



Архитектура рачунара
- јулски испитни рок -

1. (5) У посматраном рачунарском систему процесор има 4 пара линија по којима улазно/излазни уређаји могу процесору да шаљу захтеве за прекид и од процесора добијају сигнале потврда. У датом систему има 9 улазно/излазних уређаја које треба некако повезати помоћу та 4 пара линија на процесор и омогућити за сваки улазно/излазни уређај скок на одговарајућу прекидну рутину векторисаним механизмом прекида.

а) Нацртати како тих 9 улазно/излазних уређаја треба повезати помоћу та 4 пара линија на процесор.

б) Нацртати интерну структуру контролера периферије који дозвољава серијско слање прекид наредној периферији у ланцу.

2. (5) Написати оптималну секвенцу инструкција неопходних за срачунавање израза:

```
int *a, *b, c, i;
...
for (i = c-1; i >= 0; i -= 1)
    if (b[i] > 0) a[i] = 1;
    else a[i] = 0;
...
```

На располагању је процесор код кога аритметичке, логичке и померачке 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. `A`, `B`, `C` и `I` су глобалне променљиве које одговарају симболичким ознакама адреса меморијских локација у којима се налазе операнди. Садржај меморијских локација означених адресама `A`, `B` и `C` треба да остане непромењен, садржај одговарајућих регистара је дозвољено мењати. На располагању стоји 8 регистара опште намене. Претпоставити да су сви подаци и адресе исте дужине која је једнака адресибилној јединици.

3. (5) Написати оптималну секвенцу инструкција која одговара следећој стандардној библиотечкој `C` функцији која проналази прво појављивање карактера `c` у низу `str`:

```
char *strchr(const char *str, char c);
```

Функција враћа адресу на којој се налази пронађени карактер `c`, а ако не пронађе карактер `c` функција враћа нулу. Формати инструкција и података су као у задатку 2. На располагању стоје и сложене инструкције.

4. (25) У рачунарском систему се налази једноадресни процесор, меморија и периферије PER0 и PER1 са придруженим контролером периферије DMA1. Све компоненте рачунара су повезане системском магистралом са 16 битном адресном и 16 битном магистралом података. Адресирање је на нивоу 16 битних речи. Сви подаци и адресе су ширине 16 бита. Улазно-излазни адресни простор и меморијски адресни простор су раздвојени. Адресе релевантних регистара су:

PER0_CONTROL	FF00h	DMA1_CONTROL	FF20h
PER0_STATUS	FF01h	DMA1_STATUS	FF21h
PER0_DATA	FF02h	DMA1_DATA	FF22h
PER1_CONTROL	FF10h	DMA1_ADDR	FF23h
PER1_STATUS	FF11h	DMA1_CNT	FF24h
PER1_DATA	FF12h		

У управљачким регистрима бит 15 је *Start* којим се дозвољава почетак операције, бит 0 одређују тип преноса података (1 - улаз (*input*), 0 – излаз (*output*)), бит 1 је *Enable* којим се дозвољава прекид, а у статусним регистрима бит 0 је *Ready* који сигнализира спремност контролера периферије. Бит 4 управљачких регистра контролера DMA1 задаје режим рада (0-блоковски (*burst*), 1-циклус по циклус (*cycle stealing*)).

Написати главни програм, одговарајуће потпрограме и одговарајуће прекидне рутине којима се обавља следећи пренос. Периферија PER0 шаље низ података који се смешта у меморију почев од адресе 1000h. Пријем података од периферије PER0 престаје када периферија пошаље *MAX_NUM* података или три узастопна податка који имају јединице на месту највишег и најнижег бита. Проверу да ли податак садржи јединице на месту највишег и најнижег бита реализовати потпрограмом **int checkValue(int value)** који кроз акумулатор враћа јединицу ако податак *value* садржи јединице, а у супротном враћа нулу. Након пријема потребно је низ послати периферији PER1. Слање реализовати синхроним потпрограмом **void sendSyncData(int* data, int size)** који треба да пошаље низ *data* величине *size* периферији PER1.

Примање података са периферије PER0 реализовати коришћењем механизма прекида. Слање података на периферију PER1 реализовати користећи DMA1 контролер у режиму циклус по циклус. Главни програм треба да се извршава циклично.

Процесор не поседује регистре опште намене, али поседује регистре SP и BP, као и инструкције за њихову манипулацију. Стек расте од виших ка нижим локацијама, а SP указује на последњу заузету локацију. Дозвољено је користити додатне променљиве, али њихове називе треба писати описно и семантички исправно. **Обавезно је писање концизних коментара над семантичким целинама.**

Напомене: На испиту нису дозвољена никаква помоћна средства, ни калкулатори, ни литература. Испит траје 90 минута. Студент је дужан да пише читко и уредно.