

Ispit nosi 70 bodova. **1. zadatak:** 14 bodova. **2. zadatak:** 24 boda. **3. zadatak:** 32 boda.

1.a) Veća brzina rada je karakteristika \_\_\_\_\_ sabirnice. Spajanje većeg broja uređaja različitih brzina rada je karakteristika \_\_\_\_\_ sabirnice. Dva načina povezivanja memorijske i ulazno-izlazne sebirnice su \_\_\_\_\_ i \_\_\_\_\_.

1.b) Tri načina DMA-prijenosa su: \_\_\_\_\_, \_\_\_\_\_ i \_\_\_\_\_. Ako se u sklopu DMAC zada bit S=1 u registru CONFIG, onda će se nakon prijenosa svakog podatka automatski uvećati \_\_\_\_\_. Bez obzira na vrstu izvora i odredišta, nakon prijenosa svakog podatka uvijek će se za 1 smanjiti \_\_\_\_\_. Sklop DMAC će postati spreman (kada?) \_\_\_\_\_.

1.c) DMAC je na baznoj adresi **0x00FF0000**. Ako pomoću DMAC-a želimo kopirati blok memorije od 0x100 riječi s adrese **0x1000** na adresu **0x2000** **krađom ciklusa** uz prenošenje 32-bitnih podataka i **bez prekida**, onda moramo upisati sljedeće podatke na adrese DMAC-a:

Podatak _____	Adresa _____
Podatak _____	Adresa _____
Podatak _____	Adresa _____
Podatak _____	Adresa _____
Podatak _____	Adresa _____

U ovom slučaju, detekciju DMA-prijenosa napraviti ćemo čitanjem i ispitivanjem podatka s adrese \_\_\_\_\_.

1.d) Odredi trajanje ovog programskog odsječka i navedi vrstu hazarda kod kritičnih naredbi. Pretpostavite da je memorija brza. Cijeli odsječak traje \_\_\_\_\_ ciklusa. (Primjer pisanja: 2c ili 1c+3\*2c).

		Trajanje	Hazard
	B SKIP	_____	_____
TRI	DW 3	_____	_____
SKIP	LDR R6, TRI	_____	_____
	MOV R0, #9	_____	_____
LOOP	SUB R0, R0, R6	_____	_____
	BNE LOOP	_____	_____
	STR R0, REZ	_____	_____

2) Napiši potprogram **PRETVORI** koji pretvara broj iz 32-bitnog formata **BZP** (bit za predznak) u 32-bitni 2'k format. 32-bitni BZP se prima kao parametar preko lokacije **iza** naredbe **BL**. Rezultat se vraća registrom **R2**.

Napiši potprogram **SUM\_BLOK** koji računa sumu svih podataka u memorijskom bloku. **SUM\_BLOK** prima dva parametra preko stoga: **početnu adresu bloka** i **broj podataka u bloku**. Podatci u bloku su u 32-bitnom formatu BZP. Za svaki podatak treba pomoću potprograma **PRETVORI** odrediti prikaz u formatu 2'k i pribrojiti ga sumi koja također mora biti u 32-bitnom formatu 2'k. Rezultat potprograma **SUM\_BLOK** je suma svih podataka i vraća se registrom **R0**.

Napiši glavni program koji treba samo pozvati potprogram **SUM\_BLOK** za blok na adresi **0x45A0**. U ovom bloku se nalazi **0x3000** podataka u 32-bitnom formatu BZP. Rezultat (sumu) podataka treba spremiti na lokaciju **0x200**.

3) RTC je spojen na IRQ, a na ulazu mu je frekvencija od 150Hz. Na GPIO1 spojen je **temperaturni uređaj**, a na GPIO2 spojen je **pisač** (oba uređaja su jednaka kao s predavanja i labosa). Adrese RTC-a i GPIO-a odaberite sami. Program treba **čitati** temperaturu, pretvoriti je u **string** (niz bajtova koji sadrže ASCII-znakove) pomoću potprograma **TOSTR** i ispisivati dobiveni string na pisaču. Ovo se ponavlja **kontinuirano**.

**TOSTR** ne treba pisati. **TOSTR** prima temperaturu kao parametar preko **R0**, a rezultat (string) vraća preko niza bajtova od početne fiksne lokacije **0x1000**. Vraćeni string će biti terminiran znakom **NUL**, a znak **NUL** se ne šalje na pisač. Svake sekunde u lokaciji **SEK** treba osvježiti brojač proteklih sekundi (početno je nula). Nakon što protekne tih 100 sekundi, treba zaustaviti procesor (možete ga zaustaviti izravno u prekidnom potprogramu).