



Архитектура рачунара

I(15) (K2)

1. (5) Модули mod3, mod2, mod1 и mod0, од којих је mod3 највишег а mod0 најнижег приоритета, учествују у арбитражи, коришћењем серијског арбитра, да би могли да реализују циклус на магистрали.

а) Нацртати структуру арбитра и повезати ове модуле на арбитра водећи рачуна о њиховим приоритетима. Објаснити шта је додатно потребно како би се реализовала арбитра.

б) Дати систем за арбитражу не гарантује да модул који је добио право да реализује циклус на магистрали стварно и буде први следећи модул који користи магистру. Дати пример који поткрепљује ову тврдњу.

2. (5) Посматра се контролер са директним приступом меморији (DMA).

а) Навести који се регистри DMA контролера све користе да би се обавио трансфер блока са једног краја меморије на други крај меморије. За сваки од датих регистара навести улогу у датом трансферу.

Регистар	Улога у трансферу

б) Који све бити треба да постоје у управљачком регистру DMA контролера како би се омогућио овај трансфер у различитим ситуацијама.

Бит	Значење

3. (5) Контролер улазне периферије и улазна периферија за међусобно повезивање користе паралелни интерфејс који се састоји од линија података, једне статусне линије и једне управљачке линије. Статусном линијом контролер улазне периферије даје индикацију периферији да ли у његов регистар податка може да се уписује садржај са линија података, док периферија управљачком линијом обезбеђује упис податка са линија података у регистар податка контролера улазне периферије.

а) Посматра се синхронизација између процесора и контролера периферије. Навести по ком редоследу и како се размењују информације између контролера и процесора како би се обавио трансфер.

Редослед	Информација	Значење	Ко поставља

б) Након завршеног трансфера податка од контролера до процесора која линија између контролера и периферије мења вредност, која је била стара вредност и које је значење нове вредности.

Линија	Стара вредност	Нова вредност	Значење

II (10) (K2) Адресна магистрала процесора је 16 битна, магистрала података је 16 битна. Садржај дела оперативне меморије је приказан на слици:

Адреса	1000h	1001h	1002h	1003h	1004h	1005h	1006h	1007h	1008h	1009h	100Ah
Садржај	1002h	0000h	0002h	1004h	4000h	FF12h	0007h	FFFFh	4002h	FF16h	0000h
Адреса	3000h	3001h	3002h	3003h	3004h	3005h	3006h	3007h	3008h	3009h	300Ah
Садржај	0E0Ch	1000h	0804h	F100h	4080h	3002h	0000h	0104h	020Dh	FF30h	F100h
Адреса	4000h	4001h	4002h	4003h	4004h	4005h	4006h	4007h	4008h	4009h	400Ah
Садржај	0408h	FF12h	F900h	0400h	0391h	0400h	FF03h	E112h	0F0Fh	FF09h	FF04h

Процесор поседује 16 битне регистре PC (програмски бројач), SP (показивач на прву слободну локацију стека који расте према нижим адресама), и 16 битне регистре R0, R1, R2, R3. Меморијски и улазно/излазни адресни простори су раздвојени. Претпоставити да се током извршавања овог програма генерише само прекид изазван инструкцијом софтверског прекида INS. Инструкција INS је тако реализована да се у фази извршавања операције (IE) најпре на стек ставља садржај програмског бројача PC, а затим на основу броја улаза у IV табелу, који је део инструкције INS, и почетне адресе IV табеле, дате садржајем регистра IVTP, добија адреса улаза у IV табели из кога се чита адреса прекидне рутине и уписује у PC. Не пролази се кроз фазу опслуживања прекида (IS). Претпоставити да је пре почетка извршавања главног програма SP = F000h, IVTP = 0h и PC = 3000h. Навести секвенцу садржаја на адресној магистрали, магистрали података и контролној магистрали за сваки циклус на магистрали при извршавању програма приказаног на слици:

адреса	инструкција	коментар
3000h	LOAD R0, (1000h)	; меморијско индиректно адресирање
3002h	INS #4	; софтверски прекид 4
3003h	OR R0, R0	; битско или регистра R0 samog sa sobom
3004h	JNZ 3002h	; скок ако је Z!=1, апсолутан скок
3006h	HALT	; заустављање процесора
;прекидна rutina prekida 4		
4000h	OUT R0, FF12h	; IO[FF12h] = R0, директно адресирање
4002h	DEC R0	; декрементирање R0
4003h	RTI	; повратак из прекидне rutine

Решење представити у табели чије заглавље изгледа као на слици:

P6	ABUS	DBUS	RD	WR	M/ $\overline{\text{IO}}$	PC	IR _{31:16}	IR _{15:0}	TEMP	R0	SP	Коментар
----	------	------	----	----	---------------------------	----	---------------------	--------------------	------	----	----	----------

P6 представља редни број циклуса на магистрали, ABUS и DBUS садржаје адресних линија и линија података у датом циклусу. RD, WR и M/ $\overline{\text{IO}}$ вредности управљачки сигнала на магистрали. PC садржај програмског бројача, IR садржај инструкцијског регистра, TEMP садржај прихватног регистра операнда или адресе операнда након завршетка циклуса, R0 садржај регистра опште намене, и SP садржај показивача на врх стека. У коментару навести у којој од фаза дохватања инструкције (IF), декодовања инструкције (ID), извршавања (EX) или обрада прекида (INT) се реализује дати циклус.

III(15) 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 x, y, z, c, d и e.

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

2. (5) У процесору постоји инструкција за контролу петљи SOB (*Subtract One and Branch*).

а) Дати формат инструкције и навести шта се сваким од операнда у формату инструкције специфицира.

б) Дати пример програмског сегмента на неком вишем програмском језику у коме би дата инструкција могла да се искористи. Написати скуп инструкција (укључујући дату) који одговара датом сегменту.

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 2 улазно/излазна уређаја UI1 и UI0, при чему је приоритет уређаја UI1 већи од приоритета уређаја UI0. Нацртати како та 2 улазно/излазна уређаја треба повезати помоћу те линије на процесор ради слања захтева за прекид. Написати заједничку прекидну рутину у којој се софтверски обавља скок на прекидну рутину периферије која је затражила прекид, приказати изглед потребних структура података.

IV (15) Једноадресни процесор са раздвојеним улазно/излазним и меморијским адресним простором, меморија, периферије PER0, PER1 и PER2, повезани су системском магистралом са 16 битном адресном и 16 битном магистралом података. Адресирање је на нивоу 16 битних речи. Адресе релевантних регистара:

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

У управљачким регистрима бит 2 је Start којим се дозвољава почетак операције, бит 6 одређује смер операције (0-улаз, 1-излаз), бит 15 је Enable којим се дозвољава прекид, а у статусним регистрима бит 8 је Ready који сигнализира спремност контролера. Написати главни програм и одговарајуће прекидне рутине којима се обавља следећи пренос. Периферија PER0 шаље низ од 1000h података преко бафера величине 150h на адреси 2000h, из којег периферије PER1 и PER2 читају у паралели. Чим периферија PER0 упише неки елемент у бафер, периферије PER1 и PER2 могу да га прочитају. И PER1, и PER2 читају исте елементе бафера, тј. након што i -ти елемент низа буде уписан, треба да га прочитају обе периферије. Ниједна од њих не сме да настави са читањем $(i+1)$ -ог елемента, док она друга периферија није прочитала i -ти елемент. Периферија PER0 може да уписује податке неvezано за то који елемент низа читају периферије PER1 и PER2. Сви подаци пристигли са периферије PER0 треба да буду прочитани од стране периферија PER1 и PER2. Улаз са PER0 реализовати реализовати испитивањем бита спремности, излаз на PER1 механизмом прекида, излаз на PER2 испитивањем бита спремности. Писати кратке коментаре решења и претпоставити да не постоји меморијско индиректно адресирање са померајем.

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 **I** и **II** мења са колоквијумом или не.