



Архитектура рачунара  
- јулски испитни рок -

1(15) (K2)

1. (5) Посматра се синхрона магистрала са атомским циклусима. За циклус дохватања броја улаза потребно је описати које сигнале размењују газда и слуга. Одговор треба да садржи: који је то сигнал по редоследу, о којим линијама магистрале се ради, која се вредност налази на датим линијама, шта је значење датих линија у датом тренутку, ко је поставио дате линије, и колико износи односно чиме је одређено кашњење приликом постављања сигнала на дате линије у односу на претходно постављени сигнал. Одговор дати табеларно.

Редослед	Линије	Вредност	Значење	Ко поставља	Колико касније

1.2. (5) Контролер са директним приступом меморији се може искористити за пребацивање података са једног краја меморије на други крај меморије. Објаснити који све бити управљачког регистра контролера треба поставити на вредност 1 како би се ово попуњавање обавио, размотрити ситуације са преклапањем. Исто тако назначити у које остале регистре контролера треба уписати коју вредност.

1.3. (5) Контролер излазне периферије и излазна периферија за међусобно повезивање користе паралелни интерфејс који се састоји од линија података (*data*), једне статусне линије (*status*) и једне управљачке линије (*control*), као и линије за започињање рада (*start*). Статусном линијом периферија даје индикацију контролеру излазне периферије да ли се садржај са линија података може уписати у њен регистар, док контролер управљачком линијом шаље обавештење да је податка постављен на линије података.

За прва уписана два податка навести по ком редоследу се линије података, статусна линија и управљачка линија користе за синхронизацију и слање података из периферије у контролер периферије. Уколико је приступ споља неком регистру (или регистрима) контролера утицао на постављање датог сигнала назначити о ком се регистру и вредности у датом регистру ради. Уколико постављање датог сигнала утиче на упис у неки регистар (или регистре) контролера назначити тај регистар и вредност која се уписује. Одговор дати табеларно. Поље линија може да садржи само вредности *data*, *status*, *control* и *start*.

Редослед	Линије	Вредност	Значење	Ко поставља	Регистар и вредност

2 (10) (K2) Адресни простор неког рачунара је 4GB, и испуњен је са четири модула од којих је сваки капацитета 1GB и чија је ширина меморијске речи 8 бита. Први меморијски модул (M0) је повезан на линије DBUS<sub>7..0</sub>, други меморијски модул (M1) на линије DBUS<sub>15..8</sub>, трећи меморијски модул (M2) на линије DBUS<sub>23..16</sub>, а четврти меморијски модул (M3) на линије DBUS<sub>31..24</sub> магистрале података. Адресирање је на нивоу података дужине 8, 16 и 32 бита. Вишечестотни подаци се смештају тако да је нижа реч на вишој меморијској локацији. Меморијски систем и интерфејс процесора према магистрали омогућавају:

- приступ (упис/читање) једног бајта податка у трајању једног циклуса на магистрали
- приступ (упис/читање) два бајта податка у трајању до два циклуса на магистрали (у зависности од адресе, ако је могуће извршити приступ у једном циклусу)
- приступ (упис/читање) четири бајта податка у трајању до два циклуса на магистрали (у зависности од адресе, ако је могуће извршити приступ у једном циклусу)
- постављање идентификације, користећи посебне линије магистрале ( $W_{1..0}$ ), ако је  $W=00$  приступа се једној меморијској локацији, ако је  $W=01$  приступа се две суседне меморијске локације, ако је  $W=10$  приступа се три суседне локације и ако је  $W=11$  приступа се четири суседне локације.

а) (4) Потребно је пројектовати описани меморијски систем и интерфејс процесора према магистрали. На располагању су меморијски чипови са контролним улазима RD, WR и CS, и потребна логичка кола.

б) (2) Који опсег адреса обухвата сваки од коришћених меморијских модула?

в) (4) Нацртати дијаграм тока управљачких сигнала фазе извршења инструкција INC *mem* (инкрементира особитни податак са адресе *mem* и враћа га на адресу *mem*), LOADW *mem* (чита шеснаестобитни податак из меморије са адресе *mem*) и STOREDW *mem* (смешта тридесетдвобитни податак у меморију на адресу *mem*). Претпоставити да циклус уписа/читања траје једну периоду сигнала такта. Сматрати да регистар MDR има могућност инкрементирања.

### 3 (15)

**3.1. (5)** Дат је програм PRIMER током чијег извршавања се скаче на процедуру PRIM. У процедури се врши упис у регистре опште намене R0 и R1. По повратку из процедуре вредности ових регистара треба да буду исте као и пре скока на процедуру. Посматра се двоадресни процесор.

**а)** Навести које се инструкције генеришу приликом позива процедуре, при уласку у процедуру, при израчунавању датог дела израза и при изласку из процедуре. Сваки део посебно уоквирити и објаснити функцију сваке инструкције у датом блоку појединачно.

**б)** Нацртати ситуације на стеку по извршавању генерисаних инструкција приликом позива процедуре, при уласку у процедуру и при изласку из процедуре. Стек расте према нижим локацијама и указује на задњу заузету локацију. Адресе и целобројни подаци су дужине 32 бита, а адресибилна јединица је 8 битна реч.

**в)** Навести како се генеришу адресе за приступ a, b, x, y, z, c и d.

```
int a, b;
void PRIM (int x, int y, int z) {
    int c,d;
    ...
    c = x + d + b;
    ...
}
...
void main() {
    ...
    PRIM (a,b,b);
    ...
}
```

**3.2. (5)** Написати оптималну секвенцу инструкција која одговара следећој стандардној библиотечкој C функцији која копира првих num карактера низа source у низ destination:

```
char * strncpy ( char * destination, const char * source, int num );
```

На располагању је процесор код кога аритметичке инструкције имају формат: OC reg, reg, reg/imm где је OC код операције, одредишни операнд и први операнд морају бити у регистру (reg), док други може бити или у регистру или дат непосредно (reg/imm). Инструкција LOAD има формат: LOAD reg, mem где је првим операндом дат одредишни регистар (reg), а другим извориште. Инструкција STORE има формат: STORE reg, mem где је првим операндом дат изворишни регистар (reg), а другим одредиште. На располагању стоји 8 регистара опште намене, као и сложене инструкције. Претпоставити да су сви подаци и адресе исте дужине која је једнака адресибилној јединици.

**3.3. (5)** У посматраном рачунарском систему постоји само једна линија по којој улазно/излазни уређаји могу процесору да шаљу захтеве за прекид, док линија за слање сигнала потврде не постоји. У датом систему постоји 4 улазно/излазна уређаја UI3, UI2, UI1 и UI0, при чему приоритет ових уређаја опада од UI3, који има највиши приоритет, преко UI2, UI1 до UI0, који има најнижи приоритет. Периферија захтев за прекидом генерише као импулс. Нацртати како та 4 улазно/излазних уређаја треба повезати помоћу те линије на процесор ради слања захтева за прекид и објаснити како се реализује скок на одговарајућу прекидну рутину сваког од 4 улазно/излазних уређаја. Скок на одговарајућу прекидну рутину се реализује делом хардверски, а делом софтверски. Објаснити шта се и како ради хардверски, а шта и како софтверски. За онај део који се ради софтверски написати одговарајући програм. Узети да се адресе прекидних рутина уређаја (завршавају се RTI инструкцијом) UI3, UI2, UI1 и UI0 налазе у улазима 3, 2, 1 и 0, респективно, посебно формиране табеле, чија се почетна адреса налази у меморијској локацији означеној са *tabadresa*.

**4. (15)** У рачунарском систему се налази једноадресни процесор, меморија и две периферије PER0 и PER1 којој је придружен DMA контролер. Све компоненте рачунара су повезане системском магистралом са 32 битном адресном и 32 битном магистралом података. Адресирање је на нивоу 32 битних речи. Адресни простор овог рачунара и улазно/излазни адресни простор су раздвојени. Адресе релевантних регистара су:

PER0_CONTROL	FF00h	PER1_CONTROL	FF10h	DMA_CONTROL	FF30h
PER0_STATUS	FF01h	PER1_STATUS	FF11h	DMA_STATUS	FF31h
PER0_DATA1	FF02h	PER1_DATA	FF12h	DMA_DATA	FF32h
PER0_DATA2	FF03h			DMA_ADDRESS	FF33h
PER0_DATA3	FF04h			DMA_COUNT	FF34h

У управљачким регистрима бит 0 је *Start* којим се дозвољава почетак операције, бит 1 одређују тип преноса података (1 - улаз (*input*), 0 – излаз (*output*)), бит 7 је *Enable* којим се дозвољава прекид, а у статусним регистрима бит 1 је *Ready* који сигнализира спремност контролера периферије. Бит 3 управљачког регистра DMA контролера задаје режим рада (0-блоковски (*burst*), 1-циклус по циклус (*cycle stealing*)).

Написати главни програм и одговарајуће прекидне рутине којима се обавља следећи пренос. Са периферије PER0 се прихвата матрица елемената, али периферија PER0 шаље елементе у произвољном редоследу. Потребно је у меморију почев од адресе 1000h сместити матрицу по редовима. Када се PER0 укључи, она прво шаље број редова и број колона матрице преко регистара података PER0\_DATA1 и PER0\_DATA2, респективно, док садржај регистра PER0\_DATA3 није валидан. Након тога PER0 шаље елемент по елемент матрице тако што шаље редни број реда, редни број колоне и сам податак преко регистара података PER0\_DATA1, PER0\_DATA2 и PER0\_DATA3, респективно. Редни бројеви реда и колоне представљају позицију елемента у матрици и њихова нумерација креће од нуле. Када се прихвати елемент треба проверити да ли је већ примљен елемент са том позицијом. Уколико је елемент већ примљен треба одмах угасити периферију PER0 и завршити главни програм.

На располагању су две функције које су имплементиране и могу се позивати из главног програма.

Функција **int** getFlag(**int** row, **int** col) враћа информацију о томе да ли је елемент матрице већ примљен. Ако је повратна вредност нула, то значи да елемент није до сада примљен, док повратна вредност један значи да је елемент већ примљен. Функција повратну вредност враћа преко акумулатора.

Функција **void** setFlag(**int** row, **int** col) поставља информацију о томе да је елемент матрице примљен. Параметри ових функција представљају позицију елемента у матрици.

Улаз са PER0 реализовати испитивањем бита спремности. Када је овај бит постављен на јединицу тада је могуће прочитати сва три регистра података.

Када се прими цела матрица потребно је матрицу послати периферији PER1. Излаз на периферију PER1 реализовати коришћењем DMA контролера који ради у блоковском режиму рада.

**Напомене:** На испиту нису дозвољена никаква помоћна средства, ни калкулатори, ни литература. Испит траје 3 сата. Потребно је на свесци назначити да ли се задаци **1.** и **2.** мења са колоквијумом или не. Студент је дужан да пише читко и уредно.