

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники

Лабораторная работа №7
по дисциплине
«Основы профессиональной деятельности»
Вариант LDAC

Выполнил:

Студент группы Р3113
Султанов Артур Радикович

Проверил:

Блохина Елена Николаевна

г. Санкт-Петербург

2023г.

Оглавление

Оглавление	2
Задание	3
Часть 1. Исходный текст команды	4
Часть 2. Программа для тестирования синтезированной команды	4
Текст тестирующей программы	4
ОПИ, ОДЗ	6
Расположение данных	7
Адреса первой и последней выполняемой команды	8
Часть 3. Таблица трассировки микрокоманд синтезированной команды	8
Часть 4. Методика проверки синтезированной команды с использованием тестирующей программы	8
Заключение	10

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.

Команда для реализации - LDAC, безадресный вариант команды LD. Адрес берется из аккумулятора.

Часть 1. Исходный текст команды

Адрес МП	Микрокоманда	Действие	Описание
E0	0080009010	AC -> AR	Запись аргумента-адреса из аккумулятора в адресный регистр
E1	0100000000	MEM(AR) -> DR	Чтение памяти по адресу-аргументу, запись в регистр данных
E2	0010C09001	DR -> AC, N, Z, V	Запись данных в аккумулятор, выставление флагов N, Z, V
E3	80C4101040	GOTO INT @ C4	Переход к циклу прерывания

Часть 2. Программа для тестирования синтезированной команды

Текст тестирующей программы

Адрес	Метка	Мнемоника	Комментарии
020	START	CLA	Очистка аккумулятора
021		CALL TEST1	Вызов подпрограммы TEST1
022		CALL TEST2	Вызов подпрограммы TEST2
023		CALL TEST3	Вызов подпрограммы TEST3
024		LD TEST1_RESULT	Проверка на то, что результат TEST1 равен 1. Если это не так, то переход на FAIL
025		BEQ FAIL	
026		LD TEST2_RESULT	Проверка на то, что результат TEST2 равен 1. Если это не так, то переход на FAIL
027		BEQ FAIL	
028		LD TEST3_RESULT	Проверка на то, что результат TEST3 равен 1. Если это не так, то переход на FAIL
029		BEQ FAIL	
02A	SUCCESS	LD #1	Успех: если пройдены все тесты, то в ячейку результата записывается 1
02B		ST RESULT	

02C		HLT	
02D	FAIL	CLA	Провал: если не пройдены все тесты, то в ячейку результата записывается 0.
02E		ST RESULT	
02F		HLT	
052	TEST1	LD TEST1_A_PTR	Загрузка в аккумулятор адреса ячейки памяти TEST1_A
053		WORD 0x0F00 (LDAC)	Команда LDAC. Загрузка в аккумулятор значения по адресу, равному изначальному значению аккумулятора.
054		ST \$TEST1_DATA	Сохранение результат выполнения синтезированной команды
055		CMP TEST1_A	Если значение аккумулятора не равно значению ячейки TEST1_A, то переход на TEST1_FAIL
056		BNE TEST1_FAIL	
057	TEST1_SUCCESS	LD #1	Успех: тест пройден, запись 1 в ячейку результата.
058		ST \$TEST1_RESULT	
059		RET	
05A	TEST1_FAIL	CLA	Провал: тест не пройден, запись 0 в ячейку результата.
05B		ST \$TEST1_RESULT	
05C		RET	
072	TEST2	LD TEST2_A_PTR	Загрузка в аккумулятор адреса ячейки памяти TEST2_A
073		WORD 0x0F00 (LDAC)	Команда LDAC. Загрузка в аккумулятор значения по адресу, равному изначальному значению аккумулятора.
074		ST \$TEST2_DATA	Сохранение результат выполнения синтезированной команды
075		BMI TEST2_FAIL	Переход на TEST2_FAIL, если N==1
076	TEST2_SUCCESS	LD #1	Успех: тест пройден, запись 1 в ячейку результата.
077		ST \$TEST2_RESULT	
078		RET	
079	TEST2_FAIL	CLA	Провал: тест не пройден, запись 0 в

07A		ST \$TEST2_RESULT	ячейку результата.
07B		RET	
092	TEST3	LD TEST3_A_PTR	Загрузка в аккумулятор адреса ячейки памяти TEST3_A
093		WORD 0x0F00 (LDAC)	Команда LDAC. Загрузка в аккумулятор значения по адресу, равному изначальному значению аккумулятора.
094		ST \$TEST3_DATA	Сохранение результат выполнения синтезированной команды
095		BPL TEST3_FAIL	Переход на TEST3_FAIL, если N==0
096	TEST3_SUCCESS	LD #1	Успех: тест пройден, запись 1 в ячейку результата.
097		ST \$TEST3_RESULT	
098		RET	
099	TEST3_FAIL	CLA	Провал: тест не пройден, запись 0 в ячейку результата.
09A		ST \$TEST3_RESULT	
09B		RET	

ОПИ, ОДЗ

$RESULT \in [0; 1]$

$TEST1_RESULT \in [0; 1]$

$TEST2_RESULT \in [0; 1]$

$TEST3_RESULT \in [0; 1]$

$TEST1_A \in [0; 0xFFFF]$

$TEST2_A \in [0; 0x7FFF]$

$TEST3_A \in [0x8000; 0xFFFF] \sim [-0x8000; -1]$

$TEST1_A_PTR = 0x50$

$TEST2_A_PTR = 0x70$

$TEST3_A_PTR = 0x90$

$TEST1_DATA \in [0; 0xFFFF]$

$TEST2_DATA \in [0; 0x7FFF]$

$TEST3_DATA \in [0x8000; 0xFFFF] \sim [-0x8000; -1]$

Расположение данных

Адрес	Значение
010	Результат выполнения тестирующей программы
011	Результат выполнения теста №1
012	Результат выполнения теста №2
013	Результат выполнения теста №3
014	Данные, загруженные командой в рамках теста №1
015	Данные, загруженные командой в рамках теста №2
016	Данные, загруженные командой в рамках теста №3
050	Исходные данные для теста №1
051	Указатель на исходные данные для теста №1
070	Исходные данные для теста №2
071	Указатель на исходные данные для теста №2
090	Исходные данные для теста №3
091	Указатель на исходные данные для теста №3

Адреса первой и последней выполняемой команды

Адрес первой выполняемой команды	Адрес последней выполняемой команды
020	02С или 02F (в зависимости от результата)

Часть 3. Таблица трассировки микрокоманд синтезированной команды

Ячейка по адресу 0x50 равна 0x3333, с учетом этого:

MP до выборки МК	Содержимое памяти и регистров процессора после выборки и исполнения микрокоманды								
	MR	IP	CR	AR	DR	BR	AC	NZVC	MP (СчМК)
E0	0080009010	055	0F00	050	0F00	0054	0050	0000	E1
E1	0100000000	055	0F00	050	3333	0054	0050	0000	E2
E2	0010C09001	055	0F00	050	3333	0054	3333	0000	E3
E3	80C4101040	055	0F00	050	3333	0054	3333	0000	C4

Часть 4. Методика проверки синтезированной команды с использованием тестирующей программы

Исходный код в удобном для копирования формате:

https://raw.githubusercontent.com/sultanowskii/itmo-edu/master/opd/lab7/task/ommand_test.bcomp

Последовательность команд БЭВМ CLI для загрузки микрокоманд в удобном для копирования формате:

https://raw.githubusercontent.com/sultanowskii/itmo-edu/master/opd/lab7/task/mc_load.ibcomp

Проверка:

1. Запустить БЭВМ в совмещенном режиме (`java -jar -Dmode=dual bcomp-ng.jar`)
2. Загрузить тестирующую программу в БЭВМ

3. Скопировать и вставить команды БЭВМ CLI для загрузки микрокоманд
4. Поменять STOP на RUN
5. Запустить программу в автоматическом режиме (START)
6. В ячейку TEST1_A (0x050) положить значение, вписывающееся в ОДЗ
7. В ячейку TEST2_A (0x070) положить значение, вписывающееся в ОДЗ
8. В ячейку TEST3_A (0x090) положить значение, вписывающееся в ОДЗ
9. Дождаться выполнения программы
10. Проверить, что в ячейке RESULT (0x10) лежит 1.
11. Проверить, что в ячейках TEST1_RESULT (0x011), TEST2_RESULT (0x012) и TEST3_RESULT (0x013) лежат 1.
12. Проверить, что в ячейке TEST1_DATA (0x014), TEST2_DATA (0x015) и TEST3_DATA (0x016) лежат значения, соответственно равные тем, что были записаны в TEST1_A, TEST2_A, TEST3_A
13. Проверка завершена

Заключение

В рамках данной лабораторной работы я познакомился с макрокомандами в БЭВМ, его внутренним устройством, а также синтезировал собственную команду и написал методику проверки ее работоспособности.