

**Министр науки и высшего образования Российской
Федерации**

**Федеральное государственное автономное
образовательное учреждение высшего образования**

**«Национальный исследовательский университет
ИТМО»**

**Факультет информационных технологий и
программирования**

Лабораторная работа № 3

Исследование работы БЭВМ при выполнении циклических программ

Выполнил студент группы № М3101

Михеев Артем Романович

Подпись:



Проверил:

Бабич Мария Сергеевна

Санкт-Петербург
2020

Цель работы

Изучение способов организации циклических программ и исследование порядка функционирования ЭВМ при выполнении циклических программ.

Задание, вариант 6

Восстановить текст заданного варианта программы, составить описание программы, после чего занести программу в память БЭВМ и заполнить таблицу трассировки, выполняя программу по командам.

Вариант 6:

(первая команда помечена плюсиком)

Адрес	Коды команд
00A	0000
00B	0000
00C	0000
00D	0010
00E	0000
00F	0000
010	0000
011	0010
012	0000
013	0707
014	0000
015	FFFC
016	+F200
017	480D
018	B01A
019	C01D
01A	F800
01B	4011
01C	3011
01D	0015
01E	C016
01F	F000

Решение

1. Восстановим исходный код программы начиная с первой команды по адресу 016:
16: CLA # Очистить аккумулятор
17: ADD (000D) # Прибавить к аккумулятору значение по адр., записанному в ячейке 000D, Увеличить значение в 000D на 1
18: BEQ 001A # Если аккумулятор равен 0, то продолжить упр. с 001A
19: BR 001D # Продолжить упр. с 001D
1A: INC # Увеличить аккумулятор на 1
1B: ADD 0011 # Прибавить к аккумулятору значение в яч. 0011
1C: MOV 0011 # Выставить в яч. 0011 значение из аккумулятора
1D: ISZ 0015 # Прибавить к знач. в яч. 0015 единицу, если ≥ 0 , то пропустить 1 ячейку
1E: BR 0016 # Продолжить упр. с 0016

1F: HLT # Остановить программу
 Для удобства можем переписать эту программу вот так:

```
ORG 000D
POINTER: WORD 0010
```

```
ORG 0010
WORD 0000
RESULT: WORD 0010
WORD 0000
WORD 0707
```

```
ORG 0015
COUNTER: WORD FFFC
```

```
ORG 0016
BEGIN:
MAIN_LOOP:
  CLA
  ADD (POINTER)
  BEQ IF_ZERO
  BR ELSE
```

```
IF_ZERO:
  INC
  ADD RESULT
  MOV RESULT
ELSE:
  ISZ COUNTER
  BR MAIN_LOOP
```

```
HLT
```

2. Теперь составим таблицу трассировки, исполнив программу по-командно

Выполняемая команда		Содержимое регистров процессора после выполнения команды.						Ячейка, содержим. которой изменилось после вып. Программы	
Адрес	Код	СК	РА	РК	РД	А	С	Адрес	Новый код
016	F200	017	016	F200	F200	0000	0	-	-
017	480D	018	010	480D	0000	0000	0	00D	011
018	B01A	01A	018	B01A	B01A	0000	0	-	-
01A	F800	01B	01A	F800	F800	0001	0	-	-
01B	4011	01C	011	4011	0010	0011	0	-	-
01C	3011	01D	011	3011	0011	0011	0	011	0011
01D	0015	01E	015	0015	FFFD	0011	0	015	FFFD
01E	C016	016	01E	C016	C016	0011	0	-	-
016	F200	017	016	F200	F200	0000	0	-	-
017	480D	018	011	480D	0011	0011	0	00D	012
018	B01A	019	018	B01A	B01A	0011	0	-	-

019	C01D	01D	019	C01D	C01D	0011	0	-	-
01D	0015	01E	015	0015	FFFE	0011	0	015	FFFE
01E	C016	016	01E	C016	C016	0011	0	-	-
016	F200	017	016	F200	F200	0000	0	-	-
017	480D	018	010	480D	0000	0000	0	00D	013
018	B01A	01A	018	B01A	B01A	0000	0	-	-
01A	F800	01B	01A	F800	F800	0001	0	-	-
01B	4011	01C	011	4011	0010	0012	0	-	-
01C	3011	01D	011	3011	0013	0012	0	011	0012
01D	0015	01E	015	0015	FFFF	0012	0	015	FFFF
01E	C016	016	01E	C016	C016	0012	0	-	-
016	F200	017	016	F200	F200	0000	0	-	-
017	480D	018	010	480D	0000	0707	0	00D	0014
018	B01A	019	018	B01A	B01A	0707	0	-	-
019	C01D	01D	019	C01D	C01D	0707	0	-	-
01D	0015	01F	015	0015	0000	0013	0	015	0000
01F	F000	020	01F	F000	F000	0013	0	-	-

3. Составим описание программы. Как по исходному коду, так и по трассировке можно сделать вывод, что программа подсчитывает кол-во нулевых элементов в массиве, расположенном в ячейках 0010-0013. Для задания размера массива используется ячейка 0015, в которой записано -N (N = кол-во элементов начиная с ячейки 0010). Результат записывается в ячейку 0011, которая также является элементом массива, к тому же в ней изначально хранится 0010, из-за чего результатом исполнения будет число $0 \times 10 + X$ где X это количество нулевых элементов.

Выводы

При выполнении лабораторной работы появилась необходимость еще раз вспомнить принципы работы косвенной адресации, а также команды БЭВМ для переходов, с помощью которых в программе был реализован цикл. Благодаря этому далее и самому будет проще писать такие конструкции для более сложных программ, для которых нужен даже не один цикл.