

# JESENSKI ISPIT IZ ARH-a 15/16

## 1. TEORIJA(21)

1. a)(2) Broj -5 prikažite u 6-bitnom formatu 2'k: \_\_\_\_\_. Broj +13 prikažite u 6-bitnom formatu 2'k: \_\_\_\_\_. Broj +13 prikažite u 6-bitnom formatu NBC: \_\_\_\_\_. Broj +13 prikažite u 6-bitnom formatu s bitom za predznak \_\_\_\_\_.

b)(3) 4-bitna aritmetička logička jedinica oduzima binarno 0111-1100. Nakon operacije ce biti: rezultat \_\_\_\_\_, prijenos \_\_\_\_\_, posudba \_\_\_\_\_, preljev \_\_\_\_\_, ničtica \_\_\_\_\_, predznak \_\_\_\_\_.

c)(4) FRISC-GPIO za spajanje s vanjskim procesom ima 8 priključaka za podatke, te 2 sinkronizacijska priključka: \_\_\_\_\_ čiji je smjer I \_\_\_\_\_ čiji je smjer \_\_\_\_\_. Sinkronizacijski priključci koriste se u načinima rada: \_\_\_\_\_ a ne koriste u načinima \_\_\_\_\_.

d)(3) Povratna adresa iz programa se kod FRISC-a sprema \_\_\_\_\_ (gdje), a kod ARM-a \_\_\_\_\_ (gdje). Povratak iz potprograma se kod FRISC-a ostvaruje naredbom \_\_\_\_\_, a kod ARM-a naredbom \_\_\_\_\_. Ako kod ARM-a prenosimo parametre pomoći memorijskih lokacija smještenih neposredno iza naredbe za poziv potprograma, onda se unutar potprograma ovi parametri dohvaćaju pomoću registra \_\_\_\_\_. Ako kod FRISC-a prenosimo parametre stogom, onda se unutar potprograma ovi parametri dohvaćaju pomoću registra \_\_\_\_\_.

e)(4)	vrsta prekida	adresa prekidnog potprograma	povratna adresa se sprema (gdje)	naredba za povratak iz prekida
FRISC				
ARM				

f)(3) Za arhitekturu ARM 7 navedite koliko perioda traje korak izvođenja u naredbama: ADD traje \_\_\_\_\_ perioda, CDD??? \_\_\_\_\_, STR \_\_\_\_\_, BEQ s istinitim uvjetom, BEQ lažni \_\_\_\_\_. Broj protočnih razina u arhitekturi ARM 7 je \_\_\_\_\_, ARM 9 je \_\_\_\_\_. U ARM 9 javlja se \_\_\_\_\_ hazard koji ne postoji u ARM 7. Ovaaj hazard ublažava se metodom \_\_\_\_\_. U idealnom slučaju u ARM 9 kad nema spomenutog hazarda, korak izvođenja naredbe je ADD traje \_\_\_\_\_ perioda, a korak izvođenja naredbe LDR traje \_\_\_\_\_ perioda, a korak izvođenja STR \_\_\_\_\_ perioda.

g)(2) Točno zaokruži

ARM-GPIO	- može izazvat prekid	IRQ	DA	NE
-  -	-	FIQ	-  -	
-  -	- ima bistabil stanja		-  -	
-  -	- nema sinkonizacijske priključake		-  -	

2.FRISC(16) U memoriji FRISC-a su 2 bloka podataka na adresama BAJTOVI I BE\_RIJECI. Svaki blok sadrži 100<sub>(16)</sub> podataka. U bloku BAJTOVI nalaze se podaci u 8-bitnom formatu NBC. U bloku BE\_RIJECI nalaze se podatci u 32-bitnom formatu NBC, ali u redoslijedu big-endian. Svi ostali podatci pojavljuju se u redoslijedu little-endian.

Glavni program treba množiti N-ti podatak iz bloka BAJTOVI sa N-tim podatkom iz bloka BE\_RIJECI, a njihov umnožak sprema se na N-tu poziciju u trećem bloku REZULTAT. U trećem bloku REZULTAT brojevi trebaju biti u 64-bitnom formatu NBC.

Množenje treba obavljati potprogram MNOZI, koji preko stoga primi 2 32-bitna parametra u formatu NBC. Rezultat treba biti u 64-bitnom broju u formatu NBC. Rezultat se vraća preko registra R0(nizi dio umnoška) I R1(visi dio). Množenje treba koristiti algoritam uzastopnog približavanja. Množenje s 0 treba dati ispavan rezultat.

3. FRISC(20) Na FRISC su spojeni GPIO, CT, DMA I uvjetna vanjska jedinica UVJ (adresu odrediti). U glavom programu treba primiti 32-bitne NBC podatke sa uvjetne jedinice. Primljeni podatci koji su preveliki za 8-bitne NBC se zanemaruju, dok se ostali spremaju kao 8-bitni NBC u memorijski blok na adresi 1000<sub>(16)</sub>. Ovo se ponavlja dok se blok ne napuni 100<sub>(16)</sub> podataka, a tada treba privremeno obustaviti primanje podataka.

Na ulaz CT-a je spojena frekvencija 1 kHz, a CT generira prekid NMI. Nakon 5 sekundi treba provjeriti je li blok na adresi 1000<sub>(16)</sub> napunio 100<sub>(16)</sub> podataka. Ako blok nije napunio, treba pomoću DMA prijenosa prenijeti podatke iz bloka na adresi 1000<sub>(16)</sub> na GPIO. DMA treba raditi zaustavljanjem procesora. DMA prijenos traje puno kraće od 5 sekundi kad se DMA prijenosom "ispuni blok", ponovno ga se može napuniti podatkom primjenom od UVJ I to se ponavlja beskonačno.

4. ARM(14) Za procesor ARM napišite podprogram PARITET koji preko lokacije iza naredbe BL prima 32-bitni podatak I provjera mu paritet. Ako mu je paritet neparan, treba preko R0 vratiti broj 1, a ako je paran, treba preko R0 vratiti 0. Napišite podprogram OBRADI koji obrađuje blok 32-bitne podatke tako da im se provjeri paritet, a zatim podatke s neparnim paritetom zamjenjuje podatkom F0F0F0F0<sub>(16)</sub>. Podprogram OBRADI prima početnu adresu bloka kao parametar I to preko stoga. Blok uvijek ima 100<sub>(16)</sub> podataka. Potprogram OBRADI ne vraća rezultat. Potprogram OBRADI mora provjeriti paritet pomoću potprograma PARITET. U glavnom programu treba pomoću potprograma OBRADI obraditi blok podataka na adresi 1000<sub>(16)</sub>. svi potprogrami moraju čuvati registre, a parametre sa stoga ukloniti pozivatelj.

5. ARM(14) procesor ARM sa sklopovima GPIO, RTC

Na ulaz sklopa RTC spojen signal frekvencije 10 kHz. Sklop RTC spojen je na IRQ.

Na vrata B sklopa GPIO spojen je temperaturni sklop (kao na predavanjima): podsjetnik: bitovi 0-5 su ulazni iznos temperature od 0-63<sub>(10)</sub>; bit 6 ulazni -dojava da je temperatura postavljena na bitove 0-5; bit 7 pozitivan "izlazni" da je temperatura pročitana.

Na vrata A sklopa GPIO na bitovima 5, 6, 7 spojena crvena, bijela, plava koje se uključuju pomoću jedinica, a isključuju pomoću ništica.

Napišite program koji svakih 10 sekundi obavlja provjeru temperature ako temperatura < 10 treba gorjeti plava ako je temperatura > 30 treba gorjeti crvena, inače bijela.