

1. Trajanje ispita 180 minuta.
2. Odgovori se daju u vežbanci ili na formularu.
3. Na naslovnoj strani obavezno zaokružiti redne brojeve zadataka koji su rađeni.

1. KOLOKVIJUM

1. [10] Deo koda napisan u višem programskom jeziku napisati u assembleru RISC V. Podrazumevati da su označeni brojevi button i amt u registrima s0 i s1. Jasno komentarisati kod.

```
switch (button) {
    case 1: amt = 0x 0000 0020; break;
    case 2: amt = 0x FFFF FFF0; break;
    case 3: amt = 0x FEED A987; break;
    default: amt = 0;
}
```

2. [20] Nacrtati realizaciju dela višeciklusnog RISC V procesora koji izvodi sledeću instrukciju.

Address	Instruction	Type	Fields				Machine Language	
			imm _{11:0}	rs1	f3	rd	op	
0x1000	L7: lw x6, -4(x9)	I	111111111100	01001	010	00110	0000011	FFC4A303
	[x9] = 0x 0000 2004							
	[0x2000] = 0x 0000 000A							

Definisati potrebne signale koje treba da generiše kontrolna jedinica kao i njihov vremenski redosled i razmak.

3. [20] Namenski sistem koristi procesor baziran na 32bitnoj RISC-V arhitekturi instrukcijskog seta. Poznato je da je memorija povezana sa procesorom preko 32bitne magistrale koja ima odvojene putanje za podatke i za adrese. Inicijalni sadržaj dela memorije namenske platforme dat je u tabeli 3.1. Nakon dekodovanja sadržaja dela memorije uspešno su dekodovane neke assembly instrukcije predstavljene u okviru *Dissassembly 3.1*.

Dissassembly 3.1

```
0x00: -----
0x04: addi x2, x0, 52
0x08: -----
0x0C: -----
0x10: lb x5, 0(x2)
0x14: -----
0x18: addi x3, x3, -1
0x1C: addi x1, x1, -1
0x20: addi x2, x2, 1
0x24: -----
0x28: jal x0, 0
```

Tabela 3.1

Adresa	Sadržaj				Adresa	Sadržaj			
0x00	0x93	0x00	0x40	0x00	0x24	0x6f	0xf0	0x9f	0xfe
0x04	0x13	0x01	0x40	0x03	0x28	0x6f	0x00	0x00	0x00
0x08	0x93	0x01	0xb0	0x03	0x2C	0x00	0x00	0x00	0x00
0x0C	0x63	0x8e	0x00	0x00	0x30	0x00	0x00	0x00	0x00
0x10	0x83	0x02	0x01	0x00	0x34	0x05	0x07	0x01	0x03
0x14	0x23	0x80	0x51	0x00	0x38	0x00	0x00	0x00	0x00
0x18	0x93	0x81	0xf1	0xff	0x3C	...			
0x1C	0x93	0x80	0xf0	0xff					
0x20	0x13	0x01	0x11	0x00					

Ako je nakon sistemskog reseta, PC registar CPUa inicijalizovan na vrednost 0, popuniti tabele 3.2, i 3.3 (pogledati formular za odgovore) za svaku od instrukcija. U okviru tabele 3.3 potrebno je dekodovati samo one instrukcije koje nisu navedene u okviru *Dissassembly 3.1* (instrukcije na adresama 0x00, 0x08, 0x0C, 0x14, 0x24).

Napomena: Ukoliko ispred brojnih vrednosti stoji prefiks 0x smatrati da su te brojne vrednosti date u heksadecimalnom brojnom sistemu dok se u suprotnom može smatrati da su vrednosti date u decimalnom brojnom sistemu. Izvršavanje programa se analizira dok se instrukcija sa iste memorijske lokacije ne izvrši uzastopno više od 2 puta. Broj redova u tabelama 3.2 i 3.3 je proizvoljan i ne mora odgovarati broju iteracija potrebnih za izvršavanje programa.

2. KOLOKVIJUM

4. Vrednosti parametara hijerarhijski organizovanog memorijskog dela sistema, sa jednim stepenom hijerarhije, su:
- kapacitet glavne memorije (MC) = 256B;
 - vreme pristupa glavnoj memoriji ($T_{Penalty}$) = 150 T_{CLK} ;
 - veličina bloka u kešu (BS) = 4B;
 - adresibilna jedinica (AUS) = 1B;
 - kapacitet keš memorije (CC) = 16B;
 - vreme pristupa keš memoriji (T_{HIT}) = 4 T_{CLK} ;
 - keš memorija je organizovana kao **direktno mapirani keš** čiji kontroler primenjuje *write back* – *write allocate* polisu upisa;
 - keš memorija je integrisana u sistem koristeći *look through* topologiju
 - inicijalni sadržaj glavne memorije definisan je tabelom 3.1.

Za program **P** koji se izvršava na ovoj namenskoj platformi poznato je da sekvencijalno pristupa sledećim adresama (R u indeksu označava čitanje sa memorijske lokacije definisane u uglastim zagradama dok W u indeksu označava upis podatka sa desne strane znaka = na memorijsku lokaciju u uglastim zagradama):

$$M_R[0x08], M_R[0x0A], M_R[29], M_W[2] = 0x12, M_R[15], M_R[0x26], M_W[0] = 0xCD, M_W[45] = 0xDA.$$

- a) [11] U tabeli 4.2 najpre kreirati okvir tabele koji ilustruje organizaciju keša a zatim predstaviti sadržaj kreirane tabele za svaki od adresnih ciklusa generisanih od strane procesora. U tabeli 4.1 predstaviti sadržaj dela glavne memorije nakon završetka transakcije na magistrali.
- b) [2] Koliko iznosi *hit rate*? Izračunati koliko iznosi AMAT.
- c) [4] Pod pretpostavkom da je magistrala između procesora i keš memorije, kao keš memorije i glavne memorije, širine 4B (sistemska magistrala), koliko zahteva se generiše između procesora i keš memorije (NCC\$) a koliko između keš memorije i glavne memorije (NC\$MM)? Prokomentarisati da li postoji mogućnost da se smanji broj ciklusa na sistemskoj magistrali.
- d) [3] Ukoliko se hijerarhijski organizovana memorija iz tačke a) proširi sa još dva dodatna stepena hijerarhije (L2 i L3) izračunati AMAT u tom slučaju ukoliko je poznato: vreme pristupa keš memoriji na L2 nivou (T_{HIT-L2}) = 10 T_{CLK} ; globalni *miss rate* na L2 nivou ($r_{GLOBAL-L2}$) = 15%; vreme pristupa keš memoriji na L3 nivou (T_{HIT-L3}) = 56 T_{CLK} ; *miss rate* na L3 nivou ($r_{MISS-L3}$) = 0.5%.

Napomena: Ukoliko ispred brojnih vrednosti postoji prefiks 0x smatrati da su te brojne vrednosti date u heksadecimalnom brojnem sistemu dok se u suprotnom može smatrati da su vrednosti date u decimalnom brojnem sistemu. Ukoliko je potrebno izvršiti zamenu bloka u keš memoriji, iz keš memorije se izbacuje onaj blok koji je prvi dodat u keš memoriju.

5. [15] Nacrtati realizaciju memorijskog podsistema u računaru sa 32-bitnom asinhronom magistralom (asinhroni pristup je kontrolisan signalom Ready sa aktivnim nivoom) koja ima 20-bitnu adresnu magistralu i mogućnost upravljanja bajtovima. Veličina potrebne memorije je 64kB i nalazi se u najvišem delu memorijskog prostora. Na raspolaganju su standardne memorijske komponente SRAM tipa veličine 16k x 8. Dozvoljen je samo poravnat pristup memoriji. U slučaju 16-bitnog pristupa za A1A0=00, podatak ide po nižem delu magistrale, a za A1A0=10 po višem delu magistrale. Memorija je dovoljno brza. U slučaju 16-bitnog pristupa za A1A0=00, podatak ide po delu magistrale D0..D15, a za A1A0=10 po delu magistrale D16..D31. U slučaju 8-bitnog pristupa za A1A0=00, podatak ide po delu magistrale D0..D7, za A1A0=01 po delu magistrale D8..D15, za A1A0=10 po delu magistrale D16..D23 i za A1A0=11 po delu magistrale D23..D31.

6. [15] Nacrtati opštenamenski prekidni kontroler sa 8 ulaza koji može da prihvati prekide i sa aktivnim nivoom logičke jedinice i sa aktivnom usponskom ivicom. Posle svakog prihvaćenog prekida svi prekidi istog i nižeg prioriteta se automatski maskiraju.