

Федеральное государственное автономное образовательное учреждение  
высшего образования “Национальный исследовательский университет ИТМО”

Факультет Программной Инженерии И Компьютерной Техники

Лабораторная работа №3

Выполнение циклических программ

Вариант 8001

Выполнила:

Абдуллаева София Улугбековна

Группа Р3108

Проверил:

Вербовой Александр Александрович

Санкт-Петербург 2025

## Оглавление

Задание.....	3
Текст исходной программы .....	3
Назначение программы.....	4
Область представления .....	4
Область определения .....	4
Трассировка .....	5
Вывод .....	6

## Задание

По выданному преподавателем варианту восстановить текст заданного варианта программы, определить предназначение и составить описание программы, определить область представления и область допустимых значений исходных данных и результата, выполнить трассировку программы.

Введите номер варианта

57F:	0592		58D:	4EF4
580:	0200		58E:	EEF3
581:	4000		58F:	8581
582:	E000		590:	CEF8
583: +	0200		591:	0100
584:	EEFD		592:	257F
585:	AF03		593:	E58C
586:	EEFA		594:	0001
587:	4EF7			
588:	EEF7			
589:	ABF6			
58A:	0480			
58B:	F403			
58C:	0400			

## Текст исходной программы

Адрес	Код команды	Мнемоника	Описание
57F	0592	first_element	Адрес первого элемента
580	0200	current_element	Адрес текущего элемента
581	4000	num_of_elements	Размер массива
582	E000	result	Результат
583	0200	CLA	Очистка аккумулятора: 0 => AC
584	EEFD	ST(IP-3)	Прямое относительное сохранение AC => (582)
585	AF03	LD #03	Прямая загрузка 0003 => AC
586	EEFA	ST(IP-6)	Прямое относительное сохранение AC => (581)
587	4EF7	ADD(IP-9)	Прямое относительное сложение ячейки памяти (57F) с AC, в AC записать результат: (57F) + AC => AC
588	EEF7	ST(IP-9)	Прямое относительное сохранение AC => (580)

589	ABF6	LD -(IP-10)	Косвенная автодекрементная загрузка: значение адреса в ячейке памяти (580) – 1, в AC записывается значение ячейки по этому адресу
58A	0480	ROR	Циклический сдвиг вправо $AC_0 \Rightarrow C, C \Rightarrow AC_{15}$
58B	F403	BCS(IP+3)	Если $C == 1$ , то $IP+3+1 \Rightarrow IP$
58C	0400	ROL	Циклический сдвиг влево $AC_{15} \Rightarrow C, C \Rightarrow AC_0$
58D	4EF4	ADD(IP-12)	Прямое относительное сложение ячейки памяти (582) с AC, в AC записать результат: $(582) + AC \Rightarrow AC$
58E	EEF3	ST(IP-13)	Прямое относительное сохранение $AC \Rightarrow (582)$
58F	8581	LOOP 581	Запись в ячейку (581) её значение, уменьшенное на 1. Если значение ячейки $(581) \leq 0$ , то $IP+1 \Rightarrow IP$
590	CEF8	JUMP(IP-8)	Прямой относительный прыжок $IP-8+1 \Rightarrow IP$
591	0100	HLT	Останов
592	257F	-	Элементы массива
593	E58C	-	
594	0001	-	

## Назначение программы

Программа находит сумму чётных элементов массива

## Область представления

Элементы массива – 16 разрядные знаковые числа

first\_element, current\_element – 11 разрядные беззнаковые числа, адреса в БЭВМ

result – 16 разрядное знаковое число

num\_of\_elements – 8 разрядное знаковое число, так как значение в AC устанавливается при прямой загрузке, КОП и режим адресации занимают старший байт

## Область определения

$-2^{15} \leq \text{элемент массива} \leq 2^{15} - 1/\text{num\_of\_elements}$

$-2^{15} \leq \text{result} \leq 2^{15} - 1$

$$1 \leq \text{num\_of\_elements} \leq 2^7 - 1$$

current\_element  $\in$  [0; 57F – num\_of\_elements] или current\_element  $\in$  [592; 7FF]

### **Расположение программы, исходных данных и результата в памяти БЭВМ**

57F, 580, 581, 592, 593, 594 – исходные данные

582 – результат

583 – 591 – инструкции

### **Адреса первой и последней выполняемой инструкции программы**

583 – адрес первой команды

591 – адрес последней команды

### **Трассировка**

x[0] = 3032

x[1] = -1234, в доп.коде EDCC

x[2] = 23

x[3] = 1234

Выполняемая команда		Содержание регистров в процессоре после выполнения команды								Ячейка, содержащее которой изменилось после выполнения команды	
Адрес	Код	IP	CR	AR	DR	SP	BR	AC	NZVC	Адрес	Новый код
583	0200	583	0000	000	0000	000	0000	0000	0100		
583	0200	584	0200	583	0200	000	0583	0000	0100		
584	EEFD	585	EEFD	582	0000	000	FFFD	0000	0100	582	0000
585	AF04	586	AF04	585	0004	000	0004	0004	0000		
586	EEFA	587	EEFA	581	0004	000	FFFA	0004	0000	581	0004
587	4EF7	588	4EF7	57F	0592	000	FFF7	0596	0000		
588	EEF7	589	EEF7	580	0596	000	FFF7	0596	0000	580	0596
589	ABF6	58A	ABF6	594	0001	000	FFF6	3032	0000	580	0595
58A	0480	58B	0480	58A	0480	000	058A	1819	0000		
58B	F403	58C	F403	58B	F403	000	058B	1819	0000		
58C	0400	58D	0400	58C	0400	000	058C	3032	0000		
58D	4EF4	58E	4EF4	582	0000	000	FFF4	3032	000		
58E	EEF3	58F	EEF3	582	3032	000	FFF3	3032	0000	582	3032

58F	8581	590	8581	590	0003	000	0002	3032	0000	581	0003
590	CEF8	589	CEF8	590	0589	000	FFF8	3032	0000		
589	ABF6	58A	ABF6	594	EDCC	000	FFF6	EDCC	1000	580	0594
58A	0480	58B	0480	58A	0480	000	058A	76E6	0000		
58B	F403	58C	F403	58B	F403	000	058B	76E6	0000		
58C	0400	58D	0400	58C	0400	000	058C	EDCC	1010		
58D	4EF4	58E	4EF4	582	3032	000	FFF4	1DFE	0001		
58E	ABF6	58A	ABF6	582	1DFE	000	FFF3	1DFE	0001	582	1DFE
58F	8581	590	8581	581	0002	000	0001	1DFE	0001	581	0002
590	CEF8	58F	CEF8	590	0589	000	FFF8	1DFE	0001		
589	ABF6	58A	ABF6	593	0023	000	FFF6	0023	0001	580	0593
58A	0480	58B	0480	58A	0480	000	058A	8011	1001		
58B	F403	58F	F403	58B	F403	000	0003	8011	1001		
58F	8581	590	8581	581	0001	000	0000	8011	1001	581	0001
590	CEF8	589	CEF8	590	0589	000	FFF8	8011	1001		
589	ABF6	58A	ABF6	592	1234	000	FFF6	1234	0001	580	0592
58A	0480	58B	0480	58A	0480	000	058A	891A	1010		
58B	F403	58C	F403	58B	F403	000	058B	891A	1010		
58C	0400	58D	0400	58C	0400	000	058C	1234	0011		
58D	4EF4	58E	4EF4	582	1DFE	000	FFF4	3032	0000		
58E	EEF3	58F	EEF3	582	3032	000	FFF3	3032	0000	582	3032
58F	8581	591	8581	581	0000	000	FFFF	3032	0000	581	0000
591	0100	592	0100	591	0100	000	0591	3032	0000		

## Вывод

В процессе выполнения лабораторной работы я поняла, как работать с циклическими программами, изучила режимы адресации, команды ветвления и команды LOOP и JUMP. Также научилась работать с массивами.