetičko/logička operacija se zapravo uslovno izvršava: ona se izvršava samo ako je vrednost uslovnog registra koji je specifikovan jednaka 1, odnosno 0, u zavisnosti od bita 16 instrukcijske reči. Tako svaka aritmetičko-logička instrukcija ima zapravo četiri operanda: dva izvorišna registra, jedan odredišni registar i jedan uslovni registar. Na primer: ADD CC1,R3,R2,R1 znači: ako je CC1=1, onda R3:=R2+R1; ADD !CC1,R3,R2,R1 znači: ako je CC1=0, onda R3:=R2+R1. Kôd registra CCi je uvek u bitima 11..8 instrukcije.

Organizacija procesora data je na slici. ALU jedinica ima sledeće kontrolne ulaze: *add*, *sub*, *and*, *or*, *xor* i *incA*. Izlaz ALU je 32-bitni rezultat F i prenos/pozajmica C16. Pomoću odgovarajuće kombinacione mreže selektuje se CCi koji je specifikovan u instrukciji kako bi se u njega upisala odgovarajuća vrednost dobijena kombinacionom mrežom koja se nalazi na ulazu bloka označenog sa CC, odnosno da bi se u upravljačku jedinicu prosledio jednobitni signal CCi.



Zadatak:

a)(9) Nacrtati detaljnu strukturnu šemu bloka CC. Ulazi u ovaj blok su signali F (izlaz ALU), C32 (izlazni prenos/pozajmica iz ALU), IR11..IR8 (polje CCi instrukcijske reči), IR15..IR12 (polje Code instrukcijske reči), bit IR16 i CCin (upravljački signal za upis vrednosti u CCi). Blok sadrži 16 jednobitnih registara CC15..CC0. Izlaz bloka je jednobitni signal CC koji predstavlja vrednost selektovanog registra CCi koji ulazi u upravljačku jedinicu i signal cond koji definiše da li aritmetičko-logička instrukcija treba da se izvrši.

b)(13) Napisati mikroprogram za ovaj procesor, sa fazom izvršavanja **samo** za uslovne instrukcije iz date tabele i sve binarne aritmetičko-logičke instrukcije, 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. **Ne treba** pisati mikroprogram za obradu prekida. Pretpostaviti da je dohvaćanje eventualne druge reči instrukcije u fazi izvršavanja instrukcija koje poseduju te reči.

c)(9) Napisati na assembleru ovog procesora deo koda koji odgovara sledećem segmentu programa u višem programskom jeziku:

```
a:=b+c;
if (b>=c) then a:=a+5
else a:=a-6;
```

Promenljive a, b i c su celobrojne i nalaze se u memoriji na apsolutnim adresama simbolički označenim sa a, b i c redom (smestiti ih u registre R2, R3 i R4 redom).

Na ovom primeru uporediti opisani koncept uslovnog izvršavanja sa klasičnim konceptom postojanja indikatora u PSW i instrukcija uslovnog skoka prema vrednostima tih indikatora. Prokomentarisati broj instrukcija i vreme izvršavanja datog dela programa u oba slučaja.

Ispit traje 120 minuta.

Na ispitu se može osvojiti maksimalno 30 poena.