

# ПРИЛОЖЕНИЕ

## к методическим указаниям на выполнение лабораторной работы "Организация памяти ЭВМ. Устройство управления памятью"

### *Задания на проведение лабораторного практикума*

#### **Задание 1:**

1. В программе инициализации загрузить значения регистров базовых адресов и регистров прав доступа для системного режима в соответствии с вариантом задания, приведенным в таблице П4.1 приложения 4 или из файла заданий.

2. Включить устройство управления памятью.

3. Выполнить загрузку регистров базовых адресов и прав доступа для пользовательского режима в соответствии с вариантом задания, приведенным в таблице П4.2.

4. Установить пользовательский режим работы процессора.

5. Составить четыре подпрограммы, выполняющие следующие действия:

5.1. Нахождение суммы N-ых элементов сегментов данных с накоплением суммы в M-ой ячейке сегмента данных не имеющего установленных атрибутов защиты (разрешено чтение и запись). Режим работы процессора определяется выполняемой командой обращения к подпрограмме накопления суммы, заданной в п. 6. По первому и второму адресам абсолютная адресация (значения элементов N и M приведены в таблице П4.3);

5.2. Выполнение заданной в таблице П4.4 двухадресной команды с операндами из ячеек N и M (таблица 3), находящихся в сегментах со смежными номерами. Первый операнд команды (ячейка N) должен быть из сегмента с четным номером, а второй операнд (ячейка M) из сегмента с нечетным номером (то есть всего 4 команды с сочетанием операндов N-M из сегментов 0 – 1, 2 – 3, 4 – 5, 6 – 7). Виды адресаций к операндам приведены в таблице П4.4;

5.3. Выполнение одноадресных команд с заданной адресацией (таблица П4.5) для элементов с адресом N сегментов с 0 по 7 (в восьмеричной системе счисления).

6. Обратиться к разработанным в подпунктах 5.1, 5.2 и 5.3 подпрограммам по командам, приведенным в таблице: JSR, EMT, TRAP и JSR.

Вариант		1,6,11,16	2,7,12,17	3,8,13,18	4,9,14,19	5,10,15,20
Вызываемая подпрограмма	П.5.1	JSR	JSR	EMT	EMT	TRAP
	П.5.2	EMT	EMT	JSR	JSR	EMT
	П.5.3.а	TRAP	JSR	TRAP	JSR	JSR
	П.5.3.в	JSR	TRAP	JSR	TRAP	JSR

7. Выполнить изменение содержимого указанных в таблице П4.6 регистров базовых адресов и прав доступа для пользовательского режима через вектор прерывания #K и повторить выполнение подпункта 5.3.а) или в) для одноадресной команды с записью (т.е. кроме команды TST).

#### **Задание 2:**

8. Составить и выполнить тестовые программы для выработки всех типов векторов прерываний для сегментов данных, реализованных в ЛУ (таблица 7), включая вектор 250 для заданных вариантов заполнения таблицы дескрипторов и базовых адресов для сочетаний причин прерываний, приведенных в таблице 8.

9. Составить и выполнить тестовые программы для выработки вектора 250 при обращении к кодовым сегментам для сочетаний причин прерываний, приведенных в таблице 9.

#### **Примечания:**

1 В подпрограммах обработки прерываний для векторов 4, 10, 120 и 160 в регистр R0 последовательно загрузить номер вектора прерывания и содержимое программного счетчика, сохраненного в стеке, на котором выработан сигнал прерывания. Для вектора 250 в ППОП в

регистр R0 последовательно загрузить номер вектора, содержимое программного счетчика PC, содержимое регистра SR0, содержимое регистра SR2.

2 Регистры R1-R5 рекомендуется использовать при адресациях, использующих адрес, хранимый в РОН. Регистр R0 не рекомендуется использовать для этих целей, т.к. в ППОП его содержимое необходимо предварительно сохранять в стеке, а перед командой возврата - восстанавливать.

## Приложение 4

### Варианты заданий на лабораторный практикум

Таблица П4.1 – Варианты заданий базовых адресов и атрибутов защиты системных сегментов

Вариант 1					Вариант 2					Вариант 3					Вариант 4				
№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты
0	С	5016	33*		0	С	6016	24*		0	К	6716	37	В	0	С	4016	14*	
1	К	4432	101	В	1	К	6545	36	В, Ч	1	С	6432	57*		1	К	5245	47	В, Ч
2	Д	5560	24	3, Ч	2	С	6732	57*		2	Д	5560	62	3, Ч	2	Д	6432	57	3
3	С	5430	37*		3	Д	6060	72	3	3	Д	6330	11	3	3	Д	0430	72	3
4	Д	6040	57	3	4	Д	5430	11	3, Ч	4	С	6140	33*		4	С	6330	11*	
5	Д	6670	112	3	5	Д	5070	33	3	5	К	4770	101	В, Ч	5	К	4770	43	В
6	К	4250	11	В, Ч	6	К	4640	101	В	6	Д	6230	24	3	6	Д	6140	111	3, Ч
7	Д	7600	177		7	Д	7600	177		7	Д	7600	177		7	Д	7600	177	
Вариант 5					Вариант 6					Вариант 7					Вариант 8				
№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты
0	С	5616	33*		0	С	6716	34*		0	С	6216	33*		0	С	6516	64*	
1	Д	5432	101	3, Ч	1	К	5760	57	В, Ч	1	К	5432	71	В	1	Д	5245	27	3
2	Д	4560	54	3	2	Д	5432	67	3	2	К	5560	144	В, Ч	2	Д	6432	47	3, Ч
3	С	4367	27*		3	Д	5560	112	3	3	С	6330	56*		3	К	4560	76	В, Ч
4	К	5140	57	В	4	Д	6330	14	3, Ч	4	Д	6140	77	3	4	Д	5100	12	3
5	Д	4770	62	3	5	С	4750	36*		5	Д	5600	72	3	5	С	5770	36*	
6	К	6010	13	В, Ч	6	К	5140	51	В	6	Д	5010	15	3, Ч	6	К	6140	51	В
7	Д	7600	177		7	Д	7600	177		7	Д	7600	177		7	Д	7600	177	

#### Примечания:

В столбце размер \* означает расширение сегментов в сторону уменьшения адресов.

В столбце атрибутов защиты введены следующие обозначения:

- ◆ 3 - атрибут защиты по записи;
- ◆ 3, Ч - атрибут защиты по записи и чтению.
- ◆ В - означает, что чтение из кодового сегмента запрещено.
- ◆ В, Ч означает, что чтение из кодового сегмента разрешено.

Продолжение таблицы П4.1

Вариант 9					Вариант 10					Вариант 11					Вариант 12				
№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты
0	С	4716	63*		0	С	5616	54*		0	С	4250	73*		0	С	4640	64*	
1	Д	4432	111	3	1	Д	6245	105	3, Ч	1	Д	5016	101	3, Ч	1	К	6016	36	В, Ч
2	С	6560	34*		2	Д	4750	67	3	2	К	4432	24	В	2	С	6545	57*	
3	К	6330	57	В	3	Д	4560	72	3	3	С	5560	37*		3	Д	6732	72	3, Ч
4	Д	5140	77	3	4	С	5330	13*		4	Д	5430	57	3	4	Д	6060	11	3
5	Д	4770	112	3, Ч	5	К	5770	53	В, Ч	5	Д	6040	112	3	5	Д	5430	33	3
6	К	5560	11	В, Ч	6	К	6140	31	В	6	К	6670	11	В, Ч	6	К	5070	101	В
7	Д	7600	177		7	Д	7600	177		7	Д	7600	177		7	Д	7600	177	
Вариант 13					Вариант 14					Вариант 15					Вариант 16				
№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты
0	Д	6230	37	3, Ч	0	С	4770	74*		0	С	6010	63*		0	С	5140	54*	
1	С	6716	57*		1	Д	4016	47	3	1	Д	5616	101	3	1	Д	6716	57	3
2	К	6432	62	В	2	К	5245	57	В	2	К	5432	54	В, Ч	2	Д	5760	67	3
3	Д	5560	11	3	3	К	6432	72	В, Ч	3	С	4560	57*		3	Д	5432	112	3, Ч
4	С	6330	33*		4	С	0430	11*		4	Д	4367	27	3	4	К	5560	14	В
5	Д	6140	101	3	5	Д	6330	43	3, Ч	5	К	5140	62	В	5	С	6330	36*	
6	К	4770	24	В, Ч	6	Д	6140	111	3	6	Д	4770	13	3, Ч	6	К	4750	51	В, Ч
7	Д	7600	177		7	Д	7600	177		7	Д	7600	177		7	Д	7600	177	
Вариант 17					Вариант 18					Вариант 19					Вариант 20				
№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты
0	С	5010	53*		0	С	6140	64*		0	С	5560	73*		0	С	6140	64*	
1	Д	6216	71	3	1	Д	6516	27	3, Ч	1	К	4716	111	В	1	Д	5616	105	3
2	К	5432	144	В	2	К	5245	47	В	2	С	4432	34*		2	Д	6245	67	3
3	С	5560	16*		3	Д	6432	76	3	3	К	6560	57	В, Ч	3	К	4750	72	В, Ч
4	Д	6330	77	3	4	К	4560	12	В, Ч	4	Д	6330	77	3, Ч	4	С	4560	13*	
5	Д	6140	72	3, Ч	5	С	5100	36*		5	Д	5140	112	3	5	К	5330	53	В
6	К	5600	15	В, Ч	6	Д	5770	51	3	6	Д	4770	11	3	6	Д	5770	31	3, Ч
7	Д	7600	177		7	Д	7600	177		7	Д	7600	177		7	Д	7600	177	

Таблица П4.2 – Варианты заданий базовых адресов и атрибутов защиты пользовательских сегментов

Вариант 1					Вариант 2					Вариант 3					Вариант 4				
№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты
0	К	1045	23	В	0	К	0540	44	В, Ч	0	С	2245	26*		0	Д	1210	14	
1	Д	2432	111	3	1	К	1545	56	В	1	Д	2432	57		1	К	0245	55	В
2	Д	1560	34		2	Д	2732	67		2	Д	1560	42	3, Ч	2	Д	0432	67	3, Ч
3	С	3430	47*		3	Д	3060	52	3	3	Д	1330	14	3	3	К	1140	72	В, Ч
4	С	4000	67*		4	С	4000	11*		4	С	4000	53*		4	С	4000	11*	
5	К	1670	72	В, Ч	5	Д	2070	23	3, Ч	5	К	3770	121	В	5	Д	2770	73	3
6	Д	3600	12	3, Ч	6	С	3640	71*		6	К	3123	44	В, Ч	6	С	3140	55*	
7	Д	7600	77		7	Д	7600	77		7	Д	7600	77		7	Д	7600	77	
Вариант 5					Вариант 6					Вариант 7					Вариант 8				
№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты
0	Д	3245	63		0	С	1010	24*		0	С	2245	53*		0	К	1010	44	В
1	К	3432	121	В	1	Д	0540	37	3	1	Д	3432	171	3	1	С	0245	77*	
2	Д	1560	44	3, Ч	2	Д	1432	67		2	К	1560	34	В, Ч	2	Д	0432	15	
3	С	1020	65*		3	Д	2560	112	3, Ч	3	Д	1330	67	3, Ч	3	К	0560	66	В, Ч
4	С	4000	47*		4	С	4000	12*		4	С	4000	77*		4	С	4000	12*	
5	Д	1770	32	3	5	К	0750	34	В, Ч	5	К	2523	42	В	5	Д	1770	76	3
6	К	2010	13	В, Ч	6	К	3140	61	В	6	Д	3010	15		6	Д	2140	51	3, Ч
7	Д	7600	77		7	Д	7600	77		7	Д	7600	77		7	Д	7600	77	
Вариант 9					Вариант 10					Вариант 11					Вариант 12				
№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты
0	К	1245	63	В, Ч	0	К	0110	74	В	0	К	3430	23	В, Ч	0	К	3060	44	В
1	Д	0432	111	3	1	Д	0245	115	3, Ч	1	Д	1045	111	3	1	Д	0540	56	3, Ч
2	К	1560	44	В	2	С	1410	27*		2	К	2432	34	В	2	Д	1545	67	3
3	С	0030	77*		3	Д	2560	72	3	3	С	1560	47*		3	К	2732	52	В, Ч
4	Д	4000	27		4	С	4000	43*		4	С	4000	67*		4	С	4000	11*	
5	Д	2770	52	3, Ч	5	К	0770	113	В, Ч	5	Д	1670	72	3, Ч	5	Д	2070	73	
6	С	3563	11*		6	Д	1140	11		6	Д	3600	12		6	С	3640	16*	
7	Д	7600	77		7	Д	7600	77		7	Д	7600	77		7	Д	7600	77	

Продолжение таблицы П4.2

Вариант 13					Вариант 14					Вариант 15					Вариант 16				
№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты
0	С	1330	26*	3, Ч В, Ч В 3	0	Д	1140	14	3 3, Ч В, Ч В	0	Д	1020	63	3, Ч 3 В, Ч В	0	С	2560	24*	В, Ч 3, Ч 3 В
1	Д	2245	57		1	Д	1210	55		1	Д	3245	121		1	К	1010	37	
2	К	2432	42		2	Д	0245	111		2	С	3432	44*		2	Д	0540	67	
3	К	1560	14		3	К	0432	72		3	Д	1560	65		3	Д	1432	112	
4	С	4000	53*		4	С	4000	11*		4	С	4000	47*		4	С	4000	12*	
5	Д	3770	121		5	К	2770	73		5	К	1770	12		5	К	0750	34	
6	Д	3123	44		6	С	3140	65*		6	К	2010	43		6	Д	3140	61	
7	Д	7600	77	7	Д	7600	77	7	Д	7600	77	7	Д	7600	77				
Вариант 17					Вариант 18					Вариант 19					Вариант 20				
№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты
0	С	1330	53*	В, Ч 3, Ч В 3	0	К	0560	44	В 3, Ч В, Ч 3	0	К	0030	63	В, Ч 3 3, Ч В	0	Д	2560	74	В В, Ч 3 3, Ч
1	К	2245	171		1	Д	1010	27		1	Д	1245	111		1	К	0110	115	
2	Д	3432	34		2	Д	0245	157		2	Д	0432	44		2	С	0245	27*	
3	К	1560	67		3	К	0432	66		3	С	1560	37*		3	К	1410	72	
4	С	4000	77*		4	С	4000	12*		4	Д	4000	77		4	С	4000	13*	
5	Д	2523	42		5	Д	1770	76		5	К	2770	52		5	Д	0770	113	
6	Д	3010	15		6	С	2140	51*		6	С	3563	11*		6	Д	1140	41	
7	Д	7600	77	7	Д	7600	77	7	Д	7600	77	7	Д	7600	77				

Таблица П4.3 – Варианты заданий номеров используемых ячеек памяти

Вариант	N	M	Вариант	N	M
1	2000	500	11	3000	700
2	2050	600	12	3050	650
3	2100	700	13	3100	400
4	2200	750	14	3200	650
5	2400	550	15	3400	450
6	2450	650	16	4450	350
7	2500	400	17	4500	600
8	2700	450	18	4700	750
9	3000	300	19	5000	400
10	3500	350	20	5500	550

Таблица П4.4 – Варианты заданий мнемоник и адресаций для двухадресной команды

Вариант	Команда	Адресация первого операнда	Адресация второго операнда
1	MOVB	Индексная	Автодекрементная
2	BIC	Индексная	Регистровая косвенная
3	BISB	Регистровая косвенная	Абсолютная
4	SUB	Автоинкрементная	Индексная
5	MOV	Автодекрементная	Абсолютная
6	BICB	Абсолютная	Автоинкрементная
7	BIS	Автоинкрементная	Регистровая косвенная
8	XOR	Автодекрементная	Автоинкрементная
9	MOVB	Абсолютная	Индексная
10	BIC	Регистровая косвенная	Автодекрементная
11	BISB	Абсолютная	Регистровая косвенная
12	SUB	Автоинкрементная	Автодекрементная
13	MOV	Индексная	Автоинкрементная
14	BIC	Индексная	Абсолютная
15	BIS	Регистровая косвенная	Автоинкрементная
16	XOR	Автодекрементная	Индексная
17	MOVB	Абсолютная	Автодекрементная
18	BICB	Регистровая косвенная	Индексная
19	SUB	Автодекрементная	Регистровая косвенная
20	BIS	Автоинкрементная	Абсолютная

Таблица П4.5 – Варианты заданий мнемоник и адресаций для одноадресных команд

Вариант	Команда	Адресация	Команда	Адресация
1	TSTB	Абсолютная	CLR	Автодекрементная
2	TST	Регистровая косвенная	COMB	Индексная
3	TSTB	Автоинкрементная	INC	Абсолютная
4	TST	Автодекрементная	DECB	Автоинкрементная
5	TSTB	Индексная	NEG	Регистровая косвенная
6	TST	Абсолютная	ASRB	Автодекрементная
7	TSTB	Регистровая косвенная	ASL	Индексная
8	TST	Автоинкрементная	RORB	Абсолютная
9	TSTB	Автодекрементная	ROL	Автоинкрементная
10	TST	Индексная	SWAB	Регистровая косвенная
11	TST	Индексная	ROLB	Регистровая косвенная
12	TSTB	Автодекрементная	SWAB	Абсолютная
13	TST	Автоинкрементная	NEGB	Автодекрементная
14	TSTB	Регистровая косвенная	ASR	Индексная
15	TST	Абсолютная	ASLB	Автоинкрементная
16	TSTB	Индексная	ROR	Регистровая косвенная
17	TST	Автодекрементная	INCB	Абсолютная
18	TSTB	Автоинкрементная	DEC	Автодекрементная
19	TST	Регистровая косвенная	CLRB	Индексная
20	TSTB	Абсолютная	COM	Автоинкрементная

Таблица П4.6 – Варианты заданий для изменения пользовательских дескрипторов для свопинга сегментов

№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты
<b>Вариант 1</b>					<b>Вариант 2</b>					<b>Вариант 3</b>					<b>Вариант 4</b>				
0	Д	5604	10		6	Д	5073	27	3, Ч	1	Д	6140	37	3	4	К	4750	47	В
4	К	2210	66	В	5	К	1012	67	В	4	К	6700	42	В	0	Д	6310	51	
<b>Вариант 5</b>					<b>Вариант 6</b>					<b>Вариант 7</b>					<b>Вариант 8</b>				
4	К	6002	54	В, Ч	3	Д	5420	72	3	1	Д	5000	25	3, Ч	1	К	6140	54	В
6	К	4620	71	В	0	Д	5510	14	3, Ч	4	Д	5540	14	3	2	Д	5245	24	3
<b>Вариант 9</b>					<b>Вариант 10</b>					<b>Вариант 11</b>					<b>Вариант 12</b>				
1	Д	6560	7		5	К	6110	11	В, Ч	3	Д	4510	42	3, Ч	0	К	4640	37	В, Ч
3	Д	4430	77	3	2	К	5770	106	В	5	Д	4420	14		5	К	6016	35	В
<b>Вариант 13</b>					<b>Вариант 14</b>					<b>Вариант 15</b>					<b>Вариант 16</b>				
4	К	6610	54	В	1	К	4010	52	В	1	Д	4770	17	3, Ч	5	К	4750	42	В, Ч
3	Д	6430	72		4	Д	4512	61	3	4	К	5130	42	В	4	К	5560	37	В
<b>Вариант 17</b>					<b>Вариант 18</b>					<b>Вариант 19</b>					<b>Вариант 20</b>				
4	Д	6140	10	3	2	К	5245	37	В	3	К	4770	70	В	3	К	4560	11	В, Ч
5	К	6220	71	В	4	Д	5310	10	3, Ч	5	Д	5060	66	3	4	Д	4654	14	3

Таблица П4.7 – Варианты заданий на выработку векторов прерываний

Вектор	Причина прерывания
004	Нечетный адрес
010	Нелегальные или резервные инструкции процессора
014	Внутренне прерывание по биту трассировки Т слова состояния PSW
120	Обращение к неподключенному внешнему устройству
160	Физический адрес внешнего устройства больше 777776 <sub>8</sub>
250	Ошибка диспетчера памяти

Таблица П4.8 – Варианты заданий на выработку кодов ошибок для сегментов данных

Код ошибки в SR0	Причина прерывания
100	Защита по записи и чтению
010	Нарушение границ сегмента
001	Защита по записи
110	Защита по записи и чтению и нарушение границ сегмента
011	Защита по записи и нарушение границ сегмента
100	Несуществующий режим работы
111	Несуществующий атрибут защиты

Таблица 9 – Варианты заданий на выработку кодов ошибок для кодовых сегментов

Код ошибки в SR0	Причина прерывания
100	Защита по чтению из сегмента кода
010	Нарушение границ сегмента кода
001	Защита по записи в сегмент кода
110	Защита по чтению из сегмента кода и нарушение его границ
011	Защита по записи в сегмент кода и нарушение его границ
101	Защита по чтению из сегмента кода и по записи в него
111	Защита по чтению из сегмента кода, записи и нарушение его границ

# Приложение 1

## Режимы адресации и система команд процессора

### Режимы адресации, реализованные в процессоре

Мнемоника	Режим адресации	Пояснения	Пример
INC R4	Регистровая прямая	Операнд находится в РОН, указанном в команде	R4=100 Op=100
INC (R4)	Регистровая косвенная	Адрес операнда находится в РОН, указанном в команде	R4=100 Adr=100 Op=(100)
INC (R4)+	Прямая автоинкрементная (постинкрементная адресация)	Адрес операнда находится в РОН, указанном в команде. После выполнения команды содержимое регистра увеличивается на два при операциях над словами и на единицу при операциях над байтами	R4=100 Adr=100 Op=(100) R4'=102
INC -(R4)	Прямая автодекрементная (преддекрементная адресация)	До выполнения команды содержимое регистра уменьшается на два при операциях над словами и на единицу при операциях над байтами. Полученный результат является адресом операнда	R4=102 Adr=100 Op=(100) R4'=100
INC @(R4)+	Косвенная автоинкрементная (постинкрементная косвенная адресация)	В регистре, указанном в команде, находится адрес адреса операнда. После выполнения операции содержимое регистра всегда увеличивается на два, независимо от того, работает команда со словом или байтом	R4=100 Adr1=100 Adr=(100) Op=(Adr)= ((100)) R4'=102
INC @-(R4)	Косвенная автодекрементная (преддекрементная косвенная адресация)	В регистре, указанном в команде, до операции находится адрес адреса операнда, увеличенный на два. До выполнения операции содержимое регистра всегда уменьшается на два, независимо от того, работает команда со словом или байтом. Полученный результат является адресом адреса операнда	R4=102 Adr1=100 Adr=(100) Op=(Adr)= ((100)) R4'=100
INC 4(R4)	Прямая индексная	В регистре, указанном в команде, находится базовый адрес. До выполнения операции содержимое регистра суммируется с индексом, записанным перед РОН в команде. Полученная сумма является адресом операнда	R4=100 Adr=100+4=104 Op=(104)
INC @4(R4)	Косвенная индексная	Регистр, указанный в команде, содержит базовый адрес. Сумма индекса, записанного перед РОН в команде, и базового адреса определяет адрес адреса операнда	R4=100 Adr1=100+4=104 Adr=(104) Op=(Adr)=((104))
MOV #1234,R4	Непосредственная	Содержимое слова команды после знака # используется как операнд	Op=1234
INC @#400	Абсолютная	Содержимое слова команды после знака @# используется как адрес операнда	Adr=400 Op=(400)

где (100) - содержимое ячейки с адресом 100;

R4 - содержимое РОН с номером 4 до выполнения команды;

R4' - содержимое РОН с номером 4 после выполнения команды;

Adr - адрес операнда в ОП;



Adr1 - промежуточная ячейка ОП: адрес адреса операнда (адрес ячейки ОП, в которой хранится адрес операнда);

ОР - значение операнда.

Процессор выполняет команды четырех базовых форматов: одноадресные, двухадресные, передачи управления и безадресные (команды управления).

**Одноадресный формат:**

КОП	Режим	Rn
-----	-------	----

Rn=DST=источник и приемник операнда

**Двухадресный формат:**

КОП	Режим1	Rn1	Режим2	Rn2
-----	--------	-----	--------	-----

Rn1=SRC=источник операнда, Rn2=DST=приемник операнда

**Передачи управления:**

КОП	Смещение
-----	----------

Для команд передачи управления используется мнемоническая запись в формате JNE #67, где знак # означает смещение в команде относительно программного счетчика PC, 67 - номер команды, на которую выполняется переход. В процессе трансляции программы номер команды преобразуется в смещение.

В командах программного прерывания по вектору указывается номер вектора прерывания в таблице IDT в формате: EMT #120, где 120 - номер вектора прерывания в системной таблице векторов прерываний IDT.

В командах JSR, JMP и SOB в адресной части команды указывается абсолютный адрес перехода: JMP @#124.

**Безадресные:**

К О П
-------

### Система команд процессора

Мнемоника	Название	Описание
<b>Двухадресные команды процессора</b>		
MOV (B)	Пересылка	(DST):=(SRC)
CMP (B)	Сравнение	(SRC) - (DST)
ADD	Сложение	(DST):=(SRC) + (DST)
SUB	Вычитание	(DST):= (DST) - (SRC)
BIT (B)	Проверка бит	(SRC) $\wedge$ (DST)
BIC (B)	Очистка бит	(DST):=(SRC) $\wedge$ (DST)
BIS (B)	Установка бит	(DST):=(SRC) $\vee$ (DST)
XOR	Исключающее ИЛИ	(DST):=(SRC) $\oplus$ (DST)
<b>Одноадресные команды процессора</b>		
CLR (B)	Очистка	(DST) := 0
COM (B)	Инверсия	(DST):= $\sim$ (SRC)
INC (B)	Инкремент	(DST):=(SRC) + 1
DEC (B)	Декремент	(DST):=(SRC) - 1
NEG (B)	Смена знака (отрицание)	(DST):= - (SRC)
TST (B)	Проверка	(DST):=(SRC)
ASR (B)	Арифметический сдвиг вправо	(DST):=(SRC)/2
ASL (B)	Арифметический сдвиг влево	(DST):=(SRC) x 2

ROR (B)	Циклический сдвиг вправо	$C := (SRC[0]);$ $(DST) := (C.SRC[7-1])$
ROL (B)	Циклический сдвиг влево	$C := (SRC[7]);$ $(DST) := (SRC[6-0].C)$
SWAB	Перестановка байтов	$(DST) := (SRC[7-0].[15-8])$
ADC	Прибавление бита переноса	$(DST) := (SRC) + C$
SBC	Вычитание бита переноса	$(DST) := (SRC) - C$
SXT	Распространение знакового бита	$(DST) := 0$ , если $N=0$ ; $(DST) := -1$ , если $N=1$
MFPS	Перенос младшего байта из слова состояния	$(DST[7-0]) := PSW[7-0]$
MTPS	Перенос младшего байта в слово состояния	$PSW[7-0] := (STC[7-0])$
<b>Инструкции передачи управления</b>		
BR	Ветвление (безусловное) BR #124	$PC := PC + 2 \times \text{Смещение}$
<b>С учетом знака</b>		
BNE	Ветвление, если не равно нулю BNE #74	$PC := PC + 2 \times \text{Смещение}$ , если $Z=0$
BEQ	Ветвление, если равно нулю	если $Z=1$
BPL	Ветвление, если плюс	если $N=0$
BMI	Ветвление, если минус	если $N=1$
BVC	Ветвление, если бит переполнения сброшен	если $V=0$
BVS	Ветвление, если бит переполнения установлен	если $V=1$
BCC	Ветвление, если бит переноса сброшен	если $C=0$
BCS	Ветвление, если бит переноса установлен	если $C=1$
BGE	Ветвление, если больше или равно нулю	если $N \vee V=0$
BLT	Ветвление, если меньше нуля	если $N \vee V=1$
BGT	Ветвление, если больше нуля	если $[Z \vee (N \vee V)]=0$
BLE	Ветвление, если меньше или равно нулю	если $[Z \vee (N \vee V)]=1$
<b>Без учета знака</b>		
BHI	Ветвление, если выше	если $[Z \vee (N \vee V)]=1$ , $Z=C=0$
BLOS	Ветвление, если ниже или то же самое	если $[Z \vee (N \vee V)]=1$ , $C=0$
BHIS	Ветвление, если выше или то же самое	если $[Z \vee (N \vee V)]=1$ , $C=0$
BLO	Ветвление, если ниже	если $[Z \vee (N \vee V)]=1$ , $C=1$
JMP	Безусловный переход по абсолютному адресу	$PC := @\#Adr$
JSR	Переход к подпрограмме по абсолютному адресу	$PC := @\#Adr$ , $-(SP) := PC+2$
RTS	Возврат из подпрограммы	$PC := (SP)+$
SOB	Вычесть единицу и перейти (если не нуль) SOB Rn, @#Adr	$Rn := Rn-1$ , $PC := @\#Adr$ , если $Rn \neq 0$
EMT	Командное прерывание для системных программ EMT #10	$-(SP)s := PSW$ ; $-(SP)s := PC$ ; $PC := (IDT)s$
TRAP	Командное прерывание для пользовательских	$-(SP)p := PSW$ ; $-(SP)p := PC$ ;

	программ TRAP #34	PC:=(IDT)p
RTI	Возврат из системного прерывания	PC:=(SP)s+; PSW:=(SP)s+
RTT	Возврат из пользовательского прерывания	PC:=(SP)p+; PSW:=(SP)p+
<b>Безадресные команды</b>		
CLC	Очистить бит переноса	C:=0
CLV	Очистить бит переполнения	V:=0
CLZ	Очистить бит нулевого результата	Z:=0
CLN	Очистить бит отрицательного результата	N:=0
CCC	Очистить все биты условий	C=V=Z=N=0
SEC	Установить бит переноса	C:=1
SEV	Установить бит переполнения	V:=1
SEZ	Установить бит нулевого результата	Z:=1
SEN	Установить бит отрицательного результата	N:=1
SCC	Установить все биты условий	C=V=Z=N=1
NOP	Нет операции	

## Приложение 2

### Пример оформления листингов программ при домашней подготовке и последовательность выполнения экспериментальной части

Системный режим					Пользовательский				
№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты	№ сегмента	Тип сегмента	Базовый адрес	Размер	Атрибуты защиты
0	С	2016	53*		0	С	3016	34*	
1	Д	4562	101	3, Ч	1	К	3545	56	В
2	К	2760	14	В, Ч	2	С	6732	27*	
3	С	6230	27*		3	Д	4060	62	3
4	Д	5740	67		4	Д	5430	11	3, Ч
5	Д	1270	112	3	5	К	3070	23	В, Ч
6	К	4250	11	В	6	Д	5640	111	
7	Д	7600	177		7	Д	7600	77	

Исходные данные для исследований из таблиц 4.1 и 4.2 заданий:

Примечания:

В столбце тип сегмента:

С - сегмент стека;

Д - сегмент данных;

К - сегмент кода.

В столбце размер \* означает расширение сегментов в сторону уменьшения адресов (сегмент стека С).

В столбце атрибутов защиты:

♦ 3 - атрибут защиты по записи;

♦ 3, Ч - атрибуты защиты по записи и чтению;

♦ В - чтение из кодового сегмента запрещено (только выполнение);

♦ В, Ч - чтение из кодового сегмента

разрешено.

**Пункт 1.** Выполнить инициализацию регистров базовых адресов и прав доступа для системного режима работы.

MOV #2016,@#172340 ; Загрузка регистров базовых адресов

MOV #4562,@#172342 ; для системного режима

.....

MOV #7600,@#172356 ;

MOV #25436,@#172300 ; Загрузка регистров прав доступа

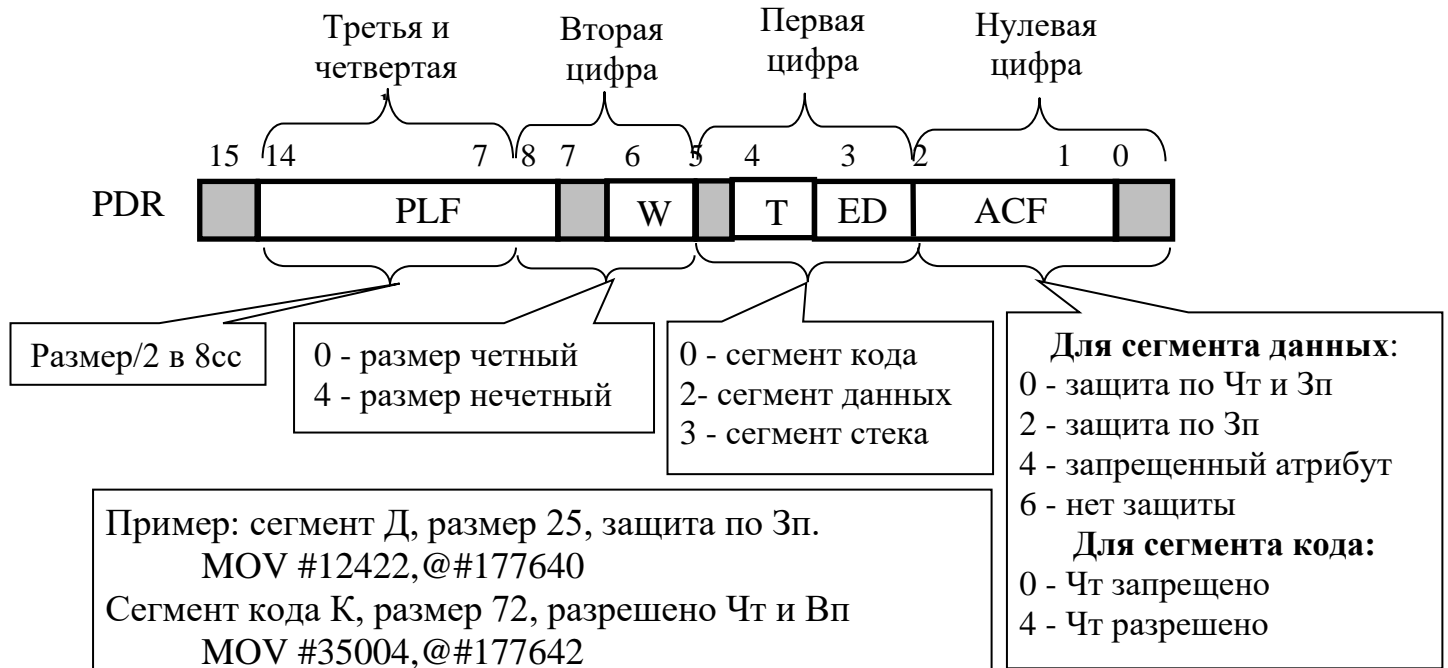
MOV #40420, @172302 ; для системного режима

.....

MOV #77426, @172316 ;

#### Рекомендации по кодированию полей регистра прав доступа:

Так как константа атрибутов защиты вводится в восьмиричной системе счисления, то рекомендуется следующая методика формирования цифр:



#### Пункт 2. Включить УУП:

MOV #1, @177572 ; SR0:=1

**Пункт 3.** Выполнить инициализацию регистров базовых адресов и прав доступа для пользовательского режима работы.

MOV #3016, @177640 ; Загрузка регистров базовых адресов

MOV #3545, @177642 ; для пользовательского режима

.....

MOV #7600, @177656 ;

MOV #16036, @177600 ; Загрузка регистров прав доступа

MOV #27000, @177602 ; для пользовательского режима

.....

MOV #37426, @177616 ;

#### Пункт 4. Установить пользовательский режим работы процессора.

MOV #14000, @177776 ; PSW:= 14XXXq, X - значение бит не  
; изменяется.

**Пункт 5.** Составить четыре подпрограммы (ПП) в соответствии с вариантом задания (значений адресов внутри сегментов N и M и видов адресаций).

Таблица 4.3 - Адреса операндов в сегменте

Вариант	N	M
21	1100	450

Таблица 4.4 – Вариант задания мнемоники и адресаций для двухадресной команды

Вариант	Команда	Адресация для операнда 1	Адресация для операнда 2
21	SUB	Индексная прямая	Автодекрементная прямая

Таблица 4.5 – Вариант задания мнемоник и адресаций для одноадресных команд

Вариант	Команда	Адресация	Команда	Адресация
21	TSTB	Абсолютная	CLR	Автодекрементная прямая

Примеры фрагментов подпрограмм.

В качестве сегмента для накопления суммы (п. 5.1) можно выбрать сегменты 0, 2 или 6, так все они являются сегментами данных для пользовательского режима работы без атрибутов защиты (обращение к сегменту по заданию по команде JSR из пользовательского режима). Выберем регистр PADR6.

ПП1: Нахождение суммы N-ых элементов сегментов:	ПП2: Вычитание элементов между соседними сегментами N и M:
CLR @#140450 ;  ADD @#1100,@# 140450 ADD @#21100,@# 140450 ..... ADD @#161100,@# 140450 RTS	MOV #1000,R1 ; Подготовка адресов MOV #20452,R2 ; операндов N0, M1 SUB 100(R1),-(R2) ; Вычитание N0-M1 ..... MOV #141000,R1 ; Подготовка адресов MOV #160452,R2 ; операндов N6, M7 SUB 100(R1),-(R2) ; Вычитание N6-M7 RTI ;
ПП3: Первая одноадресная команда:	ПП4: Вторая одноадресная команда:
TSTB @#1100 ; TSTB @#21100 ; TSTB @#41100 ; ..... TSTB @#161100 ; JMP ;	MOV #1102,R1 ; Подготовка адреса CLR -(R1) ; операнда N0 ..... MOV #161102,R1 ; Подготовка адреса CLR -(R5) ; операнда N7 RTT ;

Команда возврата из ПП зависит от варианта задания.

Из основной программы обратиться к разработанным ПП по командам:

JSR @#300

EMT #260

JSR @#320

TRAP #70

Команды JSR и TRAP не изменяют текущий режим работы процессора.

#### Пример одного из текстов ППОП:

MOV #4, R0 ; номер вектора в R0

MOV (R6),R0 ; адрес следующей команды, на которой выработано  
; прерывание: PC в R0

RTI ; возврат из прерывания

**Пункт 7.** Подпрограмму перезагрузки регистров дескрипторов предлагается выполнить самостоятельно и вновь вызвать заданную ПП по команде TRAP #70 (CLR).

Начальные адреса всех подпрограмм определяются разработчиком. После распределения адресного пространства для всех подпрограмм пп.1-7 необходимо подготовить таблицу прошивки IDT, т.е. для каждого вектора прерываний, используемого в программе, задать их начальные адреса, например:

Системный режим		Пользовательский режим	
Вектор	Адрес	Вектор	Адрес
4	120	70	200
10	130	72	220
120	140	120	250
160	150		
250	160		
260	170		

На этом домашняя подготовка заканчивается.

При лабораторных исследованиях необходимо ввести все тексты разработанных ПП инициализации, исследований пп. 5.1-5.3 и всех ППОП, заданных в таблице IDT. Открыть окно таблицы IDT в режиме "Просмотр" и ввести подготовленные начальные адреса ППОП в соответствии с номером вектора и режима работы процессора.

Установить начальные параметры лабораторной установки в окне настроек (п. "Экспериментальные исследования") и выполнить отладку разработанных программ в пошаговом режиме. В окне результатов будут фиксироваться и отображаться результаты выполнения программ только при выработке какого-либо вектора прерывания, что соответствует ведению журнала системных ошибок.

#### **Рекомендации по самоконтролю за правильностью выполнения программ**

Программы инициализации, все ППОП по векторам 4, 10, 120, 160, 250 и ПП перезагрузки регистров дескрипторов должны выполняться без выдачи сообщений об ошибках и векторов прерываний, так как они относятся к ОС и не содержат ошибок по доступу к сегментам.

При выполнении пп. 5, 6, 8 и 9 в поле вектор прерывания (рисунок 7) может появляться какой-либо номер вектора прерывания. На выполнении данной команды необходимо акцентировать внимание, выявить причину прерывания и ознакомиться с принципами работы системы защиты памяти, т.е. просмотреть содержимое ячейки ОЗУ и убедиться, что ее содержимое, например, при защите по записи в сегмент, не изменилось, либо заблокировано выполнение командного цикла процессора, связанное с доступом к ОП по чтению и записи либо по другой причине прерывания. В исходном состоянии в ячейках ОП записан их адрес. Либо достаточно проанализировать информацию в поле <ОЗУ> в выходном окне ЛУ. Цифровое значение первого, второго операндов и результата означает выполнение доступа к ОП, а значение <xxxxxx> - блокирование обращения к памяти либо по чтению, либо по записи, либо по другой причине (вектору прерывания).

В пп. 8 и 9 предлагается составить две подпрограммы для сегментов данных и сегмента кода с выработкой всех возможных причин и кодов ошибок без изменения содержимого регистров дескрипторов, т.е. путем подбора типа команды, номера сегмента и логического адреса ячейки ОП, кроме выработки векторов 160 и 250 для несуществующих атрибута защиты сегмента данных и режима работы процессора. Для выработки этих векторов предварительно необходимо изменить содержимое базового адреса одного из дескрипторов (вектор 160), атрибут защиты сегмента данных (вектор 250 для несуществующего атрибута защиты) или режим работы процессора в PSW (вектор 250 для несуществующего режима работы процессора).