Министерство высшего образования и науки Российской Федерации Национальный научно-исследовательский университет ИТМО Факультет программной инженерии и компьютерной техники

Лабораторная работа №7 по дисциплине «Основы профессиональной деятельности».

Вариант №696.

Работу выполнил: Афанасьев Кирилл Александрович, Студент группы Р3106. Преподаватель: Афанасьев Дмитрий Борисович.

Задание	3
Текст синтезированных микрокоманд	
Текст тестовой программы	
Таблица трассировки цикла исполнения разработанных микрокоманд	
Методика проверки команды с использованием тестовой программы	
Вывод	
D010VV	٠٠٠٠٠٠٠٠٠٠٠٠

Оглавление

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.

Вариант 696:

- 1. MSUB M вычитание аккумулятора из ячейки с записью результата в ячейку памяти без установки N/Z/V/C
- 2. Код операции 9...
- 3. Тестовая программа должна начинаться с адреса 040D₁₆

>>

Текст синтезированных микрокоманд

Таблица 1: Текст синтезированных микрокоманд.

Адрес МП	Микрокоманда	Действие ; Комментарий
E0 E1 E2 E3 E4 E5 E6 E7 E8	80E3084002 81E3024002 81E9014002 80E9804002 81E9404002 81E9204002 80E9104002 0001009611 8055101040	IF $CR(11) = \emptyset$ GOTO E3 ; $X(07)XX -> ha$ E4 IF $CR(9) = 1$ GOTO E3 ; $X(A,B,E,F)XX -> ha$ E4 IF $CR(8) = 1$ GOTO E9 ; $8,C -> E4/9,D -> 3a$ борт IF $CR(15) = \emptyset$ GOTO E9 ; $KO\Pi 07 -> 3a$ борт IF $CR(14) = 1$ GOTO E9 ; $KO\Pi CF -> 3a$ борт IF $CR(13) = 1$ GOTO E9 ; $KO\Pi AB -> 3a$ борт IF $CR(12) = \emptyset$ GOTO E9 ; $KO\Pi AB -> 3a$ борт $CR(12) = \emptyset$ GOTO E9 ; $CR(12) = \emptyset$ GOTO

Окончание таблицы.

Текст тестовой программы

Таблица 2: Текст тестовой программы.

Тестирование	е синтезированной команды	MSUB M (9)
	ORG 0x40D	
TCOUNT:	WORD 0x0002	; Константа: количество тестов
RESULT:	WORD 0x0001	; Результат работы тест-программы
TEMP_TC:	WORD ?	; Вспомогательная ячейка для счётчика
START:	LD TCOUNT	
	ST TEMP_TC	; Инициализируем счётчик
	LD #1	
	ST RESULT	; Инициализируем результат
	CALL \$TEST1	; Вызываем первый тест.
	CALL \$TEST2	; Вызываем второй тест.
LOOP_ADDR:	POP	; Нужно взять со стека число
	AND RESULT	; И логически умножить на результат.
	ST RESULT	; Не забываем все сохранить.

JUMP LOOP_ADDR HLT ; Tom самый выход! ORG 0x420 TEST1: LD TEST1_AC ; Загружаем тестовое значение AC. WORD 0x942C ; Вызываем синтезированную команд С. CMP TEST1_CV ; Сравниваем с эталонным. EQUITY CLA ; Нет — загрузим 0 в AC. 3 догружаем про адрес возврата регита правот уружаем обрат уружаем и уружаем и уружаем уружаем и уружаем		Остались результаты тестов?	LOOP TEMP_TC ;	
HLT ; Tom самый выход! ORG 0x420 TEST1: LD TEST1_AC ; Загружаем тестовое значение АС. WORD 0x942C ; Вызываем синтезированную комано сированную комано сирования сир				
TEST1: LD TEST1_AC ; Загружаем тестовое значение AC. WORD 0x942C ; Вызываем синтезированную команд CMP TEST1_M ; Загружаем ее результат работы. CMP TEST1_CV ; Сравниваем с эталонным. ВEQ T1_PASS ; Равен эталону? T1_FAIL: CLA ; Нет — загрузим 0 в AC. JUMP T1_RETURN ; И отправимся на возврат. Т1_PASS: LD #1 ; Да — загрузим 1 в AC. Т1_RETURN: SWAP ; Не забываем про адрес возврата РUSH ; Вернем результат теста. RET ; Возврат управления. TEST1_AC: WORD 0xFF01 ; Тестовое значение AC. TEST1_M: WORD 0x5233 ; Тестовое значение AC. TEST1_CV: WORD 0x5332 ; Ожидаемый результат. ORG 0x430 TEST2: LD TEST2_AC ; Загружаем тестовое значение AC. Bызываем синтезированную команд CLD TEST2_M ; Загружаем ее результат работы. СМР TEST2_CV ; Сравниваем с эталонным. ВEQ T2_PASS ; Равен эталону? Т2_FAIL: CLA ; Нет — загрузим 0 в AC. JUMP T1_RETURN ; И отправимся на возврат.		• •		
TEST1: LD TEST1_AC ; Загружаем тестовое значение AC. WORD 0x942C ; Вызываем синтезированную командо LD TEST1_M ; Загружаем ее результат работы. CMP TEST1_CV ; Сравниваем с эталонным. ВЕQ T1_PASS ; Равен эталону? T1_FAIL: CLA ; Нет — загрузим 0 в AC.		10m camon comoo.		
JUMP T1_RETURN T1_PASS: LD #1 ; Да — загрузим I в АС. T1_RETURN: SWAP	нду.	Вызываем синтезированную команду Загружаем ее результат работы. Сравниваем с эталонным.	LD TEST1_AC ; WORD 0x942C ; LD TEST1_M ; CMP TEST1_CV ;	TEST1:
T1_PASS: LD #1 ; Да — загрузим 1 в АС. T1_RETURN: SWAP ; Не забываем про адрес возврата PUSH ; Вернем результат теста. RET ; Возврат управления. TEST1_AC: WORD 0xFF01 ; Тестовое значение АС. TEST1_M: WORD 0x5233 ; Тестовая ячейка памяти для МЅИВ. TEST1_CV: WORD 0x5332 ; Ожидаемый результат. ORG 0x430 ; Загружаем тестовое значение АС. WORD 0x943C ; Вызываем синтезированную команд. LD TEST2_M ; Загружаем ее результат работы. CMP TEST2_CV ; Сравниваем с эталонным. BEQ T2_PASS ; Равен эталону? T2_FAIL: CLA ; Нет — загрузим 0 в АС. JUMP T1_RETURN ; И отправимся на возврат.		Hem – загрузим 0 в AC .	CLA ;	T1_FAIL:
T1_RETURN: SWAP PUSH RET RET RET RET RET REST1_AC: WORD 0xFF01 TEST1_M: WORD 0x5233 TEST1_CV: WORD 0x5332 ORG 0x430 TEST2: LD TEST2_AC WORD 0x943C LD TEST2_M CMP TEST2_CV BEQ T2_PASS T2_FAIL: CLA JUMP T1_RETURN ; He забываем про адрес возврата ; Вернем результат тестовое значение АС. ; Тестовое значение АС. ; Тестовое значение АС. ; Ожидаемый результат. ; Загружаем тестовое значение АС. ; Вызываем синтезированную команда. ; Сравниваем с эталонным. ; Равен эталону? ; Нет — загрузим 0 в АС. ; И отправимся на возврат.		И отправимся на возврат.	JUMP T1_RETURN ;	
PUSH		\mathcal{A} а – загрузим 1 в AC .	LD #1 ;	T1_PASS:
RET ; Возврат управления. TEST1_AC: WORD 0xFF01 ; Тестовое значение AC. TEST1_M: WORD 0x5233 ; Тестовая ячейка памяти для MSUB. TEST1_CV: WORD 0x5332 ; Ожидаемый результат. ORG 0x430 TEST2: LD TEST2_AC ; Загружаем тестовое значение AC. WORD 0x943C ; Вызываем синтезированную команда LD TEST2_M ; Загружаем ее результат работы. CMP TEST2_CV ; Сравниваем с эталонным. BEQ T2_PASS ; Равен эталону? T2_FAIL: CLA ; Нет — загрузим 0 в AC. JUMP T1_RETURN ; И отправимся на возврат.		Не забываем про адрес возврата	SWAP ;	T1_RETURN:
TEST1_AC: WORD 0xFF01 ; Тестовое значение AC. TEST1_M: WORD 0x5233 ; Тестовая ячейка памяти для MSUB. TEST1_CV: WORD 0x5332 ; Ожидаемый результат. ORG 0x430 ; Загружаем тестовое значение AC. WORD 0x943C ; Вызываем синтезированную команда. LD TEST2_M ; Загружаем ее результат работы. CMP TEST2_CV ; Сравниваем с эталонным. BEQ T2_PASS ; Равен эталону? T2_FAIL: CLA JUMP T1_RETURN ; И отправимся на возврат.		Вернем результат теста.	PUSH ;	
TEST1_M: WORD 0x5233 ; Тестовая ячейка памяти для MSUB. TEST1_CV: WORD 0x5332 ; Ожидаемый результат. ORG 0x430 ; Загружаем тестовое значение АС. WORD 0x943C ; Вызываем синтезированную команда. LD TEST2_M ; Загружаем ее результат работы. CMP TEST2_CV ; Сравниваем с эталонным. BEQ T2_PASS ; Равен эталону? T2_FAIL: CLA JUMP T1_RETURN ; И отправимся на возврат.		Возврат управления.	RET ;	
ТЕЅТ1_CV: WORD 0x5332 ; Ожидаемый результат. ОRG 0x430 ТЕЅТ2: LD TЕՏТ2_AC ; Загружаем тестовое значение АС. WORD 0x943C ; Вызываем синтезированную команда LD TЕЅТ2_M ; Загружаем ее результат работы. СМР TЕЅТ2_CV ; Сравниваем с эталонным. BEQ T2_PAЅ ; Равен эталону? T2_FAIL: CLA JUMP T1_RETURN ; И отправимся на возврат.		Тестовое значение АС.	WORD 0xFF01 ;	TEST1_AC:
ORG 0x430 TEST2: LD TEST2_AC ; Загружаем тестовое значение AC. WORD 0x943C ; Вызываем синтезированную комано синтезиружаем ее результат работы. Синтезиружаем тестовое значение AC. загружаем синтезированную комано синтези	В.	Тестовая ячейка памяти для MSUB.	WORD 0x5233 ;	TEST1_M:
TEST2: LD TEST2_AC ; Загружаем тестовое значение АС. WORD 0x943C ; Вызываем синтезированную команов LD TEST2_M ; Загружаем ее результат работы. CMP TEST2_CV ; Сравниваем с эталонным. BEQ T2_PASS ; Равен эталону? T2_FAIL: CLA JUMP T1_RETURN ; И отправимся на возврат.		Ожидаемый результат.	WORD 0x5332 ;	TEST1_CV:
JUMP T1_RETURN ; И отправимся на возврат.	нду.	Вызываем синтезированную команду Загружаем ее результат работы. Сравниваем с эталонным.	LD TEST2_AC ; WORD 0x943C ; LD TEST2_M ; CMP TEST2_CV ;	TEST2:
		Hет – загрузим 0 в AC .	CLA ;	T2_FAIL:
TO PASS: In #1 $\cdot \pi_a = 2a2mv_2u_M \log AC$		И отправимся на возврат.	JUMP T1_RETURN ;	
		\mathcal{A} а – загрузим 1 в AC .	LD #1 ;	T2_PASS:
Т2_RETURN: SWAP ; Не забываем про адрес возврата		Не забываем про адрес возврата	,	T2_RETURN:
PUSH ; Вернем результат теста.			,	
RET ; Возврат управления.		1 1	•	
TEST2_AC: WORD 0х6001 ; Тестовое значение АС.		Тестовое значение АС.	WORD 0x6001 ;	_
,	В.	Тестовая ячейка памяти для MSUB.	WORD 0x7FFF ;	-
TEST2_CV: WORD 0x1FFE ; Ожидаемый результат.		Ожидаемый результат.	WORD 0x1FFE ;	TEST2_CV:

Окончание таблицы.

Таблица трассировки цикла исполнения разработанных микрокоманд Таблица 3: Трассировка цикла исполнения синтезированной команды.

MР до выборки	Содержимое памяти и регистров процессора после выборки и исполнения МК								
MK	MR	IP	CR	AR	DR	BR	AC	NZVC	MP
E0	80E3084002	422	942C	42C	5233	0421	FF01	1000	E3
E3	80E9804002	422	942C	42C	5233	0421	FF01	1000	E4
E4	81E9404002	422	942C	42C	5233	0421	FF01	1000	E5
E5	81E9204002	422	942C	42C	5233	0421	FF01	1000	E6
E6	80E9104002	422	942C	42C	5233	0421	FF01	1000	E7
E7	0001009611	422	942C	42C	5332	0421	FF01	1000	E8
E8	8055101040	422	942C	42C	5332	0421	FF01	1000	55
55	0200000000	422	942C	42C	5332	0421	FF01	1000	56

56	80C4101040	422	942C	42C	5332	0421	FF01	1000	C4

Окончание таблицы.

Методика проверки команды с использованием тестовой программы

- 1) Загрузить программный комплекс в память Базовой ЭВМ
- 2) Ввести в клавишный регистр значение 0000.0100.0010.1011 (0x42B)
- 3) Нажать кнопку «ВВОД АДРЕСА»
- 4) Придумать число и записать его в клавишный регистр
- 5) Нажать кнопку «ЗАПИСЬ»
- 6) Придумать еще одно число и записать его в клавишный регистр
- 7) Нажать кнопку «ЗАПИСЬ»
- 8) Из числа, придуманного в пункте 6. вычесть число, придуманное в пункте 4. Результат ввести в клавишный регистр
- 9) Нажать кнопку «ЗАПИСЬ»
- 10) Ввести в клавишный регистр значение 0000.0100.0011.1011 (0x43B)
- 11) Последовательно повторить все действия из пунктов с 3 по 9 включительно
- 12) Ввести в клавишный регистр значение 0000.0100.0000.1111 (0x40F)
- 13) Переключить тумблер «РАБОТА/ОСТАНОВ» в режим «РАБОТА»
- 14) Нажать кнопку «ПУСК»
- 15) Дождаться остановки работы Базовой ЭВМ
- 16) Ввести в клавишный регистр значение 0000.0100.0000.1110 (0x40E)
- 17) Нажать кнопку «ВВОД АДРЕСА»
- 18) Нажать кнопку «ЧТЕНИЕ»
- 19) Убедиться, что прочитанное значение равно 0х0001

Вывод

Во время выполнения данной лабораторной работы я изучил микропрограммное устройство управления Базовой ЭВМ, а также научился синтезировать команды в Базовой ЭВМ, составлять программные комплексы для тестирования синтезированных команд и методики тестирования для операторов Базовой ЭВМ.