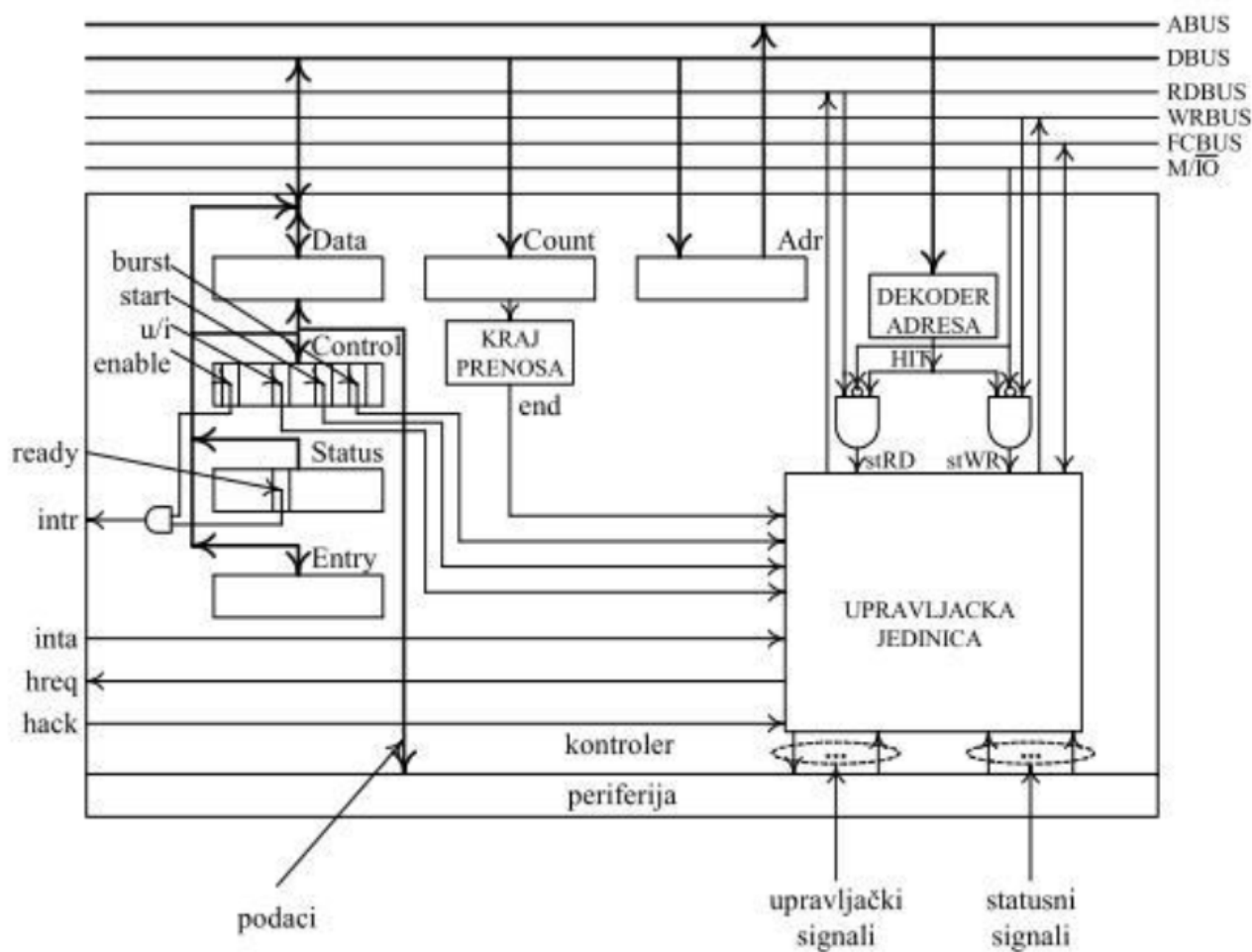


Zadaci:

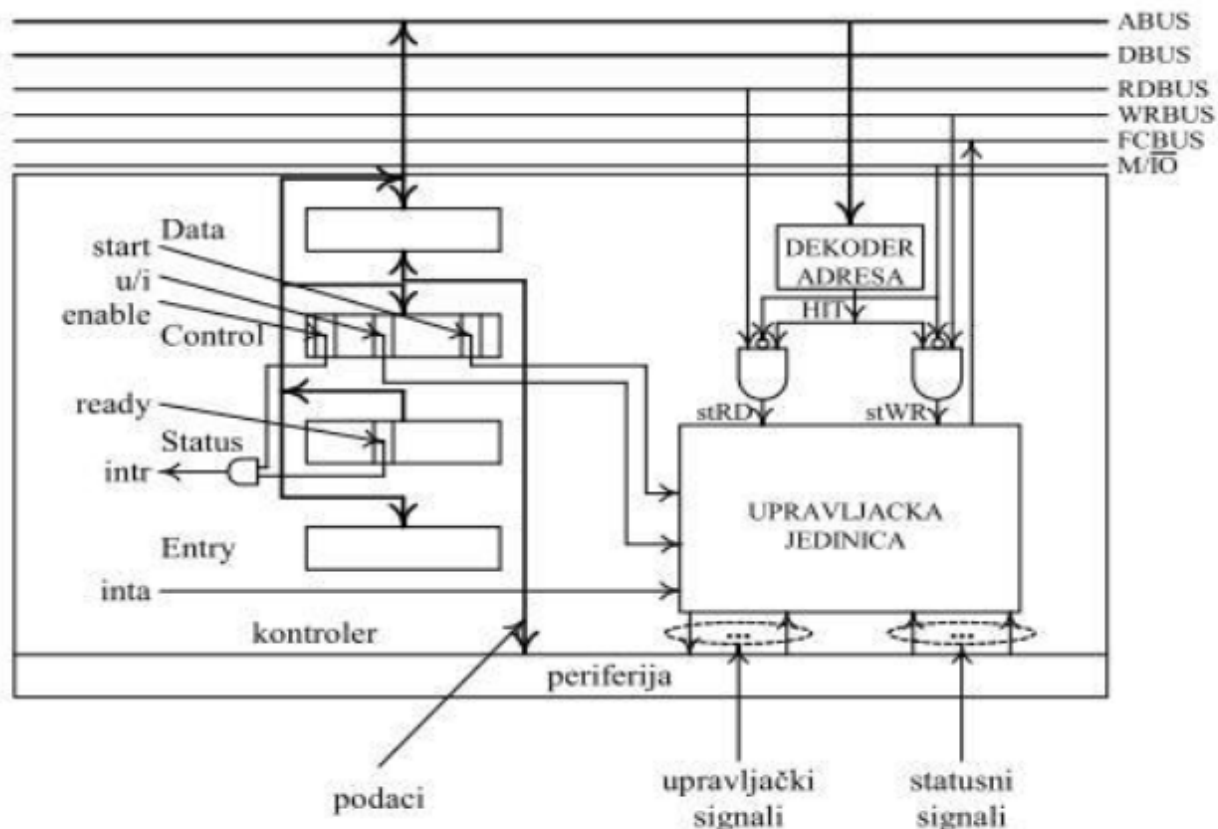
- a) Нацртати структуру контролера улазно/излазне периферије са директним приступом меморији.
- б) Објаснити функцију свих делова контролера.
- в) Објаснити којим битовима и у којим регистрима се задају могући режими рада контролера и добијају информације како се одвија пренос података.



Slika 12 Kontroler sa direktnim pristupom memoriji

jedina razlika od onog iz knjige - ima Count i Adr registar (count zbog burst rezima)?





Slika 1 Kontroler bez direktnog pristupa memoriji

?

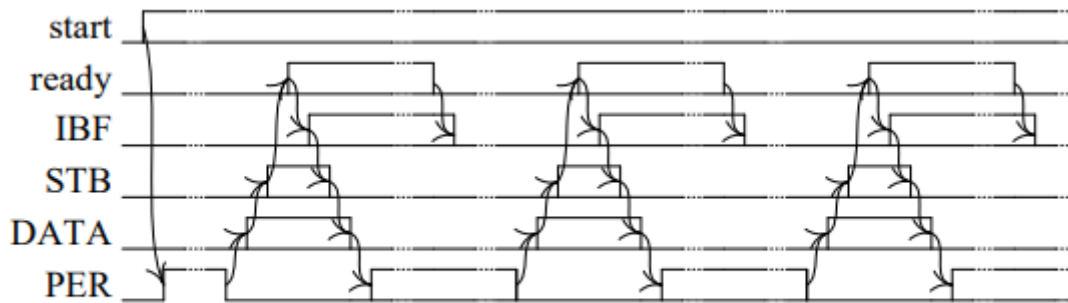
Povezivanje periferije i kontrolera periferije pomocu paralelnog interfejsa

8 linija podataka DATA, statusna linija, upravljacka linija i linija pokretanja/zaustavljanja periferije

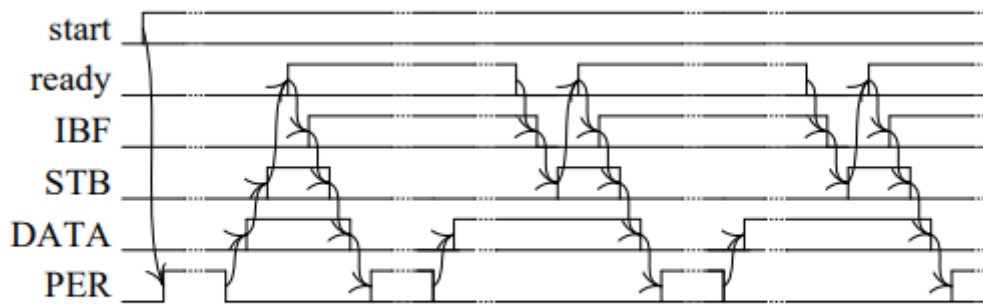
ko pokrece statusnu liniju a ko upravljacku?

## 1) Kontroler statusni signal - periferija upravljacki

- **ulazna periferija** kontroler po statusnoj liniji salje indicaciju periferiji da li je registar Data kontrolera pun ili ne, po linijama DATA periferija salje podatak kontroleru a po upravljackoj liniji periferija generise signal upisa sa DATA linije u Data registar kontrolera
  - kontroler po statusnoj liniji salje indicaciju da li je spreman za prihvatanje sled. pod. od per. (IBF tj statusna linija = 1 => registar Data kontrolera pun, IBF = 0 => kontroler moze da primi podatak)
  - periferija na DATA liniju stavlja podatak za upis u k.p. i proverava IBF
  - ako je IBF = 0, periferija generise vrednost 1 upravljackog signala STB i upisuje podatak sa DATA u Data registar k.p.
  - nakon upisa kontroler postavlja vrednost IBF na 0



Slika 4 Vremenski oblici signala za sporu ulaznu periferiju



Slika 5 Vremenski oblici signala za brzu ulaznu periferiju

- za sporu ulaznu per. vremenski trenuci:
  - kontroler postavlja start = 1
  - za vreme citanja PER = 1, a po zavrsetku PER = 0
  - procitani podatak se stavlja na linije DATA
  - utvrđuje se da IBF (statusna linija kontrolera) = 0 (spreman da primi)
  - generise se upravljacki signal STB, i zbog toga se sa DATA upisuje u Data reg. k.p.
  - ready = 1
  - IBF = 1 (k.p.)
  - STB = 0 (periferija, upravljacki signal)
  - kod spore per. (ready = 1 i IBF = 0 pre nego sto je periferija procitala sled. podatak)
  - kod brze per. (stavljanje podatka na DATA liniju se desava pre nego sto k.p. posalje podatak tj postavi IBF = 0 i ready = 1)
- **izlazna periferija** po statusnoj liniji kontroler salje znak da li je Data reg. pun ili ne, po linijama DATA kontroler salje podatak periferiji i po liniji ACK periferija salje signal da je sadrzaj sa DATA primljen

## 2) Kontroler upravljacki signal a periferija statusni

3. Јануар 2012/Јул 2011/Јануар 2011/Фебруар 2011: ЗАВРШЕНО

Улазна периферија и контролер улазне периферије (негде пише још ово: без

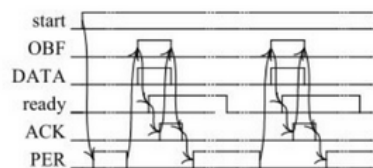
директног приступа меморији) за међусобно повезивање користе паралелни интерфејс који се састоји од линија података, једне статусне линије и једне управљачке линије. Статусном линијом улазна периферија даје индикацију да ли се у њеном излазном регистру налази садржај који се шаље по линијама података контролеру периферије, док контролер периферије управљачком линијом шаље периферији индикацију да је садржај са линија података уписан у регистар податка контролера периферије.

upis u kontroler periferije:

а) Навести како се статусна линија периферије и управљачка линија контролера периферије користе за синхронизације при слању података из периферије у контролер периферије.

- Док год нема податак, контролер чека да периферија постави активну вредност на статусну линију. Чим периферија у свој регистар података смести наредни податак за слање контролеру, она поставља активну вредност на статусну линију чиме сигнализира контролеру да може да преузме податак уколико је контролер спреман. Када контролер заврши са преносом претходног податка, уколико је статусна линија постављена на активну вредност, он почиње са преузимањем наредног податка од периферије и смешта га у свој Data регистар. Када заврши са преузимањем податка, управљачку линију (ACK) поставља на активну вредност. Периферија на активну вредност на управљачкој линији одговара постављањем неактивне вредности на статусну линију (чиме сигнализира да податак на DATA линијама није више расположив), а контролер на неактивну вредност на статусној линији одговара постављањем неактивне вредности на управљачку линију, чиме сигнализира периферији да је примио податак и да периферија може да започне припрему наредног податка.

б) За случај споре периферије нацртати временске облике сигнала и навести које сигнале и по ком редоследу генеришу периферија и контролер периферије.

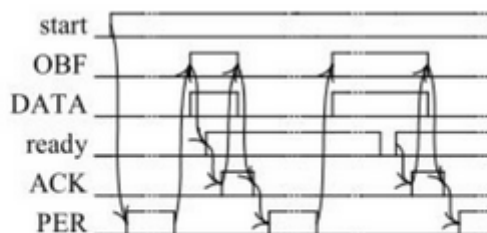


Slika 8 Vremenski oblici signala za sporu ulaznu periferiju

Редослед генерисања сигнала:

- `start = 1` (стартовање контролера периферије)
- `OBF = 1` (статусна линија)
- `ready = 1` (контролер је примио податке које је послала периферија)
- `ACK = 1` (управљачка линија)
- `OBF = 0` (статусна линија)
- `ACK = 0` (управљачка линија)
- `ready = 0` (контролер је проследио податак и спреман је за пренос наредног)
- `OBF = 1` (статусна линија) //наредни податак је спреман, а контролер је већ завршио пренос претходног
- ...

v) vremenski oblici za brzu ulaznu periferiju



Slika 9 Vremenski oblici signala za brzu ulaznu periferiju

- jedina razlika u odnosu na preth. primer je sto ce periferija statusni reg. OBF na aktivno stanje tj naredni podatak je spreman da se posalje pre nego sto je kontroler prosledio prethodno dobijen podatak, ready i dalje nije 1 u tom trenutku

#### 4. Фебруар 2012/К2 2011/Јун 2011/Септембар 2011:

Излазна периферија и контролер излазне периферије за међусобно повезивање користе паралелни интерфејс који се састоји од линија података, једне статусне линије и једне управљачке линије. Статусном линијом излазна периферија ==даје индикацију да ли у њен улазни регистар може да се уписује садржај са линија података, док ==контролер периферије управљачком линијом обезбеђује упис податка са линија података у улазни регистар периферије.

а)Навести како се статусна линија периферије и управљачка линија контролера периферије користе за синхронизације при слању података из контролера периферије у периферију.

б)За случај споре периферије нацртати временске облике сигнала и навести које сигнале и по ком редоследу генеришу периферија и контролер периферије.

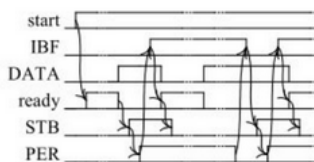
в)За случај брзе периферије нацртати временске облике сигнала и навести које сигнале и по ком редоследу генеришу периферија и контролер периферије.

ista situacija samo upis u periferiju (izlaz):

а)Контролер, када прими податак у свој регистар података, чека све док периферија на статусну линију не постави неактивну вредност, чиме сигнализира да је спремна за прихватање податка од контролера. Периферија

држи статусну линију на активној вредности све док не упише податак из свог регистра података. Када детектује неактивну вредност на статусној линији, контролер поставља управљачку линију на активну вредност, чиме започиње упис податка из свог регистра података у регистар података периферије. Периферија на активну вредност управљачке линије одговара постављањем статусне линије на активну вредност и започиње упис податка из свог регистра података. Контролер на активну вредност статусне линије одговара постављањем управљачке линије на неактивну вредност и постаје спреман да прими наредни податак који треба проследити периферији.

б) Временски облици сигнала за случај споре периферије:



Slika 10 Vremenski oblici signala za sporu izlaznu periferiju

Редослед генерисања сигнала:

- *start* = 1 (стартовање контролера периферије)
- *ready* = 1 (контролер чека да прими податак за слање периферији)
- *ready* = 0 (контролер је примио податак, спреман је да га пошаље периферији)
- *STB* = 1 (управљачка линија - почиње слање податка периферији)
- *IBF* = 1 (статусна линија - улазни бафер је пун, периферија креће са преносом)
- *ready* = 1 (контролер може да прими наредни податак)
- *STB* = 0 (управљачка линија - слање податка периферији је завршено)
- *ready* = 0 (контролер је примио наредни податак, спреман је да га пошаље периферији) //али периферија није спремна да прими податак, јер још није завршила обраду/упис претходног
- *IBF* = 0 (статусна линија - периферија је завршила са уписом претходног податка)
- *STB* = 1 (управљачка линија - почиње слање наредног податка периферији)
- ...

в) Временски облици сигнала за случај брзе периферије:



Slika 11 Vremenski oblici signala za brzu izlaznu periferiju

Редослед генерисања сигнала:

- *start* = 1 (стартовање контролера периферије)
- *ready* = 1 (контролер чека да прими податак за слање периферији)
- *ready* = 0 (контролер је примио податак, спреман је да га пошаље периферији)
- *STB* = 1 (управљачка линија - почиње слање податка периферији)
- *IBF* = 1 (статусна линија - улазни бафер је пун, периферија креће са преносом)

- *ready* = 1 (контролер може да прими наредни податак)
- *STB* = 0 (управљачка линија - слање податка периферији је завршено)
- *IBF* = 0 (статусна линија - периферија је завршила са уписом претходног податка) //периферија може да прими наредни податак, али га контролер још није добио, па мора да чека
- *ready* = 0 (контролер је примио наредни податак, спреман је да га пошаље периферији)
- *STB* = 1 (управљачка линија - почиње слање наредног податка периферији)
- ...

## 5. Јун 2011.

а) Нацртати структуру контролера улазно/излазне периферије са директним приступом меморији који може да ради у режимима периферија/меморија, меморија/периферија и меморија/меморија и објаснити функцију свих делова контролера.

б) Објаснити које све сигнале и по ком редоследу контролер размењује са арбитратором и меморијом приликом преноса блока података из једног дела меморије у други део меморије.

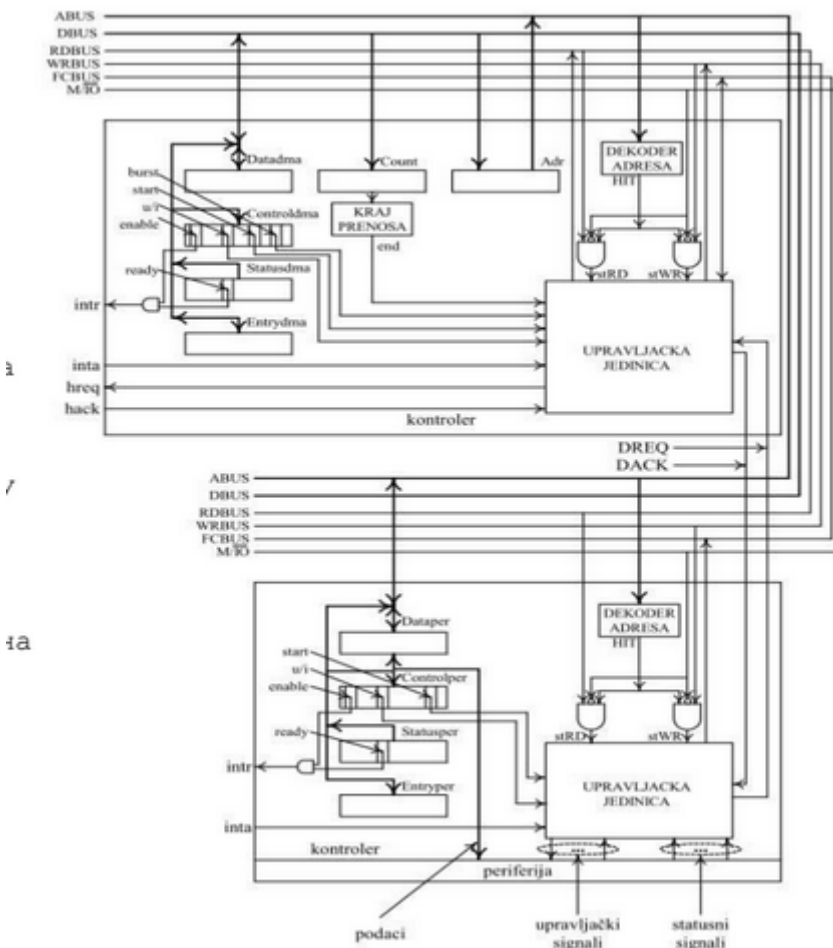
- “Режим преноса меморија меморија. Због тога постоје и додатни адресни регистар и додатни bit mem u управљачком регистру. Сада постоје изворишни и одредишни адресни регистри. Један



adresni registar se koristi za čitanje iz memorije i prebacivanje u registar podatka i drugi za upis iz registra podatka u memorijsku lokaciju. Na magistrali se realizuju dva posebna ciklusa i to ciklus čitanja i ciklus upisa.”

Остатак би требало да је исти као код DMA контролера.

7. Унос података из улазне периферије у меморију реализује се помоћу контролера без директног приступа меморији и контролера са директним приступом меморији који су повезани као на слици. Нацртати и објаснити које све сигнале и по ком редоследу размењују ови контролери између себе и са процесором и меморијом, приликом пребацивања блока података из улазне периферије у неки део меморије.



1. Контролер без директног приступа меморији (у даљем тексту nonDMA) прихвата податак добијен од периферије и смешта га у свој регистар података.
2. Након тога nonDMA контролер обавештава DMA контролер да је податак спреман постављањем сигнала DREQ на активну вредност.
3. DMA контролер затим поставља активну вредност на линију hreq чиме шаље процесору захтев за коришћење магистрале.
4. Након што добије дозволу преко активне вредности на линији hack, DMA контролер на ABUS линије поставља адресу



на коју се уписује податак, а затим поставља сигнал DACK на активну вредност, чиме сигнализира nonDMA контролеру да може да постави податак из свог регистра података на DBUS линије.

5. Затим DMA контролер стартује упис по чијем завршетку уклања адресу са ABUS линија, поставља сигнал DACK на неактивну вредност чиме сигнализира nonDMA контролеру да може да уклони податак са DBUS линија, и укида захтев за коришћење магистрале.
6. nonDMA контролер скида податак са линија података магистрале и поставља DREQ на неактивну вредност.
7. DMA контролер затим инкрементира адресни регистар и декрементира регистар броја речи преосталих за пренос, а nonDMA контролер започиње читање наредног податка из периферије.
8. претходне тачке се понављају све док се не пренесе читав блок података