

1. Trajanje ispita 180 minuta.
2. Ispit se radi u vežbanci.
3. Na naslovnoj strani **obavezno** zaokružiti redne brojeve zadataka koji su rađeni.

1. KOLOKVIJUM

1. [5] Objasniti ulogu i **načine** generisanja RESET signala u namenskom računarskom sistemu. Da li je RESET signal potreban i u baterijski napajanom uređaju? Objasniti. Koja je uloga PM (Power Management) integrisanog kola u savremenim računarima?

2. [4] Objasniti razloge uvođenja više izvora taktnih signala u namenskom računarskom sistemu. Navesti i koje su to najčešće realizacije izvora taktnih signala. Ako je potreban taktni signal 2GHz za ispravan rad uređaja kako bi realizovali takav taktni signal?

3. [7] Nacrtati vremenski dijagram rada uopštene 16 bitne **asinhronne** magistrale koja **ima upravljanje bajtovima u izlaznom 16 bitnom** ciklusu na **neparnu** memorijsku lokaciju. Označiti sve relevantne vremenske parametre na dijagramu (t_1, t_2, \dots sa naznakom da li je min ili max $t_1(\max)\dots$).

Naznačiti koji vremenski parametri su bitni za projektanta periferije koja prihvata te podatke u prihvatni registar realizovan ivičnim D flipflopovima. Koliko su potrebna setup i hold vremena flipflopa računato na osnovu označenih vremena? Šta ako ne mogu da budu postignuta?

4. [7] Nacrtati detaljnu logičku šemu vektorskog prekidnog kontrolera sa osam ulaza aktivnih sa logičkom jedinicom ili sa aktivnom logičkom nulom ili sa aktivnom usponskom ivicom ili sa aktivnom silaznom prekidnog zahteva sa sledećim osobinama:

- može da se programira nivo ili ivica aktivnog prekida;
- svaki prekid može nezavisno da se maskira;
- posle prihvaćenog prekida, taj prekid i svi prekidi nižeg prioriteta se maskiraju ako je prekid sa aktivnim nivoom;
- posle prihvaćenog prekida taj prekid se automatski resetuje ako je aktivnom ivicom.

5. [7] Definisati protokol rada za ulazni i izlazni ciklus, kao i upotrebu paralelnih portova nekog SoCa u cilju formiranja, emuliranja, 8-bitne paralelne sinhronne magistrale zajedničke za adrese i podatke koja ima mogućnost adresiranja 2^8 8-bitnih memorijskih lokacija. SoC ima dva 8-bitna bidirekciona porta.

2. KOLOKVIJUM

6. [8] Nacrtati projekat kontrole tastature sa 64 tastera. Opisati programske poslove SoCa ako je potrebno štedeti energiju i pouzdano identifikovati najviše dva pritisnuta tastera. SoC ima tri 8-bitna kvazibidirekciona porta. Biti jednog porta mogu da se konfigurise kao prekidni ulazi sa aktivnim nivoom.

7. [7] Definisati protokol rada i upotrebe paralelnih portova na kojima se emulira SPI sinhrona serijska magistrala preko koje se prenose 8-bitne poruke. SoC ima dva 8-bitna bidirekciona porta.

8. [15] Za osnovu sistema automatizacije procesa proizvodnje predviđen je jedan centralni industrijski računar, master, realizovan na bazi standardnog SoCa. U procesu proizvodnje su identifikovana različita radna mesta koja se mogu podeliti u više kategorija, ali je ustanovljeno da je za njih potrebno hardverski **ČETIRI** različite grupa uređaja. Shodno tome, radna mesta su podeljena u četiri grupe sa **maksimalno 10 uređaja po grupi**. Sa računarom komuniciraju preko zajedničke standardne **POLUDUPLEKS** asinhronne serijske magistrale.

SoC koji je na raspolaganju za realizaciju delova sistema je bez spoljne magistrale, sa četiri 8-bitna paralelna bidirekciona portova. Unutar SoCa se nalazi programabilni UART koji može da se programira da radi sa 7, 8 ili 9bitnim porukama, i van SoCa izlazi preko namenskih pinova Tx i Rx. Paralelni portovi mogu da se konfigurišu da rade kao ulazni prekidni zahtevi aktivni sa ivicom ili nivoom.

Svaka grupa uređaja je korišćenjem paralelnih portova ulančana. I svaka grupa uređaja pored linija za serijsku komunikaciju ima još jednu dodatnu liniju koja ide ka centralnom računaru, preko koga uređaji prekidom obaveštavaju centralni računar o incidentnim situacijama. U svakom uređaju se nalazi **64 8bitnih internih memorijskih lokacija** koje je moguće čitati i u njih upisati neke podatke.

Nacrtati dodatan hardver koji obezbeđuje ulančavanje i izlazak na zajedničku liniju za prekid. Kako se konfigurišu portovi mastera, na koje dolaze te zajedničke linije. Kako obezbediti mogućnost da programi za SoC unutar iste grupe uređaja budu identični? Očigledno je da treba iskoristiti ideju pseudovektorskih prekida, gde svaki OU (koji ima aktivan prekidni zahtev i najvećeg je prioriteta) odgovara svojim kodom. Ako se taj kod "hardkoduje" unutar programa onda svaki OU ima svoj program. Cilj je da imaju isti program.

Definisati formate i protokole rada na serijskoj magistrali, tako da se obezbedi što pouzdaniji upis i čitanje u svim uređajima. Takođe definisati protokol i formate podataka ako svaka grupa uređaja može o incidentnim situacijama da preko prekidnog zahteva informiše računar. Potrebno je pouzdano identifikovati uređaj(e) koji je poslao prekidni zahtev. Opisati programske zahteve za rad sa tim prekidima unutar uređaja, kao i unutar centralnog računara.

Primer kako treba definisati formate podataka

Adresna poruka

D7	D6	D5	D4	D3	D2	D1	D0
					A2	A1	A0

A2, A1 A0 ... adresa grupe uređaja
itd...

Poruka sa podacima

D7	D6	D5	D4	D3	D2	D1	D0

ACK Poruka

D7	D6	D5	D4	D3	D2	D1	D0

itd...

Primer kako treba definisati protokol razmene podataka

Ciklus upisa podataka

1. Master šalje adresnu poruku sa adresom odgovarajućeg OU
2. Master aktivira timeout logiku i čeka ACK poruku od adresiranog OU
3. Ako poruka stigne u odgovarajućem vremenskom intervalu
4. Ako poruka ne stigne u odgovarajućem vremenskom intervalu

itd...