

Федеральное государственное автономное образовательное учреждение  
высшего образования «Национальный исследовательский университет  
ИТМО»

*Факультет программной инженерии и компьютерной техники*  
*Направление подготовки: 09.03.04 – Программная инженерия,*  
*Системное и прикладное программное обеспечение.*  
*Дисциплина «Основы профессиональной деятельности»*

**Отчет**  
**По лабораторной работе №3**  
**Выполнение циклических программ**

**Вариант № 1004**

Выполнил:  
Молчанов Фёдор Денисович

Группа: Р3113

Преподаватель:  
Блохина Елена Николаевна

Г. Санкт-Петербург, 2024 г.

## Оглавление

Текст исходной программы .....	3
Описание программы .....	4
Трассировка_Допуск .....	6
Трассировка_ПолученныеДанные .....	7
Выводы .....	7

## Текст исходной программы

Адрес	Код команды	Мнемоника	Комментарий
0x596	+ AF80	LD #0x80	Загрузить 80 в AC= FF80
0x597	0740	DEC	Уменьшить AC на 1 = FF7F
0x598	0680	SWAB	Обмен старшего и мл. байта AC. AC = 7FFF
0x599	EEFB	ST (IP – 5)	Запись AC в ячейку 59A – 5. 595 = 7FFF
0x59A	AF04	LD #4	AC = 4
0x59B	EEF8	ST (IP – 8)	Запись AC в ячейку 59C – 8 = 594
0x59C	4EF5	ADD (IP – 0xB)	Добавить к AC операнд по адресу 59D – B = 592. AC= 4 + 05A8 = 05AC
0x59D	EEF5	ST (IP - 0xB)	Запись AC в ячейку 59E - B = 593
0x59E	ABF4	LD -(IP - 0xB)	В ячейке 593 указатель на верхнюю (или следующую) ячейку массива (в начале программы это указатель на 5AC-1 = 5AB). Из этой ячейки происходит запись в AC.
0x59F	0480	ROR	AC и C сдвигаются вправо. Деление на два.
0x5A0	F404	BCS (IP + 4)	Если C == 1, то IP + 4 -> IP(5A5). Проверка на чётность. Если нечет, то в loop. Иначе – след. Команда (*2)
0x5A1	0400	ROL	AC и C сдвигается влево. Умножение на два. Восстановили число после ROR.
0x5A2	7EF2	CMP (IP – 0xE)	Установление флагов по результату AC – (595).
0x5A3	F901	BGE (IP+1)	Сравнение N и V. If AC >= 595 goto 5A4 + 1 = 5A5(loop)
0x5A4	EEF0	ST (IP – 0x10)	Запись AC в ячейку 5A4 – 0x10 = 595
0x5A5	8594	LOOP 0x594	Цикл по 594 ячейке. Если там не отрицательное число, то выполняем след. Команду. Иначе выполняем команду IP + 1 + 1 = 5A7
0x5A6	CEF7	JUMP (IP – 9)	5A7 – 9 = 59E -> IP (переход к действию)
0x5A7	0100	HLT	Остановка программы

## Описание программы

**Назначение программы:** программа находит минимальный чётный элемент одномерного массива размера 4.

### Описание и назначение исходных данных:

Адрес	Данные	Мнемоника	Комментарий
592	07A8	A	Хранит адрес начала массива
593	A000	M	Указатель на ячейку в массиве (впоследствии – последнюю итерируемую)
594	E000	N	Счётчик для loop (размер массива)
595	E000	R	Значение для сравнения (результат)
В 10-формате			
5A8 ;7a8	-352 = FEA0	M[0]	Массив
5A9 ;7a9	110 = 006E	M[1]	
5AA ;7aA	201 = 00C9	M[2]	
5AB ;7aB	0001 = 0001	M[3]	

### Область представления

- Объяснение следующих мнемоник описано в секции «описание и назначение исходных данных».

A – 11 разрядное беззнаковое число  $[0; 2^{11}-1]$

M – 11 разрядное беззнаковое число  $[0; 2^{11}-1]$

M[0], M[1], M[2], M[3], R - 16-разрядные знаковые числа.  $[-2^{15}; 2^{15} - 1]$

N интерпретируется как константа, не превышающая размер массива = 4  $[0 \leq N \leq 4]$

### ОДЗ переменных и результата

Элементы массива  $[-2^{15}; 2^{15} - 1]$

Результат  $[-2^{15}; 2^{15} - 1]$  (только чётные числа). Если четных элементов в массиве нет, то результат равен  $2^{15} - 1$ .

Счётчик  $[0; 4]$

Указатели на ячейки памяти(A,M) [0; 0x592-4) или [0x596; 2<sup>11</sup>-5] (до программы и после программы)

Ячейка с адресом первого элемента массива: в завис-и от расположения массива. Такое же одз, как и массива в соотв-ии с случаем.

**Расположение в памяти БЭВМ программы, исходных данных и результатов:**

0x592 – указатель на начало массива

0x593 - указатель текущего элемента массива

0x594 - счетчик циклов

0x595 - результат

0x596 – 0x5A7 - команды программы

0x5A8 – 0x5AB - элементы массива

0x596 – адрес первой команды

0x5A7- адрес последней команды

## Трассировка\_Допуск

Выполняемая команда		Содержимое регистров процессора после выполнения команды								Ячейка, содержимое которой изменилось после выполнения команды	
Адр	Код	IP	CR	AR	DR	SP	BR	AC	NZVC	Адр	Новый код
596	AF80	597	AF80	596	FF80	0	FF80	FF80	1000		
597	0740	598	0740	597	0740	0	0597	FF7F	1001		
598	0680	599	0680	598	0680	0	0598	7FFF	0001		
599	EEFB	59A	EEFB	595	7FFF	0	FFFB	7FFF	0001	595	7FFF
59A	AF04	59B	AF04	59A	0004	0	0004	0004	0001		
59B	EEF8	59C	EEF8	594	0004	0	FFF8	0004	0001	594	0004
59C	4EF5	59D	4EF5	592	05A8	0	FFF5	05AC	0000		
59D	EEF5	59E	EEF5	593	05AC	0	FFF5	05AC	0000	593	05AC
59E	ABF4	59F	ABF4	5AB	0000	0	FFF4	0000	0100	593	05AB
59F	0480	5A0	0480	59F	0480	0	059F	0000	0100		
5A0	F404	5A1	F404	5A0	F404	0	05A0	0000	0100		
5A1	0400	5A2	0400	5A1	400	0	05A1	0000	0100		
5A2	7EF2	5A3	7EF2	595	7FFF	0	FFF2	0000	1000		
5A3	F901	5A4	F901	5A3	F901	0	05A3	0000	1000		
5A4	EEF0	5A5	EEF0	595	0000	0	FFF0	0000	1000	595	0000
5A5	8594	5A6	8594	594	0003	0	0002	0000	1000	594	0003
5A6	CEF7	59E	CEF7	5A6	059E	0	FFF7	0000	1000		
59E	ABF4	59F	ABF4	5AA	0201	0	FFF4	0201	0000	593	05AA
59F	0480	5A0	0480	59F	0480	0	059F	0100	0011		
5A0	F404	5A5	F404	5A0	F404	0	0004	0100	0011		
5A5	8594	5A6	8594	594	0002	0	0001	0100	0011	594	0002
5A6	CEF7	59E	CEF7	5A6	059E	0	FFF7	0100	0011		
59E	ABF4	59F	ABF4	5A9	1101	0	FFF4	1101	0001	593	05A9
59F	0480	5A0	0480	59F	0480	0	059F	8880	1001		
5A0	F404	5A5	F404	5A0	F404	0	0004	8880	1001		
5A5	8594	5A6	8594	594	0001	0	0000	8880	1001	594	0001
5A6	CEF7	59E	CEF7	5A6	059E	0	FFF7	8880	1001		
59E	ABF4	59F	ABF4	5A8	35A2	0	FFF4	35A2	0001	593	05A8
59F	0480	5A0	0480	59F	0480	0	059F	9AD1	1010		
5A0	F404	5A1	F404	5A0	F404	0	05A0	9AD1	1010		
5A1	0400	5A2	0400	5A1	0400	0	05A1	35A2	0011		
5A2	7EF2	5A3	7EF2	595	0000	0	FFF2	35A2	0001		
5A3	F901	5A5	F901	5A3	F901	0	0001	35A2	0001		
5A5	8594	5A7	8594	594	0000	0	FFFF	35A2	0001	594	0000
5A7	0100	5A8	0100	5A7	0100	0	05A7	35A2	0001		

## Трассировка\_ПолученныеДанные

Выполняемая команда		Содержимое регистров процессора после выполнения команды								Ячейка, содержимое которой изменилось после выполнения команды	
Адр	Код	IP	CR	AR	DR	SP	BR	AC	NZVC	Адр	Новый код
596	AF80	597	AF80	596	FF80	0	FF80	FF80	1000		
597	0740	598	0740	597	0740	0	597	FF7F	1001		
598	0680	599	0680	598	0680	0	598	7FFF	0001		
599	EEFB	59A	EEFB	595	7FFF	0	FFFB	7FFF	0001	595	7FFF
59A	AF04	59B	AF04	59A	0004	0	0004	0004	0001		
59B	EEF8	59C	EEF8	594	0004	0	FFF8	0004	0001	594	0004
59C	4EF5	59D	4EF5	592	07A8	0	FFF5	07AC	0000		
59D	EEF5	59E	EEF5	593	07AC	0	FFF5	07AC	0000	593	07AC
59E	ABF4	59F	ABF4	7AB	0001	0	FFF4	0001	0000	593	07AB
59F	0480	5A0	0480	59F	0480	0	059F	0000	0111		
5A0	F404	5A5	F404	5A0	F404	0	0004	0000	0111		
5A5	8594	5A6	8594	594	0003	0	0002	0000	0111	594	0003
5A6	CEF7	59E	CEF7	5A6	059E	0	FFF7	0000	0111		
59E	ABF4	59F	ABF4	7AA	00C9	0	FFF4	00C9	0001	593	07AA
59F	0480	5A0	0480	59F	0480	0	059F	8064	1001		
5A0	F404	5A5	F404	5A0	F404	0	0004	8064	1001		
5A5	8594	5A6	8594	594	0002	0	0001	8064	1001	594	0002
5A6	CEF7	59E	CEF7	5A6	059E	0	FFF7	8064	1001		
59E	ABF4	59F	ABF4	7A9	006E	0	FFF4	006E	1001	593	07A9
59F	0480	5A0	0480	59F	0480	0	059F	8037	1010		
5A0	F404	5A1	F404	5A0	F404	0	05A0	8037	1010		
5A1	0400	5A2	0400	5A1	0400	0	05A1	006E	0011		
5A2	7EF2	5A3	7EF2	595	7FFF	0	FFF2	006E	1000		
5A3	F901	5A4	F901	5A3	F901	0	05A3	006E	1000		
5A4	EEF0	5A5	EEF0	595	006E	0	FFF0	006E	1000	595	006E
5A5	8594	5A6	8594	594	0001	0	0000	006E	1000	594	0001
5A6	CEF7	59E	CEF7	5A6	059E	0	FFF7	006E	1000		
59E	ABF4	59F	ABF4	7A8	FEA0	0	FFF4	FEA0	1000	593	07A8
59F	0480	5A0	0480	59F	480	0	059F	7F50	0000		
5A0	F404	5A1	F404	5A0	F404	0	05A0	7F50	0000		
5A1	0400	5A2	0400	5A1	0400	0	05A1	FEA0	1010		
5A2	7EF2	5A3	7EF2	595	006E	0	FFF2	FEA0	1001		
5A3	F901	5A4	F901	5A3	F901	0	05A3	FEA0	1001		
5A4	EEF0	5A5	EEF0	595	FEA0	0	FFF0	FEA0	1001	595	FEA0
5A5	8594	5A7	8594	594	0000	0	FFFF	FEA0	1001	594	0000
5A7	0100	5A8	0100	5A7	100	0	05A7	FEA0	1001		

## Выводы

Во время выполнения работы я научился работать в БЭВМ с массивами, ветвлением и циклами. Я изучил прямую и косвенную адресации.