

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] Nacrtati realizaciju memorijskog podsistema u računaru sa 16-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 dve standardne memorijske komponente SRAM tipa veličine 32k x 8. U slučaju 8-bitnog pristupa za A0=0, podatak ide po nižem delu magistrale, a za A0=1 po višem delu magistrale. Memorija je dovoljno brza.

2. [10] 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.

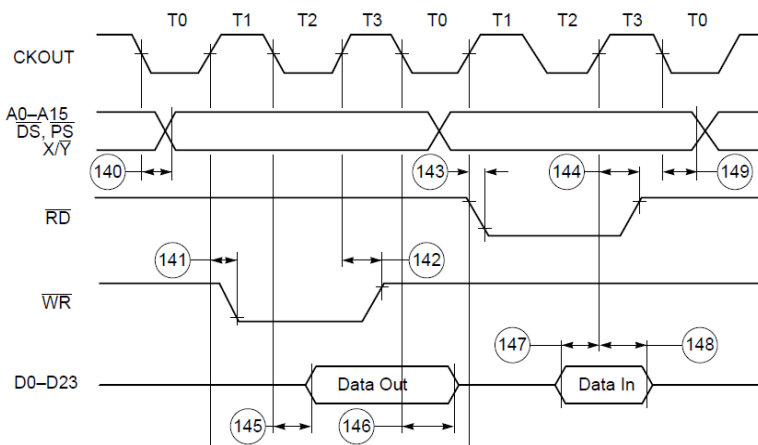
3. Dati su vremenski dijagrami rada jedne sinhronne paralelne magistrale, kao i vremenski parametri.

Izračunati:

- a) [5] maksimalno dozvoljena setup i hold vremena prihvatnog registra u koji se upisuje podatak usponskom ivicom signala upisa.
- b) [5] maksimalno dozvoljeno vreme otvaranja trostatičkih bafera u ciklusu čitanja.

Rezultate prikazati sa brojevnim oznakama sa dijagrama i za svaki broj naglasiti da li se uzima minimalno ili maksimalno vreme.

Na primer
...=143_{min}+...



4. [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 4.1. Nakon dekodovanja sadržaja dela memorije uspešno su dekodovane neke asemblerske instrukcije predstavljene u okviru *Dissassembly* 4.1.

Dissassembly 4.1

```

0x00: -----
0x04: addi x2, x0, 55
0x08: -----
0x0C: lb x4, 0(x1)
0x10: addi x5, x4, 1
0x14: -----
0x18: addi x1, x1, 1
0x1C: addi x2, x2, -1
0x20: addi x3, x3, -1
0x24: -----
0x28: jal x0, 0
    
```

Tabela 4.1

Adresa	Sadržaj(hex)				Adresa	Sadržaj(hex)			
0x00	93	00	00	03	0x24	e3	94	01	fe
0x04	13	01	70	03	0x28	6f	00	00	00
0x08	93	01	40	00	0x2C	00	00	00	00
0x0C	03	82	00	00	0x30	00	01	02	03
0x10	93	02	12	00	0x34	00	00	00	00
0x14	23	00	51	00	0x38	00	00	00	00
0x18	93	80	10	00	0x3C	...			
0x1C	13	01	f1	ff					
0x20	93	81	f1	ff					

Ako je nakon sistemskog reseta, PC registar CPUa inicijalizovan na vrednost 0, popuniti tabele 4.2, 4.3 i 4.4 za svaku od faza izvršavanja instrukcije.

Napomena: Smatrati da se program izvršava sve dok se instrukcija sa iste memorijske lokacije ne izvrši uzastopno više od 2 puta. U okviru tabele 4.4 potrebno je izvršiti dekodovanje instrukcije samo prvi put ili ukoliko već nije dekodovana u okviru *Dissassembly* 4.1.

2. KOLOKVIJUM

5. Vrednosti parametara hijerarhijski organizovanog memorijskog dela sistema, sa jednim stepenom hijerarhije, su:

- kapacitet glavne memorije (MC) = 128B;
- vreme pristupa glavnoj memoriji ($T_{Penalty}$) = $100T_{CLK}$;
- veličina bloka u kešu (BS) = 8B;
- adresibilna jedinica (AUS) = 1B;
- kapacitet keš memorije (CC) = 32B;
- vreme pristupa keš memoriji (T_{HIT}) = $3T_{CLK}$;
- keš memorija je organizovana kao *2-way set associative* keš i keš kontroler primenjuje *write back* polisu upisa;
- inicijalni sadržaj glavne memorije definisan je tabelom 4.1.

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):

$M_R[4], M_R[8], M_R[12], M_R[16], M_W[16]=54, M_R[20], M_W[22]=5, M_R[36]$.

a) [12] Kreirati okvir tabele koji ilustruje organizaciju keša u tabeli 5.2, a zatim predstaviti sadržaj kreirane tabele na osnovu tabele 5.1 i glavne memorije na lokacijama 16 i 22 za svaki od adresnih ciklusa na magistrali.

b) [3] Koliko iznosi *hit rate*? Izračunati koliko iznosi AMAT.

c) [5] 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}) = $6T_{CLK}$; *miss rate* na L2 nivou ($r_{MISS-L2}$) = 20%; vreme pristupa keš memoriji na L3 nivou (T_{HIT-L3}) = $10T_{CLK}$; *global miss rate* na L3 nivou ($r_{MISS GLOBAL-L3}$) = 0.3%.

d) [3 - Bonus] Izračunati vrednost CPU_{TIME} parametra ukoliko je poznato da: program P ima 12 instrukcija, da prosečan broj taktnih ciklusa po instrukciji iznosi 3 i da učestanost takta procesora iznosi 200MHz.

Napomena: Ukoliko je potrebno izvršiti zamenu bloka menja se onaj blok koji je prvi dovučen iz memorije.

6. [5] NOR instrukcija nije deo RISC V instrukcijskog seta. Istu funkcionalnost moguće je ostvariti sa postojećim instrukcijama. Napisati kratak asemblerski program koji ostvaruje funkcionalnost:

$$s3=s4 \text{ NOR } s5$$

7. Deo koda napisan u višem programskom jeziku napisati u assembleru RISC V. Podrazumevati da su označani brojevi g i h u registrima s0 i s1. Jasno komentarisati kod.

a) [5]	if (g > h)	b) [5]	if (g <= h)
	g = g + 1;		g = 0;
	else		else
	h = h - 1;		h = 0;

8. [15] Opisati hazarde koji mogu da nastanu prilikom izvođenja procesorskih instrukcija sa *pipeline*-om, kao i načine njihovog razrešenja.