



Архитектура рачунара  
- јулски испитни рок -

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

```
for(i = 0; i < 1000; i = i + 2)
    sum = sum + niz[i + 1];
```

за процесор код кога аритметичке, логичке и померачке инструкције имају формат: *OC reg, reg/imm* где је *OC* код операције, први операнд који је уједно и одредишни мора 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. *i*, *sum*, *niz* су глобалне променљиве које одговарају симболичким ознакама адреса меморијских локација у којима се налазе операнди. На располагању стоји 16 регистара опште намене (*R0-R15*). Сви подаци и адресе су исте дужине која је једнака адресибилној јединици. На располагању стоје и сложене инструкције.

2. (5) Написати оптималну секвенцу инструкција која одговара следећој стандардној библиотечној *C* функцији која пореди првих *n* карактера низа *str1* и низа *str2*:

```
int strcmp(const char *str1, const char *str2, int n);
```

Функција враћа -1, 1, или 0, уколико први карактер који се на поклапа има мању вредност у низу *str1*, мању вредност у низу *str2*, уколико су низови једнаки, респективно. Формати инструкција и података су као у задатку 1. На располагању стоје и сложене инструкције.

3. (5) У посматраном рачунарском систему процесор има један пар линија по којима улазно/излазни уређаји могу процесору да шаљу захтеве за прекид и од процесора добијају сигнале потврда. У датом систему има 4 улазно/излазних уређаја које треба некако повезати помоћу тог пара линија на процесор користећи посебан уређај који се назива контролер прекида тако да се омогући за сваки улазно/излазни уређај скок на одговарајућу прекидну рутину векторисаним механизмом прекида.

а) Нацртати како та 4 улазно/излазна уређаја треба повезати на процесор користећи контролер прекида и пар линија.

б) Објаснити у којим се ситуацијама понашање овог система разликује од понашања система са 4 пара линија за слање захтева за прекид и добијање сигнале потврда.

4. (15) У рачунарском систему се налази једноадресни процесор, меморија и периферије *PER0*, *PER1* и *PER2*. Све компоненте рачунара су повезане системском магистралом са 16 битном адресном и 16 битном магистралом података. Адресирање је на нивоу 16 битних речи. Адресни простор овог рачунара и улазно/излазни адресни простор су раздвојени. Адресе релевантних регистара су:

<i>PER0_CONTROL</i>	<i>FF00h</i>	<i>PER1_CONTROL</i>	<i>FF10h</i>	<i>PER2_CONTROL</i>	<i>FF20h</i>
<i>PER0_STATUS</i>	<i>FF01h</i>	<i>PER1_STATUS</i>	<i>FF11h</i>	<i>PER2_STATUS</i>	<i>FF21h</i>
<i>PER0_DATA</i>	<i>FF02h</i>	<i>PER1_DATA</i>	<i>FF12h</i>	<i>PER2_DATA</i>	<i>FF22h</i>

У управљачким регистрима бит 0 је *Start* којим се дозвољава почетак операције, бит 1 одређују тип преноса података (0 - улаз (*input*), 1 – излаз (*output*)), бит 7 је *Enable* којим се дозвољава прекид, а у статусним регистрима бит 0 је *Ready* који сигнализира спремност контролера периферије. Могуће је уписивати и читати из свих регистара.

Написати главни програм и одговарајућу прекидну рутину којима се обавља следећи пренос. Са периферија *PER0* и *PER1* упоредо се прихватају два низа, *A(i)* и *B(i)* (*i=0..99h*), респективно. Низови *A(i)* и *B(i)* се смештају у меморију почев од локација *1000h* и *2000h*, респективно. Такође, упоредо са пријемом ових низова врши се и упоредо њихово слање периферији *PER2*. Периферији *PER2* уколико су одговарајући елементи примљени треба слати један по један елемент и то тако да јој се прво шаље елемент низа *A(i)*, па елемент низа *B(i)*, а након тога јој се шаље наредни елемент низа *A(i)* итд. Постоје две процедуре које служе за дозволу и забрану прекида са периферије *PER2*, **void** *enableIntr2()* и **void** *disableIntr2()* и није могуће на неки други начин забранити прекид осим коришћења ове две процедуре. Програм треба да ради циклично (да се никада не заврши).

Прихватање низова са периферија *PER0* и *PER1* реализовати испитивањем бита спремности, а слање низа на периферију *PER2* реализовати коришћењем механизма прекида. Једном покренуте периферије не смеју се накнадно гасити. Водити рачуна да сви трансфери теку упоредо.

Процесор не поседује регистре опште намене. Називе променљивих које се користе треба писати описно и семантички исправно.

**Напомене:** На испиту нису дозвољена никаква помоћна средства, ни калкулатори, ни литература. Испит траје 150 минута. Студент је дужан да пише читко и уредно.