

Arhitektura računala 2

Međuispit, problemski dio (60% bodova)

1. (10 bodova) Skiciraj sadržaj korisničkog stoga u točkama A, B i C ako je poznato i) da se navedeni program u C-u izvodi na 32-bitnoj arhitekturi, te ii) da se parametri potprograma prenose preko stoga.

```
void* alociraj(int n){
    void * buf=malloc(n);
    memset(buf, 0xff, n);
    // B
    return buf;
}
int main(){
    // A
    alociraj(20);
    // C
}
```

2. (10 bodova) Pretpostavimo da modelu 8-instrukcijskog procesora želimo dodati jedan registar opće namjene (registar B) koji bi trebao biti u potpunosti ravnopravan registru A. Pokaži koje promjene je potrebno provesti u instrukcijskoj arhitekturi, na putu podataka te u upravljačkoj jedinici kako bi se postigao taj cilj.
3. (10 bodova) Prikažite sadržaj relevantnog dijela memorije i nacrtajte stanje na vanjskim sabirnicama pojednostavnjenog modela mikroprocesora tijekom izvođenja sljedećeg programskog odsječka:

```
$0100:  lda  #0
        add  $0200
        dec  $0200
        bne  $0100
        ...
$0200:  $03
```

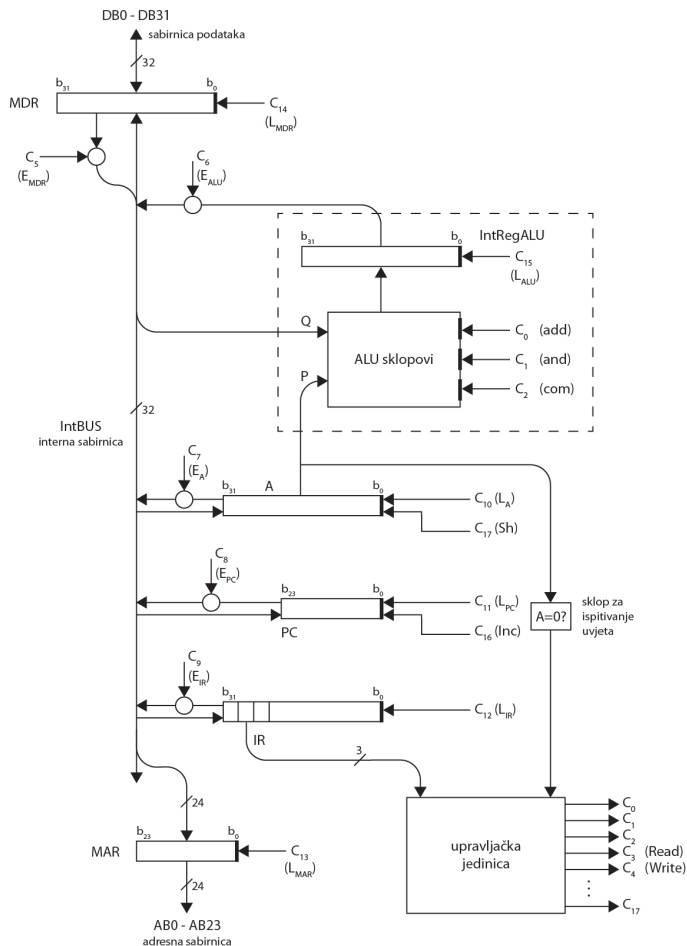
Instrukcija `lda` (op. kod: \$95) učitava 8-bitnu konstantu u akumulator. Instrukcija `add` (op. kod: \$b1) dodaje operand akumulatoru. Instrukcija `dec` (op. kod: \$12) umanjuje operand za jedan. Instrukcija `bne` (op. kod: \$48) grana na zadanu adresu ako je rezultat prethodne operacije bio različit od nule.

4. (10 bodova) Na raspolaganju su memorijski moduli sa sljedećim važnim priključcima: A0-A8, D0-D7, R/W*, CS0, CS1, CS2*.
- (a) Odredite kapacitet pojedinačnih modula.
 - (b) Nacrtajte shemu priključenja triju modula na 16-bitnu adresnu i 8-bitnu podatkovnu sabirnicu, ako moduli trebaju zauzeti kontinuirani adresni potprostor (raspon adresa) počevši od adrese \$fa00 (koristiti potpuno dekodiranje adrese).
 - (c) Koji adresni potprostor zauzimaju moduli ako ih spojimo prema prethodnom podzadatku?
5. (10 bodova) Za model mikroprogramiranog procesora koji je zadan slikom i formatom mikroriječi, napisati mikroprogram i odrediti sadržaj mikroprogramske memorije za fazu izvrši instrukcije `swap A,B`, koja zamjenjuje sadržaje u registrima A i B.
- Neka je operacijski kod instrukcije `$bc`, a početna adresa mikroprograma za fazu PRIBAVI je `$f0`.

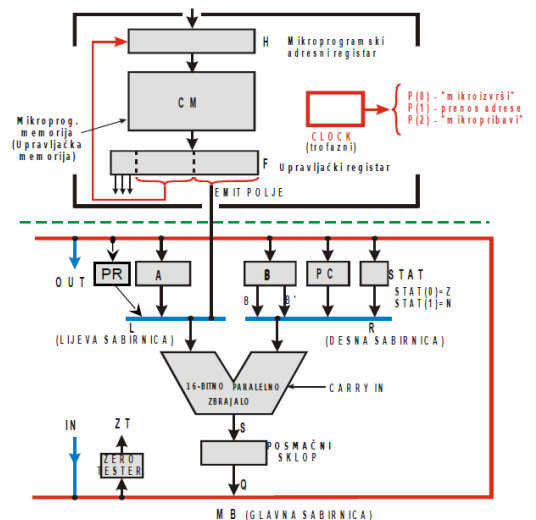
6. (10 bodova) Procesor MC68000 poziva potprogram instrukcijom \$2300: jsr \$3200. Pretpostaviti da operacijski kod odnosno operand instrukcije zauzimaju ukupno 6 bajta. Neka se parametri potprograma ne prenose preko stoga, te neka potprogram koristi dvije memorijske lokalne varijable širine po jedan bajt.

Neposredno nakon izvršavanja instrukcije potprograma \$3300: move d5,d3 procesor prima zahtjev za prekid te se poziva prekidni potprogram na adresi \$fe080.

Skicirati stanja stogova procesora prije poziva potprograma, neposredno prije primanja prekida, tijekom obrade prekida, nakon povratka iz prekidnog potprograma, te nakon povratka u glavni program.



Slika uz zadatak 2: organizacija osaminstrukcijskog procesora



Slika uz zadatak 5: organizacija mikroprogramiranog procesora

31	29	26	24	22	19	17	15	13	7	0
CA	CB	COP	CSH	CMB	CAB	CBB	CST	CNA	CEM	

CA	CB	COP	CSH
00 ... L ← PR	000 ... R ← 0	00 ... suma uz C=0	00 ... MB ← S
01 ... L ← [0, F(CEM)]	001 ... R ← B	01 ... suma uz C=1	01 ... MB ← shr S
10 ... L ← [F(CEM), 0]	010 ... R ← B*	10 ... ne koristi se	10 ... MB ← shl S
11 ... L ← A	011 ... R ← PC	11 ... ne koristi se	11 ... MB ← IN
	100 ... R ← SR		
CMB	CAB	CBB	CST
000 ... nema prijenosa	00 ... H(1) ← 0	00 ... H(0) ← 0	00 ... SR se ne mijenja
001 ... A ← MB	01 ... H(1) ← 1	01 ... H(0) ← 1	01 ... SR(0) ← ZT
010 ... B ← MB	10 ... H(1) ← SR(0)	10 ... H(0) ← SR(1)	10 ... SR(1) ← MB(15)
011 ... PC ← MB	11 ... H(1) ← MB(0)	11 ... H(0) ← MB(15)	11 ... SR(0) ← ZT
100 ... SR ← MB			SR(1) ← MB(15)
101 ... OUT ← MB			
110 ... PR ← MB			

Slika uz zadatak 5: format mikroinstrukcijske riječi