

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
“НАЦИОНАЛЬНЫЙ ИСЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО”**

Факультет Программной Инженерии и Компьютерной Техники

Дисциплина: «Основы Профессиональной Деятельности»

ОТЧЁТ

по лабораторной работе №7
Вариант №2387

Выполнил:

Студент группы Р3131

Родионов Максим Артемович

Проверила:

Остапенко Ольга Денисовна

Санкт-Петербург 2025

Оглавление

Задание	3
Выполнение	3
Тестовая программа на ассемблере	4
Методика проверки	6
Вывод.....	7

Задание

Синтезировать цикл исполнения для выданных преподавателем команд. Разработать тестовые программы, которые проверяют каждую из синтезированных команд. Загрузить в микропрограммную память БЭВМ циклы исполнения синтезированных команд, загрузить в основную память БЭВМ тестовые программы. Проверить и отладить разработанные тестовые программы и микропрограммы.

1. BGC ADDR - переход к команде, расположенной по адресу, на которую указывает адресная часть команды, если аккумулятор содержит число, большее чем 4095
2. Код операции - FDXX
3. Тестовая программа должна начинаться с адреса 01B6₁₆

Выполнение

Адрес МП	Микрокоманда	Описание	Комментарий
E0	80C4084002	if CR(11) = 0 then GOTO INT @ C4	Проверка на то, что мы работаем с FD
E1	80C4044002	if CR(10) = 0 then GOTO INT @ C4	
E2	81C4024002	if CR(9) = 1 then GOTO INT @ C4	
E3	80C4014002	if CR(8) = 0 then GOTO INT @ C4	
E4	81C4804010	if AC(15) = 1 then GOTO INT @ C4	Проверка на то, чтобы число было неотрицательным
E5	81F8404010	if AC(14) = 1 then GOTO E8	Проверка на то, чтобы число в аккумуляторе было больше, чем 4095
E6	81F8204010	if AC(13) = 1 then GOTO E8	
E7	80C4104010	if AC(12) = 0 then GOTO INT @ C4	
E8	805C101040	GOTO BR @ 5C	Переход на цикл исполнения безусловного перехода (IP + XX)

Тестовая программа на ассемблере

ORG 0x1AF

test1_number: WORD 0x0FFF ;4095

test2_number: WORD 0x1000 ;4096

test1_result: WORD 0

test2_result: WORD 0

ORG 0x1B6

START: LD \$test1_number

WORD 0xFD03 ; if command works wrong it will go to test_exception

right_execution:

LD #0x1

ST \$test1_result

JUMP second_test

test_exception:

CLA

ST \$test1_result

JUMP second_test

second_test: LD \$test2_number

WORD 0xFD03 ; if command works right it will go to the
second_part

second_right_execution:

CLA

ST \$test2_result

JUMP check_status

second_part:

LD #0x1

ST \$test2_result

CLA

```
JUMP check_status
```

```
check_status:
```

```
LD $test1_result
```

```
AND $test2_result
```

```
CMP #0x1
```

```
BEQ succed
```

```
LD #0xF
```

```
HLT
```

```
succed:
```

```
LD #0x1
```

```
HLT
```

Методика проверки

1. Запустить БЭВМ через терминал в режиме **Dual** при помощи команды **java -Dmode=dual -jar bcomp-ng.jar**
2. В терминале ввести следующую последовательность команд:

```
ma
mw 80C4084002
mw 80C4044002
mw 81C4024002
mw 80C4014002
mw 81C4804010
mw 81E8404010
mw 81E8204010
mw 80C4104010
mw 805C101040
```
3. Загрузить тестовую программу в БЭВМ. Для этого перейти во вкладку «Ассемблер», вставить скопированную программу и нажать кнопку «Компилировать»
4. Переключить тумблер «Работа/Останов» в режим «Работа»
5. Нажать кнопку пуск
6. Дождаться завершения программы
7. Посмотреть число в регистре **AC**. Если оно равно 1 (0000 0000 0000 0001 в 2-чной системе счисления), то оба теста прошли успешно. Иначе, один из тестов был провален. Для того, чтобы зафиксировать какой из, сделать следующую последовательность действий:
 1. Ввести адрес **0x1B1** (0000 0001 1011 0001) в клавишный регистр **IR** и нажать кнопку «Чтение»
 2. Посмотреть на регистр **DR**. Если его значение равно 1 (0000 0000 0000 0001), то тест на **переход при числе меньшем либо равным, чем 4095** был пройден успешно. Если его значение равно 0 (0000 0000 0000 0000), то тест был провален.
 3. Повторить последовательность действий из пункта 7.1 для адреса **0x1B2** (0000 0001 1011 0010)
 4. Посмотреть на регистр **DR**. Если его значение равно 1 (0000 0000 0000 0001), то тест на **переход при числе большем, чем 4095** был пройден успешно. Если его значение равно 0 (0000 0000 0000 0000), то тест был провален.

Вывод

В ходе выполнения лабораторной работы научился синтезировать команды в БЭВМ. Разобрался в работе ОМК и УМК.