Dekanski rok iz arha (po sjećanju) 2017

1.Teorija (23b)

-a prikaz broja -6 u 2'k i u obliku sa predznakom

-tipovi arhitektura s obzirom na način spremanja operanda (tako nešto)

-signali, koji je ulazni/izlazni/dvosmjerni, klasika

-incijaliziranje DMA, samo popunjavanje prve 4 memorijske lokacije, ako ide iz jednog dijela memorije u drugi, radi na način krađe ciklusa

-Napisati korake pri izvršavanju naredbe JP 500

-napisati kako se puni registar određenim brojevima u ARM-u, putem naredbe MOV i MVN

-navedene 4 naredbe i opisan sadržaj memorije, treba napisati što će se nalaziti u dva registra nakon tih naredbi

2.Frisc, potprogrami

Trebalo je napisati dva potprograma, jedan za množenje dva NBC 32bitna broja uzastopnim zbrajanjem, a drugi za skalarni produkt. U zadatku je dana formula za skalarni produkt koja je u principu zbroj N umnožaka. U glavnom programu se trebalo odrediti skalarni umnožak dva vektora (imaju zadane početne lokacije i N).

Znači potprogram za skalarni produkt prima početne lokacije (u zadatku su zadane A000 i B000) i njihovu duljinu, N (u zadatku 7) preko stoga, vrti petlju N (tj 7) puta u kojoj cijelo vrijeme šalje lokacije A i B u potprogram množi, to zbraja u R0 i povećava lokacije te smanjuje brojač, koji je u ovom slučaju N.

3.Frisc, vanjske jedinice

-Zadani : Gpio (bezuvjetni), CT (sa INT prekidom), prekidna vanjska jedinica (NMI)

Glavni problem u zadatku je računanje brzine motora i ako je ona veća od 50 okretaja po sekundi, zaustaviti motor na 5 sekundi (isključiti ga) i onda ga ponovo uključiti.

Gpio je bezuvjetni i ako primi 0 motor se isključuje, a ako primi 1 motor se uključuje.

Prekidna vanjska jedinica napravi prekid svaki put kad motor napravi okretaj.

Fora je u tome što se mora računat broj okretaja u sekundi i onda reinicijalizirati CT na 5 sekundi kada treba zaustaviti motor.

4.ARM, potprogrami

Trebalo je napisati potprogram Bajt koji zrcali brojeve, tj ako imaš broj 1234, trebaš na kraju dobit 4321. To se lako izvede pomoću pomaka u lijevo zatim prenošenje tog prijenosa na najviši bit novog registra te pomicanje novog registra u desno, jer LSR ne prima C na početku (ako se ne varam, slobodno me ispravite)

Zatim je trebalo napravit novi potprogram koji mijenja određene brojeve lokacije iz poretka Big endian u little endian. Ja sam slala samo te brojeve u bajt, jer kad zrcališ to napraviš u principu i spremiš ih u staru mem lokaciju. (Radi se sa 32bitnim brojevima)

Što se tiče prenošenja u potprogram, bila je klasika preko stoga, osim primanje memorijske lokacije (u potprogramu bajt) preko memorijske lokacije koja se nalazi ispod BL, ne specificiraju koji BL, pa sam stavila u drugom potprogramu.

Nemojte zaboraviti poslati R14, jer se gnijezde potprogrami.

5.ARM, vanjske jedinice.

Dakle najlakši zadatak sa ARM vj do sad.

GPIO i RTC, zadane adrese

GPIO na vratima B ima senzore, a na vratima A alarm (znači sve se inicijalzira na jedinice jer je B ulazni, a A izlazni). Ako je bilo koji od bitova u B registru u jedinici, pošalje se 1 na registar A da se uključi alarm, ako nije onda 0 da se isključi.

Treba to raditi svakih 10 sekundi i tjt.

Ako sam što krivo zapamtila (ili krivo objasnila) slobodno me ispravite, nadam se da će nekom barem malo pomoć.

Ps ostatak teorije koji nisam navela su bila neka rendom pitanja o friscu.

Sretno!