



Министерство образования Российской Федерации
МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ
им. Н.Э. БАУМАНА

Факультет: Информатика и системы управления

Кафедра: Информационная безопасность (ИУ8)

Аппаратные средства вычислительной техники

Лабораторная работа №2

**“БЛОК МИКРОПРОГРАММНОГО УПРАВЛЕНИЯ (БМУ).
ПЕРЕХОДЫ В МИКРОПРОГРАММАХ С ИСПОЛЬЗОВАНИЕМ СТЕКА”**

Преподаватель: Рафиков Андрей Гыязович

Студент: Веденеев Андрей Андреевич

Группа: ИУ8-62

Москва 2023г.

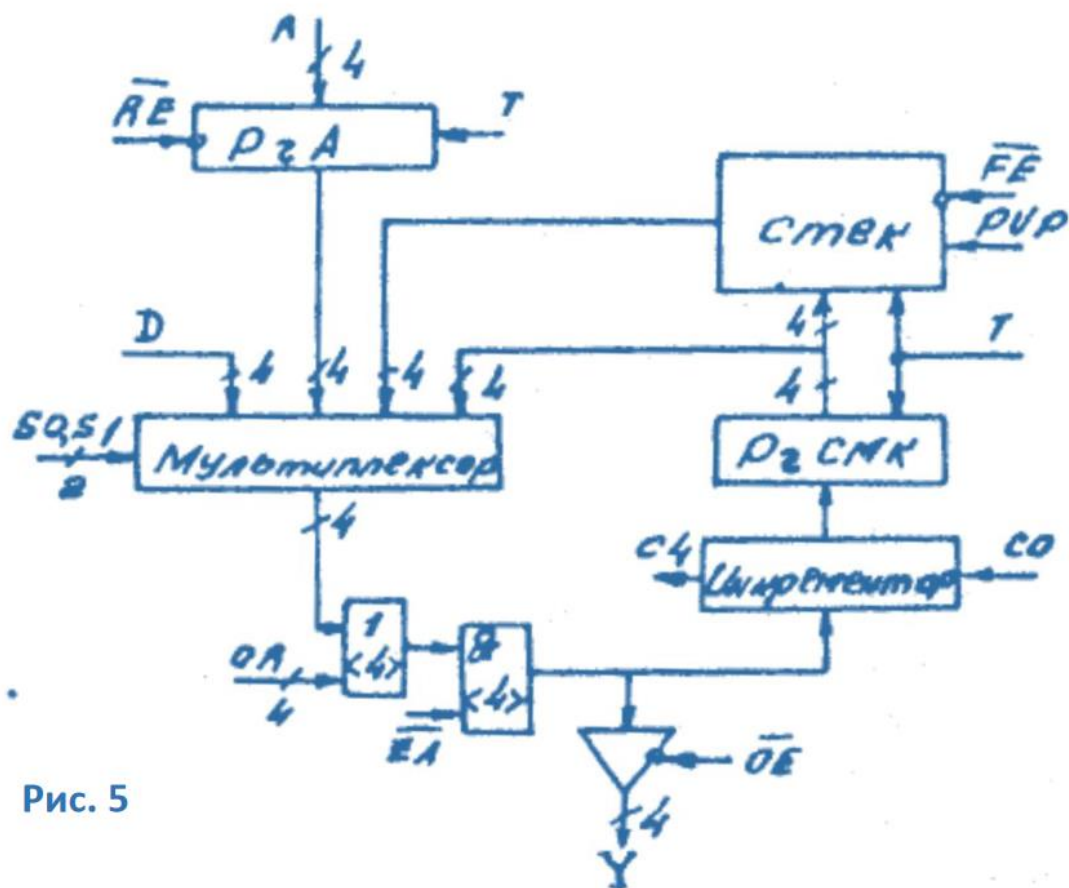
Цель работы: изучение структуры и функций БМУ К1804ВУ1, способа управления узлами БМУ с помощью микрокоманды; исследование функций перехода с использованием стека.

Теоретическая часть

Управление выборкой следующей микрокоманды из микропрограммной памяти осуществляется с помощью блока микропрограммного управления, который на схеме представлен узлами:

- БИС управления адресом микрокоманды К1804ВУ1,
- Микропрограммная память,
- Регистр микрокоманд,
- ПЗУ для управления выборкой следующего адреса.

Структурная схема БИС:



Коды, программируемые в ПЗУ в зависимости от функций перехода:

Таблица 12

Функция перехода	Входы ПЗУ		Выходной код ПЗУ					Операции в БМУ
	Код СА	Флаг	SI	$.SO$	\overline{FE}	PUP	$Pz ST$	
Переход на следующий адрес (продолжить)	0010	X	0	0	1	X	0	CMK→Y
Безусловный переход на адрес	0001	X	0	1	1	X	0	$PzA \rightarrow Y$
Переход на адрес из PгМК, если F=0*)	1100	0	0	0	1	X	1	CMK→Y
	1100	1	0	1	1	X	1	PгA→Y
Переход на адрес из PгМК, если F≠0	0000	0	0	1	1	X	1	PгA→Y
	0000	1	0	0	1	X	1	CMK→Y
Загрузить в стек (и продолжить)	1001	X	0	0	0	1	0	CMK→Y, PUSH
Вытолкнуть в стек (и продолжить)	1010	X	0	0	0	0	0	CMK→Y, POP
Переход по стеку	0111	X	1	0	1	X	0	СТЕК→Y
Окончить цикл и вытолкнуть из стека, если F=0 **)	1000 1000	0	1	0	1	X	1	СТЕК→Y
		1	0	0	0	0	1	CMK→Y, POP
Переход по адресу вектора (на переключателях адреса)	0011	X	1	1	1	X	0	D→Y
Переход к подпрограмме	0101	X	0	1	0	1	0	$PzA \rightarrow Y, PUSH$
Переход к подпрограмме, если F≠0	0100	0	0	1	0	1	1	$PzA \rightarrow Y, PUSH$
	0100	1	0	0	1	X	1	CMK→Y
Возврат из подпрограммы	0110	X	1	0	0	0	0	СТЕК→Y, POP

Практическая часть

Задание 1.

Загрузить в память (в тетрады 6, 7) программу, обеспечивающую выполнение 3 групп микрокоманд по адресам A0, A1), (A4, A5), (A14, A15) с остановом по адресу A15.

Адрес	Микрокоманда	Прыжок
0x000	Y=F=0; M1=0; M0=0	JNXT
0x001	Y=F=0; M1=0; M0=0	JMP 0x004
0x002	Y=F=0; M1=0; M0=0	JMP 0x004
0x003	Y=F=0; M1=0; M0=0	JMP 0x004
0x004	Y=F=0; M1=0; M0=0	JNXT
0x005	Y=F=0; M1=0; M0=0	JMP 0x00E
0x006	Y=F=0; M1=0; M0=0	JMP 0x00E
0x007	Y=F=0; M1=0; M0=0	JMP 0x00E
0x008	Y=F=0; M1=0; M0=0	JMP 0x00E
0x009	Y=F=0; M1=0; M0=0	JMP 0x00E
0x00A	Y=F=0; M1=0; M0=0	JMP 0x00E
0x00B	Y=F=0; M1=0; M0=0	JMP 0x00E
0x00C	Y=F=0; M1=0; M0=0	JMP 0x00E
0x00D	Y=F=0; M1=0; M0=0	JMP 0x00E
0x00E	Y=F=0; M1=0; M0=0	JNXT
0x00F	Y=F=0; M1=0; M0=0	JNXT

Задание 2.

Проверить работу программы I из таблицы, фиксируя последовательность адресов на шине Y. Изменить программу, обеспечив выход из цикла по условию.

Таблица I3

Адрес памяти	Тетрада 7 (4H)	Тетрада 6 (CA)	Выход Y	Примечания
0		0010	0001	<u>Программа I</u> Продолжить
1		1001	?	Загрузить стек
2		0010	?	Продолжить
3		0010	?	"
4		0111	?	Переход по стеку
0		0010	?	<u>Программа 2</u> Продолжить
1	1100	0101	?	Переход к подпрограмме I2
2		0010	?	Продолжить
3	0000	0001	?	Переход на 0
...				
I2		0010	?	Продолжить
I3		0110	?	Возврат

Адрес	Микрокоманда	Прыжок
0x000	POH(0)=F=5v0; M1=0; M0=0	JNXT
0x001	Y=F=0; M1=0; M0=0	PUSH
0x002	Y=F=0; M1=0; M0=0	JNXT
0x003	POH(0)=F=POH(0)-1; M1=0; M0=0	JNXT
0x004	Y=F=0; M1=0; M0=0	JSNZ

Задание 3.

Загрузить и выполнить программу условного перехода по адресу Aj, в которой проверяются 2 заданных признака, вырабатываемых в результате выполнения арифметической операции (из совокупности признаков F3, C4, OVR, F=0, F≠0). Обратить внимание на значение второго признака на выходе процессорного элемента после выполнения команды условного перехода по первому признаку. Объяснить, каким значением второго признака

(первоначальным или изменившимся) обусловлен второй переход, и почему.

Адрес	Микрокоманда	Прыжок
0x000	$POH(0) = F = 3 \vee 0; M1=0; M0=0$	JNXT
0x001	$POH(1) = F = 2 \vee 0; M1=0; M0=0$	CALL 0x010
0x002	$POH(0) = F = 0 \wedge 0; M1=0; M0=0$	JNXT
0x003	$POH(0) = F = 0 \wedge 0; M1=0; M0=0$	JNXT
0x004	$POH(0) = F = 0 \wedge 0; M1=0; M0=0$	JNXT
0x005	$POH(0) = F = 0 \wedge 0; M1=0; M0=0$	JNXT
0x006	$POH(0) = F = 0 \wedge 0; M1=0; M0=0$	JNXT
0x007	$POH(2) = F = 0 \vee PQ; M1=0; M0=0$	JNXT
0x008	$PQ = F = POH(2) + PQ; M1=0; M0=0$	JSP
0x009	$POH(0) = F = 0 \wedge 0; M1=0; M0=0$	JNXT
0x00A	$POH(0) = F = 0 \wedge 0; M1=0; M0=0$	JNXT
0x00B	$POH(0) = F = 0 \wedge 0; M1=0; M0=0$	JNXT
0x00C	$POH(0) = F = 0 \wedge 0; M1=0; M0=0$	JNXT
0x00D	$POH(0) = F = 0 \wedge 0; M1=0; M0=0$	JNXT
0x00E	$POH(0) = F = 0 \wedge 0; M1=0; M0=0$	JNXT
0x00F	$POH(0) = F = 0 \wedge 0; M1=0; M0=0$	JNXT
0x010	$PQ = F = POH(0) + POH(1); M1=0; M0=0$	CALL 0x007
0x011	$Y = F = 0 \vee PQ; M1=0; M0=0$	JSP

Задание 4.

Разработайте программу с обращением к подпрограмме из подпрограммы. Постройте диаграмму состояний БМУ в форме табл. 14, приняв уровень вложенности подпрограмм, равным 2.

Таблица 14

Выполняемый такт	T0	T1	T2	T3	T4	T5
Входы К1804ВУ1:						
<i>SO, SI</i>	00	I0	00	0I	00	I0
<i>FF, PUP</i>	IX*)	0I	IX	00	IX	IX
<i>R</i>	I2	X	X	X	0	X
Содержимое регистров:						
СМК	I	2	I3	I4	3	4
Стек 0	-	-	2	2	-	-
Стек 1,2,3	-	-	-	-	-	-
РГА	X	I2	X	X	X	0
Выход К1804ВУ1	I	I2	I3	2	3	0
Выход памяти	XSR12	MKE2	RTS	MR2	JPO	IKO
Содержимое РГМК	IKO	XSR12	MKE2	RTS	MR2	JPO

*) X – состояние безразлично.

Адрес	Микрокоманда	Прыжок
0x000	POH(1)=F=3v0; M1=0; M0=0	JNXT
0x001	POH(0)=F=2v0; M1=0; M0=0	CALL 0x010
0x002	Y=F=0v0; M1=0; M0=0	JNXT
0x003	Y=F=0v0; M1=0; M0=0	JNXT
0x004	Y=F=0v0; M1=0; M0=0	PUSH
0x005	Y=F=0v0; M1=0; M0=0	LDNXT
0x006	POH(2)=F=0vPQ; M1=0; M0=0	JNXT
0x007	PQ=F=POH(2)vPQ; M1=0; M0=0	RET
0x008	Y=F=0v0; M1=0; M0=0	JNXT
0x009	Y=F=0v0; M1=0; M0=0	JNXT
0x00A	Y=F=0v0; M1=0; M0=0	JNXT
0x00B	Y=F=0v0; M1=0; M0=0	JNXT
0x00C	Y=F=0v0; M1=0; M0=0	JNXT
0x00D	Y=F=0v0; M1=0; M0=0	JNXT
0x00E	Y=F=0v0; M1=0; M0=0	JNXT
0x00F	Y=F=0v0; M1=0; M0=0	JNXT
0x010	PQ=F=POH(0)+POH(1); M1=0; M0=0	CALL 0x006
0x011	Y=F=0v0; M1=0; M0=0	RET

Задание 5.

Изучите программу, представленную в символической записи функций перехода на рис. 6, где числами обозначены адреса ячеек.

Адрес	Микрокоманда	Прыжок
0x000	Y=F=0v0; M1=0; M0=0	CALL 0x00C
0x001	Y=F=0v0; M1=0; M0=0	CALL 0x006
0x002	Y=F=0v0; M1=0; M0=0	RET
0x003	Y=F=0v0; M1=0; M0=0	CALL 0x00C
0x004	Y=F=0v0; M1=0; M0=0	CALL 0x009
0x005	Y=F=0v0; M1=0; M0=0	RET
0x006	Y=F=0v0; M1=0; M0=0	CALL 0x00C
0x007	Y=F=0v0; M1=0; M0=0	CALL 0x003
0x008	Y=F=0v0; M1=0; M0=0	RET
0x009	Y=F=0v0; M1=0; M0=0	JNXT
0x00A	Y=F=0v0; M1=0; M0=0	RET
0x00B	Y=F=0v0; M1=0; M0=0	JNXT
0x00C	Y=F=0v0; M1=0; M0=0	RET
0x00D	Y=F=0v0; M1=0; M0=0	JNXT
0x00E	Y=F=0v0; M1=0; M0=0	CALL 0x000
0x00F	Y=F=0v0; M1=0; M0=0	CALL 0x00D

Вывод

В данной работе мы изучили устройства стека MT1804 и научились использовать его для создания нелинейных программ, обладающих большим функционалом в отличие от линейных.