



## Организација рачунара – КЗ

**1.(20)** Једноадресни процесор са раздвојеним меморијским и улазно/излазним адресним простором, меморија, периферија PER0 (adrCR=FF10h, adrSR=FF11h, adrDR=FF12h) са придруженим контролером периферије DMA (adrCR=FF00h, adrSR=FF01h, adrDR=FF02h, adrCNT=FF03h, adrAs=FF04h, adrAd=FF05h), периферија PER1 (adrCR=FF20h, adrSR=FF21h, adrDR=FF22h) и периферија PER2 (adrCR=FF30h, adrSR=FF31h, adrDR=FF32h) повезани су системском магистралом са 16-битном адресном и 16-битном магистралом података. Адресирање је на нивоу 16-битних речи. Механизам прекида је векторисан, а број улаза у IV табелу је одређен фиксно за сваку периферију. У управљачким регистрима бит 4 је Start којим се дозвољава почетак операције, бит 15 одређује смер операције (0-улаз, 1-излаз), бит 0 је Enable којим се дозвољава прекид, а у статусним регистрима бит 15 је Ready који сигнализира спремност контролера. Бит 1 управљачког регистра DMA контролера задаје режим рада (0-циклус по циклус, 1-блоковски). Написати главни програм и одговарајуће прекидне рутине којима се: упоредо врши читавање низа A(i) (i=0...FFFh) са PER0 у меморијски блок који почиње од адресе 1000h, и низа B(i) (i=0...FFFh) са PER2 у меморијски блок почев од адресе 2000h. Након читавања низова A и B врши се формирање низа C, који почиње од адресе 3000h у меморији, на следећи начин:  $C(i) = A(i) + B(i)$  (i=0...FFFh). На крају се низ C шаље на периферију PER1. Пријем са PER0 реализовати коришћењем DMA контролера у блоковском режиму рада, пријем са PER2 реализовати испитивањем бита спремности, а слање низа C на PER1 реализовати коришћењем механизма прекида.

**2. (20)** У процесору рачунара постоји кеш меморија реализована у техници асоцијативног пресликавања. Оперативна меморија је капацитета 8GB и ширине меморијске речи 1B. "Data" део кеш меморије је капацитета 128KB и ширине меморијске речи 1B. Пресликавање је на нивоу блокова величине 512B.

(5) Нацртати структурну шему кеш меморије и оперативне меморије. Приказати како се генеришу адресе кеш меморије и оперативне меморије у свим ситуацијама које могу да настану при приступу кеш меморији. Означити све капацитете и ширине поља свих делова кеш меморије и оперативне меморије.

(5) Објаснити како се утврђује да ли у кеш меморији постоји сагласност и како се адресирани бајт податка чита из кеш меморије или уписује у кеш меморију у ситуацији када је утврђено да у кеш меморији постоји сагласност.

(5) Објаснити како се генеришу адресе свих делова кеш меморије и оперативне меморије у ситуацији када је утврђено да у кеш меморији не постоји сагласност и када се блок података из улаза кеш меморије одабраног за замену враћа из кеш меморије у оперативну меморију. Објаснити и механизам којим се обезбеђује да се блок података из улаза кеш меморије одабраног за замену не враћа из кеш меморије у оперативну меморију уколико није било уписа у дати блок. Претпоставити да се број улаза кеш меморије одабраног за замену налази у посебном регистру улаза за замену.

(5) Објаснити како се генеришу адресе свих делова кеш меморије и оперативне меморије у ситуацији када је утврђено да у кеш меморији не постоји сагласност и када се блок података довлачи из оперативне меморије у улаз кеш меморије одабран за замену. Објаснити и како се том приликом врши ажурирање "Tag" дела кеш меморије. Претпоставити да се број улаза кеш меморије одабраног за замену налази у посебном регистру улаза за замену.

**Напомене:** На испиту нису дозвољена никаква помоћна средства, ни калкулатори ни литература. Испит траје 3 сата.



## Организација рачунара – К2

1. (15) Адресни простор процесора је величине 128KB, адресбилна јединица је 16-битна реч, а вишеречни бројеви се смештају тако да је на нижој адреси нижа реч. Процесор је једноадресни са раздвојеним меморијским и улазно/излазним адресним просторима, а механизам прекида је векторисан. IV (*Interrupt Vector*) табела почиње од адресе на коју указује регистар IVTP (*Interrupt Vector Table Pointer*), а регистар IVTP има вредност 0002h. Процесор има три улазне линије IRQ1, IRQ2 и IRQ3 за спољне маскирајуће прекиде, при чему је IRQ1 најнижег приоритета, а IRQ3 највишег приоритета, на које су везане периферије PER1, PER2 и PER3, респективно, којима треба доделити улазе 1, 2 и 3 у вектор табели, и којима одговарају прекидне рутине на адресама 1000h, 1006h и 100Bh, респективно. Адресе регистра у којима се чувају бројеви улаза су: 25h, 2Fh и 35h, респективно. Улаз 0 у IV табели се користи у свим осталим случајевима, којима одговара прекидна рутина на адреси 100Fh. У PSW-у постоји бит I (*Interrupt Enable*) који се брише у микропрограму за обраду прекида, као и одређен број L бита. Не прихвата се прекид истог нивоа приоритета. При прекиду се на стеку чувају PC, PSW и ACC тим редом. Стек расте према нижим локацијама. Регистар SP (*Stack Pointer*) показује на прву слободну локацију на стеку. Акумулатор је 16-битни. Дат је део главног програма на слици 1 и прекидне рутине на слици 2. Инструкција на адреси 0100h означена је као 1. (прва) по редоследу извршавања, а свака следећа инструкција која се извршава означена је следећим редним бројем. У току извршавања 2. инструкције стиже захтев за прекид по линији IRQ1, у току 5. по линији IRQ2, а у току 8. по линији IRQ3. На почетку су сви бити PSW-а постављени на 0. Не постоји регистар маске IMR. Инструкције INTE, INTD и RTI не реагују на прекид.

<u>адреса</u>	<u>наредба</u>
0100h	INTE
0101h	LOAD 0003h
0103h	INCA
0104h	STORE 0000h
0106h	DECA
0107h	DECA
0108h	INTD

слика 1

<u>адреса</u>	<u>наредба</u>
1000h	INTE
1001h	ADD #0001h
1003h	STORE 0000h
1005h	RTI
1006h	INTE
1007h	ADD #0003h
1009h	INCA

слика 2

<u>адреса</u>	<u>наредба</u>
100Ah	RTI
100Bh	DECA
100Ch	STORE 0000h
100Eh	RTI
100Fh	SUB 0007h
1011h	INCA
1012h	RTI

- a)(3) Написати део програма којим се додељују бројеви улаза наведеним периферијама.  
b)(3) Нацртати изглед првих 8 улаза у вектор табели, означити адресе релевантних локација и уписати садржаје у њих.  
c)(3) Написати секвенцу адреса наредби које се редом извршавају, почев од адресе 0100h.  
d)(3) Приказати садржај свих познатих локација на врху стека након извршавања 9. инструкције. За сачувану вредност PSW дати само вредност бита I и L. Назначити у коме смеру расте стек.  
e)(3) Која ће се вредност налазити на локацији 0000h након извршења секвенце под c)?

2. (5) Модули мод3, мод2, мод1 и мод0, од којих је мод3 највишег а мод0 најнижег приоритета, учествују у арбитражи да би могли да реализују циклус на магистрали. Нацртати структуру арбитразора за случај паралелне арбитрације и повезати ове модуле на арбитразор водећи рачуна о њиховим приоритетима. Објаснити како се реализује арбитрација.