



Архитектура рачунара  
Колоквијум

**1. (7)** У процесору са векторисаним механизмом прекида се у оквиру хардверског дела опслуживања захтева за прекид поред осталог генеришу и бројеви улаза у табелу са адресама прекидних рутина за спољашње маскирајуће прекиде који долазе од улазно/излазних уређаја по линијама (1) INTR3, (2) INTR2, (3) INTR1 и (4) INTR0. За селективно маскирање спољашњих маскирајућих прекида који долазе од улазно/излазних уређаја по линијама (1) INTR3, (2) INTR2, (3) INTR1 и (4) INTR0, користе се разреди (1) IMR3, (2) IMR2, (3) IMR1 и (4) IMR0 регистра маске IMR, респективно. Генерисани бројеви улаза су осомбитне целобројне величине без знака. Нацртати и објаснити шему за генерисање бројева улаза за ова четири прекида уколико се адресе прекидних рутина за прекиде (1), (2), (3) и (4) налазе у улазима 11, 10, 9 и 8 табеле са адресама прекидних рутина и уколико прекид (1) има највиши, прекид (2) нижи, прекид (3) још нижи и прекид (4) најнижи приоритет.

**2. (8)** Посматра се магистрала са подељеним циклусима која се састоји од адресних линија, линија података и управљачких линија и на којој се реализују циклус слање захтева за читање, циклус слање захтева за упис и циклус враћање податка.

**а)** Навести које информације по адресним линијама и линијама података и које сигнале по управљачким линијама размењују газда и слуга приликом реализације сваког од наведена три циклуса на магистрали. Одговор дати табеларно.

**б)** Навести шта све треба да постоји од хардвера на страни меморијског модула да би три наведена циклуса могла да се реализују.

**3. (25)** Адресни простор процесора је величине 128KB, адресибилна јединица је 16 битна реч. Процесор је једноадресни са раздвојеним меморијским и У/И адресним простором, механизам прекида је векторисан, табела прекидних рутина почиње од адресе на коју указује регистар IVTP, а регистар IVTP има вредност 0000h.

Процесор има две улазне линије  $IRQM_0$  и  $IRQM_1$  за спољашње маскирајуће прекиде и једну улазну линију  $IRQN$  за спољашње немаскирајуће прекиде на које су везане периферије PER0, PER1 и PER2, респективно. Адресе 16 битних регистара контролера периферија PER0, PER1 и PER2 у којима се чувају бројеви улаза у IV табелу су 5h, 10h и 15h, респективно. Придружени бројеви улаза у табели прекидних рутина за периферије PER0, PER1 и PER2 су 0, 1 и 2, док се захтеви памће у флип-флопвима  $PRM_0$ ,  $PRM_1$  и PRN. Од спољашњих маскирајућих виши приоритет има захтев који долази по линији  $IRQM_1$ . Не прихватају се прекиди истог нивоа приоритета. Процесор реагује и на прекиде због задатог режима рада прекид после сваке инструкције – ако је бит Т регистра PSW постављен на вредност 1 за који је фиксно одређен улаз 3 IV табеле, као и услед извршавања инструкције INS - захтев се чува у флип-флопу PRINS. У регистру PSW редом од нижих ка вишим битовима се налази одговарајући број L бита, па затим битови I и T. Регистри PSW и ACC су ширине 16 бита. Инструкције INTE, INTD, TRPE, TRPD и RTI не реагују на прекиде. У кораку за обраду прекида на стек се хардверски стављају PC и PSW, тим редом. Стек расте према нижим локацијама, а SP указује на последњу заузету локацију и има вредност F000h. У процесору не постоји регистар маске IMR, као ни регистри опште намене.

На слици 1, дат је део кода који је уčitан у оперативну меморију рачунара. Инструкција на адреси 1000h означена је као 1. (прва) по редоследу извршавања, а свака следећа инструкција која се извршава означена је следећим редним бројем. На слици 2, дати су тренуци пристизања спољашњих захтева за прекид.

**а) (17)** Написати секвенцу адреса наредби које се редом извршавају почев од адресе 1000h. На почетку бит I регистра PSW је постављен на 1, бит Т регистра PSW је постављен на 0, док су сви остали битови регистра PSW постављени на 0. Резултат дати након фазе извршења инструкције и уколико је у фази опслуживања прекида прихваћен прекид и након фазе опслуживања прекида. Табелу приказати до завршене инструкције која се налази на адреси 1008h. При цртању стања стека, потребно је назначити адресу сваке меморијске локације, као и на коју адресу указује регистар SP.

**б) (1)** Која вредност се налази на локацији 0100h након извршеног програма под а).

**в) (2)** Написати део програма којим се иницијализују регистри периферија PER0, PER1 и PER2 у којима се чувају бројеви улаза у IV табелу.

**г) (5)** Проширити одговарајућу прекидну рутину за маскирајући захтев за прекид који долази по линији  $IRQM_1$  тако да након повратка из прекидне рутине главни програм више не реагује ни на један маскирајући захтев за прекид до краја свог извршавања. На почетку бит I регистра PSW је постављен на 1, док су сви остали битови регистра PSW постављени на 0. Сматрати да се у главном програму никад не позивају инструкције INTE, INTD и INS. Сматрати да ће се барем једном прихвата захтев за прекид који долази по линији  $IRQM_1$ . Све адресне инструкције сматрати да су дужине 3 речи, а све безадресне 1 реч. На располагању је локација 0600h. Решење треба приказати у датој табели навођењем адресе и инструкције која ће се налазити на тој адреси.

**Напомене:** На колоквијуму нису дозвољена никаква помоћна средства, ни калкулатори, ни литература. Колоквијум траје 90 минута. **Коначно решење задатка 3, попунити искључиво на формулару.**

### Слика 1 - део оперативне меморије

### Слика 2 - пристигли захтеvi

[illegible]

Изглед стека:

Ситуација 1		
SP		F000h

Ситуација 2		
		F000h
	1005	FFFFh
SP	I=1 T=1 L=00	EF FEh

Ситуација 3		
		F000h
	1005	FFFFh
	I=1 T=1 L=00	EF FEh
	2006	EF FDh
SP	I=0 T=0 L=01	EF FCh

Ситуација 4		
		F000h
	1005	FFFFh
	I=1 T=1 L=00	EF FEh
	2006	EF FDh
	I=0 T=0 L=01	EF FCh
	5001	EF FBh
SP	I=0 T=0 L=01	EF FAh

Ситуација 5		
		F000h
	1005	FFFFh
	I=1 T=1 L=00	EF FEh
	2006	EF FDh
	I=0 T=0 L=01	EF FCh
	5001	EF FBh
	I=0 T=0 L=01	EF FAh
	4002	EF F9h
SP	I=1 T=0 L=01	EF F8h

Ситуација 6		
		F000h
	1008	FFFFh
SP	I=1 T=1 L=00	EF FEh

б) MEM[0100h] = 6h

в) Програм:

LOAD #0h  
OUT 5h  
LOAD #1h  
OUT 10h  
LOAD #2h  
OUT 15h

г)

Адреса	Инструкција
3000	ST 600h
3003	POP
3004	AND #FFFBh
3007	PUSH
3008	LD 600h
300B	DEC
300C	RTI

Могуће је решење и са постављањем L бита.  
Уместо AND #FFFBh, треба да стоји OR #3h или OR #2h