

**Министерство 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 № М3111

Гонтарь Тимур Сергеевич

Подпись:



Проверил:

Батоцыренов Павел Андреевич

Санкт-Петербург
2023

Вариант 3

- Исходные команды:

3
1
DEC
BMI 05
NOP
+ ADD 01

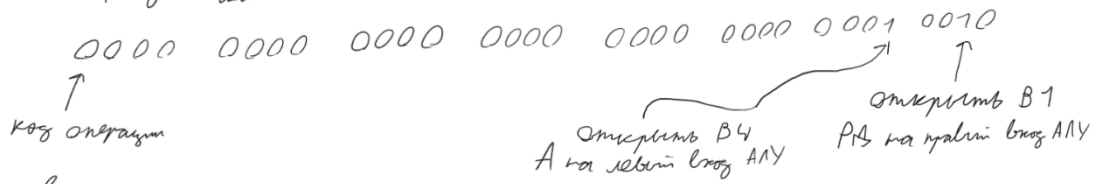
- Таблица команд с микрокомандами:

Команда	Машинный цикл	Последовательность адресов микрокоманд
DEC (F900)	- Выборка команды Декодирование безадресной команды Исполнение безадресной команды -	89 01, 02, 03, 04, 05, 06, 07, 08, 09, 0A 5E, 5F, 6C, 6F 70, 71, 72 8F 88
BMI 05 (A005)	- Выборка команды Определение вида адресации Выборка адреса операнда Исполнение адресной команды -	89 01, 02, 03, 04, 05, 06, 07, 08 0C 1D, 2D, 30, 31 4C, 4D, 47, 48, 49 8F 88
NOP (F100)	-	Микрокоманды не будут выполняться, ибо по условию предыдущей команды BMI счётчик команд перейдет сразу на 005, пропустив эту ячейку.
ADD 01 (4001) +	- Выборка команды Определение вида адресации Выборка адреса операнда Исполнение адресной команды -	89 01, 02, 03, 04, 05, 06, 07 0C 1D, 1E, 1F, 20, 27, 28, 2B 3C, 3D, 3E 8F 88

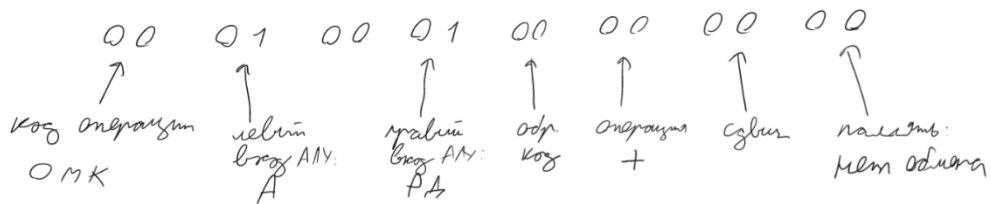
- Команды из цикла «исполнение» последней команды

микрокоманда 3С: $A + RA \rightarrow BR$

- горизонтальная: 0000 0012;

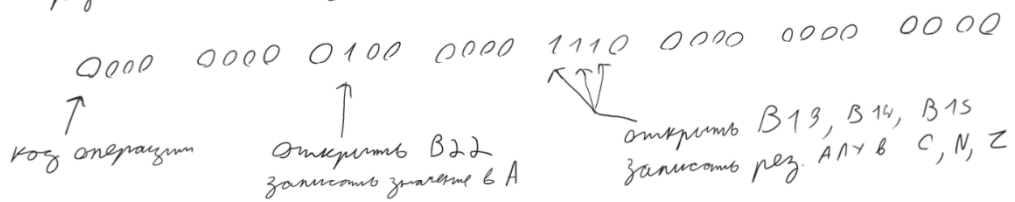


- вертикальная: 1100

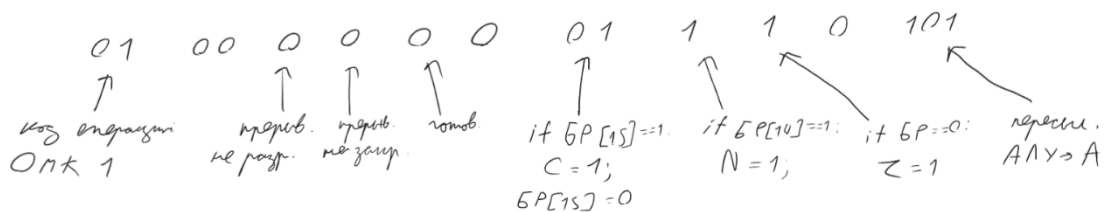


микрокоманда 3D: $BR \rightarrow A, C, N, Z$

- горизонтальная: 0040 E000



- вертикальная: 4075



mikroskopisanya 3E : 60X 8F

- копуляционная: 828 F 0008

1000 0010 1000 1111 0000 0000 0000 1000

проверка
ремонт РС

азрес перескога
8F,

↑
apogee = 4 sin PC;
if PC[3] = 0: CMK = SF

- Бермудская: 838F

[illegible]

↑
азглед перекрста: $8F$;
; $P[C[3]] = 0$;
 $C_2 M K = 8F$

микроконтроллер 8F: if PC[7] == 0:
HTL88

- координатная: 8288 0080

1000 0010 1000 1000 0000 0000 1000 0000

↑
код операции
УМК

↑
адрес регистра
PC

↑
адрес регистра
88

↑
адрес регистра
и PC
СЧМ

повторяется 8 раз PC;
it PC[7] == 0;
C7MK = 88;

- Вертикальная 8788

1
 \uparrow
 код оператора
 ymk

$0 \quad 00$
 \uparrow
 показатель
 перемещения PC

0111
 \uparrow
 показатель
 8-го разряда PC

10001000
 \nwarrow
 адрес по адресу
 it PC[7] =
 Cymk = 0

perenos no ageray 88;
 it PC[7] = 0;
 C7MK = 88;

- Микрокоманды цикла исполнение команд:

1. 7xxx – Сравнение аккумулятора и ячейки памяти (М - А, БР => C, N, Z)

Адрес микрокоманды	Микрокоманда	Комментарии
B0	1150 0001000101010000	!A + РД + 1 => БР, то же самое что БР = РД – А; В АЛУ слева подается обратный код аккумулятора, справа РД, выполняется их сложение + 1, что значит разности РД и А. Результат записывается в буферный регистр.
B1	4070 0100000001110000	C = БР[15], N = (БР < 0), Z = (БР == 0); Буферный регистр проверяется на признаки вычитания, в результате: в C записывается есть ли переполнение, в N записывается знак числа, в Z записывается 1 если БР стало 0. Сам результат из БР никуда не пересылается.
B2	838F 1000001110001111	GOTO ПРЕ(8F); Осуществляет безусловный переход к микрокомандам, завершающим исполнение команды базовой микро ЭВМ

2. Dxxx – переход по адресу (указанному в РД), если выполняется условие, что значение А > 16383. Очевидно, что это выполняется если:

- 1) Аккумулятор больше 0
- 2) 15 бит слева == 1

Адрес микрокоманды	Микрокоманда	Комментарии
D0	C28F 1100001010001111	if PC[2] == 1: GOTO 8F; Проверка условия 1), если аккумулятор меньше 0 (регистр N равен 1), то завершение выполнения команды
D1	BE8F 1011111010001111	if A[14] == 0: GOTO 8F; Проверка условия 2), если у аккумулятора 15 бит слева равен 0, то завершение выполнения команды
D2	0100 0000000100000000	БР = РД; Если оба условия для проверки аккумулятора были выполнены, то пересылка регистра данных в буферный регистр.

D3	4004 0100000000000100	СК = БР; Пересылка регистра данных в счётчик команд
D4	838F 1000001110001111	ГОТО ПРЕ(8F); Осуществляет безусловный переход к микрокомандам, завершающим исполнение команды базовой микро ЭВМ

3. FE00 – безадресная команда, получение дополнительного кода аккумулятора. FE00 – 1111 1110 0000 0000, то есть необходимо чтобы 10 бит слева был 1, а 9 был 0. Для этого в процессе дешифрации сначала исключим факт 0 в 10 бите, затем исключим факт 1 в 9 бите. Это всё нужно, чтобы выполнялась именно команда FE00, а не какая-либо другая.

Адрес микрокоманды	Микрокоманда	Комментарии
E0	A98F 1010100110001111	if PK[9] == 0: ГОТО 008F; Проверка на условие 1, если 10 бит слева равен 0, то выход из команды
E1	E88F 1110100010001111	if PK[8] == 1: ГОТО 008F; Проверка на условие 2, если 9 бит слева равен 1, то выход из команды
E2	1050 0001000001010000	БР = !А + 1; Если предыдущие условия выполняются, и команда реально FE00, то в буферный регистр записывается обратный код аккумулятора + 1
E3	4075 0100000001110101	С = БР[15], N = (БР < 0), Z = (БР == 0), А = БР; Буферный регистр проверяется на признаки построения дополнительного кода, в результате: в С записывается есть ли переполнение, в N записывается знак числа, в Z записывается 1 если БР стало 0. Сам результат из БР пересылается в аккумулятор
E4	838F 1000001110001111	ГОТО ПРЕ(8F); Осуществляет безусловный переход к микрокомандам, завершающим исполнение команды базовой микро ЭВМ

- Проверка правильности исполнения синтезированных команды базовой ЭВМ

Адрес	Код	Мнемоника	Комментарии
010	7123	-	Ячейка, которая нужна для изначального заполнения аккумулятора
011	7000	-	Ячейка М, из которой будет вычитаться аккумулятор с помощью собственной команды 7xxx
012	0000	-	Ячейка для хранения результата проверки командой Dxxx, если значение в аккумуляторе подходит под условие
013	0000	-	Ячейка для хранения значения регистра знака N, после выполнения команды 7xxx
014	0000	-	Ячейка для хранения результата проверки командой Dxxx, если значение в аккумуляторе уже не подходит подходит под условие
015	0000	-	Ячейка для хранения дополнительного кода, полученного в результате выполнения команды FE00
016	F200	CLA	Очистка аккумулятора
017	4010	ADD 010	Записать в аккумулятор значение ячейки 010
018	D01A	Dxxx	Проверка условия Dxxx (первая), в случае успеха переход на 01A
019	C01C	BR 01C	Безусловный переход на ячейку 01C
01A	0012	ISZ 012	Нарастить значение ячейки 012 на 1, СК++
01B	F100	NOP	Нет операции
01C	7011	7xxx	Вычесть из значения ячейки памяти 011 значение аккумулятора
01D	A01F	BMI 01F	Если аккумулятор меньше 0, то перейти по адресу 01F
01E	C021	BR 021	Безусловный переход по адресу 021
01F	0013	ISZ 013	Нарастить значение ячейки 013 на 1, СК++
020	F100	NOP	Нет операции
021	6011	SUB 011	А – М. После выполнения 7xxx поменялись биты РС, но сам аккумулятор не поменялся. Для повторной проверки команды Dxxx необходимо вычесть уже из самого аккумулятора это значение.
022	D024	Dxxx	Проверка условия Dxxx (вторая) , в случае успеха переход на 024

023	C026	BR 026	Безусловный переход на ячейку 026
024	0014	ISZ 014	Нарастить значение ячейки 014 на 1, СК++
025	F100	NOP	Нет операции
026	FE00	-	Выполнение собственной программы FE00, дополнительный код аккумулятора
027	3015	MOV 015	Значение аккумулятора записывается в ячейку памяти 015
028	F200	CLA	Очистка аккумулятора
	F000	HLT	Завершение программы

В ходе данной программы были выполнены действия: в аккумулятор записали число, удовлетворяющее условию Dxxx. Условие оказалось верным, в ячейку 012 записывается значение 1 как знак того что условие выполнилось. Затем выполняется команда 7xxx, в результате которой изменился регистр знака (значение аккумулятора в свою очередь не изменилось), в подтверждение этому выполняется проверка регистра N – и в ячейку 013 записывается значение 1 как правильный результат ($N == 1$). Затем из аккумулятора вычитается значение M (011), ибо в результате 7xxx аккумулятор остался неизменным, а для повторной проверки Dxxx надо его поменять. Затем происходит повторная проверка Dxxx, в этот раз она не проходит и в ячейку 014 не записывается 1. После этого выполняется безадресная команда FE00, результат записывается в ячейку 015. После этого программы завершается.

Вывод: в ходе выполнения данного домашнего задания я изучил микрокоманды базовой ЭВМ, микропрограммы выполнения отдельных команд, а так же овладел навыками составления микропрограмм для новых команд.