

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

Факультет программной инженерии и компьютерной техники

# **Отчет**

## **По лабораторной работе №3 «Исследование работы БЭВМ»**

По дисциплине «Основы профессиональной деятельности»

Вариант 1371

Выполнил: Воронов Г. А., группа Р3116

Преподаватель: Остапенко Ольга

Денисовна

Санкт-Петербург

2024

## Оглавление

Текст задания.

Текст программы.

Описание программы.

Таблица трассировки

Вариант с меньшим числом команд.

Вывод.

## Текст задания.

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.

563:	0577		571:	7EF4
564:	A000		572:	F801
565:	E000		573:	EEF2
566:	E000		574:	8565
567:	+ AF40		575:	CEF9
568:	0680		576:	0100
569:	0500		577:	0740
56A:	EEFB		578:	1200
56B:	AF05		579:	0000
56C:	EEF8		57A:	1800
56D:	4EF5		57B:	2573
56E:	EEF5			
56F:	ABF4			
570:	F003			

## Текст программы.

Адрес	Код команды	Мнемоника	Комментарий	Описание
563	0577		Переменная A	Ссылка на начало массива
564	A000		Переменная X	Ссылка на очередной элемент массива = A+Y
565	E000		Переменная Y	Счётчик цикла. Изначально равен размеру массива = 5.
566	E000		Переменная Z	Результат
567	AF40	LD #40	Прямая загрузка AC = 0040	
568	0680	SWAB	Обмен старшего и младшего байта AC	AC = 4000
569	0500	ASL	Арифметический сдвиг AC влево. AC <sub>15</sub> → CF	AC = 8000. Умножение на 2. Переполнение.
56A	EEFB	ST 566	Прямая относительная ST IP - 5 = ST 566 Z = AC	Z = 8000. В переменной результата находится минимальное число из всех возможных: -32768
56B	AF05	LD #05	Прямая загрузка	AC = 0005
56C	EEF8	ST 565	Прямая относительная ST IP - 8 = ST 565	Y = 0005. В переменной Y содержится размер массива.
56D	4EF5	ADD 563	Прямая относительная ADD IP - 11 <sub>10</sub> = ADD 563	AC = 5 + A
56E	EEF5	ST 564	Прямая относительная ST IP - 11 <sub>10</sub> = ST 564	X = AC. В переменной X содержится ссылка на очередной элемент массива. Изначально указывает на конец массива.
56F	ABF4	LD -(564)	Косвенная автодекрементная	X = X - 1 AC = [X] Загрузка очередного элемента. Итерирование с конца.

570	F003	BEQ	Переход на 574, если равенство	Если элемент = 0, то пропустить
571	7EF4	CMP 566	Прямая относительная. Сравить AC и ячейку 566 $IP - 12_{10} = 566$	$AC - Z \rightarrow NZVC$ Сравнение текущего с промежуточным максимальным.
572	F801	BLT	Переход на 574, если меньше	Если очередной элемент меньше максимального, то пропустить
573	EEF2	ST 566	Прямая относительная. $IP - 14_{10} = 566$	$Z = [X]$ Иначе — обновить максимальный
574	8565	LOOP	$M(565) - 1 \rightarrow M(565)$ . Если $M(565) \leq 0$ , то $IP + 1 \rightarrow IP$	Уменьшить счётчик Y. Проверить на $\leq 0$
575	CEF9	JUMP 56F	Прямая относительная. Прыжок на $IP - 7 = 56F$	
576	0100	HLT	Завершение программы	
577	0740			Элементы массива
578	1200			
579	0000			
57A	1800			
57B	2573			

## Описание программы.

Программа находит максимальный элемент массива, состоящего из 5 элементов и сохраняет результат.

1. Командами 567-56A программа загружает -32768 в ячейку 566 (в переменную Z).
2. Далее (56B-56C) программа устанавливает счётчик цикла равным 5 ( $Y = 5$ ) и
3. командами (56D-56E) переменную X на конец массива ( $X = A + Y = 577 + 5 = 57C$ ).
4. 56F-575 — тело цикла, в котором выполняются следующие действия:
  - 56F: загрузить очередной элемент массива в аккумулятор
  - 570: IF AC == 0: continue
  - 571-572: IF AC < Z: continue
  - 573: Z := AC

### Область представления:

- A, X – 11-ти разрядные числа, адрес БЭВМ
- Y – счётчик цикла, 16-ти разрядное беззнаковое число.
- элементы массива – 16-ти разрядные знаковые целые числа
- Z — результат, 16-ти разрядные знаковые целые числа

### Область допустимых значений:

$A \in [0; 0x563 - 5] \cup [0x577; 0x7FF - 5]$	// задаётся пользователем
Элементы массива $\in [-2^{15}; 2^{15}-1]$	// задаются пользователем
$X \in [A; A + Y]$	// указатель на очередной элемент
$Z \in [-2^{15}; 2^{15}-1]$	// результат

### Расположение в памяти БЭВМ

**исходных данных:** 563, [577; 57B]

**результатов:** 566

**инструкций:** [567; 576]

**Адреса первой и последней выполняемой инструкции программы:** 567 и 576

Таблица трассировки

A = 0x652

array: [0x1000, 0x5252, -1, -2048, 0]

Выполняема я команда		Содержимое регистров процессора после выполнения команды									Ячейка, содержимое которой изменилось после выполнения команды	
Адрес	Код	IP	CR	AR	DR	SP	BR	AC	PS	NZVC	Адрес	Новое значени
567	AF40	568	AF40	567	40	0	40	40	0	0		
568	680	569	680	568	680	0	568	4000	0	0		
569	500	56A	500	569	4000	0	569	8000	00A	1010		
56A	EEFB	56B	EEFB	566	8000	0	FFFB	8000	00A	1010	566	8000
56B	AF05	56C	AF05	56B	5	0	5	5	0	0		
56C	EEF8	56D	EEF8	565	5	0	FFF8	5	0	0	565	5
56D	4EF5	56E	4EF5	563	652	0	FFF5	657	0	0		
56E	EEF5	56F	EEF5	564	657	0	FFF5	657	0	0	564	657
56F	ABF4	570	ABF4	656	0	0	FFF4	0	4	100	564	656
570	F003	574	F003	570	F003	0	3	0	4	100		
574	8565	575	8565	565	4	0	3	0	4	100	565	4
575	CEF9	56F	CEF9	575	056F	0	FFF9	0	4	100		
56F	ABF4	570	ABF4	655	F800	0	FFF4	F800	8	1000	564	655
570	F003	571	F003	570	F003	0	570	F800	8	1000		
571	7EF4	572	7EF4	566	8000	0	FFF4	F800	1	1		
572	F801	573	F801	572	F801	0	572	F800	1	1		
573	EEF2	574	EEF2	566	F800	0	FFF2	F800	1	1	566	F800

574	8565	575	8565	565	3	0	2	F800	1	1	565	3
575	CEF9	56F	CEF9	575	056F	0	FFF9	F800	1	1		
56F	ABF4	570	ABF4	654	FFFF	0	FFF4	FFFF	9	1001	564	654
570	F003	571	F003	570	F003	0	570	FFFF	9	1001		
571	7EF4	572	7EF4	566	F800	0	FFF4	FFFF	1	1		
572	F801	573	F801	572	F801	0	572	FFFF	1	1		
573	EEF2	574	EEF2	566	FFFF	0	FFF2	FFFF	1	1	566	FFFF
574	8565	575	8565	565	2	0	1	FFFF	1	1	565	2
575	CEF9	56F	CEF9	575	056F	0	FFF9	FFFF	1	1		
56F	ABF4	570	ABF4	653	5252	0	FFF4	5252	1	1	564	653
570	F003	571	F003	570	F003	0	570	5252	1	1		
571	7EF4	572	7EF4	566	FFFF	0	FFF4	5252	0	0		
572	F801	573	F801	572	F801	0	572	5252	0	0		
573	EEF2	574	EEF2	566	5252	0	FFF2	5252	0	0	566	5252
574	8565	575	8565	565	1	0	0	5252	0	0	565	1
575	CEF9	56F	CEF9	575	056F	0	FFF9	5252	0	0		
56F	ABF4	570	ABF4	652	1000	0	FFF4	1000	0	0	564	652
570	F003	571	F003	570	F003	0	570	1000	0	0		
571	7EF4	572	7EF4	566	5252	0	FFF4	1000	8	1000		
572	F801	574	F801	572	F801	0	1	1000	8	1000		
574	8565	576	8565	565	0	0	FFFF	1000	8	1000	565	0
576	100	577	100	576	100	0	576	1000	8	1000		



## **Вывод.**

Во время выполнения лабораторной работы я научился работать в БЭВМ с массивами, ветвлением и циклами. Я изучил прямую и косвенную адресацию и цикл выполнения таких команд, как LOOP и JUMP.