

**Ј. ЂОРЂЕВИЋ, З. РАДИВОЈЕВИЋ, М. ПУНТ,
Б. НИКОЛИЋ, Д. МИЛИЋЕВ, Ј. ПРОТИЋ,
А. МИЛЕНКОВИЋ**

**АРХИТЕКТУРА
И ОРГАНИЗАЦИЈА
РАЧУНАРА**

**ПРЕКИДИ, МАГИСТРАЛА И
УЛАЗ/ИЗЛАЗ**

ЗБИРКА РЕШЕНИХ ЗАДАТАКА

Београд 2013.

САДРЖАЈ

САДРЖАЈ.....	I
1.1 СЕКВЕНЦА САДРЖАЈА НА МАГИСТРАЛИ.....	2
1.1.1 ЗАДАТАК.....	2
1.1.2 ЗАДАТАК.....	8
1.1.3 ЗАДАТАК.....	15
1.1.4 ЗАДАТАК.....	21
1.1.5 ЗАДАТАК.....	26
1.1.6 ЗАДАТАК.....	31

1.1 СЕКВЕНЦА САДРЖАЈА НА МАГИСТРАЛИ

1.1.1 ЗАДАТАК

Посматра се рачунар код кога су меморијски и улазно/излазни адресни простори раздвојени. Меморијски и улазно/излазни адресни простори су капацитета 2^{16} бајтова, а ширина адресибилне речи у оба адресна простора је 8 битова. Адресе у меморијском и улазно/излазном адресном простору су 16-битне и односе се на 8-битне речи.

Адресна магистрала 16-битна, а магистрала података 8-битна.

Процесор је са једноадресним форматом инструкција. Операције се извршавају над 8-битним целобројним величинама без знака и са знаком. Од програмски доступних регистара постоји 8-битни акумулатор.

Инструкције су променљиве дужине (слика 1) и то један, два или три бајта. Битови првог бајта су означени са 23 до 16, другог бајта са 15 до 8 и трећег бајта са 7 до 0. Структура инструкција је дефинисана на следећи начин:

1. Безадресне инструкције.

Дужина инструкција је 1 бајт. Битови 23 и 22 имају фиксне вредности 00b, док се битовима 21 до 16, означеним са ОС, специфицира код операције. Инструкцијом HALT се зауставља процесор. Остале безадресне инструкције које се не користе у овом задатку су PUSH, POP, RTS, RTI и померачке инструкције. Инструкцијама HALT, PUSH, POP, RTS и RTI су додељени кодови операција 000000b, 000001b, 000010b, 000011b и 000100b, респективно.

2. Инструкције скока и инструкција софтверског прекида.

Дужина инструкција је 3 бајта за инструкције апсолутног скока и 2 бајта за инструкције РС релативног скока и инструкцију софтверског прекида. Битови 23 до 21 имају фиксне вредности 010b, бит 20 има вредност 0 за инструкције апсолутног скока и вредност 1 за инструкције РС релативног скока и инструкцију софтверског прекида. Битовима 19 до 16, означеним са ОС, се специфицира код операције. Битови 15 до 8 и 7 до 0 представљају виши и нижи бајт 16-битне адресе код апсолутног скока, док битови 15 до 8 представљају померај као целобројну величину са знаком код РС релативног скока и број улаза у IV табелу као целобројну величину без знака код инструкције софтверског прекида. Инструкције апсолутног и РС релативног скока које се не користе су JMP, JSR и инструкције условног скока, док је инструкција софтверског прекида INS. Инструкцијама JSR и JMP су додељени кодови операција 0000b и 0001b, респективно, док је инструкцији INS додељен код 1111b.

3. IN и OUT инструкције

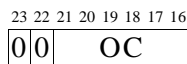
Дужина инструкција је 3 бајта. Битови 23 до 20 имају фиксне вредности 011b, док се битовима 20 до 16, означеним са ОС, специфицира код операције. Битови 15 до 8 и 7 до 0 представљају виши и нижи бајт 16-битне адресе регистра у улазно/излазном адресном простору. Инструкцијом IN се садржај регистра из улазно/излазног адресног простора пребацује акумулатор, док се инструкцијом OUT чини обрнуто. Ове инструкције се не користе у овом задатку. Инструкцијама IN и OUT су додељени кодови операција 00000b и 00001b, респективно.

4. LOAD, STORE, аритметичке и логичке инструкције

Дужина инструкција је 2 или 3 бајта. Бит 23 има фиксну вредност 1b, битовима 22 до 18, означеним са ОС, се специфицира код операције, док се битовима 17 и 16 специфицира начин адресирања и то 00b, 01b, 10b и 11b за непосредно, меморијско директно, меморијско индиректно и РС реалитивно адресирање, респективно. Битови 15 до 8 представљају 8-битну непосредну величину у случају непосредног адресирања, па је тада дужина инструкције два бајта, док битови 15 до 8 и 7 до 0 представљају виши

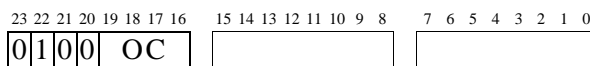
и нижи бајт или 16-битне адресе меморијске локације у случају меморијског директног и меморијског индиректног адресирања или 16-битног помераја у случају РС реалитивног адресирања, па је тада дужина инструкције три бајта. Адреса меморијске локације дужине 16 бита у случају меморијског индиректно адресирања заузима две суседне меморијске локације, при чему се виши бајт адресе налази на нижој а нижи бајт на вишој меморијској локацији. Инструкцијом LOAD се непосредна величина из инструкције или садржај локације из меморијског адресног простора пребацује у акумулатор, док се инструкцијом STORE садржај акумулатора пребацује само у локацију у меморијском адресном простору јер непосредно адресирање за одредишни операнд није дозвољено. Инструкција ADD сабира садржај акумулатора и операнда специфицираног адресним делом инструкције и резултат смешта у акумулатор. Инструкција ADD реализује логичку И операцију садржаја акумулатора и операнда специфицираног адресним делом инструкције и резултат смешта у акумулатор. На сличан начин се реализују и остале аритметичке и логичке операције које се не користе у овом задатку. Инструкцијама LOAD, STORE, ADD и AND су додељени кодови операција 00000b, 00001b, 00010b и 00011b, респективно.

1. Безадресне инструкције

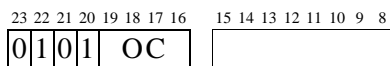


2. Инструкције скока и инструкција софтверског прекида

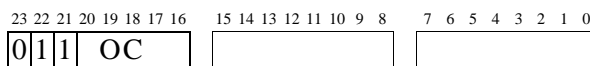
Инструкције апсолутног скока



Инструкције РС релативног скока и инструкција софтверског прекида

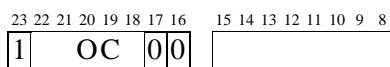


3. IN и OUT инструкције

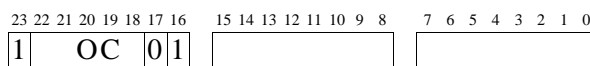


4. LOAD, STORE, аритметичке и логичке инструкције

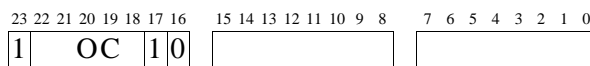
LOAD, STORE, аритметичке и логичке инструкције – непосредно адресирање



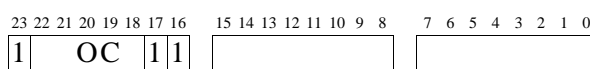
LOAD, STORE, аритметичке и логичке инструкције – меморијско директно адресирање



LOAD, STORE, аритметичке и логичке инструкције – меморијско индиректно адресирање



LOAD, STORE, аритметичке и логичке инструкције – РС релативно адресирање



Слика 1 Формати инструкција

Стек расте према нижим меморијским локацијама, а регистар SP указује на прву слободну меморијску локацију.

Садржај дела оперативне меморије је приказан на слици 2.

Адреса	0000h	0001h	0002h	0003h	0004h	0005h	0006h	0007h	0008h	0009h	000Ah	...
Садржај	02h	34h	11h	05h	42h	36h	00h	03h	05h	00h	07h	...
...
Адреса	3000h	3001h	3002h	3003h	3004h	3005h	3006h	3007h	3008h	3009h	300Ah	...
Садржај	12h	34h	80h	05h	8Ah	00h	09h	85h	00h	17h	00h	...

Слика 2 Садржај дела меморије

а) Навести секвенцу садржаја на адресној магистралаи, магистралаи података и управљачкој магистралаи за сваки циклус на магистралаи при извршавању програма приказаног на 3. Претпоставити да се током извршавања овог програма не генерише прекид. Стога се приликом извршавања инструкција пролази само кроз фазе *читање инструкције (IF)*, *декодовање инструкције (IF)* и *извршавање операције (IE)*, а не и кроз фазу *опслуживање прекида (IS)*.

адреса	инструкција	коментар
3002h	LOAD #5	; непосредно адресирање
3004h	ADD (0009h)	; меморијско индиректно адресирање
3007h	STORE 0017h	; меморијско директно адресирање
300Ah	HALT	; заустављање процесора

Слика 3 Програм

Решење:

а) Секвенца садржаја на адресној магистралаи, магистралаи података и управљачкој магистралаи за сваки циклус на магистралаи је приказана у табели са слике 4.

У табели се користе следеће ознаке:

Pb –редни број циклуса на магистралаи,

ABUS_{15..0} - садржај адресне магистрале,

DBUS_{7..0} - садржај магистрале података,

RD - управљачка линија магистрале која има вредност 1 када је у питању циклус читања,

WR - управљачка линија магистрале која има вредност 1 када је у питању циклус уписа,

M/ \overline{IO} - управљачки линија магистрале који има вредност 1 или 0 у зависности од тога да ли се циклус читања/уписа реализује са меморијским или улазно/излазним адресним простором, респективно,

PC_{15..0} - садржај програмског бројача процесора након завршетка циклуса,

IR_{23..16} - садржај првог бајта прихватног регистра инструкције процесора након завршетка циклуса на магистралаи,

IR_{15..8} - садржај другог бајта прихватног регистра инструкције процесора након завршетка циклуса на магистралаи,

IR_{7..0} - садржај трећег бајта прихватног регистра инструкције процесора након завршетка циклуса на магистралаи,

TMP_{15..0} - садржај прихватног регистра операнда или адресе операнда након уписа у дати регистар,

ACC_{7..0} - садржај акумулатора након уписа у дати регистар,

IF- фаза *читање инструкције*,

ID - фаза *декодовање инструкције* (формирања адресе операнда и читања операнда),

IE - фаза *извршавање операције* и

X- недефинисана хексадецимална вредност.

P6	ABUS _{15..0}	DBUS _{7..0}	RD	WR	M/IO	PC _{15..0}	IR _{23..16}	IR _{15..8}	IR _{7..0}	TMP _{15..0}	ACC _{7..0}	Коментар
	-	-	-	-	-	3002	-	-	-	-	-	
1	3002	80	1	0	1	3003	80	-	-	-	-	IF: LOAD #5, први бајт инс.
2	3003	05	1	0	1	3004	80	05	-	-	-	IF: LOAD #5, други бајт инс.
										XX05		ID: LOAD #5, читање опрнд. из прихв. рег. инс., TMP _{7..0} =IR _{15..8}
											05	IE: LOAD #5, ACC _{7..0} =TMP _{7..0}
3	3004	8A	1	0	1	3005	8A	-	-	-	05	IF: ADD (0009h), први бајт инс.
4	3005	00	1	0	1	3006	8A	00	-	-	05	IF: ADD (0009h), други бајт инс.
5	3006	09	1	0	1	3007	8A	00	09	-	05	IF: ADD (0009h), трећи бајт инс.
6	0009	00	1	0	1	3007	8A	00	09	00XX	05	ID: ADD (0009h), формирање адресе опрнд -читање вишег бајта, TMP _{15..8} =MEM[0009]
7	000A	07	1	0	1	3007	8A	00	09	0007	05	ID: ADD (0009h), формирање адресе опрнд -читање нижег бајта, TMP _{7..0} =MEM[000A]
8	0007	03	1	0	1	3007	8A	00	09	XX03	05	ID: ADD (0009h), читање опрнд. из мем. лок., TMP _{7..0} =MEM[0007]
											08	IE: ADD (0009h), ACC _{7..0} =ACC _{7..0} +TMP _{7..0}
9	3007	85	1	0	1	3008	85	-	-	-		IF: STORE 0017h, први бајт инс.
10	3008	00	1	0	1	3009	85	00	-	-		IF: STORE 0017h, други бајт инс.
11	3009	17	1	0	1	300A	85	00	17	-		IF: STORE 0017h, трећи бајт инс.
											0017	ID: STORE 0017h, формирање адресе опрнд., TMP _{15..8} =IR _{15..8} , TMP _{7..0} =IR _{7..0}
12	0017	08	0	1	1	300A	85	00	17	0017	08	IE: STORE 0017h, упис опрнд. у мем. лок., MEM[0017]=ACC _{7..0}
13	300A	00	1	0	1	300B	00	-	-	-	08	IF: HALT , први бајт инс.
												IE: HALT , START=0

Слика 4 Секвенца садржаја на магистрали

Извршавање програма почиње фазом *читање инструкције* (IF) прве инструкције **LOAD** #5 на коју указује тренутна вредност 3002h садржаја програмског бројача PC_{15..0}. Прва инструкција је дужине два бајта који имају вредности 80h и 05h, а налазе се у меморији на адресама 3002h и 3003h, респективно. С обзиром да је ширина меморијске речи 8 битова, процесор из меморије чита бајт по бајт. Приликом читања прве инструкције процесор на магистрали генерише два циклуса читања из меморије. У првом циклусу читања процесор на линије адресне магистрале ABUS_{15..0} пропушта вредност 3002h програмског бројача PC_{15..0} како би са те адресе прочитао први бајт инструкције 80h. Пошто је у питању циклус читања из меморије управљачки сигнали магистрале RD и M/IO имају вредност 1. Прочитани садржај 80h се појављује на линијама магистрале података DBUS_{7..0} и уписује у разреде IR_{23..16} прихватног регистра инструкције процесора, а садржај програмског бројача PC_{15..0} инкрементира на 3003h. На сличан начин се у другом циклусу читања са адресе 3003h чита други бајт инструкције 05h. Прочитани садржај се појављује на линијама магистрале података DBUS_{7..0} и уписује у разреде IR_{15..8} прихватног регистра инструкције, а садржај програмског бројача PC_{15..0} инкрементира на 3004h.

У фази *декодовања инструкције* (ID) нема циклуса на магистрали, јер је операнд због коришћења непосредног адресирања специфициран другим бајтом инструкције и налази се у разредима IR_{15..8} прихватног регистра инструкције. Фаза *декодовања инструкције* (ID) се састоји у пребацивању операнда чија је вредност 05h из разреда IR_{15..8} прихватног регистра инструкције у разреде TMP_{7..0} прихватног регистра операнда.

Са XX је означено да је садржај разреда TMP_{15:8} прихватног регистра операнда недефинисан.

У фази *извршавање операције* (IE) садржај разреда TMP_{7:0} прихватног регистра операнда се уписује у акумулатор ACC_{7:0}.

По завршетку фазе *извршавање операције* (IE) прве инструкције **LOAD #5** се прелази на фазу *читање инструкције* (IF) друге инструкције **ADD (0009h)**. Друга инструкција је дужине три бајта који имају вредности 8Ah, 00h и 09h, а налазе се у меморији на адресама 3004h, 3005h и 3006h, респективно. Приликом читања друге инструкције процесор на магистрали генерише три циклуса читања из меморије. Циклуси читања бајтова друге инструкције се реализују на идентичан начин као и циклуси читања бајтова прве инструкције. Прочитане вредности се уписују у разреде IR_{23:16}, IR_{15:8} и IR_{7:0} прихватног регистра инструкције, а садржај програмског бројача PC_{15..0} инкрементира на 3005h, 3006h и 3007h, респективно.

У фази *декодовања инструкције* (ID) постоје три циклуса на магистрали, јер због коришћења меморијског индиректног адресирања процесор најпре у два циклуса на магистрали из меморије чита два бајта 16-битне адресе операнда, а затим у трећем циклусу на магистрали чита и сам 8-битни операнд. Адреса операнда је 16-битна и заузима две суседне меморијске локације, при чему се виши бајт адресе налази на нижој а нижи бајт адресе на вишој адреси. Адреса меморијске локације на којој се налази виши бајт адресе операнда дата је другим и трећим бајтом инструкције и налази се у разредима IR_{15:8} (виши бајт) и IR_{7:0} (нижи бајт) прихватног регистра инструкције. У прва два циклуса на магистрали процесор најпре из меморије са адресе 0009h чита садржај 00h, који представља виши бајт адресе операнда, и уписује у разреде TMP_{15:8}, а затим из меморије са адресе 000Ah чита садржај 07h, који представља нижи бајт адресе операнда, и уписује у разреде TMP_{7:0}. У трећем циклусу на магистрали садржај 0007h прихватног регистра адресе операнда TMP_{15:0} се користи као адреса меморијске локације са које се чита операнд чија је вредност 03h и уписује у разреде TMP_{7:0} прихватног регистра операнда. Пошто су у питању циклуси читања из меморије управљачки сигнали магистрале RD и M/IO имају вредност 1. Са XX је означено да је садржај разреда TMP_{15:8} прихватног регистра операнда недефинисан.

У фази *извршавање операције* (IE) садржај 03h разреда TMP_{7:0} прихватног регистра операнда се сабира са садржајем 05h акумулатора ACC_{7:0} и добијена сума 08h уписује у акумулатор ACC_{7:0}.

По завршетку фазе *извршавање операције* (IE) друге инструкције **ADD (0009h)** се прелази на фазу *читање инструкције* (IF) треће инструкције **STORE 0017h**. Трећа инструкција је дужине три бајта који имају вредности 85h, 00h и 07h, а налазе се у меморији на адресама 3007h, 3008h и 3009h, респективно. Приликом читања треће инструкције процесор на магистрали генерише три циклуса читања из меморије. Циклуси читања бајтова треће инструкције се реализују на идентичан начин као и циклуси читања бајтова прве инструкције. Прочитане вредности се уписују у разреде IR_{23:16}, IR_{15:8} и IR_{7:0} прихватног регистра инструкције, а садржај програмског бројача PC_{15..0} инкрементира на 3008h, 3009h и 300Ah, респективно.

У фази *декодовање инструкције* (ID) нема циклуса на магистрали, јер се због коришћења меморијског директног адресирања у овој фази само формира адреса меморијске локације у коју треба да се упише 8-битни операнд дат садржајем акумулатора ACC_{7:0}. Фаза *декодовање инструкције* се састоји у пребацивању адресе меморијске локације чија је вредност 0017h и која је дата другим и трећим бајтом инструкције из разреда IR_{15:8} (виши бајт) и IR_{7:0} (нижи бајт) прихватног регистра инструкције у разреде TMP_{15:8} и TMP_{7:0} прихватног регистра адресе операнда.

У фази *извршавања операције* (IE) процесор на магистралаи генерише циклус уписа у меморију. На линије адресне магистрале ABUS_{15:0} се пропушта вредност 0017h прихватног регистра адресе операнда TMP_{15:0} а на линије магистрале података DBUS_{7:0} вредност 08h акумулатора ACC_{7:0}. Пошто је у питању циклус уписа у меморију управљачки сигнали магистрале WR и $\overline{M/\overline{IO}}$ имају вредност 1.

По завршетку фазе *извршавања операције* (IE) треће инструкције **STORE** 0017h се прелази на фазу *читања инструкције* (IF) четврте инструкције **HALT**. Четврта инструкција је дужине један бајт који има вредности 00h, а налази се у меморији на адреси 300Ah. Приликом читања четврте инструкције процесор на магистралаи генерише један циклус читања из меморије. Циклус читања једног бајта четврте инструкције се реализују на идентичан начин као и циклуси читања бајтова прве инструкције. Прочитана вредност се уписује у разреде IR_{23:16} прихватног регистра инструкције, а садржај програмског бројача PC_{15:0} инкрементира на 300Bh.

Инструкција **HALT** је безадресна инструкција, па се после фазе читања инструкције одмах прелази на фазу *извршавања операције* (IE) у оквиру које се зауставља процесор.

1.1.2 ЗАДАТАК

Посматра се рачунар код кога су меморијски и улазно/излазни адресни простори раздвојени. Меморијски и улазно/излазни адресни простори су капацитета 2^{16} бајтова, а ширина адресибилне речи у оба адресна простора је 8 битова. Адресе у меморијском и улазно/излазном адресном простору су 16-битне и односе се на 8-битне речи.

Адресна магистрала 16-битна, а магистрала података 8-битна.

Подаци над којима се извршавају операције су 8-битне целобројне величине без знака и са знаком. Процесор има четири 8-битна регистра опште намене R0 до R3.

Инструкције су променљиве дужине (слика 5) и то један или три бајта. Битови првог бајта су означени са 23 до 16, другог бајта са 15 до 8 и трећег бајта са 7 до 0. Структура инструкција је дефинисана на следећи начин:

1. Безадресне инструкције.

Дужина инструкција је 1 бајт. Битови 23 до 20 имају фиксне вредности 0000b, док се битовима 19 до 16, означеним са C, специфицира код операције. Инструкцијом HALT се зауставља процесор. Остале безадресне инструкције које се не користе су RTS и RTI. Инструкцијама HALT, RTS и RTI су додељени кодови операција 0000b, 0001b и 0010b, респективно.

2. Инструкције скока и инструкција софтверског прекида.

Дужина инструкција је 3 бајта за инструкције апсолутног скока и 2 бајта за инструкције PC релативног скока и инструкцију софтверског прекида. Битови 23 до 20 имају фиксне вредности 0001b, бит 19 има вредност 0 за инструкције апсолутног скока и вредност 1 за инструкције PC релативног скока и инструкцију софтверског прекида. Битовима 18 до 16, означеним са C, се специфицира код операције. Битови 15 до 8 и 7 до 0 представљају нижи и виши бајт 16-битне адресе скока код инструкција апсолутног скока, док битови 15 до 8 представљају померај као целобројну величину са знаком код инструкција PC релативног скока и број улаза у IV табелу код инструкције софтверског прекида. Инструкције апсолутног и PC релативног скока које се не користе су JMP, JSR и инструкције условног скока, док је инструкција софтверског прекида INS. Инструкцијама JSR и JMP су додељени кодови операција 000b и 001b, респективно, док је инструкцији INS додељен код 111b.

3. LOAD/STORE инструкције.

Дужина инструкција је 3 бајта. Битови 23 до 20 имају фиксне вредности 0010b, док се вредностима 0 и 1 бита 19, означеног са C, специфицирају кодови операција LOAD и STORE, респективно, вредностима 0 и 1 бита 18, означеног са A, специфицирају начини адресирања и то меморијско директно и меморијско индиректно адресирање у меморијском адресном простору, респективно, и вредностима 0 до 3 битова 17 и 16, означених са SD, специфицира један од четири регистра опште намене R0 до R3. Битови 15 до 8 и 7 до 0 представљају, у случају меморијског директног адресирања, нижи и виши бајт 16-битне адресе локације у меморијском адресном простору у којој се налази 8-битни операнд. Битови 15 до 8 и 7 до 0 представљају, у случају меморијског индиректног адресирања, нижи и виши бајт 16-битне адресе локације у меморијском адресном простору на којој се налази нижи а на првој следећој виши бајт 16-битне адресе локације у меморијском адресном простору у којој се налази 8-битни операнд. Инструкцијом LOAD се садржај локације из меморијског адресног простора пребацује у један од регистара опште намене, док се инструкцијом STORE чини обрнуто. Инструкција LOAD се користи, док се инструкција STORE не користи.

4. IN/OUT инструкције.

Дужина инструкција је 3 бајта. Битови 23 до 20 имају фиксне вредности 0011b, док се вредностима 0 и 1 бита 19, означеног са C, специфицирају кодови операција IN и

OUT, респективно, вредностима 0 и 1 бита 18, означеног са А, специфицирају начини адресирања и то у/и директно и у/и индиректно адресирање регистра у улазно/излазном адресном простору, респективно, и вредностима 0 и 3 битова 17 и 16, означених са SD, специфицира један од четири регистра опште намене R0 до R3. Битови 15 до 8 и 7 до 0 представљају, у случају у/и директног адресирања, нижи и виши бајт 16-битне адресе регистра у улазно/излазном адресном простору у коме се налази 8-битни операнд. Битови 15 до 8 и 7 до 0 представљају, у случају у/и индиректног адресирања, нижи и виши бајт 16-битне адресе локације у меморијском адресном простору на којој се налази нижи а на првој следећој виши бајт 16-битне адресе регистра у улазно/излазном адресном простору у коме се налази 8-битни операнд. Инструкцијом IN се садржај регистра из улазно/излазног адресног простора пребацује у један од регистара опште намене, док се инструкцијом OUT чини обрнуто. Инструкција OUT се користи, док се инструкција IN не користи.

5. Аритметичке, логичке и померачке инструкције.

Дужина инструкција је 1 бајт. Битови 23 до 20, означени са C, вредностима 0100b до 1111b специфицирају кодове аритметичких, логичких и померачких операција, битови 19 и 18, означени са SD, вредностима 0 до 3 специфицирају један од регистара опште намене R0 до R3 који представља прво извориште и одредиште операнда и битови 17 и 16, означени са S2, вредностима 0 до 3 специфицирају један од регистара опште намене R0 до R3 који представља друго извориште операнда. Инструкција ADD сабира садржаје регистара опште намене специфицираних са SD и S2 и резултат смешта у регистар опште намене специфициран са SD. На сличан начин се реализују и остале аритметичке и логичке операције које се не користе. Померачке операције, које се не користе, померају садржај регистара опште намене специфициран са S2 и резултат смештају у регистар опште намене специфициран са SD. Инструкцији ADD је додељен код операције 0100b.

1. Безадресне инструкције

23	22	21	20	19	18	17	16
0	0	0	0				C

2. Инструкције скока и инструкција софтверског прекида

Инструкције апсолутног скока

23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	1	0			C																

Инструкције PC релативног скока и инструкција софтверског прекида

23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8
0	0	0	1	1			C								

3. LOAD/STORE инструкције

23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	1	0	C	A	S	D																

4. IN/OUT инструкције – директно или индиректно адресирање

23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	1	1	C	A	S	D																

5. Аритметичке, логичке и померачке инструкције

23	22	21	20	19	18	17	16
	C	S	D	S	2		

Слика 5 Формати инструкција

Стек расте према вишим меморијским локацијама, а регистар SP указује на последњу заузету меморијску локацију.

Садржај дела оперативне меморије почев од адресе 003Dh је приказан на слици 6.

Адреса	...	003Dh	003Eh	003Fh	0040h	0041h	0042h	0043h	0044h	0045h	0046h
Садржај	...	04h	3Dh	00h	25h	3Eh	00h	22h	3Dh	00h	46h
Адреса	0047h	0048h	0049h	004Ah	...						
Садржај	39h	40h	00h	00h	...						

Слика 6 Садржај дела меморије

а) Навести секвенцу садржаја на адресној магистралу, магистралу података и управљачкој магистралу за сваки циклус на магистралу при извршавању програма приказаног на 7. Претпоставити да се током извршавања овог програма не генерише прекид. Стога се приликом извршавања инструкција пролази само кроз фазе *читање инструкције (IF)*, *декодовање инструкције (IF)* и *извршавање операције (IE)*, а не и кроз фазу *опслуживање прекида (IS)*.

адреса	инструкција	коментар
0040h	LOAD R1, (3Eh)	; меморијско индиректно адресирање; R1=MEM[MEM[3Eh]]
0043h	LOAD R2, 3Dh	; меморијско директно адресирање; R2=MEM[3Dh]
0046h	ADD R1, R2	; R1=R1+R2
0047h	OUT R1, 40h	; у/и директно адресирање; IO[40h]=R1
004Ah	HALT	; заустављање процесора

Слика 7 Програм

б) Навести који је садржај локације на адреси 0040h у меморијском адресном простору, а који регистра на адреси 0040h у улазно/излазном адресном простору након извршења датог програма.

Решење:

а) Секвенца садржаја на адресној магистралу, магистралу података и управљачкој магистралу за сваки циклус на магистралу је приказана у табели са слике 8.

У табели се користе следеће ознаке:

Pb –редни број циклуса на магистралу,

ABUS_{15:0} - садржај адресне магистрале,

DBUS_{7:0} - садржај магистрале података,

RD - управљачка линија магистрале која има вредност 1 када је у питању циклус читања,

WR - управљачка линија магистрале која има вредност 1 када је у питању циклус уписа,

M/IO- управљачки линија магистрале који има вредност 1 или 0 у зависности од тога да ли се циклус читања/уписа реализује са меморијским или улазно/излазним адресним простором, респективно,

PC_{15:0} - садржај програмског бројача процесора након завршетка циклуса,

IR_{23:16} - садржај првог бајта прихватног регистра инструкције процесора након завршетка циклуса на магистралу,

IR_{15:8} - садржај другог бајта прихватног регистра инструкције процесора након завршетка циклуса на магистралу,

IR_{7:0} - садржај трећег бајта прихватног регистра инструкције процесора након завршетка циклуса на магистралу,

TMP_{15:0} - садржај прихватног регистра операнда или адресе операнда након уписа у дати регистар,

R1_{7:0} и R2_{7:0} садржаји регистара опште намене након уписа у дате регистре,

IF- фаза читање инструкције,
 ID - фаза декодовање инструкције (формирања адресе операнда и читања операнда),
 IE - фаза извршавање операције и
 X- недефинисана хексадецимална вредност.

P6	ABUS _{15..0}	DBUS _{7..0}	RD	WR	M/O	PC _{15..0}	IR _{23..16}	IR _{15..8}	IR _{7..0}	TMP _{15..0}	R _{17..0}	R _{27..0}	Коментар
	-	-	-	-	-	0040	-	-	-	-	-	-	
1	0040	25	1	0	1	0041	25	-	-	-	-	-	IF: LOAD R1 , (3Eh), први бајт инс.
2	0041	3E	1	0	1	0042	25	3E	-	-	-	-	IF: LOAD R1 , (3Eh), други бајт инс.
3	0042	00	1	0	1	0043	25	3E	00	-	-	-	IF: LOAD R1 , (3Eh), трећи бајт инс.
4	003E	3D	1	0	1	0043	25	3E	00	XX3D	-	-	ID: LOAD R1 , (3Eh), формирање адресе опрнд. - читање нижег бајта, TMP _{7..0} =MEM[003E]
5	003F	00	1	0	1	0043	25	3E	00	003D	-	-	ID: LOAD R1 , (3Eh), формирање адресе опрнд. - читање вишег бајта, TMP _{15..8} =MEM[003F]
6	003D	04	1	0	1	0043	25	3E	00	XX04	-	-	ID: LOAD R1 , (3Eh), читање опрнд. из мем. лок., TMP _{7..0} =MEM[003D]
											04	-	IE: LOAD R1 , (3Eh), R _{17..0} =TMP _{7..0}
7	0043	F9	1	0	1	0044	22	-	-	-	04	-	IF: LOAD R2 , 3Dh, први бајт инс.
8	0044	3D	1	0	1	0045	22	3D	-	-	04	-	IF: LOAD R2 , 3Dh, други бајт инс.
9	0045	00	1	0	1	0046	22	3D	00	-	04	-	IF: LOAD R2 , 3Dh, трећи бајт инс.
										003D	04	-	ID: LOAD R2 , 3Dh, формирање адресе опрнд., TMP _{15..8} =IR _{7..0} , TMP _{7..0} =IR _{15..8}
10	003D	04	1	0	1	0046	22	3D	00	XX04	04	-	ID: LOAD R2 , 3Dh, читање опрнд. из мем. лок., TMP _{7..0} =MEM[003D]
											04	-	IE: LOAD R2 , 3Dh, R _{27..0} =TMP _{7..0}
11	0046	06	1	0	1	0047	06	-	-	-	04	04	IF: ADD R1 , R2, први бајт инс.
											08	04	IE: ADD R1 , R2, R _{17..0} =R _{17..0} +R _{27..0}
12	0047	C1	1	0	1	0048	39	-	-	-	08	04	IF: OUT R1 , 40h, први бајт инс.
13	0048	40	1	0	1	0049	39	40	-	-	08	04	IF: OUT R1 , 40h, други бајт инс.
14	0049	00	1	0	1	004A	39	40	00	-	08	04	IF: OUT R1 , 40h, трећи бајт инс.
										0040	08	04	ID: OUT R1 , 40h, формирање адресе опрнд., TMP _{15..8} =IR _{7..0} , TMP _{7..0} =IR _{15..8}
15	0040	08	0	1	0	004A	39	40	00	-	08	04	IE: OUT R1 , 40h, упис опрнд. у уи рег., UI[0040]=R _{17..0}
16	004A	00	1	0	1	004B	00	-	-	-	08	04	IF: HALT , први бајт инс.
													IE: HALT , START=0

Слика 8 Секвенца садржаја на магистрали

Извршавање програма почиње фазом *читање инструкције* (IF) прве инструкције **LOAD R1**, (3Eh) на коју указује тренутна вредност 0040h садржаја програмског бројача PC_{15..0}. Прва инструкција је дужине три бајта који имају вредности 25h, 3Eh и 00h, а налазе се у локацијама меморије на адресама 0040h, 0041h и 0042h, респективно. С обзиром да је ширина меморијске речи 8 битова, процесор из меморије чита бајт по бајт. Приликом читања прве инструкција процесор на магистрали генерише три циклуса читања из меморије. У првом циклусу читања процесор на линије адресне магистрале ABUS_{15..0} пропушта вредност 0040h програмског бројача PC_{15..0} како би са те адресе прочитао први бајт инструкције F6h. Пошто је у питању циклус читања из меморије управљачки сигнали магистрале RD и M/O имају вредност 1. Прочитани садржај 25h се појављује на линијама магистрале података DBUS_{7..0} и уписује у разреде IR_{23..16} прихватног регистра инструкције процесора, а садржај програмског бројача PC_{15..0} инкрементира на 0041h. На сличан начин се у другом и трећем циклусу читања

са адреса 0041h и 0042h читају други и трећи бајтови бајт инструкције 3Eh и 00h, респективно. Прочитани садржаји се појављује на линијама магистрале података DBUS_{7..0} и уписује у разреде IR_{15..8} и IR_{7..0} прихватног регистра инструкције, а садржај програмског бројача PC_{15..0} инкрементира на 0042h и 0043h, респективно.

У фази *декодовања инструкције* (ID) постоје три циклуса на магистрали, јер због коришћења меморијског индиректног адресирања процесор најпре у два циклуса на магистрали из меморије чита два бајта 16-битне адресе операнда, а затим у трећем циклусу на магистрали чита и сам 8-битни операнд. Адреса операнда је 16-битна и заузима две суседне меморијске локације, при чему се нижи бајт адресе налази на нижој а виши бајт на вишој адреси. Адреса меморијске локације на којој се налази нижи бајт адресе операнда дата је другим и трећим бајтом инструкције и налази се у разредима IR_{15..8} (нижи бајт) и IR_{7..0} (виши бајт) прихватног регистра инструкције. У прва два циклуса на магистрали процесор најпре из меморије са адресе 003Eh чита садржај 3Dh, који представља нижи бајт адресе операнда, и уписује у разреде TMP_{7..0}, а затим из меморије са адресе 003Fh чита садржај 00h, који представља виши бајт адресе операнда, и уписује у разреде TMP_{15..8}. У трећем циклусу на магистрали садржај 003Dh прихватног регистра адресе операнда TMP_{15..0} се користи као адреса меморијске локације са које се чита операнд чија је вредност 04h и уписује у разреде TMP_{7..0} прихватног регистра операнда. Пошто су у питању циклуси читања из меморије управљачки сигнали магистрале RD и M/ \overline{IO} имају вредност 1. Са XX је означено да је садржај разреда TMP_{15..8} прихватног регистра операнда недефинисан.

У фази *извршавање операције* (IE) садржај разреда TMP_{7..0} прихватног регистра операнда се уписује у регистар R1_{7..0}.

По завршетку фазе *извршавање операције* (IE) прве инструкције **LOAD R1**, (3Eh) се прелази на фазу *читање инструкције* (IF) друге инструкције **LOAD R2**, 3Dh. Друга инструкција је дужине три бајта који имају вредности 22h, 3Dh и 00h, а налазе се у меморији на адресама 0043h, 0044h и 0045h, респективно. Приликом читања друге инструкције процесор на магистрали генерише три циклуса читања из меморије. Циклуси читања бајтова друге инструкције се реализују на идентичан начин као и циклуси читања бајтова прве инструкције. Прочитане вредности се уписују у разреде IR_{23..16}, IR_{15..8} и IR_{7..0} прихватног регистра инструкције, а садржај програмског бројача PC_{15..0} инкрементира на 0044h, 0045h и 0046h, респективно.

У фази *декодовања инструкције* (ID) постоји један циклус на магистрали, јер процесор због коришћења меморијског директног адресирања у једном циклусу на магистрали из меморије чита 8 битни операнд. Најпре се адреса меморијске локације, чија је вредност 003Dh и која је дата другим и трећим бајтом инструкције из разреда IR_{15..8} (нижи бајт) и IR_{7..0} (виши бајт), пребацује у разреде TMP_{7..0} и TMP_{15..8} прихватног регистра адресе операнда, респективно. Затим се у циклусу на магистрали садржај 003Dh прихватног регистра адресе операнда TMP_{15..0} користи као адреса меморијске локације са које се чита операнд чија је вредност 04h и уписује у разреде TMP_{7..0} прихватног регистра операнда. Пошто је у питању циклус читања из меморије управљачки сигнали магистрале RD и M/ \overline{IO} имају вредност 1. Са XX је означено да је садржај разреда TMP_{15..8} прихватног регистра операнда недефинисан.

У фази *извршавање операције* (IE) садржај разреда TMP_{7..0} прихватног регистра операнда се уписује у регистар R2_{7..0}.

По завршетку фазе *извршавање операције* (IE) друге инструкције **LOAD R2**, 3Dh се прелази на фазу *читање инструкције* (IF) треће инструкције **ADD R1, R2**. Трећа инструкција је дужине један бајт који има вредности 06h, а налази се у меморији на адреси 0046h. Приликом читања треће инструкције процесор на магистрали генерише један циклус читања из меморије. Циклус читања једног бајта треће инструкције се

реализују на идентичан начин као и циклуси читања бајтова прве инструкције. Прочитана вредност се уписује у разреде $IR_{23:16}$ прихватног регистра инструкције, а садржај програмског бројача $PC_{15..0}$ инкрементира на 0047h.

Ова инструкција нема фазу *декодовање инструкције* (ID) јер се као изворишта и одредиште операнда имплицитно користе регистри опште намене.

У фази *извршавање операције* (IE) садржај 04h регистра опште намене $R1_{7..0}$ се сабира са садржајем 04h регистра опште намене $R2_{7..0}$ и добијена сума 08h уписује у регистар опште намене $R1_{7..0}$.

По завршетку фазе *извршавање операције* (IE) треће инструкције **ADD** $R1, R2$ се прелази на фазу *читање инструкције* (IF) четврте инструкције **OUT** $R1, 40h$. Четврта инструкција је дужине три бајта који имају вредности 39h, 40h и 00h, а налазе се у меморији на адресама 0047h, 0048h и 0049h, респективно. Приликом читања четврте инструкције процесор на магистрали генерише три циклуса читања из меморије. Циклуси читања бајтова четврте инструкције се реализују на идентичан начин као и циклуси читања бајтова прве инструкције. Прочитане вредности се уписују у разреде $IR_{23..16}$, $IR_{15..8}$ и $IR_{7..0}$ прихватног регистра инструкције, а садржај програмског бројача $PC_{15..0}$ инкрементира на 0048h, 0049h и 004Ah, респективно.

У фази *декодовање инструкције* (ID) нема циклуса на магистрали, јер се због коришћења у/и директног адресирања у овој фази само формира адреса регистра улазно/излазног адресног простора у који треба да се упише 8-битни операнд дат садржајем регистра опште намене $R1_{7..0}$. Фаза *декодовање инструкције* (ID) се састоји у пребацивању адресе регистра улазно/излазног адресног простора, чија је вредност 0040h и која је дата трећим и другим бајтом инструкције, из разреда $IR_{7..0}$ (виши бајт) и $IR_{15..8}$ (нижи бајт) прихватног регистра инструкције у разреде $MP_{15..8}$ и $MP_{7..0}$ прихватног регистра адресе операнда.

У фази *извршавање операције* (IE) процесор на магистрали генерише циклус уписа у регистар улазно/излазног адресног простора. На линије адресне магистрале $ABUS_{15..0}$ се пропушта вредност 0040h прихватног регистра адресе операнда $MP_{15..0}$ а на линије магистрале података $DBUS_{7..0}$ вредност 08h регистра опште намене $R1_{7..0}$. Пошто је у питању циклус уписа у регистар улазно/излазног адресног простора управљачки сигнали магистрале WR и M/\overline{IO} имају вредности 1 и 0, респективно.

По завршетку фазе *извршавање операције* (IE) четврте инструкције **OUT** $R1, 0040h$ се прелази на фазу *читање инструкције* (IF) пете инструкције **HALT**. Пета инструкција је дужине један бајт који има вредности 00h, а налази се у меморији на адреси 004Ah. Приликом читања пете инструкције процесор на магистрали генерише један циклус читања из меморије. Циклус читања једног бајта пете инструкције се реализују на идентичан начин као и циклуси читања бајтова прве инструкције. Прочитана вредност се уписује у разреде $IR_{23:16}$ прихватног регистра инструкције, а садржај програмског бројача $PC_{15..0}$ инкрементира на 004Bh.

Инструкција **HALT** је безадресна инструкција, па се после фазе *читање инструкције* (IF) одмах прелази на фазу *извршавање операције* (IE) у оквиру које се зауставља процесор.

в) Са слике 6 се добија да садржај локације са адресе 40h у меморијском адресном простору износи 25h.

Садржај локације са адресе 40h у улазно/излазном простору се добија као резултат извршавања инструкција програма са слике 7.

Првом инструкцијом **LOAD** $R1, (3Eh)$ се, због меморијског индиректног адресирања, најпре са адресе 003Eh и прве следеће адресе (003Fh) у меморијском адресном простору чита нижи и виши бајт 16 битне адресе 003Dh са које се затим из меморијског адресног простора чита 8-битни операнд чија је вредност 4h и уписује у регистар $R1$.

Другом инструкцијом **LOAD R2**, 3Dh се, због директног меморијског адресирања, са адресе 003Dh из меморијског адресног простора чита 8-битни операнд чија је вредност 4h и уписује у регистар R2.

Трећа инструкција **ADD R1, R2** се сабира садржај регистра R1, чија је вредност 4h, и регистра R2, чија је вредност 4h, и резултат, чија је вредност 8h, уписује у регистар R1.

Инструкцијом **OUT** се, због у/и директног адресирања, садржај регистра R1, чија је вредност 8h, уписује у регистар на адреси 40h у улазно/излазном адресном простору.

Према томе, садржај локације са адресе 40h у меморијском адресном простору је 25h, а садржај регистра са адресе 40h у улазно/излазном адресном простору је 08h.

1.1.3 ЗАДАТАК

Посматра се процесор описан у задатку 1.1.1.

Садржај дела оперативне меморије је приказан на слици 9, при чему су све вредности дате у хексадецималном облику.

Адреса	0000	0001	0002	0003	0004	0005	0006	0007	0008	0009	000A	000B
Садржај	02	34	11	05	42	36	00	03	05	00	07	07
...
Адреса	5000	5001	5002	5003	5004	5005	5006	5007	5008	5009	500A	
Садржај	82	00	09	89	00	0B	50	03	00	17	00	
Адреса	500B	500C	500D	500E	500F	5010	5011	5012	5013	5014	5015	...
Садржај	02	85	00	06	01	03	50	03	00	17	00	...

Слика 9 Садржај дела меморије

а) Навести секвенцу садржаја на адресној магистрали, магистрали података и управљачкој магистрали за сваки циклус на магистрали при извршавању програма приказаног на слици 10. Претпоставити да се током извршавања овог програма не генерише прекид. Стога се приликом извршавања инструкција пролази само кроз фазе *читање инструкције (IF)*, *декодовање инструкције (IF)* и *извршавање операције (IE)*, а не и кроз фазу *опслуживање прекида (IS)*.

Претпоставити да је пре почетка извршавања главног програма $SP = E000h$. Навести вредност почетне адресе потпрограма означене са X.

адреса	инструкција	коментар
; главни програм		
5000h	LOAD (9)	; меморијско индиректно адресирање $ACC=MEM[MEM[0009h]]$
5003h	ADD 0Ah	; меморијско директно $ACC1=ACC1+MEM[000Ah]$
5006h	JSR (PC)3	; скок у потпрограм PC релативан скок
5008h	HALT	; заустављање процесора
5009h	...	
; потпрограм		
X	POP	; упис садржаја врха стека у акумулатор; $MEM[-SP]=ACC$
X+1	STORE 0006h	; меморијско директно адресирање $MEM[0006h]=ACC$
X+4	PUSH	; упис садржаја акумулатора на стек $MEM[SP++]=ACC$
X+5	RTS	; повратак из потпрограма
X+6	...	

Слика 10 Програм

Решење:

а) Секвенца садржаја на адресној магистрали, магистрали података и управљачкој магистрали за сваки циклус на магистрали је приказана у табели са слике 11. У табели се користе исте ознаке као и у табели са слике 4 из задатка 1.1.1, при чему се јавља и колона $SP_{15..0}$ - указивач на врх стека након ажурирања његовог садржаја.

При извршавању инструкције JSR (PC)3h, вредност PC је 5008h, јер PC указује на наредну инструкцију, померај је 3h, па се, због PC релативног скока, за адресу скока добија $PC + 3 = 500Bh$. Због тога је почетна адреса потпрограма X = 500Bh.

P6	ABUS _{15.0}	DBUS _{7.0}	RD	WR	MIO	PC _{15.0}	IR _{23.16}	IR _{15.8}	IR _{7.0}	TMP _{15.0}	ACC _{7.0}	SP _{15.0}	Коментар
	-	-	-	-	-	5000	-	-	-	-	-	E000	
1	5000	82	1	0	1	5001	82	-	-	-	-	E000	IF: LOAD (9h), први бајт инс.
2	5001	00	1	0	1	5002	82	00	-	-	-	E000	IF: LOAD (9h), други бајт инс.
3	5002	09	1	0	1	5003	82	00	09	-	-	E000	IF: LOAD (9h), трећи бајт инс.
4	0009	00	1	0	1	5003	82	00	09	00XX	-	E000	ID: LOAD (9h), формирање адресе опрнд. - читање вишег бајта, TMP _{15.8} =MEM[0009]
5	000A	07	1	0	1	5003	82	00	09	0007	-	E000	ID: LOAD (9h), формирање адресе опрнд. - читање нижег бајта, TMP _{7.0} =MEM[000A]
6	0007	03	1	0	1	5003	82	00	09	XX03	-	E000	ID: LOAD (9h), читање опрнд. из мем. лок., TMP _{7.0} =MEM[0007]
											03	E000	IE: LOAD (9h), ACC _{7.0} =TMP _{7.0}
7	5003	89	1	0	1	5004	89	-	-	-	03	E000	IF: ADD 0Bh, први бајт инс.
8	5004	00	1	0	1	5005	89	00	-	-	03	E000	IF: ADD 0Bh, други бајт инс.
9	5005	0A	1	0	1	5006	89	00	0B	-	03	E000	IF: ADD 0Bh, трећи бајт инс.
										000B	03	E000	ID: ADD 0Bh, формирање адресе опрнд., TMP _{15.8} =IR _{15.8} , TMP _{7.0} =IR _{7.0}
10	000B	07	1	0	1	5006	89	00	0B	XX07	03	E000	ID: ADD 0Bh, читање опрнд. из мем. лок., TMP _{7.0} =MEM[000B]
											0A	E000	IE: ADD 0Bh, ACC _{7.0} =ACC _{7.0} +TMP _{7.0}
11	5006	50	1	0	1	5007	50	-	-	-	0A	E000	IF: JSR (PC)3, први бајт инс.
12	5007	03	1	0	1	5008	50	03	-	-	0A	E000	IF: JSR (PC)3, други бајт инс.
13	E000	08	0	1	1	5008	50	03	-	-	0A	E000	IE: JSR (PC)3, нижи бајт PC на стек
											0A	DFFF	IE: JSR (PC)3, SP=SP-1
14	DFFF	50	0	1	1	5008	50	03	-	-	0A	DFFF	IE: JSR (PC)3, виши бајт PC на стек
						500B					0A	DFFE	IE: JSR (PC)3, SP=SP-1, PC=PC+3=5008+3=500Bh
15	500B	02	1	0	1	500C	02	-	-	-	0A	DFFE	IF: POP , први бајт инс.
											0A	DFFF	IE: POP , SP=SP+1,
16	DFFF	50	1	0	1	500C	02	-	-	-	50	DFFF	IE: POP , бајт са стека у ACC
17	500C	85	1	0	1	500D	85	-	-	-	50	DFFF	IF: STORE 0006h, први бајт инс.
18	500D	00	1	0	1	500E	85	00	-	-	50	DFFF	IF: STORE 0006h, други бајт инс.
19	500E	06	1	0	1	500F	85	00	06	-	50	DFFF	IF: STORE 0006h, трећи бајт инс.
										0006	50	DFFF	ID: STORE 0006h, формирање адресе опрнд. TMP _{15.8} =IR _{15.8} , TMP _{7.0} =IR _{7.0}
20	0006	50	0	1	1	500F	85	00	06	-	50	DFFF	IE: STORE 0006h, упис опрнд. у мем. лок., MEM[0006]=ACC _{7.0}
21	500F	01	1	0	1	5010	01	-	-	-	50	DFFF	IF: PUSH , први бајт инс.
22	DFFF	50	0	1	1	5010	01	-	-	-	50	DFFF	IE: PUSH , бајт из ACC на стек
											50	DFFE	IE: PUSH , SP=SP-1
23	5010	03	1	0	1	5011	03	-	-	-	50	DFFE	IF: RTS , први бајт инс.
											50	DFFF	IE: RTS , SP=SP+1,
24	DFFF	50	1	0	1	50XX	03	-	-	-	50	DFFF	IE: RTS , бајт са стека у виши бајт PC
											50	E000	IE: RTS , SP=SP+1,
25	E000	08	1	0	1	5008	03	-	-	-	50	E000	IE: RTS , бајт са стека у нижи бајт PC
26	5008	00	1	0	1	0009	00	-	-	-	50	E000	IF: HALT , први бајт инс.
													IE: HALT , START=0

Слика 11 Секвенца садржаја на магистрали

Извршавање програма почиње фазом *читање инструкције* (IF) прве инструкције **LOAD** (9h) на коју указује тренутна вредност 5000h садржаја програмског бројача PC_{15.0}. Прва инструкција је дужине три бајта који имају вредности 82h, 00h и 09h, а

налазе се у локацијама меморије на адресама 5000h, 5001h и 5002h, респективно. С обзиром да је ширина меморијске речи 8 битова, процесор из меморије чита бајт по бајт. Приликом читања прве инструкција процесор на магистралаи генерише три циклуса читања из меморије. У првом циклусу читања процесор на линије адресне магистрале ABUS_{15:0} пропушта вредност 5000h програмског бројача PC_{15..0} како би са те адресе прочитао први бајт инструкције 82h. Пошто је у питању циклус читања из меморије управљачки сигнали магистрале RD и M/\overline{IO} имају вредност 1. Прочитани садржај 82h се појављује на линијама магистрале података DBUS_{7:0} и уписује у разреде IR_{23:16} прихватног регистра инструкције процесора, а садржај програмског бројача PC_{15..0} инкрементира на 5001h. На сличан начин се у другом и трећем циклусу читања са адреса 5001h и 5002h читају други и трећи бајтови бајт инструкције 00h и 09h, респективно. Прочитани садржаји се појављује на линијама магистрале података DBUS_{7:0} и уписује у разреде IR_{15:8} и IR_{7:0} прихватног регистра инструкције, а садржај програмског бројача PC_{15..0} инкрементира на 5002h и 5003h, респективно.

У фази *декодовања инструкције* (ID) постоје три циклуса на магистралаи, јер због коришћења меморијског индиректног адресирања процесор најпре у два циклуса на магистралаи из меморије чита два бајта 16-битне адресе операнда, а затим у трећем циклусу на магистралаи чита и сам 8-битни операнд. Адреса операнда је 16-битна и заузима две суседне меморијске локације, при чему се нижи бајт адресе налази на вишој а виши бајт на нижој адреси. Адреса меморијске локације на којој се налази виши бајт адресе операнда дата је другим и трећим бајтом инструкције и налази се у разредима IR_{15:8} (виши бајт) и IR_{7:0} (нижи бајт) прихватног регистра инструкције. У прва два циклуса на магистралаи процесор најпре из меморије са адресе 0009h чита садржај 00h, који представља виши бајт адресе операнда, и уписује у разреде TMP_{15:8}, а затим из меморије са адресе 000Ah чита садржај 07h, који представља нижи бајт адресе операнда, и уписује у разреде TMP_{7:0}. У трећем циклусу на магистралаи садржај 0007h прихватног регистра адресе операнда TMP_{15:0} се користи као адреса меморијске локације са које се чита операнд чија је вредност 03h и уписује у разреде TMP_{7:0} прихватног регистра операнда. Пошто су у питању циклуси читања из меморије управљачки сигнали магистрале RD и M/\overline{IO} имају вредност 1. Са XX је означено да је садржај разреда TMP_{15:8} прихватног регистра операнда недефинисан.

У фази *извршавање операције* (IE) садржај разреда TMP_{7:0} прихватног регистра операнда се уписује у регистар ACC_{7:0}.

По завршетку фазе *извршавање операције* (IE) прве инструкције **LOAD** (9h) се прелази на фазу *читање инструкције* (IF) друге инструкције **ADD** 0Bh. Друга инструкција је дужине три бајта који имају вредности 89h, 00h, 0Bh, а налазе се у меморији на адресама 5003h, 5004h и 5005h, респективно. Приликом читања друге инструкције процесор на магистралаи генерише три циклуса читања из меморије. Циклуси читања бајтова друге инструкције се реализују на идентичан начин као и циклуси читања бајтова прве инструкције. Прочитане вредности се уписују у разреде IR_{23:16}, IR_{15:8} и IR_{7:0} прихватног регистра инструкције, а садржај програмског бројача PC_{15..0} инкрементира на 5004h, 5005h и 5006h, респективно.

У фази *декодовања инструкције* (ID) постоји један циклус на магистралаи, јер процесор због коришћења меморијског директног адресирања у једном циклусу на магистралаи из меморије чита 8 битни операнд. Најпре се адреса меморијске локације, чија је вредност 000Bh и која је дата другим и трећим бајтом инструкције из разреда IR_{15:8} (виши бајт) и IR_{7:0} (нижи бајт), пребацује у разреде TMP_{15:8} и TMP_{7:0} прихватног регистра адресе операнда, респективно. Затим се у циклусу на магистралаи садржај 000Bh прихватног регистра адресе операнда TMP_{15:0} користи као адреса регистра из улазно/излазног адресног простора са које се чита операнд чија је вредност 07h и

уписује у разреде $TM\overline{P}_{7..0}$ прихватног регистра операнда. Пошто је у питању циклус читања из регистра улазно/излазног адресног простора управљачки сигнали магистрале RD и M/\overline{IO} имају вредности 1 и 0, респективно. Са XX је означено да је садржај разреда $TM\overline{P}_{15..8}$ прихватног регистра операнда недефинисан.

У фази *извршавање операције* (IE) садржај разреда $TM\overline{P}_{7..0}$ прихватног регистра операнда се сабира са садржајем регистра $ACC_{7..0}$ и резултат уписује у регистар $ACC_{7..0}$.

По завршетку фазе *извршавање операције* (IE) друге инструкције **ADD** $0Bh$ се прелази на фазу *читање инструкције* (IF) треће инструкције **JSR** (PC) $3h$. Трећа инструкција је дужине два бајта који имају вредности $50h$ и $03h$, а налазе се у меморији на адресама $5006h$ и $5007h$. Приликом читања треће инструкције процесор на магистрали генерише два циклуса читања из меморије. Циклуси читања бајтова треће инструкције се реализују на идентичан начин као и циклуси читања бајтова прве и друге инструкције. Прочитане вредности се уписују у разреде $IR_{23:16}$ и $IR_{15:8}$ прихватног регистра инструкције, а садржај програмског бројача $PC_{15..0}$ инкрементира на $5007h$ и $5008h$, респективно.

Ова инструкција нема фазу *декодовање инструкције* (ID).

У фази *извршавање операције* (IE) најпре се на стеку чува програмски бројач $PC_{15..0}$, а затим се адреса потпрограма уписује у програмски бројач $PC_{15..0}$.

Садржај регистра $PC_{15..0}$ смешта се на стек у два циклуса уписа на магистрали. Пошто SP указује на прву слободну локацију и стек расте ка нижим адресама, најпре се на адресу $E000h$ уписује нижи бајт PC , $PC_{7..0}=08h$ па се SP декрементира, а затим се на адресу $DFFFh$ уписује виши бајт PC , $PC_{15..8}=50h$ па се SP декрементира на вредност $DFFEh$. Пошто је у питању циклус уписа у меморију управљачки сигнали магистрале WR и M/\overline{IO} имају вредност 1. Адреса скока се рачуна као збир $PC_{15..0}$, чија је вредност $5008h$, и 8-битног помераја садржаног у другом бајту инструкцијског регистра $IR_{15:8}$, чија је вредност 3. Добијени збир $500Bh$ се смешта у $PC_{15..8}$.

По завршетку фазе *извршавање операције* (IE) треће инструкције **JSR** (PC) 3 се прелази на фазу *читање инструкције* (IF) четврте инструкције **POP**. Четврта инструкција је дужине један бајт који има вредности $02h$, а налази се у меморији на адреси $500Bh$. Приликом читања четврте инструкције процесор на магистрали генерише један циклус читања из меморије. Циклус читања једног бајта четврте инструкције се реализују на идентичан начин као и циклуси читања бајтова прве инструкције. Прочитана вредност се уписује у разреде $IR_{23:16}$ прихватног регистра инструкције, а садржај програмског бројача $PC_{15..0}$ инкрементира на $500Ch$.

Ова инструкција нема фазу *декодовање инструкције* (ID).

У фази *извршавање операције* (IE) се реализују један циклуса читања на магистрали. Најпре се садржај SP , који указује на прву слободну локацију на стеку инкрементира, па се користи да се у циклусу читања на магистрали, са стека, са адресе $DFFFh$, чита вредност $50h$ и уписује у акумулатор $ACC_{7..0}$. Пошто је у питању циклус читања из меморије управљачки сигнали магистрале RD и M/\overline{IO} имају вредност 1.

По завршетку фазе *извршавање операције* (IE) четврте инструкције **POP** се прелази на фазу *читање инструкције* (IF) пете инструкције **STORE** $0006h$. Пета инструкција је дужине три бајта који имају вредности $85h$, $00h$ и $06h$, а налазе се у меморији на адресама $500Ch$, $500Dh$ и $500Eh$, респективно. Приликом читања пете инструкције процесор на магистрали генерише три циклуса читања из меморије. Циклуси читања бајтова пете инструкције се реализују на идентичан начин као и циклуси читања бајтова прве инструкције. Прочитане вредности се уписују у разреде $IR_{23:16}$, $IR_{15:8}$ и $IR_{7:0}$ прихватног регистра инструкције, а садржај програмског бројача $PC_{15..0}$ инкрементира на $500Dh$, $500Eh$ и $500Fh$, респективно.

У фази *декодовања инструкције* (ID) нема циклуса на магистралаи, јер се због коришћења меморијског директног адресирања у овој фази само формира адреса меморијске локације у коју треба да се упише 8-битни операнд дат садржајем акумулатора ACC_{7.0}. Фаза *декодовања инструкције* (ID) се састоји у пребацивању адресе меморијске локације чија је вредност 0006h и која је дата другим и трећим бајтом инструкције из разреда IR_{15:8} (виши бајт) и IR_{7:0} (нижи бајт) прихватног регистра инструкције у разреде TMP_{15:8} и TMP_{7:0} прихватног регистра адресе операнда.

У фази *извршавање операције* (IE) процесор на магистралаи генерише циклус уписа у меморију. На линије адресне магистрале ABUS_{15.0} се пропушта вредност 0006h прихватног регистра адресе операнда TMP_{15.0} а на линије магистрале података DBUS_{7.0} вредност 50h акумулатора ACC_{7.0}. Пошто је у питању циклус уписа у меморију управљачки сигнали магистрале WR и $\overline{M}/\overline{IO}$ имају вредност 1.

По завршетку фазе *извршавање операције* (IE) пете инструкције **STORE** 0006h се прелази на фазу *читање инструкције* (IF) шесте инструкције **PUSH**. Шеста инструкција је дужине један бајт који има вредности 01h, а налази се у меморији на адреси 500Fh. Приликом читања шесте инструкције процесор на магистралаи генерише један циклус читања из меморије. Циклус читања једног бајта четврте инструкције се реализују на идентичан начин као и циклуси читања бајтова прве инструкције. Прочитана вредност се уписује у разреде IR_{23:16} прихватног регистра инструкције, а садржај програмског бројача PC_{15.0} инкрементира на 5010h.

Ова инструкција нема фазу *декодовање инструкције* (ID).

У фази *извршавање операције* (IE) се реализују један циклуса уписа на магистралаи. Пошто SP указује на прву слободну локацију и стек расте ка нижим адресама, најпре се на адреси DFFFh уписује садржај акумулатора ACC_{7.0}=50h, па се затим SP декрементира, Пошто је у питању циклус уписа у меморију управљачки сигнали магистрале WR и $\overline{M}/\overline{IO}$ имају вредност 1.

По завршетку фазе *извршавање операције* (IE) шесте инструкције **PUSH** се прелази на фазу *читање инструкције* (IF) седме инструкције **RTS**. Седма инструкција је дужине један бајт који има вредности 03h, а налази се у меморији на адреси 5010h. Приликом читања седме инструкције процесор на магистралаи генерише један циклус читања из меморије. Циклус читања једног бајта седме инструкције се реализују на идентичан начин као и циклуси читања бајтова прве инструкције. Прочитана вредност се уписује у разреде IR_{23:16} прихватног регистра инструкције, а садржај програмског бројача PC_{15.0} инкрементира на 5011h.

Ова инструкција нема фазу *декодовање инструкције* (ID).

У фази *извршавање операције* (IE) се реализују два циклуса читања на магистралаи. Најпре се садржај SP, који указује на прву слободну локацију на стеку, инкрементира, па се затим, у првом циклусу читања на магистралаи, са врха стека, са адресе DFFFh, чита вредност 50h, која представља виши бајт адресе повратка из потпрограма, и уписује у виши бајт регистра PC, PC_{15.8}. Потом се садржај SP, који указује на прву слободну локацију на стеку, поново инкрементира, па се затим, у другом циклусу читања на магистралаи, са врха стека, са адресе E000h, чита вредност 08h, која представља нижи бајт адресе повратка из потпрограма, и уписује у нижи бајт регистра PC, PC_{7.0}. Пошто су у питању циклуси читања из меморије управљачки сигнали магистрале RD и $\overline{M}/\overline{IO}$ имају вредност 1.

По завршетку фазе *извршавање операције* (IE) седме инструкције **RTS** се прелази на фазу *читање инструкције* (IF) осме инструкције **HALT**. Осма инструкција је дужине један бајт који има вредности 00h, а налази се у меморији на адреси 5008h. Приликом читања осме инструкције процесор на магистралаи генерише један циклус читања из меморије. Циклус читања једног бајта осме инструкције се реализују на идентичан

начин као и циклуси читања бајтова прве инструкције. Прочитана вредност се уписује у разреду $IR_{23..16}$ прихватног регистра инструкције, а садржај програмског бројача $PC_{15..0}$ инкрементира на $5009h$.

Инструкција **HALT** је безадресна инструкција, па се после фазе *читање инструкције* (IF) одмах прелази на фазу *извршавање операције* (IE) у оквиру које се зауставља процесор.

1.1.4 ЗАДАТАК

Посматра се процесор описан у задатку 1.1.2.

Садржај дела оперативне меморије је приказан на слици 12, при чему су све вредности дате у хексадецималном облику.

Адреса	0000	0001	0002	0003	0004	0005	0006	0007	0008	0009	000A	...
Садржај	37	A1	B5	08	02	36	00	33	56	07	00	...
...
Адреса	3000	3001	3002	3003	3004	3005	3006	3007	3008	3009	300A	
Садржај	21	00	00	39	09	00	10	0D	30	00	00	
Адреса	300B	300C	300D	300E	300F	3010	3011	3012	3013	3014	3015	...
Садржај	02	85	2D	09	00	01	50	03	00	17	00	...

Слика 12 Садржај дела меморије

а) Навести секвенцу садржаја на адресној магистралу, магистралу података и управљачкој магистралу за сваки циклус на магистралу при извршавању програма приказаног на слици 13. Претпоставити да се током извршавања овог програма не генерише прекид. Стога се приликом извршавања инструкција пролази само кроз фазе *читање инструкције (IF)*, *декодовање инструкције (IF)* и *извршавање операције (IE)*, а не и кроз фазу *опслуживање прекида (IS)*.

Претпоставити да је пре почетка извршавања главног програма $SP = D000h$.

адреса	инструкција	коментар
; главни програм		
3000h	LOAD R1, 0h	; меморијско директно адресирање, R1=MEM[0h]
3003h	OUT R1, 9h	; у/и директно адресирање, IO[9h]=R1
3006h	JSR 300Dh	; скок у потпрограм апсолутан скок
3009h	HALT	; заустављање процесора
300Ah	...	
; потпрограм		
300Dh	STORE R1, (9h)	; меморијско индиректно адресирање, MEM[MEM[9]]= R1
3010h	RTS	; повратак из потпрограма
3011h	...	

Слика 13 Програм

Решење:

а) Секвенца садржаја на адресној магистралу, магистралу података и управљачкој магистралу за сваки циклус на магистралу је приказана у табели са слике 14. У табели се користе исте ознаке као и у табели са слике 8 из задатка 1.1.2, при чему нема колоне за $R2_{7.0}$, а јавља се и колона $SP_{15.0}$ - указивач на врх стека након ажурирања његовог садржаја.

Pb	ABUS _{15..0}	DBUS _{7..0}	RD	WR	M/I/O	PC _{15..0}	IR _{23..16}	IR _{15..8}	IR _{7..0}	TMP _{15..0}	SP _{15..0}	R1 _{7..0}	Коментар
	-	-	-	-	-	3000	-	-	-		D000	-	
1	3000	21	1	0	1	3001	21	-	-	-	D000	-	IF: LOAD R1, 0h, први бајт инс.
2	3001	00	1	0	1	3002	21	00	-	-	D000	-	IF: LOAD R1, 0h, други бајт инс.
3	3002	00	1	0	1	3003	21	00	00	-	D000	-	IF: LOAD R1, 0h, трећи бајт инс.
										0000	D000		ID: LOAD R1, 0h, формирање адресе опрнд., TMP _{15..8} =IR _{7..0} , TMP _{7..0} =IR _{15..8}
4	0000	37	1	0	1	3003	21	00	00	XX37	D000	-	ID: LOAD R1, 0h, читање опрнд. из мем. лок., TMP _{7..0} =MEM[0000]
												37	IE: LOAD R1, 0h, R1 _{7..0} =TMP _{7..0}
5	3003	39	1	0	1	3004	39	-	-	-	D000	37	IF: OUT R1, 9h, први бајт инс.
6	3004	09	1	0	1	3005	39	09	-	-	D000	37	IF: OUT R1, 9h, други бајт инс.
7	3005	00	1	0	1	3006	39	09	00	-	D000	37	IF: OUT R1, 9h, трећи бајт инс.
										0009	D000	37	IF: OUT R1, 9h, формирање адресе опрнд. TMP _{15..8} =IR _{7..0} , TMP _{7..0} =IR _{15..8}
8	0009	37	0	1	0	3006	39	09	00	-	D000	37	IE: OUT R1, 9h, упис опрнд. у уи рег., UI[0009]=R1 _{7..0}
9	3006	10	1	0	1	3007	10	-	-	-	D000	37	IF: JSR 300Dh, први бајт инс.
10	3007	0D	1	0	1	3008	10	0D	-	-	D000	37	IF: JSR 300Dh, други бајт инс.
11	3008	30	1	0	1	3009	10	0D	30	-	D000	37	IF: JSR 300Dh, трећи бајт инс.
											D001	37	IE: JSR 300Dh, SP=SP+1,
12	D001	09	0	1	1	3009	10	0D	30	-	D001	37	IE: JSR 300Dh, нижи бајт PC на стек
											D002	37	IE: JSR 300Dh, SP=SP+1,
13	D002	30	0	1	1	3009	10	0D	30	-	D002	37	IE: JSR 300Dh, виши бајт PC на стек
						300D				-	D002	37	IE: JSR 300Dh, PC=300Dh
14	300D	2D	1	0	1	300E	2D	-	-	-	D002	37	IF: STORE R1,(9h), први бајт инс.
15	300E	09	1	0	1	300F	2D	09	-	-	D002	37	IF: STORE R1,(9h), други бајт инс.
16	300F	00	1	0	1	3010	2D	09	00	-	D002	37	IF: STORE R1,(9h), трећи бајт инс.
17	0009	07	1	0	1	3010	2D	09	00	XX07	D002	37	ID: STORE R1,(9h), формирање адресе опрнд. - читање нижег бајта, TMP _{7..0} =MEM[0009]
18	000A	00	1	0	1	3010	2D	09	00	0007	D002	37	ID: STORE R1,(9h), формирање адресе опрнд. - читање вишег бајта, TMP _{15..8} =MEM[000A]
19	0007	37	0	1	1	3010	2D	09	00	-	D002	37	IE: STORE R1,(9h), упис опрнд. у мем. лок., MEM[0007]=R1 _{7..0}
20	3010	01	1	0	1	3011	01	-	-	-	D002	37	IF: RTS , први бајт инс.
21	D002	30	1	0	1	30XX	01	-	-	-	D002	37	IE: RTS , бајт са стека у виши бајт PC
											D001	37	IE: RTS , SP=SP-1
22	D001	09	1	0	1	3009	01	-	-	-	D001	37	IE: RTS , бајт са стека у нижи бајт PC
											D000		IE: RTS , SP=SP-1
23	3009	00	1	0	1	300A	00	-	-	-	D000	37	IF: HALT , први бајт
											D000	37	IE: HALT , START=0

Слика 14 Секвенца садржаја на магистрали

Извршавање програма почиње фазом *читање инструкције* (IF) прве инструкције **LOAD** R1, 0h на коју указује тренутна вредност 3000h садржаја програмског бројача PC_{15..0}. Прва инструкција је дужине три бајта који имају вредности 21h, 00h и 00h, а налазе се у локацијама меморије на адресама 3000h, 3001h и 3002h, респективно. С обзиром да је ширина меморијске речи 8 битова, процесор из меморије чита бајт по бајт. Приликом читања прве инструкција процесор на магистрали генерише три циклуса читања из меморије. У првом циклусу читања процесор на линије адресне

магистрале $ABUS_{15.0}$ пропушта вредност $3000h$ програмског бројача $PC_{15.0}$ како би са те адресе прочитао први бајт инструкције $21h$. Пошто је у питању циклус читања из меморије управљачки сигнали магистрале RD и M/\overline{IO} имају вредност 1. Прочитани садржај $21h$ се појављује на линијама магистрале података $DBUS_{7.0}$ и уписује у разреде $IR_{23..16}$ прихватног регистра инструкције процесора, а садржај програмског бројача $PC_{15.0}$ инкрементира на $3001h$. На сличан начин се у другом и трећем циклусу читања са адреса $3001h$ и $3002h$ читају други и трећи бајт инструкције $00h$ и $00h$, респективно. Прочитани садржаји се појављује на линијама магистрале података $DBUS_{7.0}$ и уписује у разреде $IR_{15..8}$ и $IR_{7..0}$ прихватног регистра инструкције, а садржај програмског бројача $PC_{15.0}$ инкрементира на $3002h$ и $3003h$, респективно.

У фази *декодовања инструкције* (ID) постоји један циклус на магистрали, јер процесор због коришћења меморијског директног адресирања у једном циклусу на магистрали из меморије чита 8 битни операнд. Најпре се адреса меморијске локације, чија је вредност $0000h$ и која је дата другим и трећим бајтом инструкције из разреда $IR_{15..8}$ (нижи бајт) и $IR_{7..0}$ (виши бајт), пребацује у разреде $TMP_{7.0}$ и $TMP_{15..8}$ прихватног регистра адресе операнда, респективно. Затим се у циклусу на магистрали садржај $0000h$ прихватног регистра адресе операнда $TMP_{15..0}$ користи као адреса меморијске локације са које се чита операнд чија је вредност $37h$ и уписује у разреде $TMP_{7..0}$ прихватног регистра операнда. Пошто је у питању циклус читања из меморије управљачки сигнали магистрале RD и M/\overline{IO} имају вредност 1. Са XX је означено да је садржај разреда $TMP_{15..8}$ прихватног регистра операнда недефинисан.

У фази *извршавање операције* (IE) садржај разреда $TMP_{7..0}$ прихватног регистра операнда се уписује у регистар $R1_{7..0}$.

По завршетку фазе *извршавање операције* (IE) прве инструкције **LOAD** $R1$, $0h$ се прелази на фазу *читање инструкције* (IF) друге инструкције **OUT** $R1$, $9h$. Друга инструкција је дужине три бајта који имају вредности $39h$, $09h$ и $00h$, а налазе се у меморији на адресама $3003h$, $3004h$ и $3005h$, респективно. Приликом читања друге инструкције процесор на магистрали генерише три циклуса читања из меморије. Циклуси читања бајтова друге инструкције се реализују на идентичан начин као и циклуси читања бајтова прве инструкције. Прочитане вредности се уписују у разреде $IR_{23..16}$, $IR_{15..8}$ и $IR_{7..0}$ прихватног регистра инструкције, а садржај програмског бројача $PC_{15.0}$ инкрементира на $3004h$, $3005h$ и $3006h$, респективно.

У фази *декодовање инструкције* (ID) нема циклуса на магистрали, јер се због коришћења у/и директног адресирања у овој фази само формира адреса регистра улазно/излазног адресног простора у који треба да се упише 8 битни операнд дат садржајем регистра опште намене $R1_{7..0}$. Фаза *декодовање инструкције* (ID) се састоји у пребацивању адресе регистра улазно/излазног адресног простора чија је вредност $0009h$ и која је дата трећим и другим бајтом инструкције из разреда $IR_{7..0}$ (виши бајт) и $IR_{15..8}$ (нижи бајт) прихватног регистра инструкције у разреде $TMP_{15..8}$ и $TMP_{7..0}$ прихватног регистра адресе операнда.

У фази *извршавање операције* (IE) процесор на магистрали генерише циклус уписа у регистар улазно/излазног адресног простора. На линије адресне магистрале $ABUS_{15.0}$ се пропушта вредност $0009h$ прихватног регистра адресе операнда $TMP_{15.0}$ а на линије магистрале података $DBUS_{7.0}$ вредност $37h$ регистра опште намене $R1_{7..0}$. Пошто је у питању циклус уписа у регистар улазно/излазног адресног простора управљачки сигнали магистрале WR и M/\overline{IO} имају вредности 1 и 0, респективно.

По завршетку фазе *извршавање операције* (IE) друге инструкције **OUT** $R1$, $9h$ се прелази на фазу *читање инструкције* (IF) треће инструкције **JSR** $300Dh$. Трећа инструкција је дужине три бајта који имају вредности $10h$, $0Dh$, $30h$, а налазе се у меморији на адресама $3006h$, $3007h$, $3008h$. Приликом читања треће инструкције

процесор на магистралаи генерише три циклуса читања из меморије. Циклуси читања бајтова треће инструкције се реализују на идентичан начин као и циклуси читања бајтова прве и друге инструкције. Прочитане вредности се уписују у разреде IR_{23..16}, IR_{15..8} и IR_{7..0} прихватног регистра инструкције, а садржај програмског бројача PC_{15..0} инкрементира на 3007h, 3008 и 3009h респективно.

Ова инструкција нема фазу *декодовање инструкције* (ID).

У фази *извршавање операције* (IE) садржај регистра PC_{15..0} смешта се на стек у два циклуса уписа на магистралаи. Пошто SP указује на последњу заузету локацију и стек расте ка вишим адресама, најпре се SP инкрементира па се на адресу D001h уписује нижи бајт PC, PC_{7..0}=09h, а затим се SP поново инкрементира па се на адресу D002h уписује виши бајт PC, PC_{15..8}=30h. Пошто је у питању циклус уписа у меморију управљачки сигнали магистрале WR и M/IO имају вредност 1. Адреса апсолутног скока, чија вредност 300Dh је садржана у разредима IR_{7..0} (виши бајт) и IR_{15..8} (нижи бајт) прихватног регистра инструкције, се смешта у регистар PC_{15..0}.

По завршетку фазе *извршавање операције* (IE) треће инструкције JSR 300Dh се прелази на фазу *читање инструкције* (IF) четврте инструкције STORE R1, (9h). Четврта инструкција је дужине три бајта који имају вредности 2Dh, 09h и 00h, а налазе се у меморији на адресама 300Dh, 300Eh и 300Fh, респективно. Приликом читања четврте инструкције процесор на магистралаи генерише три циклуса читања из меморије. Циклуси читања бајтова четврте инструкције се реализују на идентичан начин као и циклуси читања бајтова прве инструкције. Прочитане вредности се уписују у разреде IR_{23..16}, IR_{15..8} и IR_{7..0} прихватног регистра инструкције, а садржај програмског бројача PC_{15..0} инкрементира на 300Eh, 300Fh и 3010h, респективно.

У фази *декодовања инструкције* (ID) постоје два циклуса на магистралаи, јер због коришћења меморијског индиректног адресирања процесор у два циклуса на магистралаи из меморије чита два бајта 16 битне адресе операнда. На нижој меморијској адреси 0009h, садржаној у разредима IR_{7..0} (виши бајт) и IR_{15..8} (нижи бајт) прихватног регистра инструкције, се налази нижи бајт адресе операнда 07h, а на првој следећој вишој меморијској адреси 000Ah се налази виши бајт адресе операнда 00h на коју је потребно уписати садржај регистра R1_{7..0} у фази *извршавање операције* (IE). Најпре се у првом циклусу на магистралаи из меморије са адресе 0009h чита нижи бајт адресе операнда 07h и смешта у разреде TMP_{7..0} прихватног регистра адресе операнда, а затим се у другом циклусу на магистралаи из меморије са адресе 000Ah чита виши бајт адресе операнда 00h и смешта у разреде TMP_{15..8} прихватног регистра адресе операнда. У циклусу читања управљачки сигнали RD и M/IO имају вредност 1.

У фази *извршавање операције* (IE) процесор на магистралаи генерише циклус уписа у меморију. На линије адресне магистрале ABUS_{15..0} се пропушта вредност 0007h прихватног регистра адресе операнда TMP_{15..0} а на линије магистрале података DBUS_{7..0} вредност 37h регистра R1_{7..0}. Пошто је у питању циклус уписа у меморију управљачки сигнали магистрале WR и M/IO имају вредност 1.

По завршетку фазе *извршавање операције* (IE) четврте инструкције STORE R1, (9h) се прелази на фазу *читање инструкције* (IF) пете инструкције RTS. Пета инструкција је дужине један бајт који има вредности 01h, а налази се у меморији на адреси 3010h. Приликом читања пете инструкције процесор на магистралаи генерише један циклус читања из меморије. Циклус читања једног бајта пете инструкције се реализују на идентичан начин као и циклуси читања бајтова прве инструкције. Прочитана вредност се уписује у разреде IR_{23..16} прихватног регистра инструкције, а садржај програмског бројача PC_{15..0} инкрементира на 3011h.

Ова инструкција нема фазу *декодовање инструкције* (ID).

У фази *извршавање операције* (IE) се реализују два циклуса читања на магистралаи. Најпре се садржај SP, који указује на задњу заузету локацију на стеку, користи да се у првом циклусу читања на магистралаи, са стека, са адресе D002h, чита вредност 30h, која представља виши бајт адресе повратка из потпрограма, и уписује у виши бајт регистра PC, PC_{15..8}, па се затим садржај SP декрементира. Потом се, садржај SP, који указује на задњу заузету локацију на стеку, користи да се у другом циклусу читања на магистралаи, са стека, са адресе D001h, чита вредност 09h, која представља нижи бајт адресе повратка из потпрограма, и уписује у нижи бајт регистра PC, PC_{7..0}, па се затим садржај SP поново декрементира. Пошто је у питању циклус читања из меморије управљачки сигнали магистрале RD и M/ \overline{IO} имају вредност 1.

По завршетку фазе *извршавање операције* (IE) пете инструкције RTS се прелази на фазу *читање инструкције* (IF) шесте инструкције HALT. Шеста инструкција је дужине један бајт који има вредности 00h, а налази се у меморији на адреси 3009h. Приликом читања шесте инструкције процесор на магистралаи генерише један циклус читања из меморије. Циклус читања једног бајта шесте инструкције се реализују на идентичан начин као и циклуси читања бајтова прве инструкције. Прочитана вредност се уписује у разреде IR_{23..16} прихватног регистра инструкције, а садржај програмског бројача PC_{15..0} инкрементира на 300Ah.

Инструкција HALT је безадресна инструкција, па се после фазе *читање инструкције* (IF) одмах прелази на фазу *извршавање операције* (IE) у оквиру које се зауставља процесор.

1.1.5 ЗАДАТАК

Посматра се процесор описан у задатку 1.1.2.

Садржај дела оперативне меморије је приказан на слици 15, при чему су све вредности дате у хексадецималном облику.

Адреса	0000	0001	0002	0003	0004	0005	0006	0007	0008	0009	000A	...
Садржај	37	A1	B5	08	0E	30	00	33	56	07	00	...
...
Адреса	1000	1001	1002	1003	1004	1005	1006	1007	1008	1009	100A	...
Садржај	00	30	06	00	07	00	30	20	56	07	00	...
...
Адреса	3000	3001	3002	3003	3004	3005	3006	3007	3008	3009	300A	...
Садржај	25	02	10	26	04	10	1F	02	00	25	03	...
Адреса	300B	300C	300D	300E	300F	3010	3011	3012	3013	3014	3015	...
Садржај	02	85	2D	46	02	01	50	03	00	17	00	...

Слика 15 Садржај дела меморије

а) Навести секвенцу садржаја на адресној магистралу, магистралу података и управљачкој магистралу за сваки циклус на магистралу при извршавању програма приказаног на слици 16.

Претпоставити да се током извршавања овог програма генерише само прекид изазван инструкцијом софтверског прекида INS. Инструкција INS је тако реализована да се у фази *извршавање операције* (IE) на најпре на стек ставља садржај програмског бројача PC, а затим на основу броја улаза у IV табелу, датог адресним делом 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. Стога се приликом извршавања инструкција пролази само кроз фазе *читање инструкције* (IF), *декодовање инструкције* (IF) и *извршавање операције* (IE), а не и кроз фазу *опслуживање прекида* (IS).

Претпоставити да је пре почетка извршавања главног програма SP=4000h, IVTP=0h и PC=3000h.

адреса	инструкција	коментар
	; главни програм	
3000h	LOAD R1, (1002h)	; memorijsko indirektno adresiranje R1=MEM[MEM[1002h]]
3003h	LOAD R2, (1004h)	; memorijsko indirektno adresiranje R2=MEM[MEM[1004h]]
3006h	INS #2	; softverski prekid
3008h	HALT	; zaustavljanje procesora
3009h	...	
...		
	; прекидна рутина	
300Eh	ADD R2, R1	; R2=R2+R1
300Fh	RTI	; повратак из прекидне рутине
3010h	...	

Слика 16 Програм

Решење:

а) Секвенца садржаја на адресној магистралу, магистралу података и управљачкој магистралу за сваки циклус на магистралу је приказана у табели са слике 17. У табели се користе исте ознаке као и у табели са слике 8 из задатка 1.1.2, при чему се јавља и колона SP_{15..0} - указивач на врх стека након ажурирања његовог садржаја.

P6	ABUS _{15..0}	DBUS _{7..0}	RD	WR	M/I/O	PC _{15..0}	IR _{23..16}	IR _{15..8}	IR _{7..0}	TMP _{15..0}	R1 _{7..0}	R2 _{7..0}	SP _{15..0}	Коментар		
	-	-	-	-	-	3000	-	-	-	-	-	-	4000			
1	3000	25	1	0	1	3001	25	-	-	-	-	-	4000	IF: LOAD R1 ,(1002h), први бајт инс.		
2	3001	02	1	0	1	3002	25	02	-	-	-	-	4000	IF: LOAD R1 ,(1002h), други бајт инс.		
3	3002	10	1	0	1	3003	25	02	10	-	-	-	4000	IF: LOAD R1 , (1002h), трећи бајт инс.		
4	1002	06	1	0	1	3003	25	02	10	XX06	-	-	4000	ID: LOAD R1 , (1002h), формирање адресе опрд - читање нижег бајта, TMP _{7..0} =MEM[1002]		
5	1003	00	1	0	1	3003	25	02	10	0006	-	-	4000	ID: LOAD R1 , (1002h), формирање адресе опрд - читање вишег бајта, TMP _{15..8} =MEM[1003]		
6	0006	00	1	0	1	3003	25	02	10	XX00	-	-	4000	ID: LOAD R1 ,(1002h),читање опрд. из мем. лок., TMP _{7..0} =MEM[0006]		
											00			IE: LOAD R1 , (1002h), R1 _{7..0} =TMP _{7..0}		
7	3003	26	1	0	1	3004	26	-	-	-	00	-	4000	IF: LOAD R2 , (1004h), први бајт инс.		
8	3004	04	1	0	1	3005	26	04	-	-	00	-	4000	IF: LOAD R2 , (1004h), други бајт инс.		
9	3005	10	1	0	1	3006	26	04	10	-	00	-	4000	IF: LOAD R2 , (1004h), трећи бајт инс.		
10	1004	07	1	0	1	3006	26	04	10	XX07	00	-	4000	ID: LOAD R2 , (1004h), формирање адресе опрд - читање нижег бајта, TMP _{7..0} =MEM[1004]		
11	1005	00	1	0	1	3006	26	04	10	0007	00	-	4000	ID: LOAD R2 , (1004h), формирање адресе опрд - читање вишег бајта, TMP _{15..8} =MEM[1005]		
12	0007	33	1	0	1	3006	26	04	10	XX33	00	-	4000	ID: LOAD R2 ,(1004h),читање опрд. из мем. лок., TMP _{7..0} =MEM[0007]		
												33		IE: LOAD R2 , (1004h), R2 _{7..0} =TMP _{7..0}		
13	3006	1F	1	0	1	3007	1F	-	-	-	00	33	4000	IF: INS #2 , први бајт инс.		
14	3007	02	1	0	1	3008	1F	02	-	-	00	33	4000	IF: INS #2 , други бајт инс.		
													4001	IE: INS #2 , SP=SP+1		
15	4001	08	0	1	1	3008	1F	02	-	-	00	33	4001	IE: INS #2 , нижи бајт PC на стек		
													4002	IE: INS #2 , SP=SP+1		
16	4002	30	0	1	1	3008	1F	02	-	-	00	33	4002	IE: INS #2 , виши бајт PC на стек		
										0004	00		4002	IE: INS #2 , израчунавање адресе улаза 2 у IV табели (0h+2*2=4h)		
17	0004	0E	1	0	1	XX0E	1F	02	-	-	00	33	4002	IE: INS #2 , читање нижег бајта адресе прек. рутине и упис у нижи бајт PC		
18	0005	30	1	0	1	300E	1F	02	-	-	00	33	4002	IE: INS #2 , читање вишег бајта адресе прек. рутине и упис у виши бајт PC		
19	300E	46	1	0	1	300F	46	-	-	-	00	33	4002	IF: ADD R2 , R1, први бајт инс.		
													33	4002	IE: ADD R2 , R1, R2 _{7..0} =R2 _{7..0} +R1 _{7..0}	
20	300F	02	1	0	1	3010	02	-	-	-	00	33	4002	IF: RTI , први бајт инс.		
21	4002	30	1	0	1	30XX	02	-	-	-	00	33	4002	IE: RTI , бајт са стека у виши бајт PC		
													00	33	4001	IE: RTI , SP=SP-1
22	4001	08	1	0	1	3008	C0	-	-	-	00	33	4001	IE: RTI , бајт са стека у нижи бајт PC		
													4000		IE: RTI , SP=SP-1	
23	3008	00	1	0	1	3009	00	-	-	-	00	33	4000	IF: HALT , први бајт		
													4000		IE: HALT , START=0	

Слика 17 Секвенца садржаја на магистрали

Извршавање програма почиње фазом *читање инструкције* (IF) прве инструкције **LOAD R1**, (1002h) на коју указује тренутна вредност 3000h садржаја програмског бројача PC_{15..0}. Прва инструкција је дужине три бајта који имају вредности 25h, 02h и 10h, а налазе се у локацијама меморије на адресама 3000h, 3001h и 3002h, респективно.

С обзиром да је ширина меморијске речи 8 битова, процесор из меморије чита бајт по бајт. Приликом читања прве инструкција процесор на магистралаи генерише три циклуса читања из меморије. У првом циклусу читања процесор на линије адресне магистрале $ABUS_{15..0}$ пропушта вредност 3000h програмског бројача $PC_{15..0}$ како би са те адресе прочитао први бајт инструкције 25h. Пошто је у питању циклус читања из меморије управљачки сигнали магистрале RD и M/\overline{IO} имају вредност 1. Прочитани садржај 25h се појављује на линијама магистрале података $DBUS_{7..0}$ и уписује у разреде $IR_{23..16}$ прихватног регистра инструкције процесора, а садржај програмског бројача $PC_{15..0}$ инкрементира на 3001h. На сличан начин се у другом и трећем циклусу читања са адреса 3001h и 3002h читају други и трећи бајт инструкције 02h и 10h, респективно. Прочитани садржаји се појављују на линијама магистрале података $DBUS_{7..0}$ и уписују у разреде $IR_{15..8}$ и $IR_{7..0}$ прихватног регистра инструкције, а садржај програмског бројача $PC_{15..0}$ инкрементира на 3002h и 3003h, респективно.

У фази *декодовања инструкције (ID)* постоје три циклуса на магистралаи, јер због коришћења меморијског индиректног адресирања процесор најпре у два циклуса на магистралаи из меморије чита два бајта 16-битне адресе операнда, а затим у трећем циклусу на магистралаи чита и сам 8-битни операнд. Адреса операнда је 16-битна и заузима две суседне меморијске локације, при чему се нижи бајт адресе налази на нижој а виши бајт адресе на вишој адреси. Адреса меморијске локације на којој се налази нижи бајт адресе операнда дата је другим и трећим бајтом инструкције и налази се у разредима $IR_{15..8}$ (нижи бајт) и $IR_{7..0}$ (виши бајт) прихватног регистра инструкције. У прва два циклуса на магистралаи процесор најпре из меморије са адресе 1002h чита садржај 06h, који представља нижи бајт адресе операнда, и уписује у разреде $TMP_{7..0}$, а затим из меморије са адресе 1003h чита садржај 00h, који представља виши бајт адресе операнда, и уписује у разреде $TMP_{15..8}$. У трећем циклусу на магистралаи садржај 0006h прихватног регистра адресе операнда $TMP_{15..0}$ се користи као адреса меморијске локације са које се чита операнд чија је вредност 00h и уписује у разреде $TMP_{7..0}$ прихватног регистра операнда. Пошто су у питању циклуси читања из меморије управљачки сигнали магистрале RD и M/\overline{IO} имају вредност 1. Са XX је означено да је садржај разреда $TMP_{15..8}$ прихватног регистра операнда недефинисан.

У фази *извршавање операције (IE)* садржај разреда $TMP_{7..0}$ прихватног регистра операнда се уписује у регистар $R1_{7..0}$.

По завршетку фазе *извршавање операције (IE)* прве инструкције **LOAD R1**, (1002h) се прелази на фазу *читање инструкције (IF)* друге инструкције **LOAD R2**, (1004h). Друга инструкција је дужине три бајта који имају вредности 26h, 04h и 10h, а налазе се у меморији на адресама 3003h, 3004h и 3005h, респективно. Приликом читања друге инструкције процесор на магистралаи генерише три циклуса читања из меморије. Циклуси читања бајтова друге инструкције се реализују на идентичан начин као и циклуси читања бајтова прве инструкције. Прочитане вредности се уписују у разреде $IR_{23..16}$, $IR_{15..8}$ и $IR_{7..0}$ прихватног регистра инструкције, а садржај програмског бројача $PC_{15..0}$ инкрементира на 3004h, 3005h и 3006h, респективно.

У фази *декодовања инструкције (ID)* постоје три циклуса на магистралаи, јер због коришћења меморијског индиректног адресирања процесор најпре у два циклуса на магистралаи из меморије чита два бајта 16-битне адресе операнда, а затим у трећем циклусу на магистралаи чита и сам 8-битни операнд. Истим поступком као и у првој инструкцији најпре се са адреса 1004h и 1005h читају нижи бајт 07h и виши бајт 00h 16 битне адресе операнда и смештају у разреде $TMP_{7..0}$ и $TMP_{15..8}$, прихватног регистра адресе операнда, а затим се садржај 0007h прихватног регистра адресе операнда $TMP_{15..0}$ користи као адреса меморијске локације са које се чита операнд чија је вредност 33h и уписује у разреде $TMP_{7..0}$ прихватног регистра операнда. Пошто су у

питању циклус читања из меморије управљачки сигнали магистрале RD и M/\overline{IO} имају вредност 1. Са XX је означено да је садржај разреда TMP_{15..8} прихватног регистра операнда недефинисан.

У фази *извршавање операције* (IE) садржај разреда TMP_{7..0} прихватног регистра операнда се уписује у регистар R_{27..0}.

По завршетку фазе *извршавање операције* (IE) друге инструкције **LOAD R2**, (1004h) се прелази на фазу *читање инструкције* (IF) треће инструкције **INS #2**. Трећа инструкција је дужине два бајта који имају вредности 1Fh и 02h, а налазе се у меморији на адресама 3006h и 3007h, респективно. Приликом читања треће инструкције процесор на магистрали генерише два циклуса читања из меморије. Циклуси читања бајтова треће инструкције се реализују на идентичан начин као и циклуси читања бајтова прве инструкције. Прочитане вредности се уписују у разреде IR_{23..16} и IR_{15..8} прихватног регистра инструкције, а садржај програмског бројача PC_{15..0} инкрементира на 3007h и 3008h, респективно.

Ова инструкција је нема фазу *декодовање инструкције* (ID).

У фази *извршавање операције* (IE) најпре се на стеку чува програмски бројач PC_{15..0}, а затим се из улаза два IV табеле чита адреса прекидне рутине и уписује у програмски бројач PC_{15..0}. Садржај регистра PC_{15..0} смешта се на стек у два циклуса уписа на магистрали. Пошто SP указује на последњу заузету локацију и стек расте ка вишим адресама, најпре се SP инкрементира па се на адресу 4001h уписује нижи бајт PC, PC_{7..0}=08h, а затим се SP поново инкрементира па се на адресу 4002h уписује виши бајт PC, PC_{15..8}=30h. Пошто је у питању циклус уписа у меморију управљачки сигнали магистрале WR и M/\overline{IO} имају вредност 1. Нижи бајт адресе прекидне рутине се налази у IV табели на адреси која се рачуна по формули

$$IVTP+i*size=0h+2*2=4h,$$

где је IVTP почетна адресе IV табеле, дата садржајем регистра IVTP, i број улаза у IV табелу, задат адресним делом инструкције софтверског прекида, а size број бајтова које заузима адреса прекидне рутине у IV табели. Виши бајт адресе прекидне рутине се налази у IV табели на првој следећој адреси. Стога се у два циклуса читања на магистрали, најпре са адресе 0004h чита вредност 0Eh и уписује у разреде PC_{7..0}, а затим са адресе 0005h и вредност 30h и уписује у разреде PC_{15..8} програмског бројача PC_{15..0}. Пошто је у питању циклус читања из меморије управљачки сигнали магистрале RD и M/\overline{IO} имају вредност 1.

По завршетку фазе *извршавање операције* (IE) треће инструкције **INS #2** се прелази на фазу *читање инструкције* (IF) четврте инструкције **ADD R2, R1**. Четврта инструкција је дужине један бајт који има вредности 46h, а налази се у меморији на адреси 300Eh. Приликом читања четврте инструкције процесор на магистрали генерише један циклус читања из меморије. Циклус читања једног бајта четврте инструкције се реализују на идентичан начин као и циклуси читања бајтова прве инструкције. Прочитана вредност се уписује у разреде IR_{23..16} прихватног регистра инструкције, а садржај програмског бројача PC_{15..0} инкрементира на 300Fh.

Ова инструкција нема фазу *декодовање инструкције* (ID) јер се као изворишта и одредиште операнда имплицитно користе регистри опште намене.

У фази *извршавање операције* (IE) садржај 33h регистра опште намене R_{27..0} се сабира са садржајем 00h регистра опште намене R_{17..0} и добијена сума 33h уписује у регистар опште намене R_{27..0}.

По завршетку фазе *извршавање операције* (IE) четврте инструкције **ADD R2, R1** се прелази на фазу *читање инструкције* (IF) пете инструкције **RTI**. Пета инструкција је дужине један бајт који има вредности 02h, а налази се у меморији на адреси 300Fh. Приликом читања пете инструкције процесор на магистрали генерише један циклус

читања из меморије. Циклус читања једног бајта пете инструкције се реализују на идентичан начин као и циклуси читања бајтова прве инструкције. Прочитана вредност се уписује у разреде $IR_{23..16}$ прихватног регистра инструкције, а садржај програмског бројача $PC_{15..0}$ инкрементира на 3010h.

Ова инструкција нема фазу *декодовање инструкције (ID)*.

У фази *извршавање операције (IE)* се реализују два циклуса читања на магистралаи. Најпре се садржај SP , који указује на задњу заузету локацију на стеку, користи да се у првом циклусу читања на магистралаи, са стека, са адресе 4002h, чита вредност 30h, која представља виши бајт адресе повратка из потпрограма, и уписује у виши бајт регистра PC , $PC_{15..8}$, па се затим садржај SP декрементира. Потом се, садржај SP , који указује на задњу заузету локацију на стеку, користи да се у другом циклусу читања на магистралаи, са стека, са адресе 4001h, чита вредност 08h, која представља нижи бајт адресе повратка из потпрограма, и уписује у нижи бајт регистра PC , $PC_{7..0}$, па се затим садржај SP поново декрементира. Пошто је у питању циклус читања из меморије управљачки сигнали магистрале RD и M/\overline{IO} имају вредност 1.

По завршетку фазе *извршавање операције (IE)* пете инструкције **RTI** се прелази на фазу *читање инструкције (IF)* шесте инструкције **HALT**. Шеста инструкција је дужине један бајт који има вредности 00h, а налази се у меморији на адреси 3008h. Приликом читања шесте инструкције процесор на магистралаи генерише један циклус читања из меморије. Циклус читања једног бајта шесте инструкције се реализују на идентичан начин као и циклуси читања бајтова прве инструкције. Прочитана вредност се уписује у разреде $IR_{23..16}$ прихватног регистра инструкције, а садржај програмског бројача $PC_{15..0}$ инкрементира на 3009h.

Инструкција **HALT** је безадресна инструкција, па се после фазе *читање инструкције (IF)* одмах прелази на фазу *извршавање операције (IE)* у оквиру које се зауставља процесор.

1.1.6 ЗАДАТАК

Посматра се процесор описан у задатку 1.1.1.

Садржај дела оперативне меморије је приказан на слици 18, при чему су све вредности дате у хексадецималном облику.

Адреса	0000	0001	0002	0003	0004	0005	0006	0007	0008	0009	000A	...
Садржај	37	A1	00	08	2A	30	40	00	56	07	00	...
...
Адреса	1000	1001	1002	1003	1004	1005	1006	1007	1008	1009	100A	...
Садржај	00	30	06	00	40	08	40	00	40	05	00	...
...
Адреса	3000	3001	3002	3003	3004	3005	3006	3007	3008	3009	300A	...
Садржај	82	00	02	8F	02	1F	03	00	00	25	03	...
Адреса	4000	4001	4002	4003	4004	4005	4006	4007	4008	4009	400A	...
Садржај	60	00	08	04	02	01	50	03	00	17	00	...

Слика 18 Садржај дела меморије

а) Навести секвенцу садржаја на адресној магистрали, магистрали података и управљачкој магистрали за сваки циклус на магистрали при извршавању програма приказаног на слици 19.

Претпоставити да се током извршавања овог програма генерише само прекид изазван инструкцијом софтверског прекида INS. Инструкција INS је тако реализована да се у фази *извршавање операције* (IE) на најпре на стек ставља садржај програмског бројача PC, а затим на основу броја улаза у IV табелу, датог адресним делом 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. Стога се приликом извршавања инструкција пролази само кроз фазе *читање инструкције* (IF), *декодовање инструкције* (IF) и *извршавање операције* (IE), а не и кроз фазу *опслуживање прекида* (IS).

Претпоставити да је пре почетка извршавања главног програма SP=6036h, IVTP=0h и PC=3000h.

адреса	инструкција	коментар
	; главни програм	
3000h	LOAD (02h)	; memorijski indirektno adresiranje ACC = MEM[02h]
3003h	AND #2	; neposredno adresiranje ACC = ACC & 2
3005h	INS #3	; softverski prekid
3007h	HALT	; zaustavljanje procesora
3008h	...	
...		
	; прекидна рутина	
4000h	IN 8h	; ACC=IO[8h]
4003h	RTI	; povratak iz prekidne rutine
4004h	...	

Слика 19 Програм

Решење:

а) Секвенца садржаја на адресној магистрали, магистрали података и управљачкој магистрали за сваки циклус на магистрали је приказана у табели са слике 20. У табели се користе исте ознаке као и у табели са слике 4 из задатка 1.1.1, при чему се јавља и колона SP_{15.0} - указивач на врх стека након ажурирања његовог садржаја.

P6	ABUS _{15..0}	DBUS _{7..0}	RD	WR	MIO	PC _{15..0}	IR _{23..16}	IR _{15..8}	IR _{7..0}	TMP _{15..0}	ACC _{7..0}	SP _{15..0}	Коментар
	-	-	-	-	-	3000	-	-	-	-	-	6036	
1	3000	82	1	0	1	3001	82	-	-	-	-	6036	IF: LOAD (02h), први бајт инс.
2	3001	00	1	0	1	3002	82	00	-	-	-	6036	IF: LOAD (02h), други бајт инс.
3	3002	02	1	0	1	3003	82	00	02	-	-	6036	IF: LOAD (02h), трећи бајт инс.
4	0002	00	1	0	1	3003	82	00	02	00XX	-	6036	ID: LOAD (02h), формирање адресе опрнд. - читање вишег бајта, TMP _{15..8} =MEM[0002]
5	0003	08	1	0	1	3003	82	00	02	0008	-	6036	ID: LOAD (02h), формирање адресе опрнд. - читање нижег бајта, TMP _{7..0} =MEM[0003]
6	0008	56	1	0	1	3003	82	00	02	XX56	-	6036	ID: LOAD (02h), читање опрнд. из мем. лок., TMP _{7..0} =MEM[0008]
											56	6036	IE: LOAD (02h), ACC _{7..0} =TMP _{7..0}
7	3003	8F	1	0	1	5004	8F	-	-	-	56	6036	IF: AND #2h, први бајт инс.
8	3004	02	1	0	1	5005	8F	02	-	-	56	6036	IF: AND #2h, други бајт инс.
										XX02	56	6036	ID: AND #2h, читање опрнд. из прихв. рег. инс., TMP _{7..0} =IR _{15..8}
											02	6036	IE: AND #2h, ACC _{7..0} =ACC _{7..0} &TMP _{7..0}
9	3005	5F	1	0	1	3006	5F	-	-	-	02	6036	IF: INS #3h, први бајт инс.
10	3006	03	1	0	1	3007	5F	03	-	-	02	6036	IF: INS #3h, други бајт инс.
11	6036	07	0	1	1	3007	5F	03	-	-	02	6036	IE: INS #3h, нижи бајт PC на стек
												6035	IE: INS #3h, SP=SP-1
12	6035	30	0	1	1	3007	5F	03	-	-	02	6035	IE: INS #3h, виши бајт PC на стек
												6034	IE: INS #3h, SP=SP-1
									0006	02	6034	IE: INS #3h, израчунавање адресе улаза 3 у IV табели (0h+3*2=6h)	
13	0006	40	1	0	1	40XX	5F	03	-	-	02	6034	IE: INS #3h, читање вишег бајта адресе прек. рутине и упис у виши бајт PC
14	0007	00	1	0	1	4000	5F	03	-	-	02	6034	IE: INS #3h, читање нижег бајта адресе прек. рутине и упис у нижи бајт PC
15	4000	60	1	0	1	4001	60	-	-	-	02	6034	IF: IN 8h, први бајт инс.
16	4001	00	1	0	1	4002	60	00	-	-	02	6034	IE: IN 8h, други бајт инс.
17	4002	08	1	0	1	4003	60	00	08	-	02	6034	IF: IN 8h, трећи бајт инс.
										0008	02	6034	IF: IN 8h, формирање адресе опрнд., TMP _{15..8} =IR _{15..8} , TMP _{7..0} =IR _{7..0}
18	0008	56	1	0	0	4003	60	00	08	XX56	02	6034	ID: IN 8h, читање опрнд. из уи рег. TMP _{7..0} =UI[0008]
											56	6034	IE: IN 8h, ACC _{7..0} =TMP _{7..0}
19	4003	04	1	0	1	4004	04	-	-	-	56	6034	IF: RTI , први бајт инс.
												6035	IE: RTI , SP=SP+1
20	6035	30	1	0	1	30XX	04	-	-	-	56	6035	IE: RTI , бајт са стека у виши бајт PC
												6036	IE: RTI , SP=SP+1
21	6036	07	1	0	1	3007	04	-	-	-	56	6036	IE: RTI , бајт са стека у нижи бајт PC
22	3007	00	1	0	1	3008	00	-	-	-	56	6036	IF: HALT , први бајт инс.
													IE: HALT , START=0

Слика 20 Секвенца садржаја на магистрали

Извршавање програма почиње фазом *читање инструкције* (IF) прве инструкције **LOAD** (2h) на коју указује тренутна вредност 3000h садржаја програмског бројача PC_{15..0}. Прва инструкција је дужине три бајта који имају вредности 82h, 00h и 02h, а налазе се у локацијама меморије на адресама 3000h, 3001h и 3002h, респективно. С обзиром да је ширина меморијске речи 8 битова, процесор из меморије чита бајт по

бајт. Приликом читања прве инструкција процесор на магистралаи генерише три циклуса читања из меморије. У првом циклусу читања процесор на линије адресне магистрале ABUS_{15.0} пропушта вредност 3000h програмског бројача PC_{15.0} како би са те адресе прочитао први бајт инструкције 82h. Пошто је у питању циклус читања из меморије управљачки сигнали магистрале RD и M/ \overline{IO} имају вредност 1. Прочитани садржај 82h се појављује на линијама магистрале података DBUS_{7.0} и уписује у разреде IR_{23.16} прихватног регистра инструкције процесора, а садржај програмског бројача PC_{15.0} инкрементира на 3001h. На сличан начин се у другом и трећем циклусу читања са адреса 3001h и 3002h читају други и трећи бајтови бајт инструкције 00h и 02h, респективно. Прочитани садржаји се појављује на линијама магистрале података DBUS_{7.0} и уписује у разреде IR_{15.8} и IR_{7.0} прихватног регистра инструкције, а садржај програмског бројача PC_{15.0} инкрементира на 3002h и 3003h, респективно.

У фази *декодовања инструкције* (ID) постоје три циклуса на магистралаи, јер због коришћења меморијског индиректног адресирања процесор најпре у два циклуса на магистралаи из меморије чита два бајта 16-битне адресе операнда, а затим у трећем циклусу на магистралаи чита и сам 8-битни операнд. Адреса операнда је 16-битна и заузима две суседне меморијске локације, при чему се нижи бајт адресе налази на вишој а виши бајт на нижој адреси. Адреса меморијске локације на којој се налази виши бајт адресе операнда дата је другим и трећим бајтом инструкције и налази се у разредима IR_{15.8} (виши бајт) и IR_{7.0} (нижи бајт) прихватног регистра инструкције. У прва два циклуса на магистралаи процесор најпре из меморије са адресе 0002h чита садржај 00h, који представља виши бајт адресе операнда, и уписује у разреде TMP_{15.8}, а затим из меморије са адресе 0003h чита садржај 08h, који представља нижи бајт адресе операнда, и уписује у разреде TMP_{7.0}. У трећем циклусу на магистралаи садржај 0008h прихватног регистра адресе операнда TMP_{15.0} се користи као адреса меморијске локације са које се чита операнд чија је вредност 56h и уписује у разреде TMP_{7.0} прихватног регистра операнда. Пошто је у питању циклуси читања из меморије управљачки сигнали магистрале RD и M/ \overline{IO} имају вредност 1. Са XX је означено да је садржај разреда TMP_{15.8} прихватног регистра операнда недефинисан.

У фази *извршавање операције* (IE) садржај разреда TMP_{7.0} прихватног регистра операнда се уписује у регистар ACC_{7.0}.

По завршетку фазе *извршавање операције* (IE) прве инструкције **LOAD** (2h) се прелази на фазу *читање инструкције* (IF) друге инструкције **AND** #2h. Друга инструкција је дужине два бајта који имају вредности 8Fh и 02h, а налазе се у меморији на адресама 3003h и 3004h. Приликом читања друге инструкције процесор на магистралаи генерише два циклуса читања из меморије. Циклуси читања бајтова друге инструкције се реализују на идентичан начин као и циклуси читања бајтова прве инструкције. Прочитане вредности се уписују у разреде IR_{23.16} и IR_{15.8} прихватног регистра инструкције, а садржај програмског бројача PC_{15.0} инкрементира на 3004h и 3005h, респективно.

У фази *декодовање инструкције* (ID) нема циклуса на магистралаи, јер је операнд због коришћења непосредног адресирања специфициран другим бајтом инструкције и налази се у разредима IR_{15.8} прихватног регистра инструкције. Фаза *декодовања инструкције* (ID) се састоји у пребацивању операнда чија је вредност 02h из разреда IR_{15.8} прихватног регистра инструкције у разреде TMP_{7.0} прихватног регистра операнда. Са XX је означено да је садржај разреда TMP_{15.8} прихватног регистра операнда недефинисан.

У фази *извршавање операције* (IE) се примењује логичка И операција над садржајима акумулатора ACC_{7.0} и разреда TMP_{7.0} прихватног регистра операнда и резултат уписује у акумулатор ACC_{7.0}.

По завршетку фазе *извршавање операције* (IE) друге инструкције **AND #2h** се прелази на фазу *читање инструкције* (IF) треће инструкције **INS #3**. Трећа инструкција је дужине два бајта који имају вредности 1Fh и 03h, а налазе се у меморији на адресама 3005h и 3006h, респективно. Приликом читања треће инструкције процесор на магистрали генерише два циклуса читања из меморије. Циклуси читања бајтова треће инструкције се реализују на идентичан начин као и циклуси читања бајтова прве инструкције. Прочитане вредности се уписују у разреде $IR_{23..16}$ и $IR_{15..8}$ прихватног регистра инструкције, а садржај програмског бројача $PC_{15..0}$ инкрементира на 3006h и 3007h, респективно.

Ова инструкција је нема фазу *декодовање инструкције* (ID).

У фази *извршавање операције* (IE) најпре се на стеку чува програмски бројач $PC_{15..0}$, а затим се из улаза 3 IV табеле чита адреса прекидне рутине и уписује у програмски бројач $PC_{15..0}$. Садржај регистра $PC_{15..0}$ смешта се на стек у два циклуса уписа на магистрали. Пошто SP указује на прву слободну локацију и стек расте ка нижим адресама, најпре се на адресу 6036h уписује нижи бајт PC, $PC_{7..0}=07h$ па се SP декрементира, а затим се на адресу 6035h уписује виши бајт PC, $PC_{15..8}=30h$ па се SP декрементира на вредност 6034h. Пошто је у питању циклус уписа у меморију управљачки сигнали магистрале WR и M/\overline{IO} имају вредност 1. Виши бајт адресе прекидне рутине се налази у IV табели на адреси која се рачуна по формули

$$IVTP+i*size=0h+2*3=6h,$$

где је IVTP почетна адресе IV табеле, дата садржајем регистра IVTP, i број улаза у IV табелу, задат адресним делом инструкције софтверског прекида, а size број бајтова које заузима адреса прекидне рутине у IV табели. Нижи бајт адресе прекидне рутине се налази у IV табели на првој следећој адреси. Стога се у два циклуса читања на магистрали, најпре са адресе 0006h чита вредност 40h и уписује у разреде $PC_{15..8}$, а затим са адресе 0007h и вредност 00h и уписује у разреде $PC_{7..0}$ програмског бројача $PC_{15..0}$. Пошто је у питању циклус читања из меморије управљачки сигнали магистрале RD и M/\overline{IO} имају вредност 1.

По завршетку фазе *извршавање операције* (IE) треће инструкције **INS #3** се прелази на фазу *читање инструкције* (IF) четврте инструкције **IN 8h**. Четврта инструкција је дужине три бајта који имају вредности 60h, 00h и 08h, а налазе се у меморији на адресама 4000h, 4001h и 4002h, респективно. Приликом читања четврте инструкције процесор на магистрали генерише три циклуса читања из меморије. Циклуси читања бајтова четврте инструкције се реализују на идентичан начин као и циклуси читања бајтова прве инструкције. Прочитане вредности се уписују у разреде $IR_{23..16}$, $IR_{15..8}$ и $IR_{7..0}$ прихватног регистра инструкције, а садржај програмског бројача $PC_{15..0}$ инкрементира на 4001h, 4002h и 4003h, респективно.

У фази *декодовања инструкције* (ID) постоји један циклус на магистрали, јер процесор због коришћења у/и директног адресирања у једном циклусу на магистрали из улазно/излазног адресног простора чита 8 битни операнд. Најпре се адреса регистра из улазно/излазног адресног простора, чија је вредност 0008h и која је дата другим и трећим бајтом инструкције из разреда $IR_{15..8}$ (виши бајт) и $IR_{7..0}$ (нижи бајт), пребацује у разреде $TMP_{15..8}$ и $TMP_{7..0}$ прихватног регистра адресе операнда, респективно. Затим се у циклусу на магистрали садржај 0008h прихватног регистра адресе операнда $TMP_{15..0}$ користи као адреса регистра из улазно/излазног адресног простора са које се чита операнд чија је вредност 56h и уписује у разреде $TMP_{7..0}$ прихватног регистра операнда. Пошто је у питању циклус читања из регистра улазно/излазног адресног простора управљачки сигнали магистрале RD и M/\overline{IO} имају вредности 1 и 0, респективно. Са XX је означено да је садржај разреда $TMP_{15..8}$ прихватног регистра операнда недефинисан.

У фази *извршавање операције* (IE) садржај разреда $TM\overline{P}_{7:0}$ прихватног регистра операнда се уписује у регистар $ACC_{7:0}$.

По завршетку фазе *извршавање операције* (IE) четврте инструкције **IN** 8h се прелази на фазу *читање инструкције* (IF) пете инструкције **RTI**. Пета инструкција је дужине јадан бајт који има вредности 04h, а налази се у меморији на адреси 4003h. Приликом читања пете инструкције процесор на магистрали генерише један циклус читања из меморије. Циклус читања једног бајта пете инструкције се реализују на идентичан начин као и циклуси читања бајтова прве инструкције. Прочитана вредност се уписује у разреде $IR_{23..16}$ прихватног регистра инструкције, а садржај програмског бројача $PC_{15..0}$ инкрементира на 4004h.

Ова инструкција нема фазу *декодовање инструкције* (ID).

У фази *извршавање операције* (IE) се реализују два циклуса читања на магистрали. Најпре се садржај **SP**, који указује на прву слободну локацију на стеку инкрементира, па се користи да се у првом циклусу читања на магистрали, са стека, са адресе 6035h, чита вредност 30h, која представља виши бајт адресе повратка из потпрограма, и уписује у виши бајт регистра **PC**, $PC_{15..8}$, па се затим садржај **SP** инкрементира. Потом се, садржај **SP** користи да се у другом циклусу читања на магистрали, са стека, са адресе 6036h, чита вредност 07h, која представља нижи бајт адресе повратка из потпрограма, и уписује у нижи бајт регистра **PC**, $PC_{7..0}$. Пошто је у питању циклус читања из меморије управљачки сигнали магистрале **RD** и M/\overline{IO} имају вредност 1.

По завршетку фазе *извршавање операције* (IE) пете инструкције **RTI** се прелази на фазу *читање инструкције* (IF) шесте инструкције **HALT**. Шеста инструкција је дужине јадан бајт који има вредности 00h, а налази се у меморији на адреси 3007h. Приликом читања шесте инструкције процесор на магистрали генерише један циклус читања из меморије. Циклус читања једног бајта шесте инструкције се реализују на идентичан начин као и циклуси читања бајтова прве инструкције. Прочитана вредност се уписује у разреде $IR_{23..16}$ прихватног регистра инструкције, а садржај програмског бројача $PC_{15..0}$ инкрементира на 3008h.

Инструкција **HALT** је безадресна инструкција, па се после фазе *читање инструкције* (IF) одмах прелази на фазу *извршавање операције* (IE) у оквиру које се зауставља процесор.