Министр науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

«Национальный исследовательский университет ИТМО»

Факультет информационных технологий и программирования

Домашняя работа № 4

Расширение системы команд ЭВМ

Выполнил студент группы № М3101 Михеев Артем Романович

Подпись:

Проверил:

Бабич Мария Сергеевна

Цель работы

Написать Изучение микрокоманд базовой ЭВМ, микропрограмм выполнения отдельных команд, а так же овладение навыками составления микропрограмм для новых команд.

Задание, вариант 6

Часть І

Написать последовательность адресов микрокоманд, которые должны быть выполнены при реализации заданного фрагмента программы, начинающегося с команды, расположенной по адресу 0002. По полученным результатам составить таблицу а так же описать последние шесть микрокоманд цикла "ИСПОЛНЕНИЯ" для выделенной команды в виде рисунков.

Вариант 6:

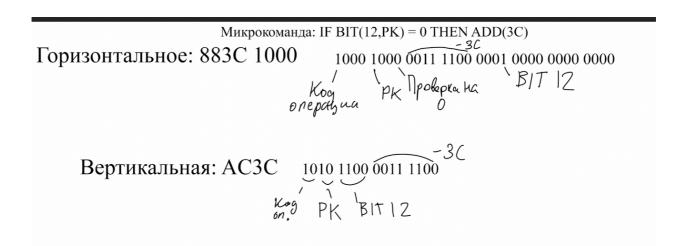
01:	1	0001
02:	CMC	F500
03:	BCS 05	8005
04:	NOP	F100
05:	+ADC 01	5001

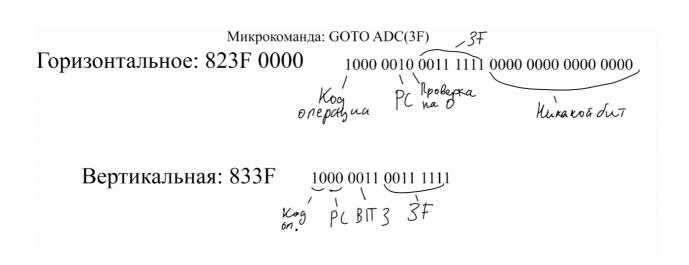
Решение

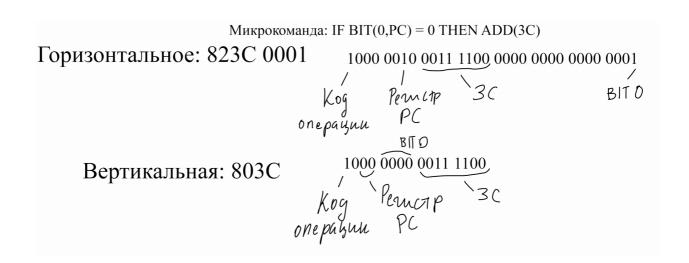
1. Результаты первой части задания в виде таблицы

Команда	Машинный цикл	Последовательность адресов	
		микрокоманд	
	_	89	
CMC	Выборка команды	01, 02, 03, 04, 05, 06, 07, 08, 09, 0A	
(F500)	Исполнение	5E, 61, 62, 65, 66, 7E, 80, 81	
,		8F	
		88	
	_	89	
BCS 05	Выборка команды	01, 02, 03, 04, 05, 06, 07, 08, 0C	
(8005)	Исполнение	1D, 2D, 30, 33, 46, 47, 48, 49	
	_	8F	
		88	
ADC 01	_	89	
(5001)	Выборка команды	01, 02, 03, 04, 05, 06, 07, 0C	
	Исполнение	1D, 1E, 1F, 20, 27, 28, 2B, 2C, 3F, 40,	
		41, 42	
	_	8F	
		88	

2. Последние 6 микрокоманд цикла "исполнение" команды ADC 01

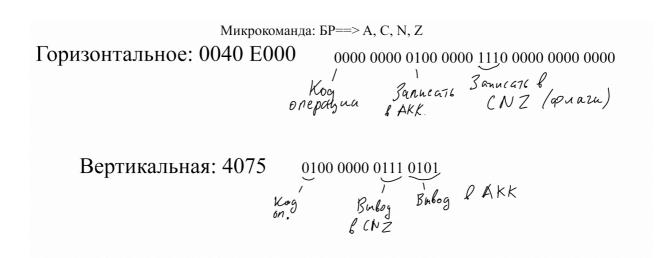


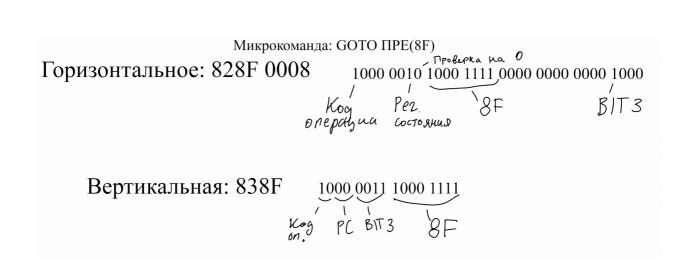




Микрокоман	лда: A + РД+ 1 ==> БР	
Горизонтальное: 0000 0412	0000 0000 0000 0	0000 0000 0100 0001 0010
0	Kog nepagua	+1 AKK PA HA ned npabwū brog brog
Вертикальная: 1110	0001 0001 0001 0000	







Часть II

Написать завершающие вертикальные микрокоманды цикла "исполнение" следующих команл:

Команда 7ххх – пересылка удвоенная (записать в ячейку памяти, на которую указывает адресная часть команды, удвоенное содержимое аккумулятора)

Команда Dxxx – организовывать переход к команде, расположенной по адресу, на которую указывает адресная часть команды, если 7-й бит аккумулятора равен единице Команда FE00 – циклический сдвиг вправо с очисткой регистра С

После этого написать тестовые программы для проверки правильности исполнения всех синтезированных команд БЭВМ.

Решение

WORD D045

```
Команда 7ххх:
     1000
                        # АКК → БР
B0:
B1:
     4002
                        # БР → РД
                        # АКК + РД → БР
B2:
     1100
B3:
     4002
                        #БР → РД
B4:
      0002
                        # РД → (РА)
     838F
B5:
Команда Dxxx:
D0:
     B7D3
                        # If bit7 is not set then go to next command
D1:
     0100
                        # РД → БР
D2:
     4004
                        # БР → СК
D3:
     838F
Команда FE00:
E0:
     0004
                        # RAR, сдвиг вправо
                        # BP \rightarrow N, Z, clear C and BP \rightarrow AKK
E1:
      40B5
E2:
     838F
Для тестирования была написана вот такая программа:
ORG 0000
BEGIN:
      CLA
      ADD TEST7
      WORD 7101
      CLA
      ADD TESTD 1
      WORD D040
      DEC.
      MOV TESTD_1_RESULT
CONTINUE 1:
      CLA
      ADD TESTD_2
```

```
INC
     MOV TESTD_2_RESULT
CONTINUE_2:
     CLA
     ADD TESTF
     WORD FE00
     CMC
     BCS 0050
     CMA
     MOV TESTF_RESULT
CONTINUE_3:
     HLT
ORG 0040
     MOV TESTD_1_RESULT
     BR CONTINUE_1
ORG 0045
     MOV TESTD_2_RESULT
     BR CONTINUE_2
ORG 0050
     MOV TESTF_RESULT
     BR CONTINUE_3
ORG 0100
TEST7:
          WORD 0003
TEST7_RESULT: WORD 0000
TESTD 1: WORD 0040
TESTD_1_RESULT: WORD 0000
TESTD_2: WORD 0080
TESTD_2_RESULT: WORD 0000
TESTF:
          WORD E01F
TESTF_RESULT: WORD 0000
```

Вывод

В ходе домашнего задания нужно было изучить два разных вида кодировки команд – горизонтальный и вертикальный, детальнее понять разные циклы ЭВМ, а также разобраться с существующими микрокомандами и научится писать свои для того чтобы расширять возможности ЭВМ. Несмотря на то что в случае с БЭВМ это относительно

упрощено, по сравнению с настоящими процессорами, такие навыки всё равно являются необходимыми и полезными.					