


"УТВЕРЖДАЮ"

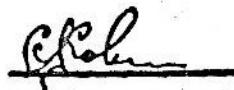
ЗАМ. РУКОВОДИТЕЛЯ ПРЕДПРИЯТИЯ

 Ю. А. ПЛАТОНОВ

1988 г.

МИКРОСХЕМЫ ИНТЕГРАЛЬНЫЕ
КМ1801ВМЗА, КМ1801ВМЗБ, КМ1801ВМЗВ, КН1801ВМЗА,
КН1801ВМЗБ, КН1801ВМЗВ
ТЕХНИЧЕСКОЕ ОПИСАНИЕ
ШИЗ.480.167 ТО

/ Главный конструктор ОКР

 П. Р. МАШЕВИЧ

1988

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата
903942	"30.08.88"	90394		

от 25.08.88 / 19.08.88 / 60.08.88

СОДЕРЖАНИЕ

	Лист
I. ВВЕДЕНИЕ	3
2. НАЗНАЧЕНИЕ	4
3. ТЕХНИЧЕСКИЕ ДАННЫЕ	6
4. ОПИСАНИЕ ВЫВОДОВ ПРЦ	65
5. ОПИСАНИЕ СТРУКТУРНОЙ СХЕМЫ ПРЦ	25
6. ФУНКЦИОНАЛЬНОЕ ОПИСАНИЕ ИНТЕРФЕЙСА ПРЦ	40
7. РАБОТА ПРЦ	64
8. СИСТЕМА КОМАНД	72
9. ДИСПЕТЧЕР ПАМЯТИ	155
10. ПРЕРЫВАНИЕ ПРЦ	176
II. РАЗЛИЧИЕ В РАБОТЕ МИКРОСХЕМ КМ1801ВМ2 И КМ1801ВМ3	179
ПРИЛОЖЕНИЕ I. УСЛОВНЫЕ ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ	181
ПРИЛОЖЕНИЕ 2. СИСТЕМА КОМАНД	184

Проб. примен.
ШМЗ.480.167

справ. №

Подп. и дата

ШМЗ. № докум.

Взам. инв. №

Подп. и дата

ШМЗ. № подл.

90394
3.06.88
90394

ШМЗ.480.167 ТО

Изм.	Лист	№ докум.	Подп.	Дата	Микросхемы интегральные	Лит.	Лист	Листов
Разр.	Давыдов	90394	Горский	23.88	КМ1801ВМ3А, КМ1801ВМ3Б, КМ1801ВМ3В, КМ1801ВМ3А	А	2	192
Проб.	Горский	90394	Горский	23.88	КМ1801ВМ3Б, КМ1801ВМ3В			
Контр.	Сомичева	90394	Горский	23.88	КМ1801ВМ3Б, КМ1801ВМ3В			
Упр.					Техническое описание			

426-9000 от 21.11.88

I. ВВЕДЕНИЕ

Настоящее техническое описание (ТО) предназначено для изучения микросхем КМ1801ВМЗА, КМ1801ВМЗБ, КМ1801ВМЗВ и КН1801ВМЗА, КН1801ВМЗБ, КН1801ВМЗВ.

Условные обозначения и сокращения, принятые в настоящем ТО, приведены в приложении I.

Система команд приведена в приложении 2.

Инв. № подл. 903942	Подп. и дата "3.06.88" В.В.	Взам инв. № 90394	Инв. № дубл.	Подп. и дата
2	Зал	ЦН502-88	4502	13.03
Изм	Лист	№ докум.	Подп.	Дата
ШИЗ.480.167 ТО				Лист 3

2. НАЗНАЧЕНИЕ

2.1. Микросхемы КМ1801ВМЗА, КМ1801ВМЗБ, КМ1801ВМЗВ и КН1801ВМЗА, КН1801ВМЗБ, КН1801ВМЗВ - однокристалльный шестнадцатизрядный микропроцессор (далее по тексту ПРЦ), помещенный в разных корпусах:

- КМ1801ВМЗА - в корпусе 2136.64-1,
- КМ1801ВМЗБ - в корпусе 2136.64-1,
- КМ1801ВМЗВ - в корпусе 2136.64-1,
- КН1801ВМЗА - в корпусе Н1864-2В,
- КН1801ВМЗБ - в корпусе Н1864-2В,
- КН1801ВМЗВ - в корпусе Н1864-2В,

предназначен для обработки цифровой информации, ПРЦ изготовлен по п-канальной технологии на МОП-транзисторах.

ПРЦ предназначен для встраивания в аппаратуру потребителя и может применяться:

- в составе технологического оборудования;
- в контрольно-измерительных комплексах;
- в системах обработки цифровой информации общего назначения.

2.2. Допустимые воздействующие факторы при эксплуатации

Микросхемы допускают воздействие на них механических нагрузок в соответствии с таблицей.

Механические нагрузки	:КМ1801ВМЗА:КН1801ВМЗА :КМ1801ВМЗБ:КН1801ВМЗБ :КМ1801ВМЗВ:КН1801ВМЗВ	
Вибрационные нагрузки		
диапазон частот, Гц	I-500	I-5000
максимальное ускорение, м/с ² (g)	100	400

Инв. № подл. 90394/2
 Подп. и дата "3.06.88" З.В.С.
 Взам инв. № 90394
 Инв. № док. № дубл.
 Подп. и дата

Изм	Лист	№ докум	Подп.	Дата
		111502-88		

ШИЗ.480.167 ТО

Механические нагрузки		:КМ1801ВМЗА:КН1801ВМЗА :КМ1801ВМЗБ:КН1801ВМЗБ :КМ1801ВМЗВ:КН1801ВМЗВ
Многokратные удары:		
максимальное ускорение, м/с ² (g)	750(75)	1500(150)
длительность удара, мс	-	1-5
Одиночные удары:		
максимальное ускорение, м/с ² (g)	1500(150)	15000(150)
длительность удара, мс	0,1-2,0	0,1-2,0
Линейные центробежные нагрузки, м/с²		
максимальное ускорение, м/с ² (g)	500	5000
	500(50)	5000(500)
Акустический шум:		
диапазон частот, Гц	-	50-10000
уровень звукового давления, дБ (относительно 2·10 ⁻⁵ Па)	-	170

Допускается эксплуатация микросхем в условиях воздействия на нее следующих климатических факторов:

Климатические факторы		:КМ1801ВМЗА:КН1801ВМЗА :КМ1801ВМЗБ:КН1801ВМЗБ :КМ1801ВМЗВ:КН1801ВМЗВ
Верхнее значение температуры окружающей среды, °С (°К)	+70(343)	+85(358)
Нижнее значение температуры окружающей среды, °С (°К)	минус 10 (263)	минус 45 (228)

Инв. № подл.	Взам. инв. №	Инв. № дубл.	Подп. и дата.
90394/2	3.06.88	8803/1	90394

Климатические факторы	КМ1801ВМЗА	КН1801ВМЗА
	КМ1801ВМЗБ	КН1801ВМЗБ
	КМ1801ВМЗВ	КН1801ВМЗВ
Многократное циклическое изменение температуры, °С (°К)	от +70(343) до минус 60(263)	от +85(358) до минус 60(263)
Верхнее значение атмосферного давления, Па	$0,27 \cdot 10^6$	$3 \cdot 10^5$
Нижнее значение атмосферного давления, Па	$0,66 \cdot 10^3$	$1,3 \cdot 10^{-4}$

2.3. Минимальная наработка микросхем в допустимых режимах и условиях составляет для:

- КМ1801ВМЗА - 50000 часов,
- КМ1801ВМЗБ - 50000 часов,
- КМ1801ВМЗВ - 50000 часов,
- КН1801ВМЗА - 50000 часов,
- КН1801ВМЗБ - 50000 часов,
- КН1801ВМЗВ - 50000 часов.

2.4. Питание ПРЦ производится от одного источника напряжения для:

- КМ1801ВМЗА - $+5В \pm 5\%$,
- КМ1801ВМЗБ - $+5В \pm 5\%$,
- КМ1801ВМЗВ - $+5В \pm 5\%$,
- КН1801ВМЗА - $+5В \pm 10\%$,
- КН1801ВМЗБ - $+5В \pm 10\%$,
- КН1801ВМЗВ - $+5В \pm 10\%$.

Инв. № подл.	Подп. и дата
90394	3.06.88
Взам. инв. №	Инв. № дубл.
90394	

Лист	№ докум.	Подп.	Дата
5а			

ШИЗ.480.167 ТО

Лист
5а

3. ТЕХНИЧЕСКИЕ ДАННЫЕ

3.1. ПРЦ выполнен по п-канальной МОП-технологии. Кристалл, содержащий около двухсот тысяч элементов, имеет размер 6,65x8,0 мм и помещен в шестидесятичетырех выводной металлокерамический корпус:

2I36.64-I для КМ1801ВМЗА

2I36.64-I для КМ1801ВМЗБ

2I36.64-I для КМ1801ВМЗВ

типа HI864-2В для КН1801ВМЗА

типа HI864-2В для КН1801ВМЗБ

типа HI864-2В для КН1801ВМЗВ

Условное графическое обозначение ПРЦ КМ1801ВМЗА, КМ1801ВМЗБ, КМ1801ВМЗВ приведено на рис. I, а КН1801ВМЗА, КН1801ВМЗБ, КН1801ВМЗВ - рис. Ia.

3.2. Информация представляется в дополнительном двоичном коде с фиксированной запятой.

3.3. Разрядность для чисел и команд - шестнадцать двоичных разрядов с возможностью представления и обработки восьми и тридцатидвух разрядных слов.

3.4. Система команд - безадресная, одноадресная, дваадресная.

3.5. Виды адресации:

регистравая, косвенно-регистравая, автоинкрементная, автодекрементная, косвенно-автодекрементная, индексная, косвенно-индексная.

3.6. Число регистров общего назначения - 8.

3.7. Количество каналов передачи информации - I.

3.8. Количество уровней запроса прерывания - 4.

Обработка внешних и внутренних прерываний выполняется с

Инв. № подл.	Подп. и дата
903942	3.06.88
Взам. инв. №	Инв. № дубл.
90394	

Изм.	Лист	№ докум.	Подп.	Дата	ШИЗ.480.167 ТО	Лист
						6

помощью памяти магазинного типа (стека).

3.9. Количество команд 74.

3.10. Максимальный объем адресуемой памяти 4М байта.

3.11. Тактовая частота:

микросхема КМ1801ВМ3А - не более 6 МГц

микросхема КМ1801ВМ3Б - не более 5 МГц

микросхема КМ1801ВМ3В - не более 4 МГц

микросхема КН1801ВМ3А - не более 8 МГц

микросхема КН1801ВМ3Б - не более 6 МГц

микросхема КН1801ВМ3В - не более 4 МГц

3.12. Время выполнения команды типа "сложение" при регистровом методе адресации - 3Т (Т- период тактовой частоты).

3.13. Время выполнения команды "умножение" - 60Т.

3.14. Время выполнения команды "деление" - 93Т.

3.15. Электрические параметры ПРЦ при температуре +25°C приведены в табл. I для КН1801ВМ3 и в табл. Ia для КМ1801ВМ3.

3.16. Предельно-допустимые значения электрических параметров в диапазоне температур для КМ1801ВМ3 от минус 10°C до плюс 70°C:

максимальное напряжение источника питания $U_{CC} = +7В$;

максимальное входное напряжение U_{IA} не более U_{CC} ;

минимальное входное напряжение U_{IB} не менее 0В;

емкость нагрузки C_L - не более 100 пФ.

3.17. Для КН1801ВМ3 в диапазоне температур от минус 45°C до плюс 85°C:

напряжение источника питания не более 5,5В;

напряжение входного сигнала низкого уровня U_{IL} не более 0,4В;

напряжение входного сигнала высокого уровня U_{IH} не

Инв. № подл	Подп. и дата	Взам. инв. №	Инв. № док.	Подп. и дата
903942	3.06.88	8808	90394	

Э	Зам	ИИ/502-88	ВЗМ	3.06.88
Изм	Лист	№ док.	Подп.	Дата

ШИЗ.480.167 ТО

Лист
6а

менее 2,2В;

емкость нагрузки C_L - не более 150 пФ.

4. ОПИСАНИЕ ВЫВОДОВ ПРЦ

Обозначение и наименование выводов представлены в табл.2.
Электрические схемы входных и выходных элементов ПРЦ при-
ведены на рис. 2-7.

Инв. № подл.	Подп. и дата	Взам инв. №	Инв. № дубл.	Подп. и дата
903942	" 3.06.88	5002/13		
2	Нов.	44502-88	6602/13	13.06.88
Изм	Лист	№ докум	Подп.	Дата

ИИЗ.480.167 ТО

Лист
60

Условное - графическое обозначение
 КМ1801ВМ3А, КМ1801ВМ3Б, КМ1801ВМ3В

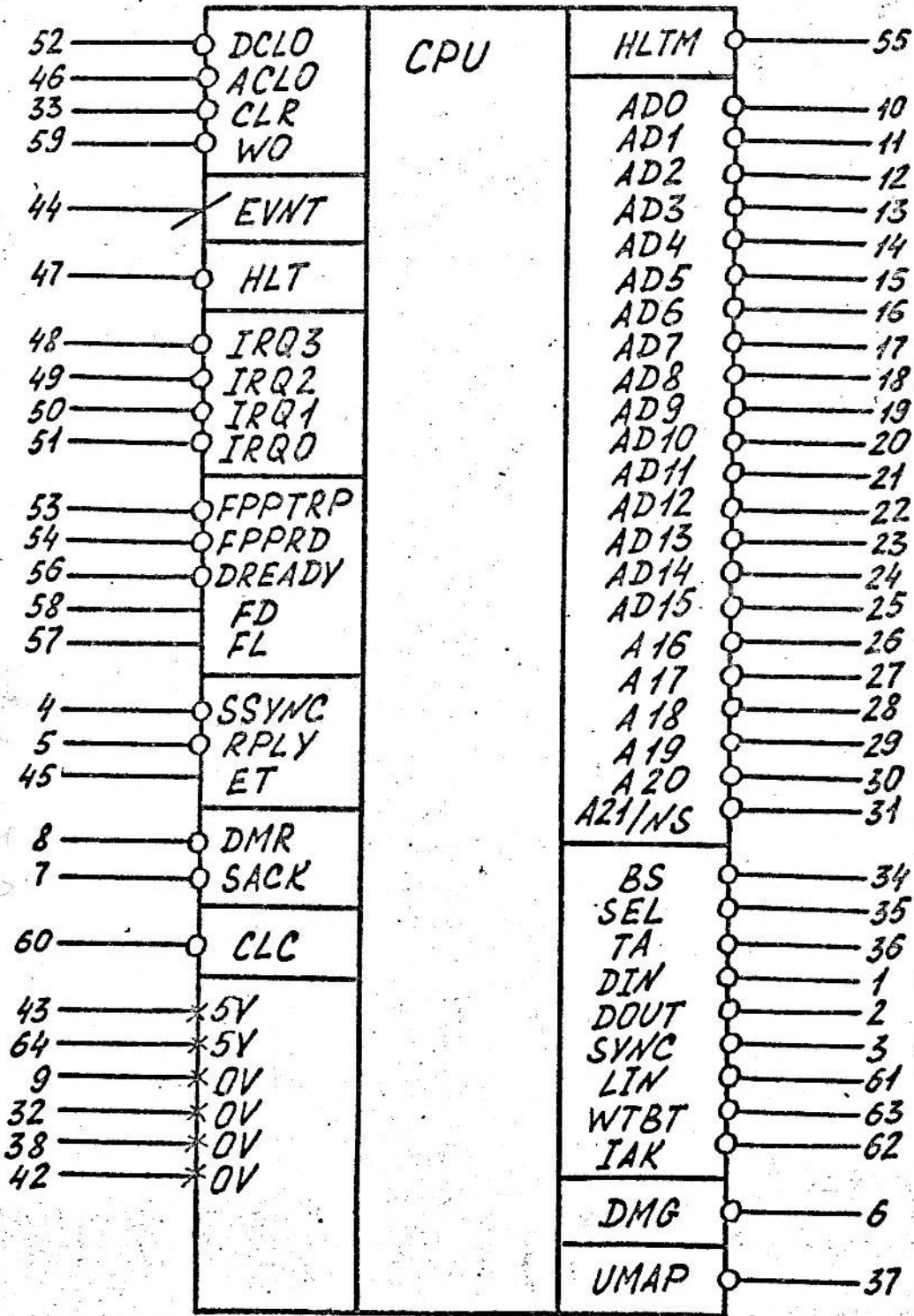


Рис. 1

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата
903942	3.06.88	90394		

Условное - графическое обозначение
 КН1801ВМ3А, КН1801ВМ3Б, КН1801ВМ3В

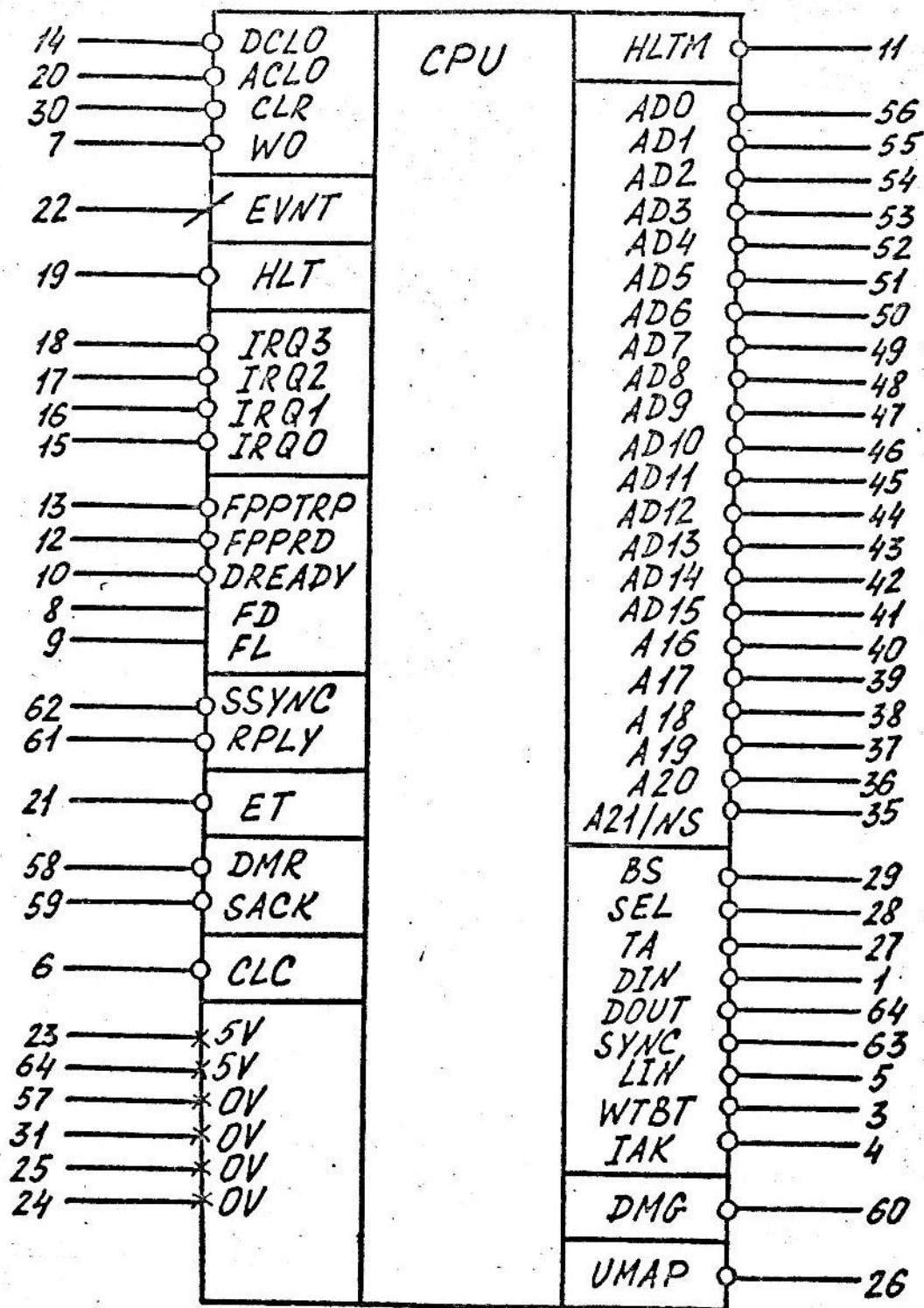


Рис. 1а

№ док. по подл.	№ док. по дата	№ док. по дата	№ док. по дата
90394/2	3.06.88	90394	

2	Зам	ИИ502-88	ИИ502-88	ИИ502-88
ИИ502-88	ИИ502-88	ИИ502-88	ИИ502-88	ИИ502-88

ИИ5.480.167 Т0

Лист
7а

Таблица I

Наименование параметра, единица измерения	Буквен- ное обозна- чение	Норма		Темпера- тура, °C
		не менее	не более	
Ток потребления, мА	I_{cc}	-	290	25±10 -45 85
Выходное напряжение низкого уровня, В при $I_{OL} = 1,6 \text{ мА} \pm 2\%$	U_{OL}	-	0,4	25±10 -45 85
Выходное напряжение высокого уровня, В при $I_{OH} = -0,15 \text{ мА} \pm 5\%$	U_{OH}	2,4	-	25±10 -45 85
Ток утечки по входам, отклю- чаемым входам/выходам, мА.	I_{L1} I_{L0}	-	50	25±10 -45 85
Входное напряжение низкого уровня, В	U_{IL}	-	0,8	25±10 -45 85
Входное напряжение высокого уровня, В	U_{IH}	2,2	-	25±10 -45 85
Максимальная тактовая час- тота функционирования, МГц	$f_c \text{ max}$	6	-	25±10 -45 85

Инв. № подл. 903942 Подп. и дата. 3.06.88
 Взам инв. № 90394
 Инв. № дубл.

Инв. № 2
 Изм. Лист № докум. Подп. Дата.

ИИЗ.480.167 ТО

Лист
8

Таблица 1а

Наименование параметра	Буквенное обозначение	Норма		Выходной ток мА	Температура, °C
		не менее	не более		
Выходное напряжение низкого уровня, В	U_{OL}	-	0,45	$3,2 \pm 2\%$	$+25 \pm 10$
Выходное напряжение высокого уровня, В	U_{OH}	2,45	-	$-0,2 \pm 15\%$	$+25 \pm 10$
Выходное напряжение низкого уровня, В	U_{OL}	-	0,5	$3,2 \pm 2\%$	-10 ± 3 $+70 \pm 3$
Выходное напряжение высокого уровня, В	U_{OH}	2,4	-	$0,2 \pm 15\%$	-10 ± 3 $+70 \pm 3$
Ток утечки низкого уровня на входе, мкА	I_{LIL}	-	1	-	$+25 \pm 10$
Ток утечки высокого уровня на входе, мкА	I_{LIH}	-	1	-	$+25 \pm 10$
Ток утечки низкого уровня на выходе, мкА	I_{LOL}	-	10	-	$+25 \pm 10$
Ток утечки высокого уровня на выходе, мкА	I_{LOH}	-	10	-	$+25 \pm 10$
Ток потребления, мА	I_{CC}	-	310	-	$+25 \pm 3$
Входное напряжение низкого уровня, В	U_{IL}	-	0,7	-	-10 ± 3 $+70 \pm 3$
Ток утечки низкого уровня на входе, мкА	I_{LIL}	-	10	-	-10 ± 3 $+70 \pm 3$
Ток утечки высокого уровня на входе, мкА	I_{LIH}	-	10	-	-10 ± 3 $+70 \pm 3$

№ подл.	№ подл.	№ подл.	№ подл.	№ подл.
90394	10.07.8708	10.07.8708	10.07.8708	10.07.8708

ШИЗ.480.167 ТО

Лист

9

Наименование параметра	Буквенное обозначение	Норма		Выходной ток, мА	Температура, °C
		не менее	не более		
Ток утечки низкого уровня на выходе, мкА	I_{LOL}	-	50	-	$-10_{\pm 3}$ $+70_{\pm 3}$
Ток утечки высокого уровня на выходе, мкА	I_{LOH}	-	50	-	$-10_{\pm 3}$ $+70_{\pm 3}$
Входное напряжение высокого уровня, В	U_{IH}	2,2	-	-	$+25_{\pm 10}$ $-10_{\pm 3}$ $+70_{\pm 3}$

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата
90394	12.07.87	8708		

ИИЗ.480.167 ТО

Лист

10

Таблица 2

Номер вывода ПРЦ		Обозна- чение	Тип вывода	Активный уровень	Наименование
КН1801ВМ3А:КМ1801ВМ3А:					
КН1801ВМ3Б:КМ1801ВМ3Б:					
КН1801ВМ3В:КМ1801ВМ3В:					
I	I	<i>DIN</i>	выход	низкий	Сигнал управления вводом данных
64	2	<i>DOUT</i>	выход	низкий	Сигнал управления выводом данных
63	3	<i>SYNC</i>	выход	низкий	Сигнал синхронизации обмена
62	4	<i>SSYNC</i>	вход	низкий	Сигнал синхронизации устройства
61	5	<i>RPLY</i>	вход	низкий	Сигнал ответа приемника информации
60	6	<i>DMG</i>	выход	низкий	Сигнал разрешения прямого доступа к памяти
59	7	<i>SACK</i>	вход	низкий	Сигнал подтверждения запроса прямого доступа к памяти
58	8	<i>DMR</i>	вход	низкий	Сигнал запроса прямого доступа к памяти
57	9	<i>OV</i>			Общий
56	10	<i>ADD</i>	вход /выход/	низкий	Нулевой разряд СМ
55	11	<i>AD1</i>	вход /выход/	низкий	Первый разряд СМ

Инв. № подл.	Подп. и дата.	Взам. инв. №	Инв. № дубл.	Подп. и дата.
903942	3.06.88	90394		

ЩИЗ.480.167 ТО

Лист

11

Номер вывода ПРЦ		Обозначение	Тип вывода	Активный уровень	Наименование
КН1801ВМ3А:КМ1801ВМ3А:	КН1801ВМ3Б:КМ1801ВМ3Б:				
КН1801ВМ3В:КМ1801ВМ3В:					
54	I2	AD2	ВХОД /ВЫХОД/	низкий	Второй разряд СМ
53	I3	AD3	ВХОД /ВЫХОД/	низкий	Третий разряд СМ
52	I4	AD4	ВХОД /ВЫХОД/	низкий	Четвертый разряд СМ
51	I5	AD5	ВХОД /ВЫХОД/	низкий	Пятый разряд СМ
50	I6	AD6	ВХОД /ВЫХОД/	низкий	Шестой разряд СМ
49	I7	AD7	ВХОД /ВЫХОД/	низкий	Седьмой разряд СМ
48	I8	AD8	ВХОД /ВЫХОД/	низкий	Восьмой разряд СМ
47	I9	AD9	ВХОД /ВЫХОД/	низкий	Девятый разряд СМ
46	20	AD10	ВХОД /ВЫХОД/	низкий	Десятый разряд СМ
45	21	AD11	ВХОД /ВЫХОД/	низкий	Одиннадцатый разряд СМ
44	22	AD12	ВХОД /ВЫХОД/	низкий	Двенадцатый разряд СМ
43	23	AD13	ВХОД /ВЫХОД/	низкий	Тринадцатый разряд СМ
42	24	AD14	ВХОД /ВЫХОД/	низкий	Четырнадцатый разряд СМ

Инв. № подл. 903942
 Подп. и дата. 3.06.88
 Возм. инв. №. 90394
 Инв. № дубл.
 Подп. и дата.

Номер вывода ПРЦ		Обозначение	Тип вывода	Активный уровень	Наименование
КН1801ВМ3А:КМ1801ВМ3А	КН1801ВМ3Б:КМ1801ВМ3Б				

41	25	AD 15	ВХОД /ВХОД/	низкий	Пятнадцатый разряд СМ
40	26	A 16	ВЫХОД	низкий	Шестнадцатый разряд адреса
39	27	A 17	ВЫХОД	низкий	Семнадцатый разряд адреса
38	28	A 18	ВЫХОД	низкий	Восемнадцатый разряд адреса
37	29	A 19	ВЫХОД	низкий	Девятнадцатый разряд адреса
36	30	A 20	ВЫХОД	низкий	Двадцатый разряд адреса
35	31	A 21/NS	ВЫХОД	низкий	Сигнал - адрес /инструкция
31	32	OV			Общий
30	33	CLR	ВХОД /ВЫХОД/	низкий	Сигнал установки
29	34	BS	ВЫХОД	низкий	Сигнал обращения к банку внешних устройств
28	35	SEL	ВЫХОД	низкий	Сигнал выборки при моде
27	36	TA	ВЫХОД	низкий	Сигнал выдачи адреса
26	37	UMAP	-	-	Разрешение преобразования адресов "Общая шина"
25	38	OV	-	-	Общий
34	39	SP	-	-	Свободный выход

Инв. № подл.	Подп. и дата	Взам инв. №	Инв. № докл.	Подп. и дата
903942	3.06.88	90394		

Инв. № подл.	Подп. и дата	Взам инв. №	Инв. № докл.	Подп. и дата	ШИЗ.480.167 Т0	Лист
903942	3.06.88	90394				13

Номер вывода ПРЦ	Обозначение	Тип вывода	Активный уровень	Наименование
КН1801ВМ3А:КМ1801ВМ3А:				
КН1801ВМ3Б:КМ1801ВМ3Б:				
КН1801ВМ3В:КМ1801ВМ3В:				

33	40	SP	-	-	Свободный выход
32	41	SP	-	-	Свободный выход
24	42	OV	-	-	Общий
23	43	Ucc	-	-	Вывод питания от источника напряжения
22	44	EVNT	вход	-	Сигнал радиального прерывания
21	45	ET	вход	низкий	Сигнал разрешения зависания
20	46	ACLO	вход	низкий	Сигнал включения источника питания переменного напряжения
19	47	HALT	вход	низкий	Сигнал останова
18	48	IRQ3	вход	низкий	Сигнал запроса на прерывание с уровнем приоритета 7
17	49	IRQ2	вход	низкий	Сигнал запроса на прерывание с уровнем приоритета 6
16	50	IRQ1	вход	низкий	Сигнал запроса на прерывание с уровнем приоритета 5
15	51	IRQ0	вход	низкий	Сигнал запроса на прерывание с уровнем приоритета 4

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата
903942	3.06.88 В.м.	90394		

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата
2	30.04.88	441502-88		
Изм	Лист	№ докум.	Подп.	Дата

ШИЗ.480.167 ТО

Номер вывода ПРЦ	Обозна- чение	Тип вывода	Активный уровень	Наименование
КН1801ВМ3А:КМ1801ВМ3А				
КН1801ВМ3Б:КМ1801ВМ3Б				
КН1801ВМ3В:КМ1801ВМ3В				

14	52	DCLO	вход	низкий	Сигнал включения источника питания постоянного напряжения
13	53	FPTRP	вход	низкий	Сигнал прерывания ШИЗ- процессора
12	54	FPPRD	вход	низкий	Сигнал готовности ШИЗ -процессора
11	55	HALTM	выход	низкий	Сигнал отладочного режима
10	56	DREADY	вход	низкий	Сигнал готовности данных
9	57	FL	вход	высокий	Сигнал длинного целого ШИЗ-процессора
8	58	FD	вход	высокий	Сигнал двойной точности ШИЗ-процессора
7	59	WO	вход	-	Сигнал режима начального включения
6	60	CLC	вход	-	Тактовый импульс
5	61	LIN	выход	низкий	Сигнал нагрузки команды
4	62	IAC	выход	низкий	Сигнал разрешения запроса на прерывание
3	63	WTBT	выход	низкий	Сигнал управления запись-байт
2	64	Ucc	-	-	Вывод питания от источника напряжения

Инв. № подл. Подп. и дата. 90394/2 3.06.88 0802/115

Взам инв. №. Инв. № докл. Подп. и дата. 90394

Инв. № подл.	Подп.	Дата	Инв. № докл.	Подп.	Дата	ШИЗ.480.167 TO	Лист 15
90394/2	3.06.88	0802/115					

Вывод X, \overline{DIN}

Выход сигнала "чтение данных". Этот сигнал используется в двух процедурах СМ:

1) во время сигнала \overline{SYNC} , в момент, когда ПРЦ готов принять данные /процедура чтения информации/;

2) при отсутствии сигнала \overline{SYNC} , низкий уровень этого сигнала фиксирует состояние запросов на прерывание от устройств на СМ /процедура приема адреса вектора прерывания/.

Вывод X, \overline{DOVT}

Выход сигнала "запись данных". Появление этого сигнала /низкий уровень/ свидетельствует о том, что при выполнении ПРЦ процедуры записи данных, на $AD\ 0-15$ выставлены истинные данные.

Вывод X, \overline{SYNC}

Выход сигнала "обмен". Наличие этого сигнала указывает на то, что адрес обмена выставлен на шину $AD\ 0-15$ и $AI6-2I$. Обмен продолжается до тех пор, пока \overline{SYNC} равен нулю.

Вывод X, \overline{SSYNC}

Вход сигнала "синхронизация устройства". Выставляется ведомым устройством при опознании им своего адреса.

Вывод X, \overline{RPLY}

Вход сигнала "ответ". Выставляется /низким уровнем/ в ответ на сигналы \overline{DIN} , или \overline{DOVT} , или во время обмена, посредством сигнала \overline{TAK} . Указывает на то, что данные выставлены на шине $AD0-15$ или приняты во время записи.

Вывод X, \overline{DMG}

Выход сигнала разрешения на захват магистрали. Низкий уровень этого сигнала выставляется процессором в случае свободной магистрали.

Вывод X, \overline{SACK}

Имя, № подл.	Подп. и дата
90394	10.07.870Вел.
Взам.имб. №	Имб. № дубл.
Подп. и дата	Подп. и дата

ШИЗ.480.167 Т0

Лист 16

Вход сигнала подтверждения захвата магистрали. Выставляется /низким уровнем/ устройством в ответ на разрешение на захват магистрали /сигнал *DMG 1*.

Вывод 8, *DMR*

Вход сигнала "запрос на захват магистрали". Устройство выставляет этот сигнал /низким уровнем/ и ожидает разрешения от ПРЦ на захват СМ.

Вывод 9, *OV*

Общий вывод (земля)

Выводы 10-25, *AD0-15*

16 входов /выходов/ совмещенной по адресам и данным СМ. Предназначены для передачи и приема адресов и данных. Совмещение использования одних и тех же выводов для передачи как адресов, так и данных, происходит на основе разделения во времени. Низкий уровень на одном из этих выводов соответствует передаче логической единицы.

Выводы 26-30, *A16-20*

5 старших разрядов адреса. Используются при обращении к внешним регистрам, при работе с длинным адресом.

Вывод 31, *A21/NS*

Во время переднего фронта сигнала *SYNC* вывод несет информацию об адресе, во время наличия сигнала *DIN* - о том, принимается команда или данные.

Выводы 32, *OV*

Общий вывод (земля).

Вывод 33, *CLR*

Этот сигнал используется для установки периферийной части системы в начальное состояние.

Вывод 34, *BS*

Низкий уровень сигнала *BS* вызывает обращение к области

Изм. №	Изм. №	Изм. №	Изм. №	Изм. №
90394	12.07.87	12.07.87	12.07.87	12.07.87
Подп.	Дата	Подп.	Дата	Подп.

ИИЗ.480.167 ТО

1457

17

памяти внешних устройств.

Вывод 35, *SEL*

Выход сигнала обращения к системной области памяти. Появление этого сигнала (низкий уровень) в фазе выдачи адреса свидетельствует о том, что обмен идет не с основной, а с дополнительной системной памятью.

Вывод 36, *TA*

Выход сигнала сопровождения выдачи адреса в цикле чтения. При наличии низкого уровня сигнала *TA* на магистраль выдается адрес.

Вывод 37, *UMAP*. Преобразование адресов *UNIBUS*

Выводы 39-41 представляют собой резервные входы/выходы ПРЦ.

Вывод 42, *33 OV*

Общий вывод (земля).

Вывод 43, *Vcc*

Источник питания

Вывод 44, *EVNT*

Вход сигнала запроса прерывания от таймера. Переход сигнала на этом выводе из низкого в высокий уровень свидетельствует о том, что таймер выставил запрос на прерывание.

Вывод 45, *ET*

Вход сигнала разрешения зависания. При наличии низкого уровня на этом выводе в случае зависания *CM* возникает прерывание.

Вывод 46, *ACLO*

Переход сигнала из высокого уровня в низкий уровень на этом выводе вызывает прерывание программы ПРЦ и переход на подпрограмму обработки прерывания по сбоям питания. Появление высокого уровня этого сигнала свидетельствует о нормальном состоянии сетевого питания и вызывает переход к выполнению микропрограммы начального пуска.

Инв. № подл.	90394	Полн. и дата	10.07.87
	90394		10.07.87
Узм. лист		Инв. № докл.	
№ док.ум.		Инв. №	
Полн.		Дата	

ИИЗ.480.167 ТО

Лист 18

Вывод 47, HLT

Вход сигнала "стоп". При наличии этого сигнала ПРЦ выходит в пультовой режим.

Выводы 48-51, IRQ3-0

Входы запросов на прерывание. Переход уровня сигналов на этих входах из высокого в низкий вызывает установку запросов на прерывание на соответствующих входах блоков прерываний ПРЦ.

Вывод 52, DCLO

Низкий уровень сигнала на этом входе вызывает установку в начальное состояние всех блоков ПРЦ.

Вывод 53, FPRTRP

Этот сигнал возникает в случае фатальных ошибок, в том числе в ШИЗ при выполнении инструкций обработки чисел с плавающей запятой.

Вывод 54, FPRRD

Во время начального включения низкий уровень этого сигнала указывает на то, что в системе существует ШИЗ. Во время выполнения программы установление низкого уровня этого сигнала указывает главному ПРЦ на то, что процессор готов начать выполнение следующей команды.

Вывод 55, HLTM

Выход сигнала "отладочный режим". Указывает на то, что процессор находится в пультовом режиме.

Вывод 56, DREADY

Вход сигнала "данные готовы". Этот сигнал выставляется при наличии истинных данных во время записи из ШИЗ.

Вывод 57, FL

Выставление /высоким уровнем/ этого сигнала говорит о том, что ШИЗ работает с длинными числами /целыми/.

Вывод 58, FD

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № докл.	Псал. и дата
90394	10.07.87			

ШИЗ.480.167 ТО

Этот сигнал определяет точность, с которой работает ШИЗ. Выставляется /высоким уровнем/, когда ШИЗ обрабатывает числа с двойной точностью.

Выход 59, W0

Если сигнал W0 выставлен /высоким уровнем/, то ПРЦ начинает выполнение программы, загружая счетчик команд содержимым ячейки 24, если W0 выставлен /низким уровнем/, то ПРЦ начинает выполнение программы, расположенной по адресу 173000.

Выход 60, CLC

Вход тактовых импульсов. Используется для синхронизации работы всех блоков ПРЦ.

Выход 61, LIM

Выход сигнала загрузки команды. Сигнал для ШИЗ загрузить очередную команду.

Выход 62, IAK

Выставляется ПРЦ в ответ на запрос на прерывание при возникновении условий обмена по СМ.

Выход 63, WTBT

Выход сигнала "запись/байт". Во время переднего фронта сигнала SYNC указывает на то, какой цикл обмена будет происходить: чтение или запись. Сигнал WTBT выставляется низким уровнем при процедуре записи. При процедуре записи во время выдачи данных /сигнал DOUT / этот сигнал выставляется /низким уровнем/, если выдается байт, а не слово.

Выход 64, Ucc

Источник питания.

Уменьш - 6.07.84

Инв. № подл.	90394	Подп. и дата	10.07.87	Взам. инв. №		Инв. № докл.		Подп. и дата	
--------------	-------	--------------	----------	--------------	--	--------------	--	--------------	--

ШИЗ.480.167 ТО

Лист
20

Выходной элемент выводов АДО-15, А16-21

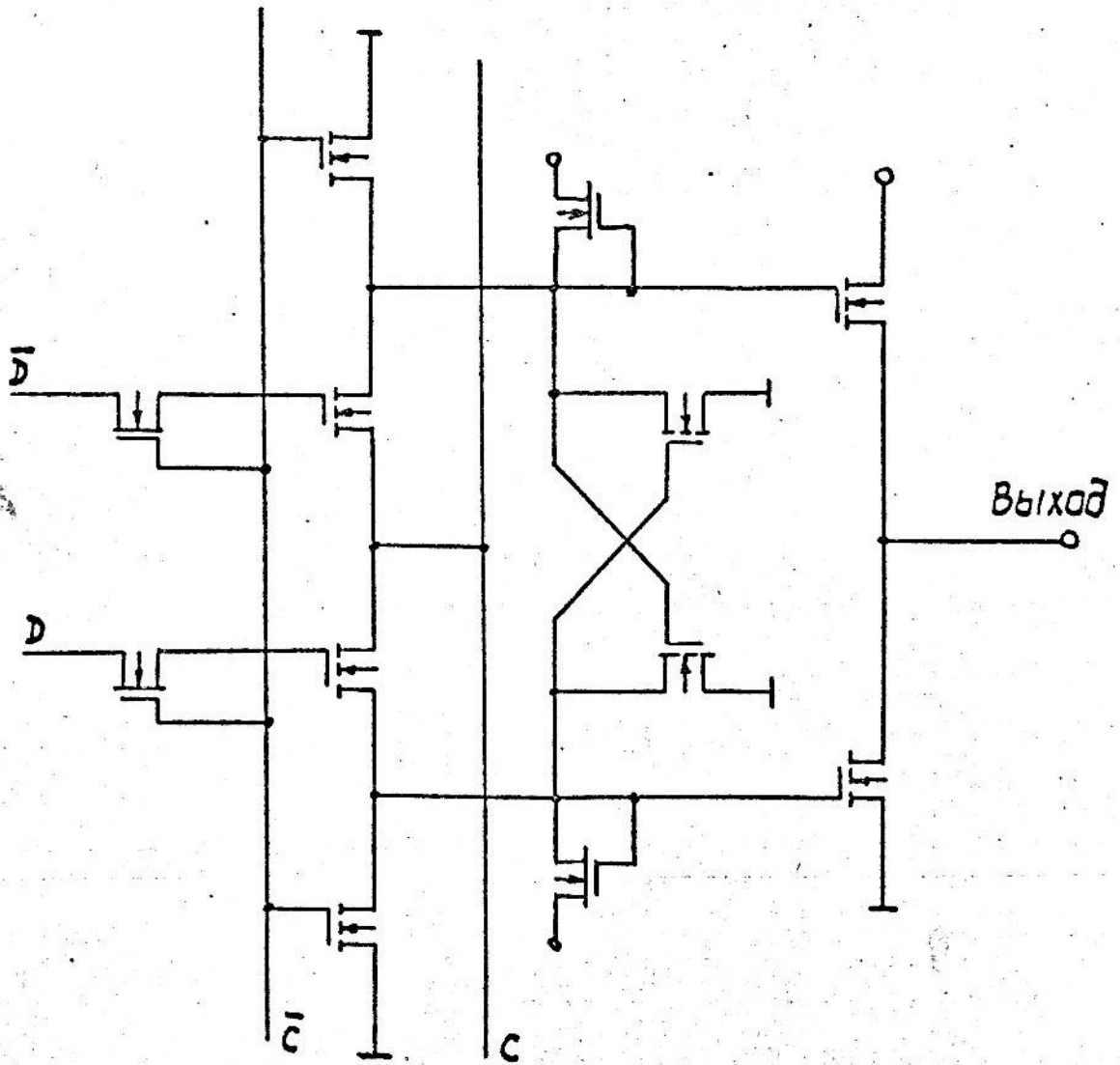


Рис. 2

Инв. № подл.	Лист	Взам. инв. №	Инв. № дубл.	Подп. и дата
90394		70.07.870Вн		
Инв. № подл.	Лист	№ докум.	Подп.	Дата
ИИЗ.480.167 ТО				Лист
0.57 ГОСТ 2.105-68				21
				СформатИ

Выходной элемент вывода А21/NS

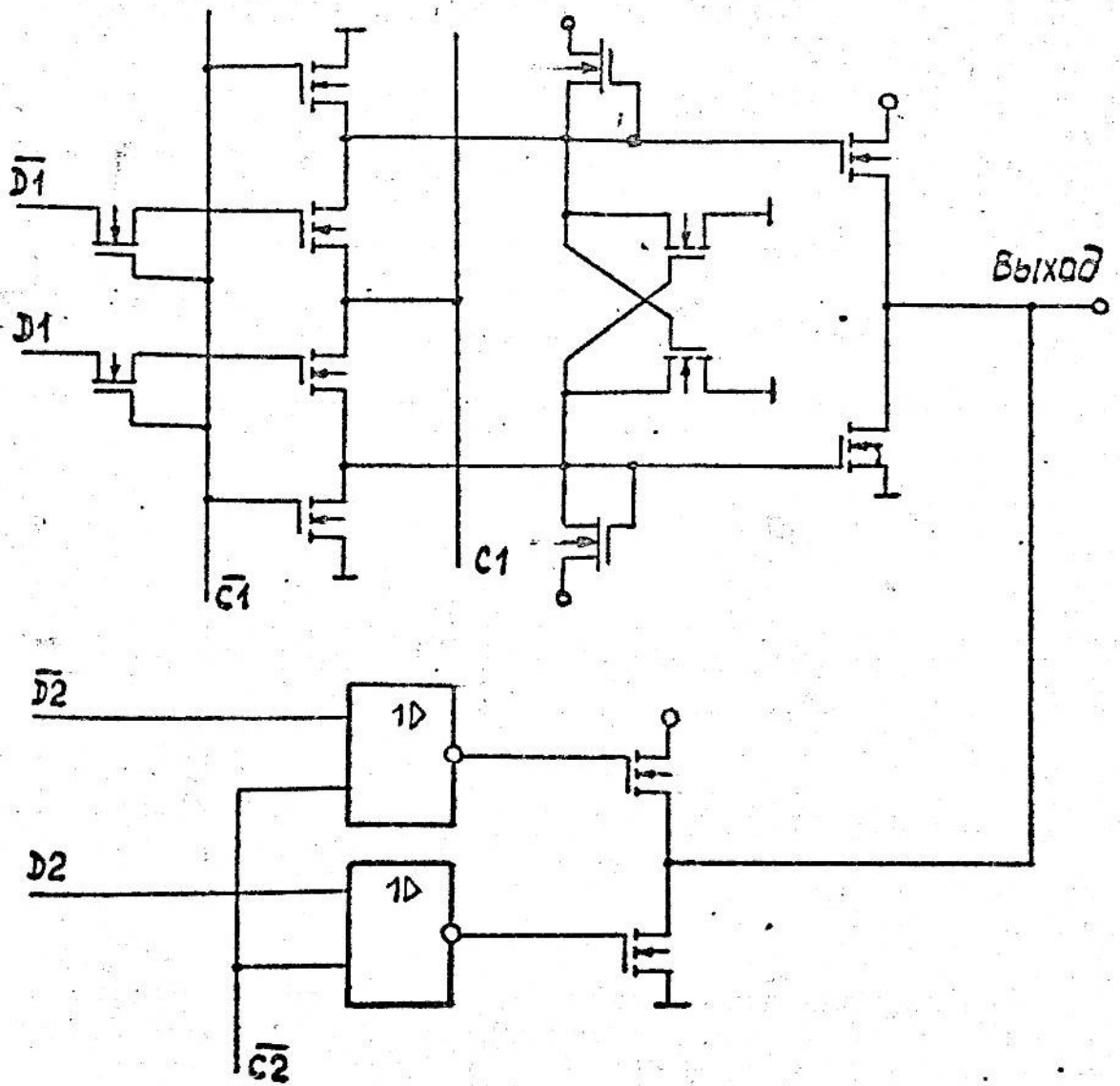


Рис. 3

Инв. № подл.	Подп. и дата	Узм. инв. №	Инв. № докл.	Подп. и дата
50394	12.07.87 Вш			

Узм. лист	№ докум.	Подп.	Дата	ИИЗ.480.167 ТО	Лист
					22

Выходной элемент выводов SYNC, DOUT, RPLY,
WTBT, DIN

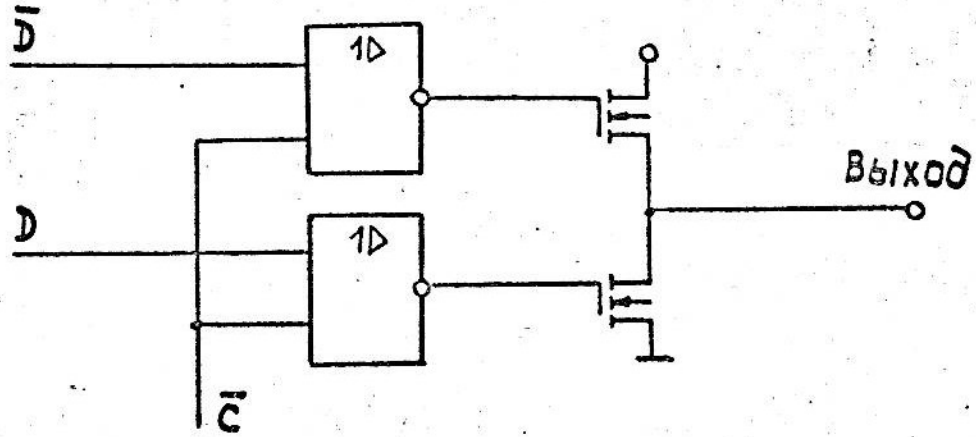


Рис. 4

Выходной элемент вывода FPPTRP

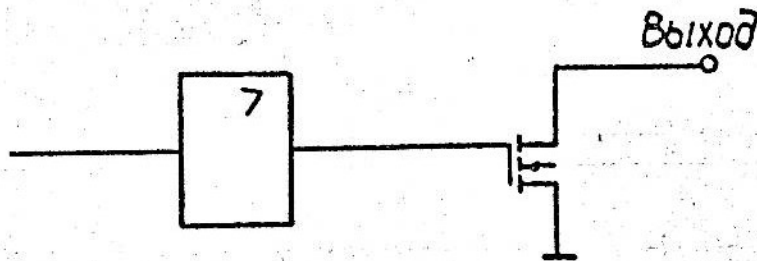


Рис. 5

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата
90394	10.07.870Вч.			

Изм.	Лист	№ докум.	Подп.	Дата

ИИЗ.480.167 ТО

Лист
23

Выходной элемент выводов DMG, HLTM,
IAK, LIN, UMAP, SEL, TA, BS

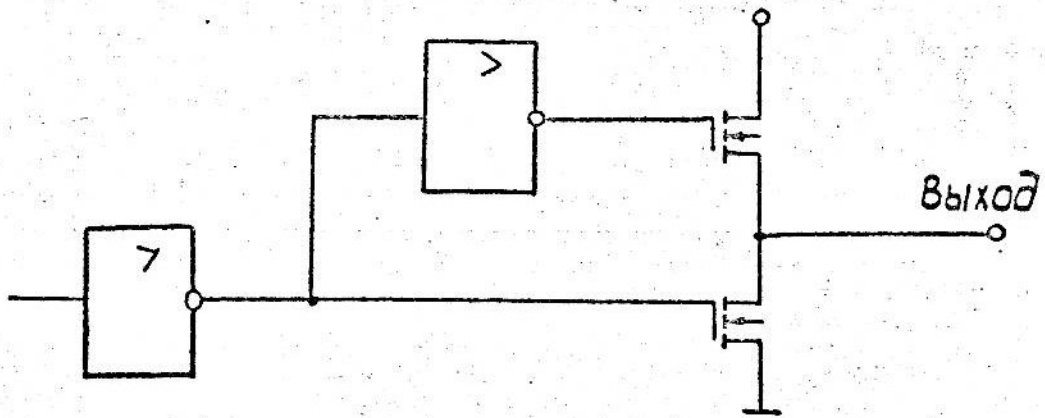


Рис. 6

Входной элемент выводов ADO-15, SSYNC, RPLY,
DMR, SACK, DCLO, ACLO, W0, HLT, IRQ3-0, FPPTRA,
FPPRD, FD, FL, DREADY, CLC, ET

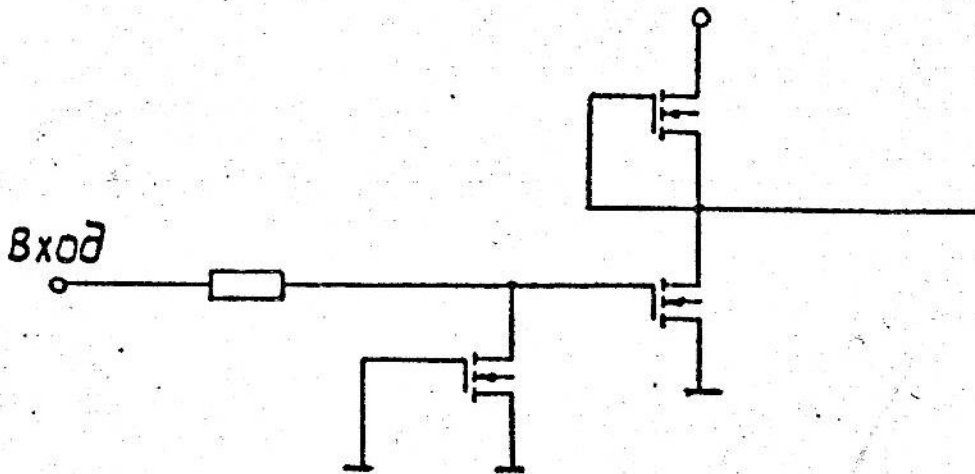


Рис. 7

Инв. № подл.	Подл. и дата	Взам. инв. №	Инв. № докл.	Подл. и дата
90394	10.07.8703н.			

ШИЗ.480.167 ТО

Лист
24

5. ОПИСАНИЕ СТРУКТУРНОЙ СХЕМЫ ПРЦ

5.1. Общая структурная схема ПРЦ представлена на рис. 8

ПРЦ состоит из следующих блоков:

операционный блок /ОБ/;

блок микропрограммного управления /БМУ/;

блок прерываний /БПР/;

диспетчер памяти /ДП/;

контроллер системной магистрали /КСМ/.

Блоки соединены между собой шиной адреса и данных, кото-
рая предварительно преобразовывается в блоке контроллера систем-
ной магистрали /КСМ/. Кроме того, блоки связаны специальными ин-
формационными и управляющими сигналами.

5.2. Описание структурной схемы операционного блока

ОБ /рис. 9 / предназначен для выполнения следующих функций:

прием данных и их хранение в регистрах;

прием и расшифровка микрокоманды;

выполнение арифметико-логических операций между регистрами
и константами, между парой регистров;

выдача данных в системную магистраль;

формирование адресов векторов прерывания;

формирование состояний.

Микропроцессор содержит шестнадцать основных регистров, из ко-
торых восемь являются регистрами общего назначения, один регистр
PSW - слово состояния процессора, ^{семь} ~~шесть~~ - внутренние служебные
регистры процессора. Программно доступными являются все регистры
общего назначения и регистр *PSW*. Остальные регистры доступны
микропрограммно, причем, регистры R0...R4 доступны лишь из поля
адресации команд. Назначения регистров и их назначение приведены
в табл. 3

Инв. № подл.	90394
Побл. и дата	10.07.87 Вч.
Взам. инв. №	
Инв. № док.	
Побл. и дата	

Инв. № подл.	90394	Лист	25
Лист	№ докум.	Побл.	Дата

ИИЗ.480.167 ТО

Структурная схема ПРЦ

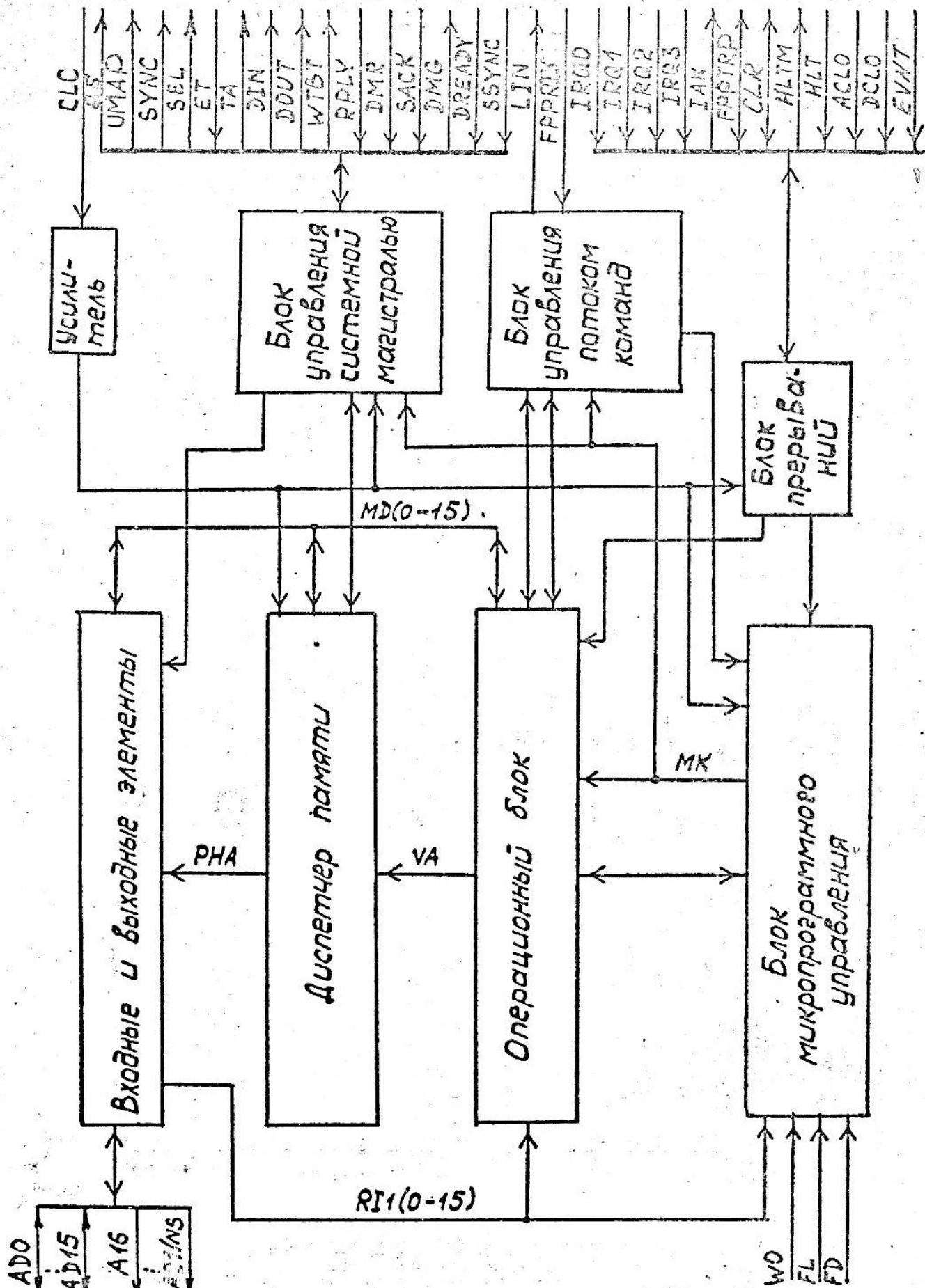


Рис. 8

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата
90394	10.07.87	87		

ИИЗ.480.167 Т0

Лист
26

Инв. № подл.	Подп. и дата	Экз. инв. №	Инв. № док. бл.	Подп. и дата
90394	10.07.87 <i>Вил</i>			

Структурная схема операционного блока

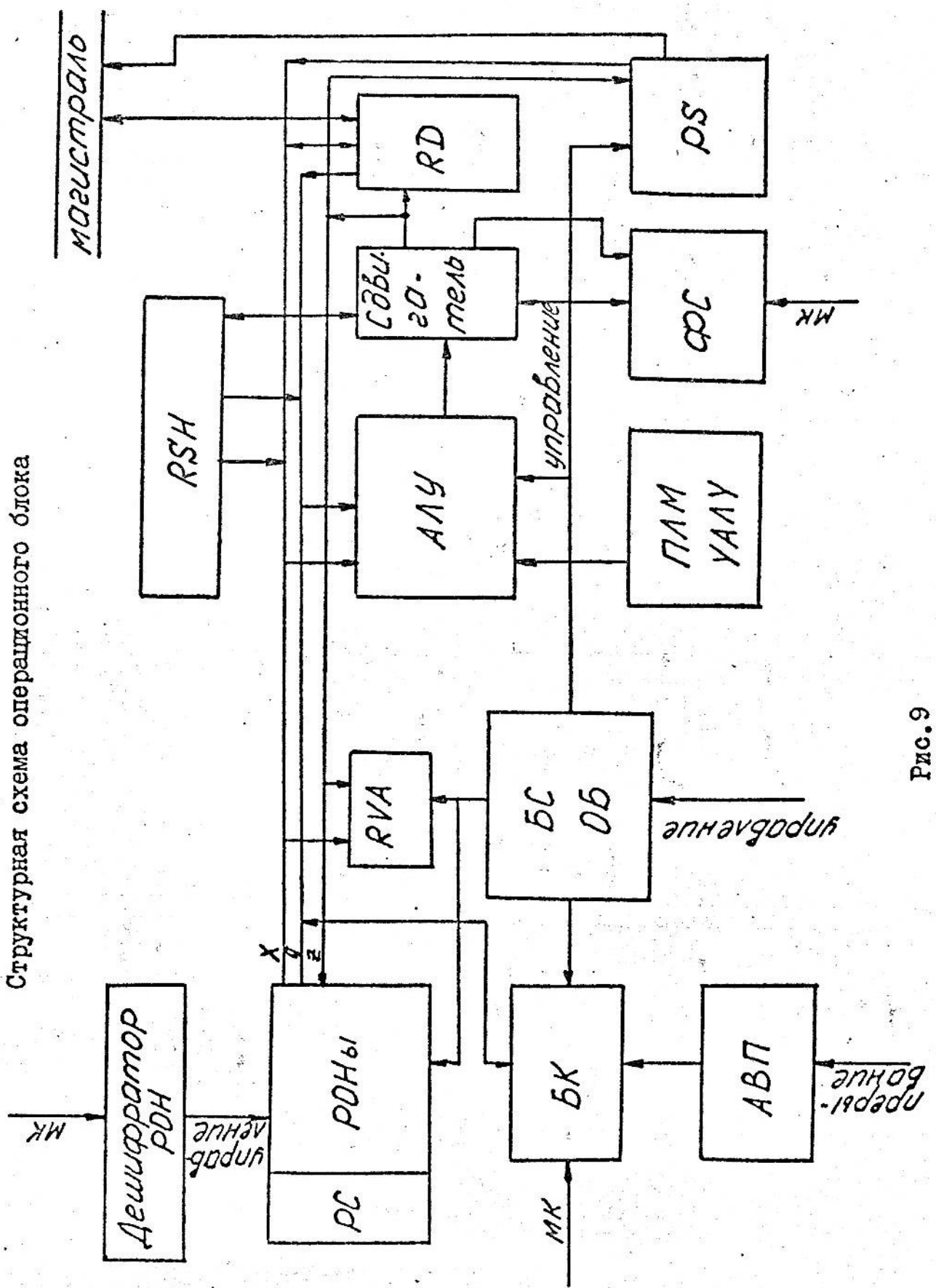


Рис. 9

ИИЗ.480.167 ТО

Изм. лист. № докум. Подп. Дата

9.54 ГОСТ 2.105-68

Лист
27

ГОСМАТ 114

Название регистра	Назначение
<i>R0... R5</i>	Регистры общего назначения
<i>R6 (KSP)</i>	Регистр-указатель стека в режиме операционной системы
<i>R16 (USP)</i>	Регистр-указатель стека в режиме пользователя
<i>R7 (PC)</i>	Регистр-счетчик команд
<i>R10</i>	Регистр для хранения промежуточных результатов вычислений; в режиме <i>HALT</i> - моды является указателем стека
<i>R11</i>	Регистр для хранения промежуточных результатов вычислений
<i>R14</i>	Регистр-счетчик тактов при выполнении команд расширенной арифметики
<i>RSH</i>	Сдвиговый регистр. Используется при выполнении команд расширенной арифметики и как аккумулятор
<i>PSW</i>	Регистр: слова состояния ПРЦ
<i>RVA</i>	Регистр виртуального адреса
<i>RD</i>	Буферный регистр данных

Регистры *R0...R6, R16, R10, R11* расположены в блоке РОН, который содержит дешифратор и накопитель. Дешифратор РОН представляет собой блок, который получая на входе МК, обрабатывает ее и получает на выходе сигналы, управляющие работой блока РОН.

Регистр *RVA* предназначен для хранения виртуального

Инв. № подл.	90394
Полн. и дата	10.07.87обл.
Узм. инв. №	
Инв. № дубл.	
Полн. и дата	

Изм.	Лист	№ докум.	Полн.	Дата

ШИЗ.480.167 Т0

адреса на время преобразования его в диспетчере памяти или в случае занятости регистра физического адреса. Загрузка *RVA* возможна как в фазе чтения, так и в фазе записи ОБ.

Регистр *RI4* является шестиразрядным регистром счетчиком, доступным только по записи. В регистр заносится параметр, определяющий число тактов работы ОБ при выполнении команд расширенной арифметики.

Регистр *RSH* используется в качестве аккумулятора, доступного по чтению на обе шины чтения *X* и *У* и как расширитель сдвигателя АЛУ до тридцатидвухразрядного. Как сдвиговый регистр, *RSH* используется в командах расширенной арифметики.

Важную роль в работе процессора играет регистр состояния *PSW*. Информация, содержащаяся в *PSW*, влияет на режимы выполнения команд и прерываний, определяет режим работы диспетчера памяти.

Регистр данных *RD* является буферным регистром при выдаче данных с шины результата *Z* в магистраль данных, в цикле записи по адресу и при приеме данных на шину *X* и *У*, в цикле чтения по адресу.

АЛУ выполняет все арифметические и логические операции между операндами, кроме того, здесь вырабатываются некоторые признаки, необходимые для формирования состояний. Результат операции поступает на сдвигатель /СДВ/.

ПЛМ управления АЛУ вырабатывает последовательность сигналов управления АЛУ в соответствии с обобщенной микрокомандой, поступающей из ПЛМ ОСН. Все микрокоманды, кроме трех из шестнадцати возможных, выполняются за I такт ОБ. Три микрооперации /параметрического сдвига, умножения и деления/ - многотактные.

ЛИН. №: 0394 / Подп. и дата: 10.07.87 г. /
 ЛИН. №: 0394 / Подп. и дата: 10.07.87 г. /
 ЛИН. №: 0394 / Подп. и дата: 10.07.87 г. /

ИИЗ.480.167 ТО

Лист
29

Блок констант (БК) представляет собой специально организованный блок для формирования констант, используемых при микропрограммировании.

Блок формирования адреса вектора прерываний (АВП) служит для формирования адреса вектора прерываний. Схема формирования АВП представляет собой программируемую логическую матрицу ПЛМ АВП. Значение адреса вектора прерываний определяет адреса как нового значения регистра слова состояния процессора (PSW), так и значения счетчика команд, что предопределяет начальное состояние процессора при вхождении в программу обработки данного прерывания, так и собственно программу. Количество входов - 12, выходов - 4.

Блок синхронизации ОБ (БС ОБ) представляет собой блок, который управляет работой всего ОБ. Здесь вырабатывается последовательность синхроимпульсов, обеспечивающая выполнение микрокоманды.

Блок формирования состояния ПРЦ (ФС) выполнен в виде ПЛМ. Анализируя микрокоманду и выдвигающийся из сдвигателя разряд, ПЛМ ФС формирует новые значения четырех разрядов регистра слова состояния ПРЦ N, Z, V, C , т.е. признаков, по которым можно определить знак операнда, равенство его нулю, переполнение и перенос.

Все блоки ОБ объединены внутренними шинами "X" и "Y", чтение на которые адресуется соответствующими полями микрокоманды. Выходом АЛУ является шина результата "Z", с которой производится запись. Все регистры, доступные микропрограммно, могут читаться по двум направлениям, на шину "X" и "Y"; константы читаются только на шину "Y".

5.3. Описание структурной схемы блока микропрограммного управления (БМУ).

Изм. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата
90394	10.07.87	87064		
				Лист
				30

ШИЗ.480.167 Т0

БМУ (рис.10) предназначен для преобразования команды в последовательность микрокоманд. В БМУ входят:

программируемая логическая матрица предварительного разбора (ПЛМ ПРД);

программируемая логическая матрица - основная (ПЛМ ОСН);

программируемая логическая матрица анализа состояния (ПЛМ АС);

блок синхронизации БМУ (БС БМУ).

ПЛМ предварительного разбора команды (ПЛМ ПРД) многофункциональна. Это следующие функции:

- 1) уменьшение числа произведений основной ПЛМ;
- 2) устранение так называемой "прокрутки" (холостых микрокоманд) для выявления необходимости обращения к блоку адреса - ции, для выборки и записи операндов с формированием признаков, детализирующих это обращение (необходимость записи, выборки байта, слова, двойного слова или четырех слов), а также установка начального адреса микропрограммы обработки команд;

3) выдача кодов команд с плавающей запятой с формированием признаков по биту пункта 2, в случае наличия процессора арифметики с плавающей запятой и отнесение их к резервным кодам в противном случае;

4) устранение ненужных пересылок операндов из регистра в регистр в некоторых формах адресации;

5) упрощение анализа деталей адресации, использующей счетчик команд;

6) определение необходимости отказа от очереди команд при опережающем просмотре программы, вследствие изменения значения счетчика команд в результате выполнения операции;

7) выработка четырехразрядного кода, определяющего класс команды по типу формирования признаков операции (для ПЛМ АС);

Инв. № посл.	Подп. и дата	Взам. инв. №	Инв. № докум.	Подп. и дата
90394	12.07.87			

ИИЗ.480.167 ТО

Лист
31

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № д/зл.	Подп. и дата
90394	10.07.87 Вил -			

Структурная схема БС БМУ

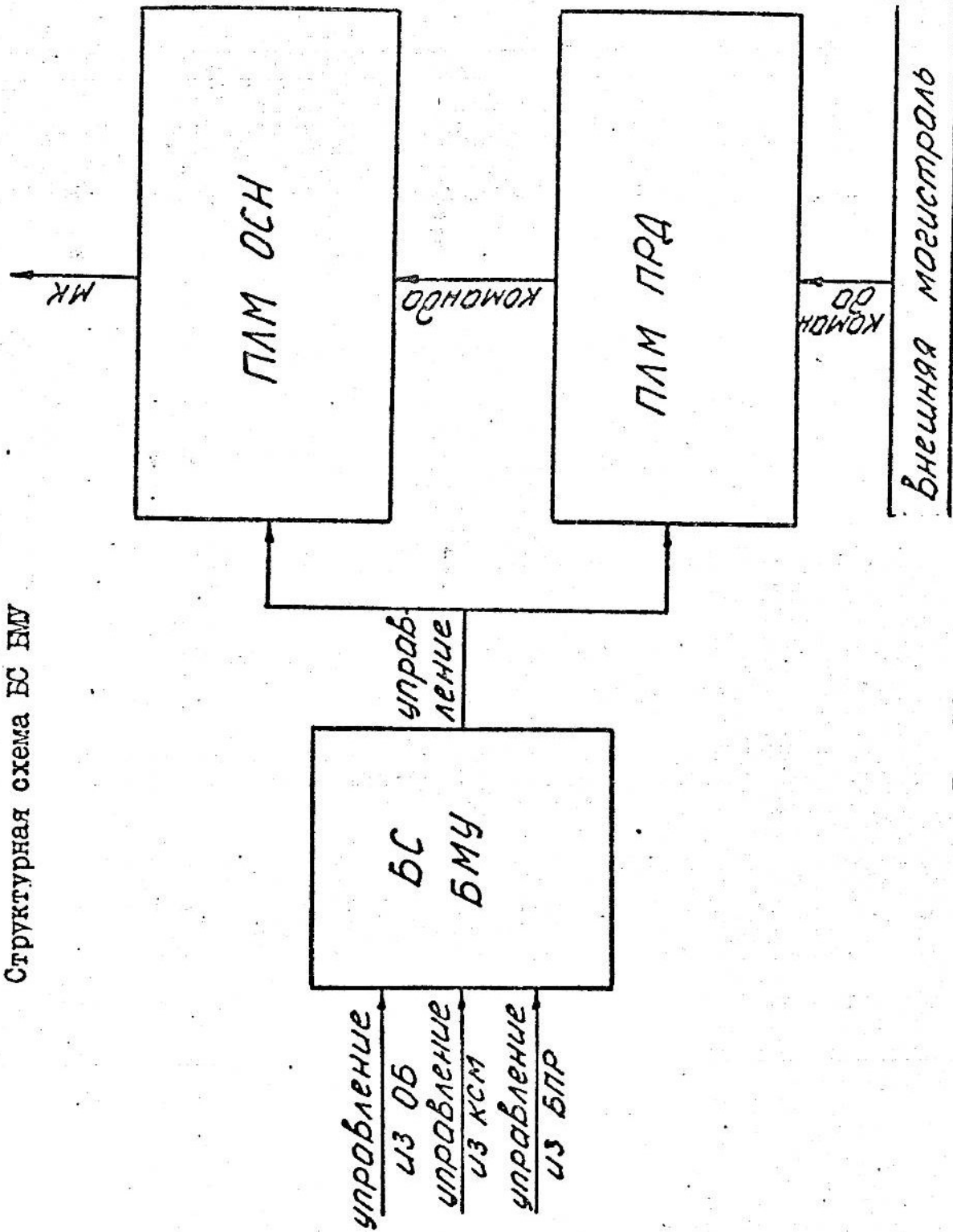


Рис. 10

ШИЗ.480.167 Т0

Лист

32

Инв. № подл.	Подп. и дата	Узач. инв. №	Инв. № д/бл.	Подп. и дата
90394	10.07.87 В.И.			

Структурная схема блока прерываний

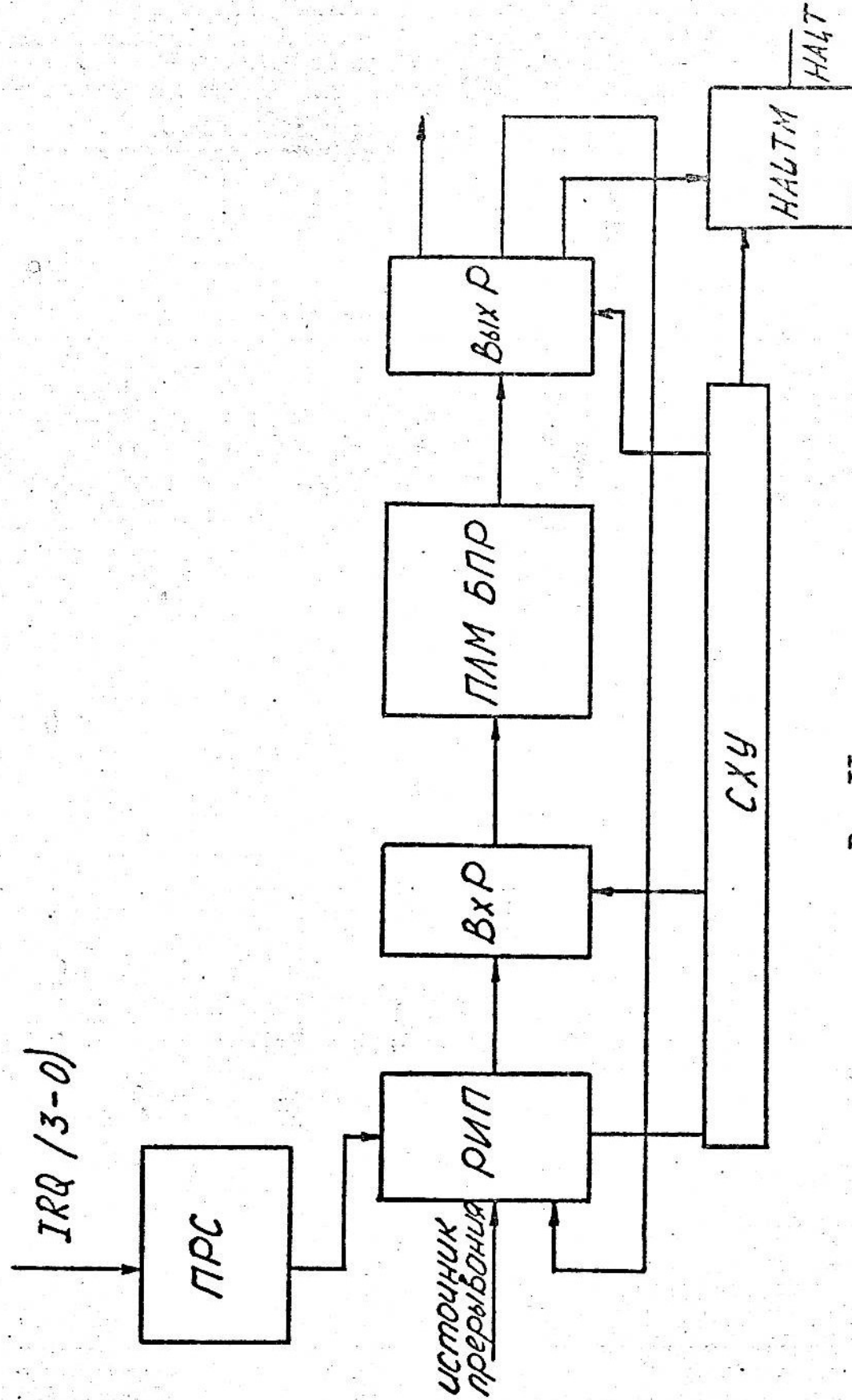


Рис. II

ШИЗ.480.167 ТО.

схему формирования признака пультового режима (HALTM);

БПР является приоритетной схемой для арбитража внутренних и внешних прерываний (рис. II). Он формирует признак наличия прерывания, указывает адрес вектора прерывания, вырабатывает сигнал HALT - сигнал перехода к пультовому режиму. ПРС и ПЛМ БПР непосредственно арбитражируют все прерывания; РИП служит для запоминания источника прерывания и его сброса после обработки. Регистры ВхР и ВыхР служат для временной синхронизации работы самого БПР и его работы совместно с другими блоками.

5.5. Описание структурной схемы диспетчера памяти (ДП)

ДП обеспечивает расширение емкости адресуемой памяти с 64К до 4М байт; перераспределение виртуального адреса в физический и защиту памяти в системах с разделением времени; использование различных областей адресов для режима работы ПРЦ. В ДП входят следующие функциональные блоки (рис. I2):

схема синхронизации ДП;

сумматор и схема сравнения;

регистр адреса страницы/регистр признака страницы (PAR/PDR);

регистры состояния ДП (SR0, SR1, SR2, SR3).

Сумматор производит сложение виртуального адреса с соответствующим регистром адреса страницы, а схема сравнения определяет доступность адреса в соответствии с регистром - описанием страницы. Блок регистров PAR/PDR содержит шестнадцать двадцатисемиразрядных регистров, доступных по адресу для чтения и записи. Регистр SR0 хранит признаки работы с текущей страницей, регистр SR1 не используется, SR2 - хранит виртуальный адрес текущей страницы, SR3 - режим работы ДП.

5.6. Описание структурной схемы контроллера системной магистрали (КСМ)

Уч. № подл.	Подп. и дата	Взам. Учв. №	Учв. № дубл.	Подп. и дата
50394	19.07.87			

Изм.	Лист	№ докум.	Подп.	Дата	ИЗМ. 480.167 ТО	Лист
						35

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № докл.	Подп. и дата
90394	10.07.87.03ч			
Изм.	Лист	№ докум.	Подп.	Дата
0.5d				

Структурная схема диспетчера памяти

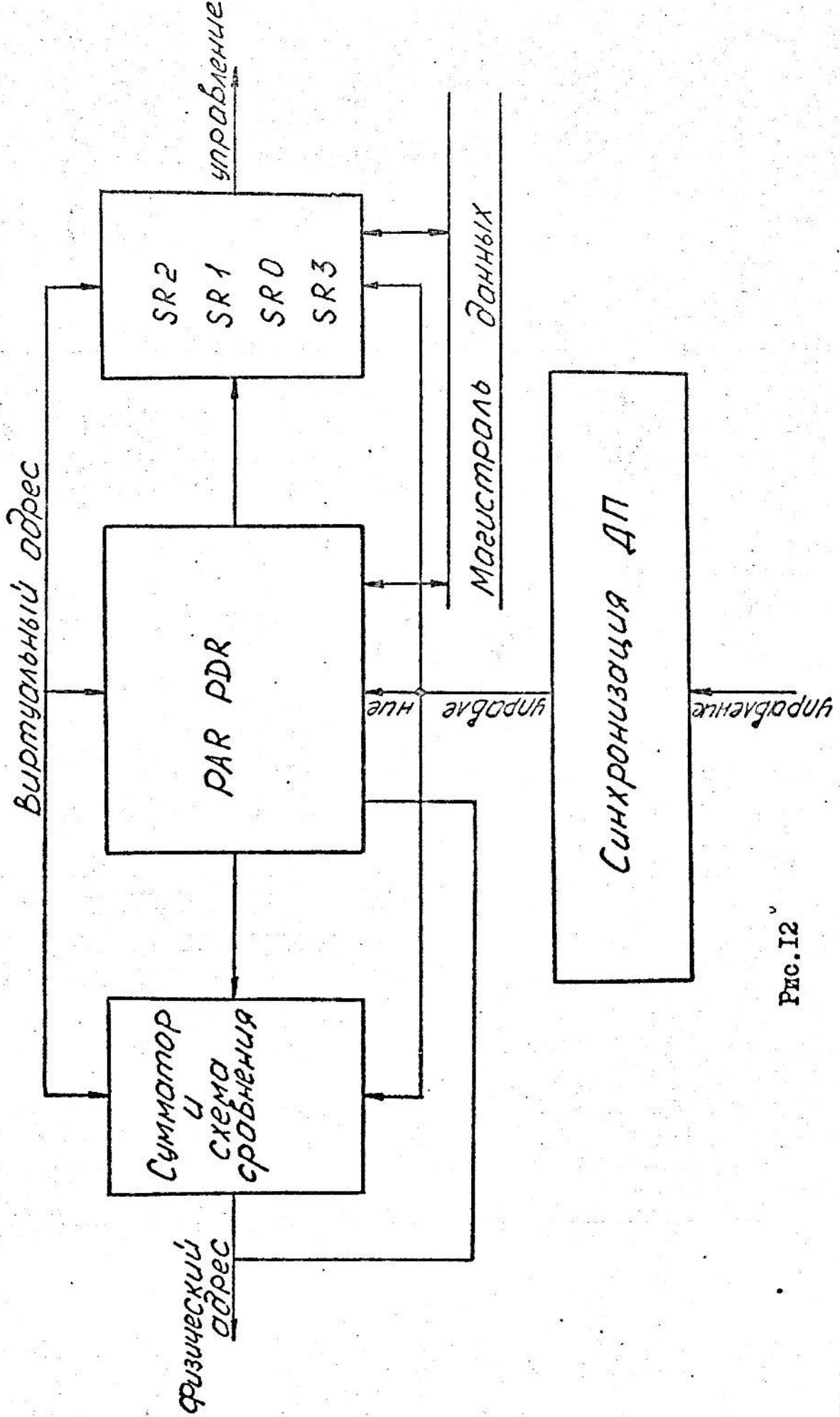


Рис. 12

КСМ предназначен для управления выборкой последовательности команд из памяти; для управления совмещения операций и согласования работы блоков синхронизации ОБ, БМУ, ДП; для организации циклов обмена данными с внешней памятью и внутренними регистрами.

КСМ выполняет роль арбитра для устройств первого доступа системной магистрали. КСМ включает в себя следующие блоки (рис. 13):

- выходной элемент;
- входной элемент;
- блок управления внутренними регистрами (БУРЕГ);
- устройство управления потоком команд (УУПК);
- устройство управления системной магистралью (УУСМ);
- счетчик зависания (СЧЗВС);
- дешифратор обмена (ДС ОБМ);
- контроллер прямого доступа в память (КПДП).

Работа процессора распараллелена: одновременно могут осуществляться выполнение i -й команды, предварительный разбор и преобразование в последовательность микрокоманд ($i+1$)-й команды, выборка ($i+2$)-й команды, подготовка (инкремент счетчика команд РС и релокация РС) к выборке ($i+3$)-й команды. УУПК синхронизирует эти четыре, идущих параллельно процесса, вырабатывает признаки торможения блоков синхронизации ДП, ОБ, БМУ. ДС ОБМ дешифрирует код обмена (четырёхразрядное поле микрокоманды) и управляет работой УУСМ при циклах обмена данными. В выходном элементе находится компаратор адреса, вырабатывающий признаки обращения к регистрам диспетчера памяти, к регистру *PSW*. В случае появления этих признаков УУСМ не вырабатывает сигналов обмена на системной магистрали, а начинает работать БУРЕГ, вырабатывающий сигналы управления для обмена по

Исх. № подл.	Подп. и дата
90394	10.07.87
Взам. исх. №	Подп. и дата
Исх. № дубл.	Подп. и дата

ИИЗ.480.167 ТО

Лист
37

внутренней магистрали. Контроллер ЦДП предоставляет разрешение на захват магистрали устройствам системной магистрали в промежутки времени, свободные от обменов процессора.

Инв. № подл.	Подп. и дата	Узлом. инв. №	Инв. № докл.	Подп. и дата
90394	10.07.87 03ч.			

ЦЗМ	Лист	№ докум.	Подп.	Дата

ШИЗ.480.167 ТО

Лист
38

УИВ. № подл.	Подп. и дата	Взам. инв. №	ИНВ. № докл.	Подп. и дата
90394	10.07.87	Взам.		

Структурная схема контроллера системной магистрали

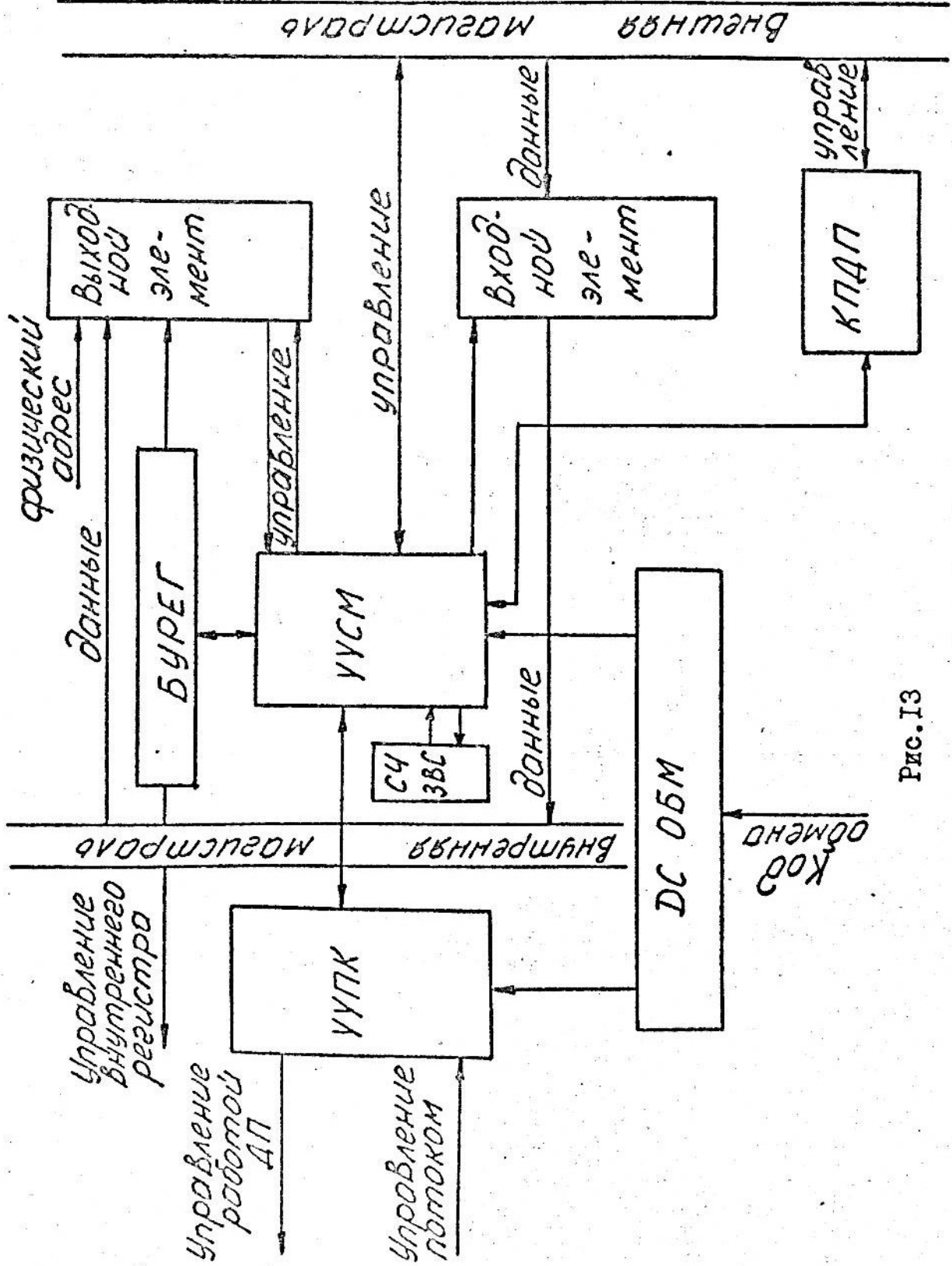


Рис. 13

ИИ3.480.167 ТО

6. ФУНКЦИОНАЛЬНОЕ ОПИСАНИЕ ИНТЕРФЕЙСА ПРЦ

6.1. Работа ПРЦ при включении питания

Временная диаграмма сигналов при включении питания показана на рис. 14

При сигналах \overline{ACLO} и \overline{DCLO} равных "0" происходит обнуление ПРЦ и установка начального адреса микропрограммы начального пуска. При включении питания поднимается в "1" сигнал \overline{DCLO} и процессор переходит в режим ожидания перехода в "1" сигнала \overline{ACLO} или перехода в "0" сигнала \overline{HALT} . Как только сигнал \overline{ACLO} становится равным "1", ПРЦ переходит к выполнению одной из двух микропрограмм начального пуска в зависимости от состояния на входе $W0$:

1) вход $W0 = 1$. Процессор загружает в регистр слово состояния PSW содержимое ячейки памяти с адресом 26_8 , счетчик команд PC - содержимым 24_8 ячейки памяти и начинает выполнение программы с адреса, находящегося в PC ;

2) вход $W0 = 0$. ПРЦ загружает в регистр слово состояния ПРЦ константу - 000340 , в регистр счетчика команд PC - 173000_8 и начинает выполнение программы с этого адреса.

Если после поднятия в "1" сигнала \overline{DCLO} входной сигнал \overline{HALT} равен "0", то процессор переходит в режим \overline{HALT} -моды.

6.2. Прерывание процессора

Прерывания процессора подразделяются на аппаратные и командные. Аппаратные прерывания могут вызвать немедленное прекращение всех процессов и переход к микропрограмме обработки прерывания. Это так называемые фатальные ошибки. Они возникают при ошибках системной магистрали, диспетчера памяти, процессора

ИЗМ. Лист УДОБ. Подп. Дата
90394 10.07.87

ШИЗ.480.167 ТО

Лист
40

с плавающей запятой. Анализ других аппаратных прерываний, таких как авария источника питания, внешний сигнал \overline{HALT} , прерывания IRQ (0-3), переполнение стека в моде операционной системы, производится после завершения выполнения очередной команды в соответствии с приоритетом:

Таблица 4

Приоритет	Действие
<p style="text-align: center;">Наивысший</p> <p style="text-align: center;">↑</p> <p style="text-align: center;">Наинизший</p> <p style="writing-mode: vertical-rl; transform: rotate(180deg);">Возрастание приоритета</p>	<p>HALT - команда</p> <p>Ошибка нечетного адреса</p> <p>Ошибка ДП</p> <p>Зависание системной магистрали</p> <p>Ошибка ШЗ</p> <p>Команды прерываний</p> <p>T-бит</p> <p>Переполнение стека</p> <p>Авария источника питания</p> <p>Сигнал HALT</p> <p>Запрос таймера $EVNT$</p> <p>IRQ 3</p> <p>IRQ 2</p> <p>IRQ 1</p> <p>IRQ 0</p> <p>Выполнение очередной команды</p>

Инв. № подл.	90394	Побл. и дата	10.07.87
Взам. инв. №		Инв. № докл.	
Заст. инв. №		Побл. и дата	

Изд.	Лист	№ докум.	Побл.	Дата

ШЗ.480.167 TO

Лист

41

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата
90394	19.07.87	Взм.		

Временная диаграмма сигналов ПРЦ при включении и выключении питания

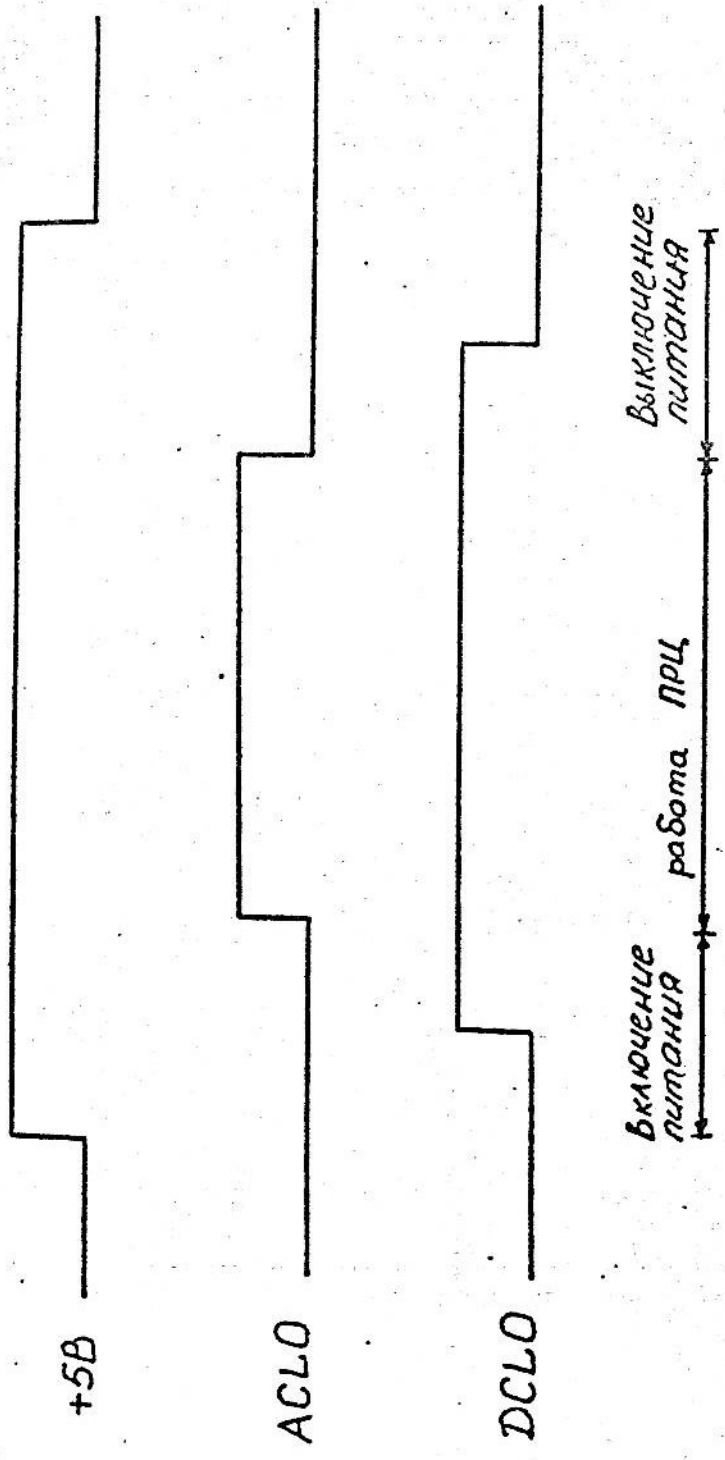


рис. 14

К командам прерывания относятся команды EMT, TRAP, BPT, IOT, а также, если учитывать алгоритм выполнения и резервные коды. Вектора прерываний приведены в табл. 5

Таблица 5

АВП	Вектор ₈	Комментарии
ЗДИО		
0	100	Прерывание от таймера
I	4	Фатальная ошибка, переполнение стека
2	250	Ошибка ДП
3	244	Ошибка ППЗ
4	24	Включение питания
5	10	Неход, запрещенная инструкция
6	-	Не используется
7	24	Выключение питания
10	30	EMT
11	14	BPT, прерывание по T-биту
12	34	TRAP
13	20	IOT
14	-	Не используется
15	I73000	Адрес начального пуска при включении питания по $WO=0$.
16	I00000	Значение указателя стека при входе в HALT-моду
17	RI(0-4)	Вектор безадресного прерывания IRQ (0-3)

При помощи вектора прерывания загружаются новые значения регистров R3W и PC, причем загрузка ведется из области

ЧИЗ. № ПОЗ. ПОЗ. И ОСТА. ВЗАМ. ИНЫ. И ОУОЛ. ПОДП. И ДАТА
 90394 40.07.87 В.м.

ЧИЗ.480.167 ТО

Лист

43

ИЗ-Лист 100301. ПОДП. ДАТА

памяти операционной системы.

Особый случай представляет собой возникновение фатальной ошибки (кроме ошибки ДП) в ходе выполнения микропрограммы обработки прерывания, так называемая, "двойная ошибка". В этом случае процессор переходит в режим \overline{HALT} -моды. Для упрощения аппаратуры начальный адрес микропрограммы обработки ситуации двойной ошибки совпадает с адресом начального пуска.

6.3. Режим \overline{HALT} -моды

В режим \overline{HALT} -моды процессор выходит в следующих случаях:
в результате выполнения команды \overline{HALT} в моде операционной системы;

в результате двойной ошибки;

при возникновении внешнего сигнала $\overline{HALT} = 0$ и отсутствии в этот момент других более приоритетных прерываний.

При входе в \overline{HALT} -моду процессор загружает в регистр RIO блока PОН константу 100000_8 , используя его в качестве указателя стека, загружает в стек регистры PSW и PC ; заносит в регистр PSW значение 340_8 в регистр PC - "0"; включает ДП на преобразование виртуальных адресов в 22-разрядные физические и переходит к выполнению программы, расположенной по виртуальному адресу "0".

В реальном физическом адресном пространстве виртуальные адреса, находящиеся в регистрах PC и RIO располагаются в области свободной от других программ. В ДП с целью реализации режима \overline{HALT} -моды введены четыре дополнительных регистра адреса страницы PARN:

Инв. № подл.	Подл. и дата	Экз. инв. №	Инв. № докум.	Полн. и дата	
90394	10.07.870Вм -				
Изм.	Лист	№ докум.	Подл.	Дата	
ШИЗ.480 I67 TO					Лист
					44

Таблица 6

Номер регистра RARN	VA13	VA14	Доступность	Значение	Назначение
0	0	0	Программно недоступен	0	Для формирования начального адреса программы НАЛТ-моды (ПЗУ)
1	0	1	Программно недоступен	0	Для формирования начального адреса ОЗУ для программы НАЛТ-моды
2	1	0	По чтению и записи по адресу $I725I2_8$	-	Регистр адреса страницы для адресации в режиме НАЛТ моды по всей области памяти
3	1	1	Программно недоступен	$I77600_8$	Для формирования начального адреса страницы старшего банка

Адресация к этим регистрам производится при помощи двух старших разрядов виртуального адреса. Для программной реализации режима НАЛТ-моды отводится адресная область памяти объемом 8К. При входе в НАЛТ-моду в регистр R10, служащий в этом случае указателем стека, заносится значение $I00000_8$, т.е. запись в стек регистров PSW и PC производится по виртуальным адресам 77776 и 77774 соответственно. Поэтому для организации режима НАЛТ-моды необходимо предусмотреть область ОЗУ в физической области памяти, соответствующей этим виртуальным адресам с ис-

ЧИЗ. № подл. 90394
 Дата. 10.07.87
 Подп. и дата. Взам. инв. № инв. Подп. и дата.

ЧИЗ. № подл.	90394	Дата.	10.07.87	Подп.		Дата.	
ЧИЗ.480.167 ТО							Лист 45

пользованием регистра $PARH1$.

Регистр $PARH2$ позволяет осуществить обращение ко всем 4 М-байтам памяти. Это производится путем записи в этот регистр соответствующего значения и затем обращения через него к нужной ячейке памяти. Адрес регистра $PARH2 - 172512$.

В режиме H/LT -моды существуют некоторые отличия при выполнении команд и возникновении прерываний. В этом режиме нельзя использовать команды прерываний EMT , $TRAP$, IOT , BPT , команду $WAIT$, $HALT$, команды с ПЗ, команды расширенной арифметики MUL , DIV , ASH , $ASHC$, команды $MFPD(I)$, $MTRPD(I)$. Эти ограничения обусловлены организацией регистра указателя стека в $R10$, а также тем, что выход из режима H/LT -моды осуществляется по командам RTI и RTT . При этом команды RTI и RTT выполняются без учета значения T -бита.

В режиме H/LT -моды блокируются прерывания от ШП, путем загрузки в PSW значения 340, замаскированы прерывания $IRQ (0-3)$, невозможно прерывание от ШПЗ. В случае возникновения ошибки системной магистрали не происходит обычной процедуры обработки прерывания, а в регистр PC загружается виртуальное значение "4" и процессор переходит к выполнению программы по этому адресу.

6.4. Циклы обращения к памяти или устройствам

Для выполнения любой команды ПРЦ требует выполнить хотя бы одну операцию обращения к каналу. Для некоторых команд требуется выполнить несколько операций. Первой такой операцией для всех команд является ввод данных из ячейки памяти, адрес которой определяется счетчиком команд (PC). Все операции обращения к каналу для ввода и вывода данных называются циклами обращения

Имя: _____
№: _____
Подп. и дата: _____
Взам. инж. инв. и дата: _____
Подп. и дата: _____
14.02.87 г. вв

ШИЗ.480.167 ТО

Лист

45

Имя: _____
Подп. _____
Дата: _____

Формат 5а по ГОСТ 2.105-68

Формат 44

к каналу. Если для выполнения команды не требуется обращаться за операндами к памяти или внешним устройствам, дополнительных циклов канала не требуется. Однако, если выполняется команда с обращением к памяти или устройствам, то в этом случае могут выполняться любые из следующих циклов: ЧТЕНИЕ, ЗАПИСЬ, ЧТЕНИЕ-МОДИФИКАЦИЯ-ЗАПИСЬ.

Формат адреса всегда двадцатидвухразрядный. Данные могут быть как шестнадцатиразрядными словами, так и восьмиразрядными байтами. Перед каждым циклом канала ПРЦ осуществляет процедуру захвата канала (CM). Требование прерываний может быть удовлетворено ПРЦ только в промежутках между выполнениями команд.

6.4.1. Цикл "Чтение"

При выполнении цикла "Чтение" данные передаются от пассивного устройства к активному. Временная диаграмма сигналов, которые выдает ПРЦ, показана на рис.15. Порядок выполнения операций следующий:

на линии $AD0-15$, $A16-21$ выставляется адрес
через время $T/2$ выставляется сигнал \overline{SYNC} , по которому пассивное устройство запоминает адрес и выставляет сигнал \overline{SSYNC} ;

через время T после сигнала \overline{SSYNC} адрес снимается с выводов $AD0-15$. Через $1,5T$ после \overline{SSYNC} выставляется сигнал \overline{DIN} . ПРЦ переключает элементы входа (выхода) на выходах $AD0-15$ на прием и ожидает поступление сигнала $\overline{RP4Y}$ от пассивного устройства. Если в течение 128 тактов частоты синхронизации сигнал $\overline{RP4Y}$ не появляется, ПРЦ переходит к обслуживанию внутреннего прерывания по ошибке обращения к каналу с адресом вектора 4;

после приема сигнала $\overline{RP4Y}$ ПРЦ принимает данные и снимает

Инв. № подл.	Подп. и дата	Инв. № докл.	Подп. и дата
90394	10.07.87		

Инв. № подл.	Подп. и дата	Инв. № докл.	Подп. и дата

ИМЗ.480.167 ТО

Лист

47

мает сигналы \overline{DIN} и \overline{SYNC} ;

по снятии сигнала \overline{DIN} пассивное устройство снимает сигнал \overline{RPLY} ;

если магистраль не захватывается другим устройством, ПРЦ сразу после снятия сигнала \overline{SYNC} может начать новый цикл обращения к каналу. Во время цикла ЧТЕНИЕ сигнал \overline{WTBT} не вырабатывается.

6.4.2. Цикл ЗАПИСЬ

При выполнении цикла ЗАПИСЬ данные передаются от ПРЦ к пассивному устройству, например, происходит запись данных в память. Временная диаграмма сигналов, которые выдает ПРЦ, показана на рис.16.

Порядок выполнения операций следующий:

на выходы $ADD-15$, $A16-21$ ПРЦ выставляет адрес;

через $T/2$ выдается сигнал \overline{SYNC} . Пассивное устройство запоминает и дешифрирует адрес и выдает сигнал \overline{SSYNC} ;

через время T после приема сигнала \overline{SSYNC} адрес снимается. Во время выдачи адреса выдается сигнал \overline{WTBT} , который снимается одновременно со снятием адреса. В этой фазе сигнал сообщает, что происходит цикл ЗАПИСЬ, а не ЧТЕНИЕ;

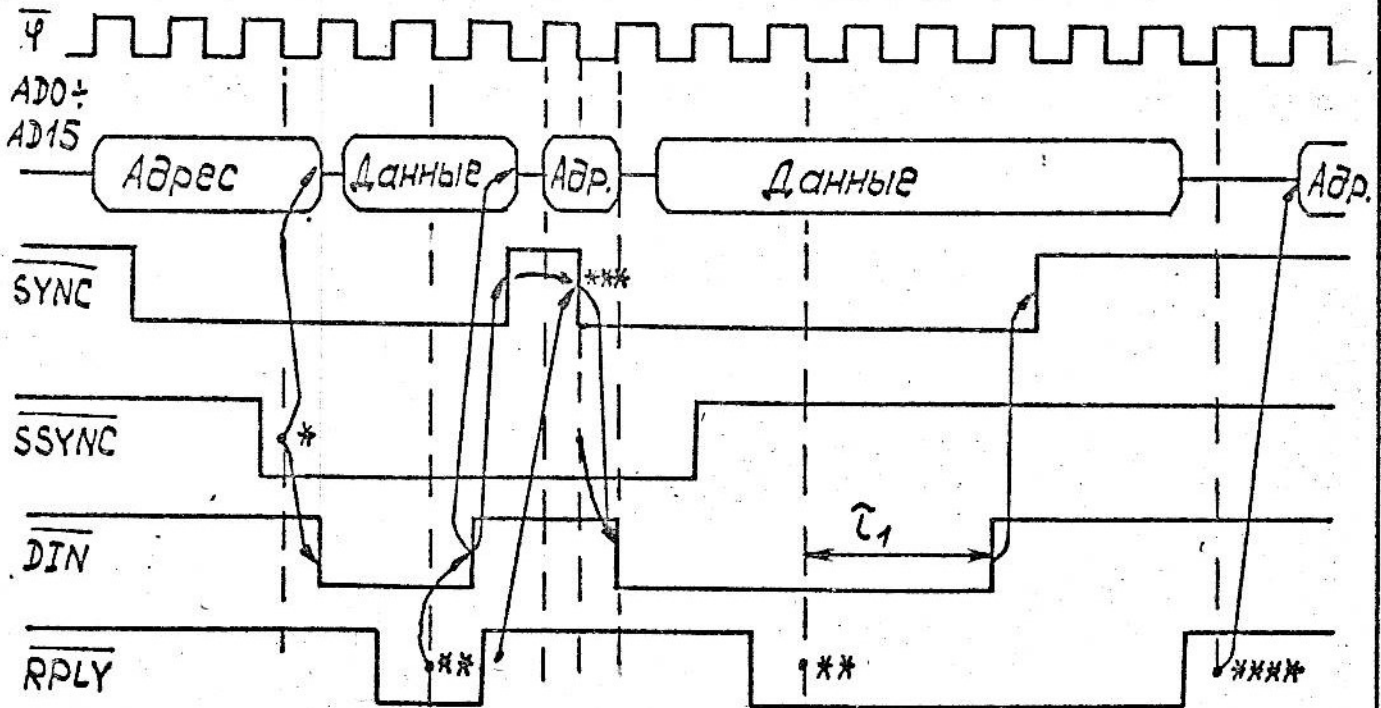
после снятия адреса ПРЦ выставляет на выходы $ADD-15$ данные и через время $T/2$ после выдачи данных сигнал \overline{DOUT} . Если происходит запись байта, то одновременно с началом выдачи данных выдается сигнал \overline{WTBT} , который в этой фазе сообщает, что происходит выдача байта, а не слова. При выдаче байта второй байт на остальных 8-ми выводах AD выдается логическим нулем (активный высокий уровень). Если после 128 тактов частоты синхронизации пассивное устройство не выдает сигнал \overline{RPLY} то как и в процедуре ЧТЕНИЯ, ПРЦ переходит к обработке внутреннего прерывания по ошибке обращения к каналу с адресом вектора 4;

Полн. и дата	
Инв. № б/н	
Взам. инв. №	
Полн. и дата	10.07.87
Инв. № подл.	46394

ШИЗ.480.167 Т0

Лист 43

Временная диаграмма чтения по адресу.



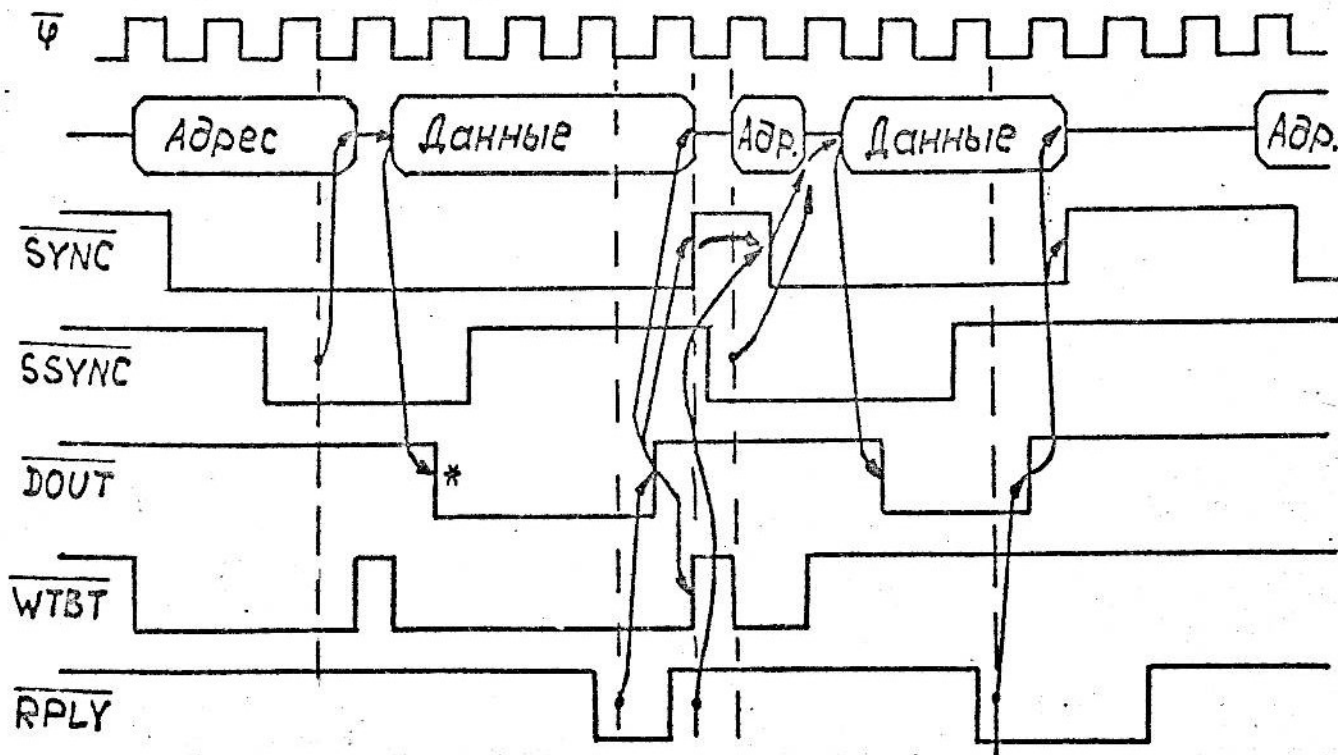
1. Через 0,5 такта после выполнения условия $\overline{SYNC} \& \overline{SSYNC} \& (\Sigma \Phi)$ сбрасывается адрес и выставляется сигнал DIN .
2. Данные с магистрали читаются через 0,5 такта после выполнения условия $\overline{DIN} \& \overline{RPLY} \& (\Sigma \Phi)$. После чтения данных снимается сигнал DIN , $\tau_{1max} = 20$ тактов.
3. Новый адресный обмен начинается, если отсутствует запрос прямого доступа, сигнал захвата магистрали $SACK$. Если в момент выдачи адреса сигнал $SSYNC$ уже присутствует, то DIN выставляется через 0,5 такта после $SYNC$.
4. Если после снятия $SYNC$, $RPLY$ еще не снят, то новый обмен начинается не раньше чем через 0,5 такта после $\overline{RPLY} \& (\Sigma \Phi)$.

Примечание: Новый обмен (выдача адреса) начинается через 0,5 такта после выполнения условия $\overline{RPLY} \& \overline{SACK} \& \overline{DMR} \& (\Sigma \Phi)$

Рис. 15

№ п. подл.	№ п. подл.	№ п. подл.	№ п. подл.	№ п. подл.
90394	10.07.87	07.87	07.87	07.87

Временная диаграмма записи по адресу.



* Выдача адреса квитируется сигналами SYNC и SSYNC аналогично процедуре чтения по адресу.

Данные выставляются через такт после условия $SYNC \& SSYNC \& (\Sigma \varphi)$, при условии готовности данных.

Рис. 16

Инв. № подл.	100394
Полн. и дата	10.07.87
Взам. инв. №	
Инв. № докл.	
Полн. и дата	

после приема сигнала \overline{RPLV} ПРЦ через один такт минимум снимает сигнал \overline{DOVT} и еще через $T/2$ данные с выводов $ADD-15$ сигнал \overline{SYNC} и сигнал \overline{WTBT} , если он был выставлен;

по снятии сигнала \overline{DOVT} пассивное устройство снимает сигнал \overline{RPLV} .

6.4.3. Цикл ЧТЕНИЕ-МОДИФИКАЦИЯ-ЗАПИСЬ

Временная диаграмма цикла ЧТЕНИЕ-МОДИФИКАЦИЯ-ЗАПИСЬ показана на рис.17. Адресная часть и фаза чтения аналогична циклу ЧТЕНИЕ. Однако после окончания чтения данных сигнал \overline{SYNC} не снимается, что позволяет записать данные во второй фазе цикла без повторения адресной части цикла. Фаза записи начинается только после того, как пассивное устройство сняло сигнал \overline{RPLV} в ответ на снятие сигнал \overline{DIN} . В случае записи байта во время записи вырабатывается сигнал \overline{WTBT} . Как в фазе чтения, так и в фазе записи в случае отсутствия сигнала \overline{RPLV} после выдачи сигнала \overline{DIN} или \overline{DOVT} в течение 128 тактов частоты синхронизации ПРЦ переходит к обработке внутреннего прерывания по ошибке обращения к каналу с адресом вектора "4".

6.4.4. Цикл ЗАПРОС - ПРЕДОСТАВЛЕНИЕ ПРЯМОГО ДОСТУПА К СМ

Временная диаграмма цикла ЗАПРОС - ПРЕДОСТАВЛЕНИЕ ПРЯМОГО ДОСТУПА К СМ представлена на рис.18.

В случае необходимости выполнить захват магистрали устройство системной магистрали выставляет сигнал на выводе \overline{DMR} . ПРЦ в ответ на этот сигнал выставляет сигнал разрешения захвата магистрали \overline{DMGO} , который принимается на вывод \overline{DMGI} устройства. Приняв сигнал разрешения, устройство блокирует дальнейшее прохождение этого сигнала по цепочке устройств и выставляет сигнал подтверждения захвата магистрали на выводе \overline{SACK} и начинает цикл канала, выставляя адрес на выводах $ADD-15$ и $A16-21$. Сигнал \overline{DMR} снимается после начала выдачи

Изм. №	Дата	Взам. инв. №	Инв. № дубл.	Подп. и дата
90394	10.07.87			

ИИЗ.480.167 Т0

Лист 51

адреса. Сигнал \overline{SACK} снимается одновременно со снятием сигнала \overline{SYNC} .

6.4.5. Прерывание программы по запросу на векторное прерывание

Прерывание программы - это временное прекращение выполнения текущей программы и переход к выполнению программы обслуживания устройства, вызвавшего прерывание. Каждое устройство, способное вызвать прерывание, должно иметь регистр состояния, содержащий разряд разрешения прерывания. Этот разряд должен быть программно установлен в "1", если устройству разрешается прерывание программы. Кроме того ПРЦ будет предоставлять прерывание и, следовательно, выполнять программу обслуживания устройства только в том случае, если в пятом-седьмом разрядах PSW будет снята маска прерывания. Каждое устройство, требующее прерывание, имеет программу обслуживания, вход в которую осуществляется автоматически с помощью вектора прерывания.

Временная диаграмма процедуры "ЧТЕНИЕ ВЕКТОРА ПРЕРЫВАНИЯ" показана на рис. 19.

Последовательность операций следующая:

устройство, которому необходимо обслуживание, выставляет сигнал запроса прерывания, который поступает на вывод $IRQ(3-0)$ ПРЦ;

если прерывание разрешается, то ПРЦ переходит в режим его обработки. Он помещает в стек содержимое счетчика команд (PC) и регистра состояния (PSW). Затем выставляет сигнал \overline{DIV} и через $T/2$ сигнал разрешения прерывания на выводе \overline{IAKO} ;

устройство принимает сигнал разрешения на свой вход \overline{IAKI} и запрещает распространение этого сигнала к другим устройствам. Затем оно помещает адрес вектора прерывания на шину $AD 0-7$,

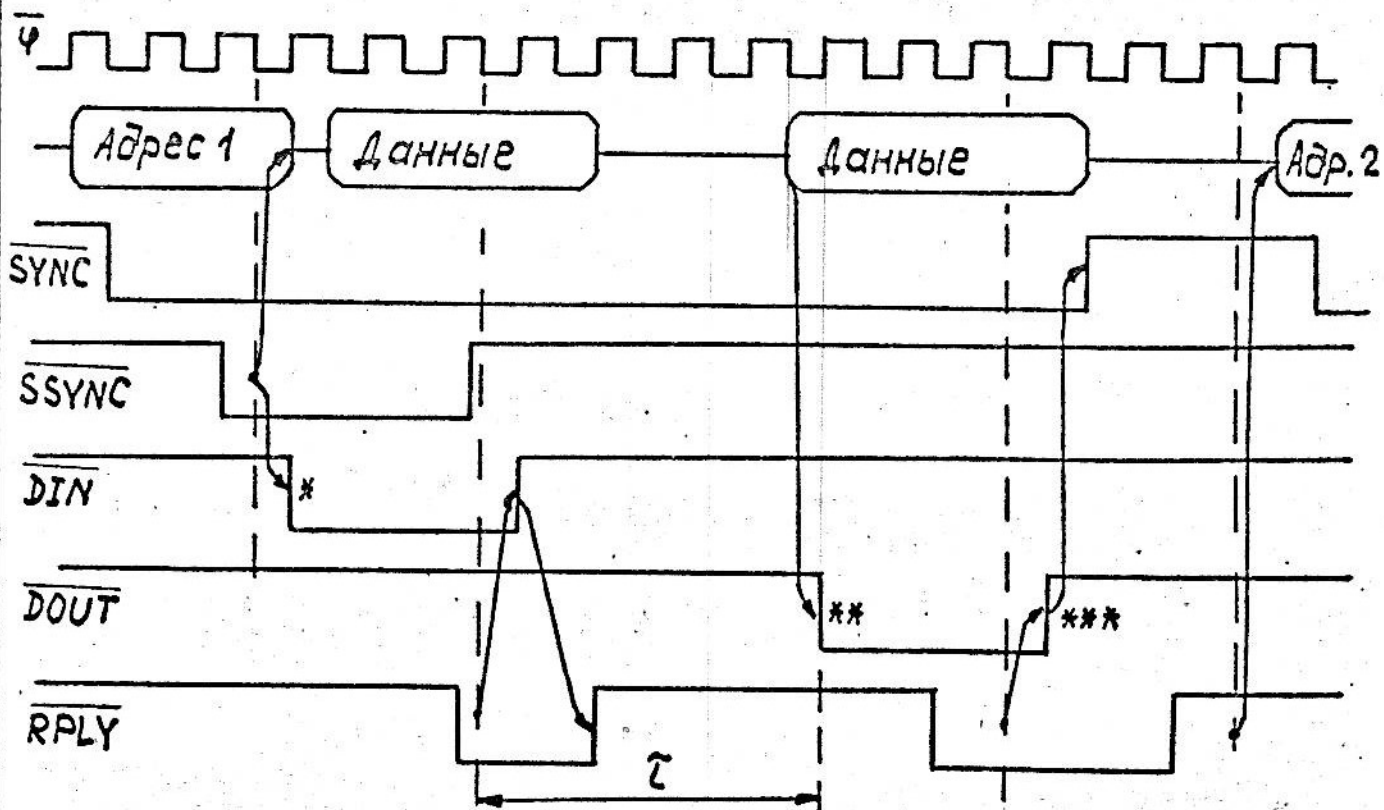
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № д/сл.	Подп. и дата
00394	10.07.87	87034		

ШИЗ.480.167 Т0

Лист

52

Временная диаграмма чтение-модификация-запись.



1. Временная диаграмма процесса считывания аналогична временной диаграмме чтения по адресу.
2. Минимальная задержка между $RPLY \& (\Phi)$ и выдачей сигнала $DOUT$ $\tau_{min} = 4,5$ тактов.
3. Окончание обмена аналогично окончанию записи по адресу.

Рис. 17

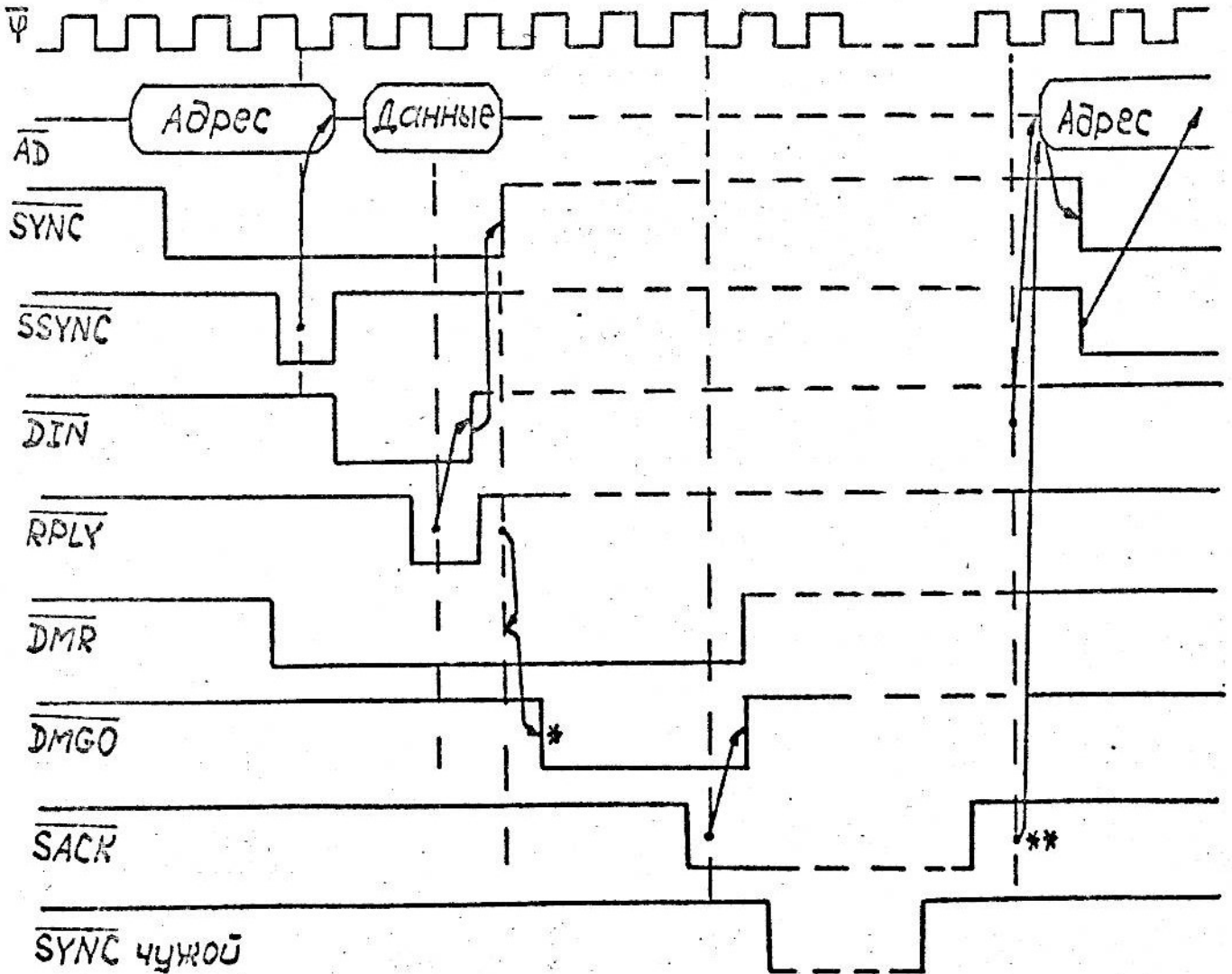
Инв. № позн.	Подп. и дата	Зам. инв. №	Инв. № дубл.	Подп. и дата
90394	10.07.87			

ИИЗ.480.167 ТО

Лист

53

Захват магистрали.



1. Формирование DMGO происходит через 0,5 такта после $\overline{DMR} \ \& \ \overline{RPLY} \ \& \ (\Sigma \Psi)$
2. Очередной обмен может произойти через 0,5 такта после $\overline{SACK} \ \& \ (\Sigma \Psi) \ \& \ \overline{RPLY} \ \& \ \overline{DMR}$

Рис. 18

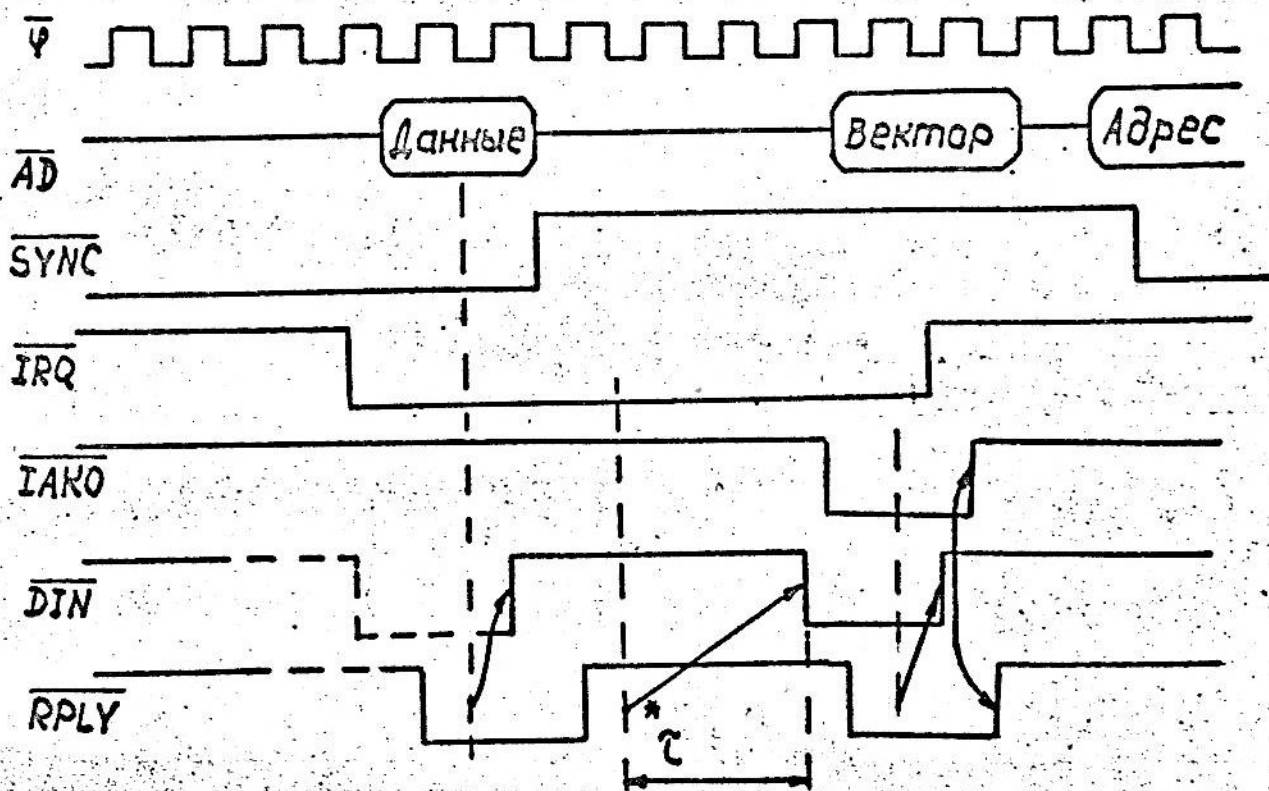
Инв. № подл.	Взам. инв. №	Инв. № дубл.	Подп. и дата
90394	10.07.87	Соб. 10.07.87	

ИИЗ.480.167 ТО

Лист

54

Чтение вектора прерывания.



1. Выдача \overline{DIN} может произойти через $\geq 2,5$ такта после освобождения магистрали $\overline{RPLY} \& (\Sigma \Psi)$, но не раньше 2 тактов после $\overline{SYNC} \& (\Sigma \Psi)$
2. Начало следующего обмена аналогично временной диаграмме чтения.

Рис. 19

Инв. № подл.	Подл. и дата
90394	12.07.87
Инв. № докл.	Подл. и дата
Взят. инв. №	

Инв. № подл.	Лист	№ докум.	Подп.	Дата

ИИЗ.480.167 Т0

Лист
55

вырабатывает сигнал $RPLY$ и снимает сигнал IRQ ;

ПРЦ принимает вектор прерывания (его адрес) и снимает сигнал DIN и через $T/2$ сигнал $IAKO$;

устройство завершает передачу вектора и снимает сигнал $RPLY$;

ПРЦ загружает новое содержимое PC и PSW из двух последовательных ячеек, первая из которых определяется адресом вектора прерывания, после чего переходит к выполнению программ обслуживания данного устройства.

По завершении программы обслуживания ПРЦ возобновляет выполнение прерванной программы с помощью команды возврата из прерывания (RTI), по которой из стека выбираются два слова и записываются в PC и PSW

Если имеется несколько устройств, способных запрашивать векторное прерывание, то они соединяются в цепочку по цепи разрешения прерывания.

6.5. Работа процессора с ШИЗ

6.5.1. Прием команды ШИЗ

ПРЦ выставляет обычную последовательность при обмене с памятью в цикле чтения. На выходе $A2I/NS$, во время выдачи сигнала DIN устанавливается низкий уровень, сообщающий ШИЗ, что происходит прием команды. ШИЗ принимает команду в свой регистр и затем, если команда из множества команд плавающей запятой, приступает к ее выполнению. Временная диаграмма цикла приема команды ШИЗ представлена на рис. 20.

6.5.2. Прием данных ШИЗ из памяти

При приеме данных ШИЗ из ОЗУ ПРЦ выставляет адрес, сигналы $SYNC$ и DIN в соответствии с временной диаграммой чтения, но сам ПРЦ данные не принимает. Данные, после прихода

ИНВ. № 90394
ИЗМ. № 1
ИЗМ. № 2
ИЗМ. № 3
ИЗМ. № 4
ИЗМ. № 5
ИЗМ. № 6
ИЗМ. № 7
ИЗМ. № 8
ИЗМ. № 9
ИЗМ. № 10
ИЗМ. № 11
ИЗМ. № 12
ИЗМ. № 13
ИЗМ. № 14
ИЗМ. № 15
ИЗМ. № 16
ИЗМ. № 17
ИЗМ. № 18
ИЗМ. № 19
ИЗМ. № 20
ИЗМ. № 21
ИЗМ. № 22
ИЗМ. № 23
ИЗМ. № 24
ИЗМ. № 25
ИЗМ. № 26
ИЗМ. № 27
ИЗМ. № 28
ИЗМ. № 29
ИЗМ. № 30
ИЗМ. № 31
ИЗМ. № 32
ИЗМ. № 33
ИЗМ. № 34
ИЗМ. № 35
ИЗМ. № 36
ИЗМ. № 37
ИЗМ. № 38
ИЗМ. № 39
ИЗМ. № 40
ИЗМ. № 41
ИЗМ. № 42
ИЗМ. № 43
ИЗМ. № 44
ИЗМ. № 45
ИЗМ. № 46
ИЗМ. № 47
ИЗМ. № 48
ИЗМ. № 49
ИЗМ. № 50

ШИЗ.480.167 ТО

Лист

56

ИЗМ. Лист. № об. кум. Подп. Дата

ФОРМА 50. ГОСТ 2.406-68

ФОРМА ЛТ А4

Инв. № подл.	Подп. и дата	Взят. инв. №	Инв. № докум.	Подп. и дата
90394	10.07.87	Вен.		

Прием команды ЛПЗ

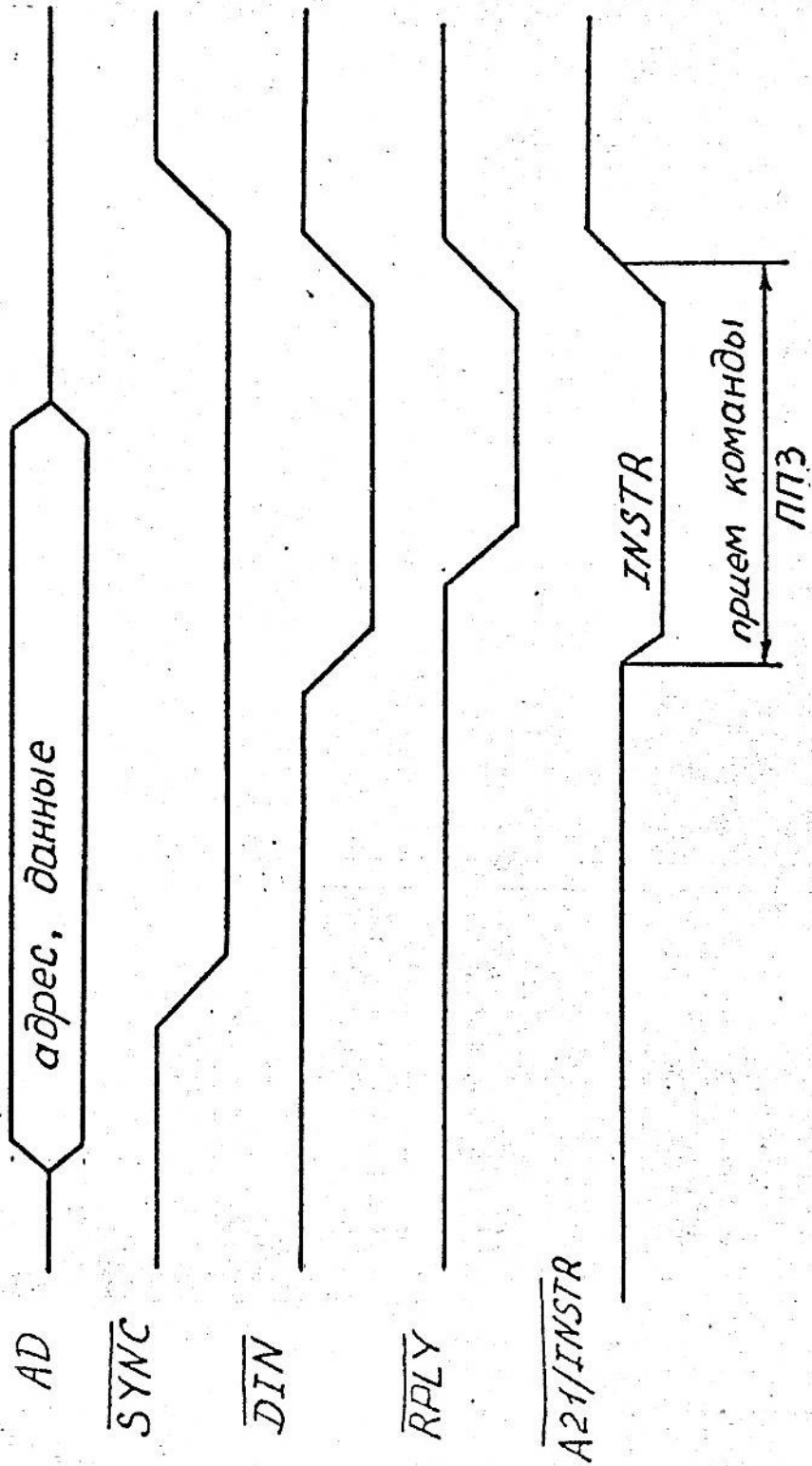


Рис. 20

ИИИ3.480.167 Т0

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата
90394	10.07.87 вв.			

Чтение данных в ППЗ из памяти.

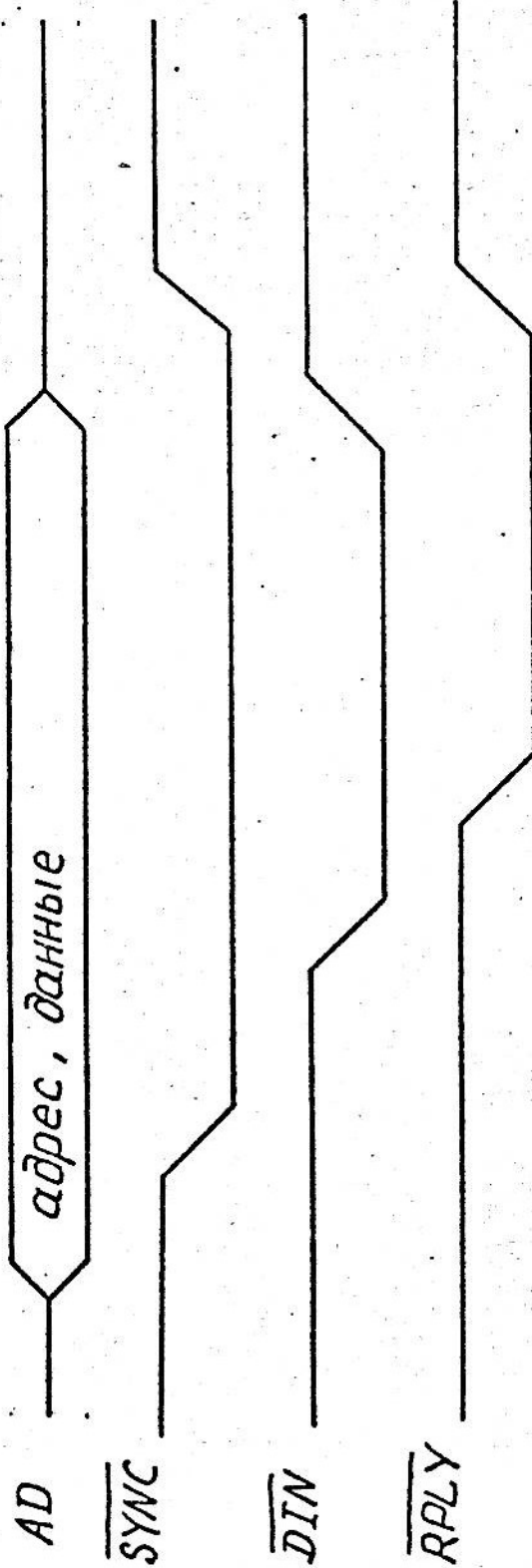


Рис. 21

Изм.	Лист	№ докум.	Подп.	Дата

ИИЗ.480.167 Т0

$RPLV$ принимает ШЗ, который должен успеть их принять синхронно до снятия сигнала \overline{DIN} процессором. Завершается обмен как обычный цикл чтения (рис.21).

6.5.3. Запись данных из ШЗ в ОЗУ

ПРЦ развертывает начало временной диаграммы цикла записи, но сигнал \overline{DOVT} не выставляет до тех пор, пока ШЗ не подготовит записываемые данные, о чем он сообщает низким уровнем сигнала \overline{DREADY} . После этого, цикл записи продолжается и завершается обычным образом. ШЗ снимает данные после снятия сигнала \overline{DOVT} (рис.22).

6.5.4. Чтение из ШЗ в ПРЦ

При выполнении команд над числами с плавающей запятой, ПРЦ выставляет сигнал \overline{DIN} низким уровнем, сохраняя сигнал \overline{SYNC} равным "1" и ожидая от ШЗ данных. Так как ПРЦ и ШЗ выполняют одну и ту же команду над числами с плавающей запятой, то обмены между ними заранее определены и ШЗ выставляет соответствующие данные на шине A/D , сопровождая их низким уровнем сигнала \overline{RPLV} . После этого через $3T/2$ максимум снимается сигнал \overline{DIN} (рис. 23).

6.5.5. Запись из ПРЦ в ШЗ

Как и при чтении, сигнал \overline{SYNC} находится в "1", ПРЦ выставляет данные на шине A/D , через время $T/2$ выставляет сигнал \overline{DOVT} низким уровнем и ожидает сигнала \overline{RPLV} от ШЗ. Приняв данные, ШЗ выставляет сигнал \overline{RPLV} , ПРЦ через время $3T/2$ максимум снимает сигнал \overline{DOVT} и ШЗ снимает сигнал \overline{RPLV} (рис. 24).

6.6. Значения времен задержек сигналов системной магистрали

Значения времен задержек сигналов на системной магистрали

Инв. № докум.	Подп. и дата
Взам. инв. №	
Подп. и дата	10.07.87
Инв. № докум.	90394

ШИЗ.480.167 ТО

Лист

59

Инд. № подл.	Подп. и дата	Взам. инв. №	Инд. № докум.	Подп. и дата
90394	10.07.87 <i>В.В.</i>			

Запись данных ППЗ в память

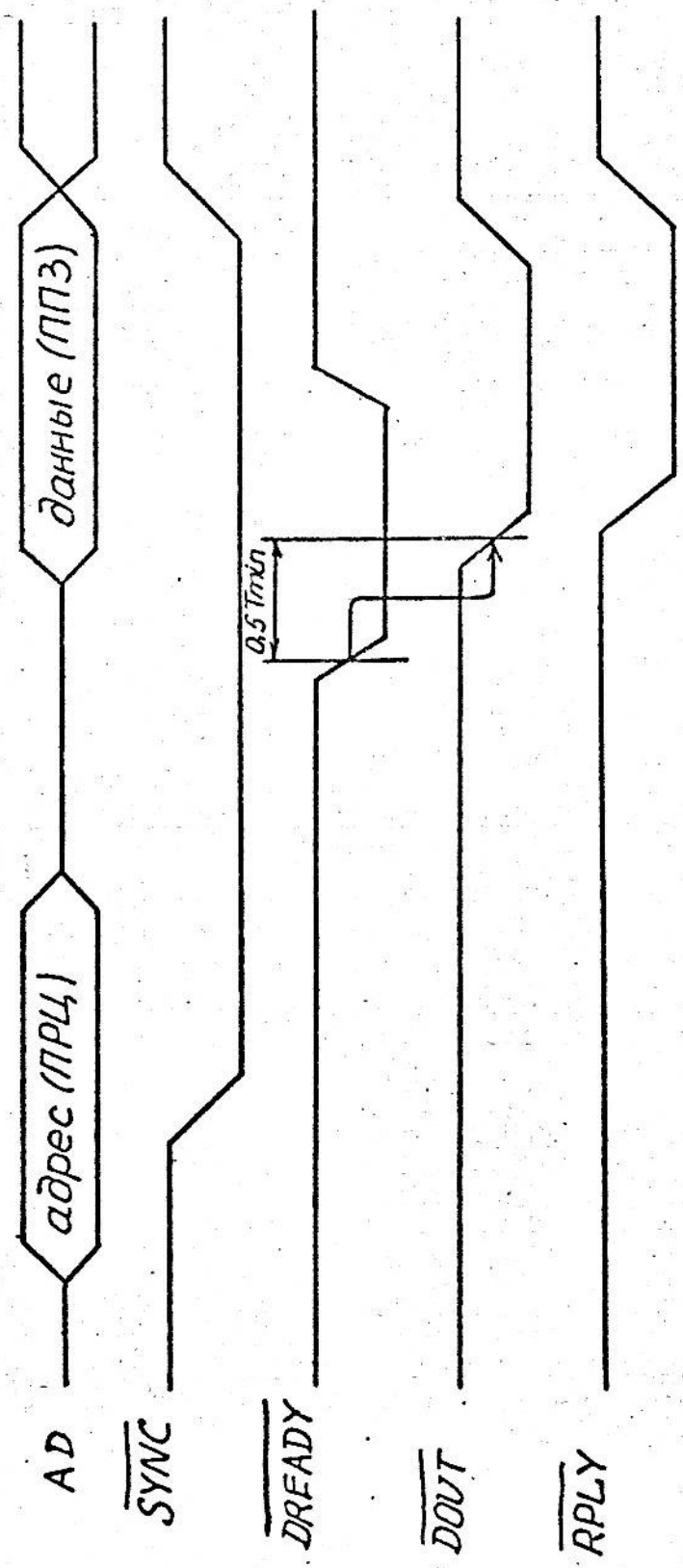


Рис. 22

ИИЗ.480.167 ТО

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № докл.	Подп. и дата
90394	10.07.87 <i>Вм.</i>			

Чтение из ППЗ.

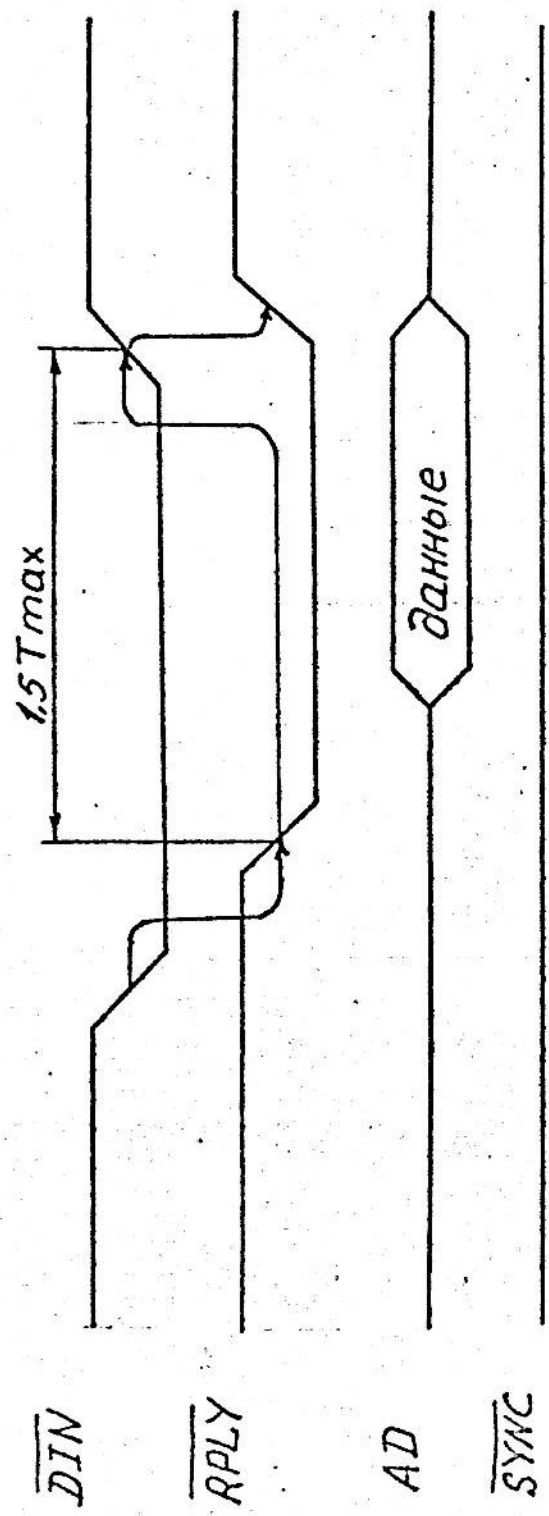


Рис. 23

ИИЗ.480.167 ТО

Лист
61

Инд. № подл.	Подп. и дата	Взам. инв. №	Инд. № докум.	Подп. и дата
90394	10.07.870Вм.			

Запись в ППЗ

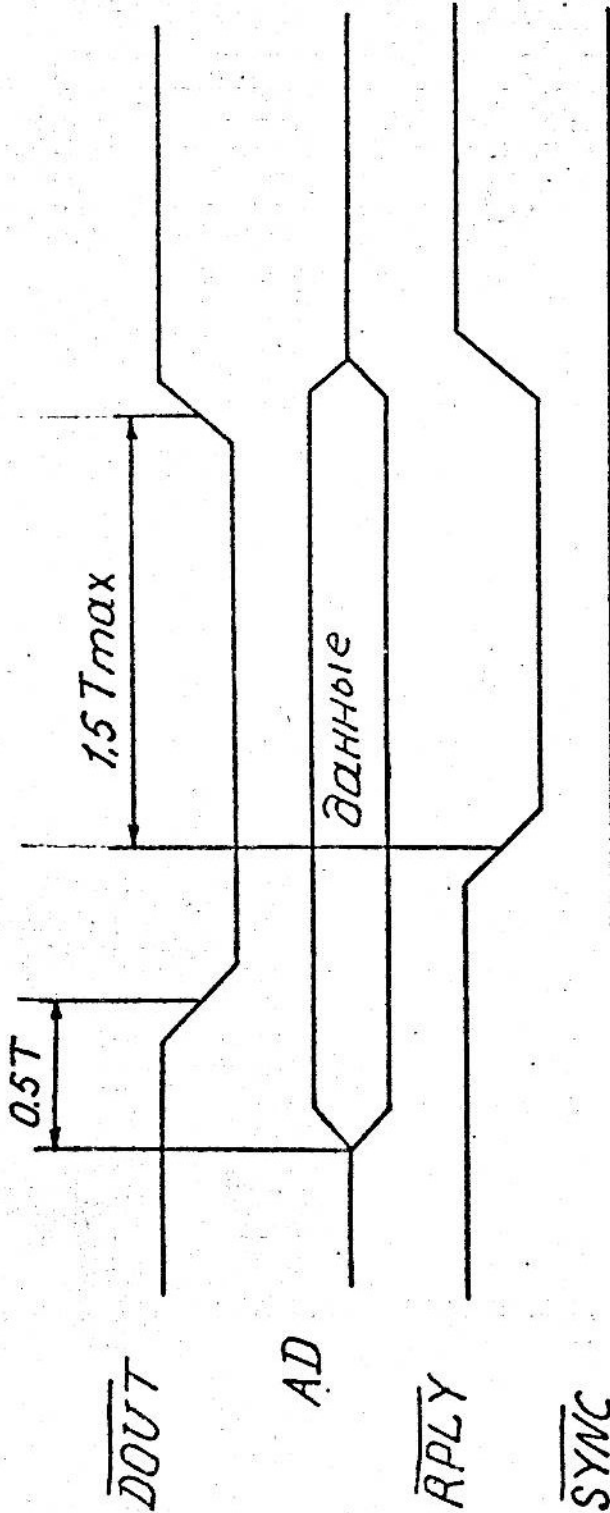


Рис. 24

ИИЗ.480.167 ТО

Лист
62

в процедурах обмена (рис. 15-19) представлены в табл. 7

Таблица 7

Обозначение		Максимальное значение, нс	Минимальное значение, нс
t_{SA}	Время установки адреса	160 60	
t_{RA}	Время снятия адреса	100 60	
t_{SD}	Время установки данных	80 60	
t_{RD}	Время снятия данных	80 60	
t_{SOAD}	Время установки отключенного состояния на выводах AD (0-15)	80 60	
t_{ROAD}	Время снятия отключенного состояния на выводах AD (0-15)	60 60	
	Задержка выдачи сигналов SYNC, DIN, DOUT относительно синхросигналов	100 45	
	Задержка снятия сигналов SYNC, DIN, DOUT относительно синхросигналов	100 15	10
	Время предустановки сигналов SSYNC, RPLY относительно синхросигналов	100 20	
	Время от снятия сигнала RPLY до синхросигнала		10

ЧИВ. № подл. 90394
 Подп. и дата 10.07.87
 ЧИВ. № инв. 10.07.87
 Подп. и дата

ИИЗ.480.167 ТО

Лист
63

7. РАБОТА ПРЦ

ПРЦ в составе ЭВМ управляет распределением времени использования канала внешними устройствами и выполняет все необходимые арифметико-логические операции для обработки информации. Он содержит восемь быстродействующих регистров общего назначения (РОН), которые широко используются при выполнении различных операций. ПРЦ может обрабатывать как шестнадцатиразрядные слова, так и восьмиразрядные байты. Возможность использования восьми методов адресации позволяет вести высокоэффективную обработку данных, хранимых в любой ячейке памяти или в регистре. В состав ПРЦ входят шесть программируемых логических матриц, каждая из которых имеет свое назначение. Особо нужно отметить наличие в ПРЦ блока диспетчера памяти, который значительно расширяет возможности ПРЦ.

7.1. Регистры общего назначения (РОН)

ПРЦ содержит восемь шестнадцатиразрядных регистров общего назначения, один регистр *PSW* - слово состояния процессора, шесть-внутренние служебные регистры процессора. Программно доступными являются все регистры общего назначения и регистр *PSW*. Остальные регистры доступны микропрограммно, причем регистры *R0...R4* доступны лишь из поля адресации команды.

РОН общего назначения могут служить в качестве накопительных регистров, индексных регистров, регистров автоинкрементной и автодекрементной адресации, так называемых указателей стека и для других целей. РОН используется для выборки операндов и записи результатов при выполнении арифметико-логических операций аналогично ячейкам памяти и регистрам внешних

Инв. № подл.	90394
Подл. и дата	10.07.87 Звн-
Взам. инв. №	
Инв. № док. инв.	
Подл. и дата	

Инв. № подл.	90394	Подл. и дата	10.07.87 Звн-	Инв. № док. инв.		Взам. инв. №		Подл. и дата		Инв. № подл.		Лист	64
Инв. № подл.		Подл. и дата		Инв. № док. инв.		Взам. инв. №		Подл. и дата		Инв. № подл.		Лист	

ИИЗ. 480. I67 TO

устройств. Два из 16 РОН R6 и R7 имеют, кроме того, специальное назначение. Регистр R6 (R16) используется как указатель стека (SP) и содержит адрес последней заполненной ячейки стека. Регистр R7 служит счетчиком команд (PC) и содержит адрес очередной выполняемой команды. Обычно он используется для целей адресации и не используется как накопительный регистр.

Операции по выполнению команд с регистровым методом адресации являются внутренними по отношению к ПРЦ и не требуют выполнения циклов обращения к каналу (за исключением цикла выборки команды). Обмен же данными с памятью и внешними устройствами выполняется через канал и занимает более длительное время. Таким образом, использование РОН для хранения операндов при процессорных операциях повышает быстродействие ЭВМ.

7.2. Слово состояния процессора (PSW).

Слово состояния ПРЦ содержит информацию о текущем состоянии ПРЦ. Это информация о текущем приоритете ПРЦ, об операционной моде ПРЦ, значении кодов условий ветвления, зависящих от результатов выполнения команд и состояния Т-разряда, используемого при отладке программ и вызывающего прерывание программ. Формат PSW показан на рис. 25.

Тринадцатый, двенадцатый разряды PSW содержат информацию о предыдущей моде микропроцессора, а пятнадцатый, четырнадцатый разряды, — о текущей моде микропроцессора. Возможность и способы загрузки различных разрядов приведены в табл. 5.

Инв. № подл.	Подп. и дата
90394	10.07.87
Взам. инв. №	Инв. № дубл.
Подп. и дата	Подп. и дата

ИИЗ.480.167 Т0

Лист
65

Слово состояния процессора

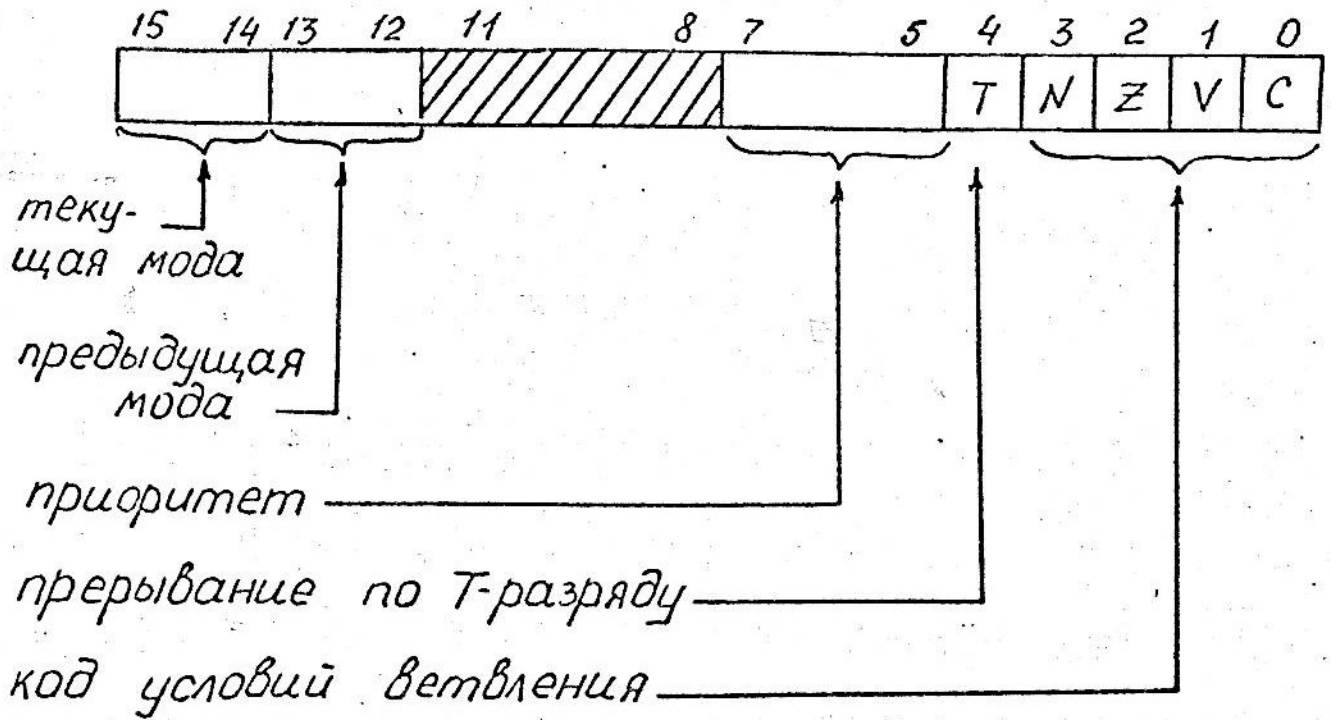


рис. 25

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № докл.	Посл. и дата
90394	19.07.87			

Изм.	Лист	№ докум.	Подп.	Дата

ШИЗ.480.167 ТО

Лист

66

Разряды PSW	RTI, RTT		Команды прерываний и внешние прерывания	
	Пользователь	ОС	Пользователь	ОС
Код признаков PSW (3-0)	Загружается из стека	Загружается из стека	Загружается из вектора прерываний	Загружается из вектора прерываний
T-бит PSW (4)	Загружается из стека	Загружается из стека	Загружается из вектора прерываний	Загружается из вектора прерываний
Уровень приоритета PSW (7-5)	Не изменяется	Загружается из стека	Загружается из вектора прерываний	Загружается из вектора прерываний
Предыдущая мода PSW (I3-I2)	Не изменяется	Загружается из стека	Переписывается из PSW (I5-I4)	Переписывается из PSW (I5-I4)
Текущая мода PSW (I5-I4)	Не изменяется	Загружается из стека	Загружается из вектора прерываний	Загружается из вектора прерываний

№ п/п подл. 90324
 Подп. и дата 10.07.87
 Взам. инв. №
 Инв. № дубл.
 Подп. и дата

Таблица 5

	Запись в PSW по адресу I77776		MTPS	
	Пользователь	ОС	Пользователь	ОС
ся из реры-	Загружается из источника	Загружается из ис- точника	Загружается из источника	Загружается из источника
ся из реры-	Не изменяется	Не изменяется	Не изменяется	Не изменяется
ся из реры-	Загружается из источника	Загружается из источника	Загружается из источника	Загружается из источника
вается	Загружается из источника	Загружается из источника	Не доступны	Не доступны
тся ра и	Загружается из источника	Загружается из источника	Не доступны	Не доступны

Илл	Лист	№ докум.	Подп.	Дата

ИДЗ.480.167 ТО

Лист

67

Так как выполнение команды **MTPS** в моде пользователя аппаратурой разрешено, то необходимо в математическом обеспечении предусмотреть запрет обращения по адресу **PSW** пользователю.

ПРЦ работает в любом из восьми уровней приоритета, от 0 до 7. Когда код приоритета равен "7", внешние устройства не могут вызвать прерывания программы. Если код приоритета больше или равен "4", то работа процессора может прерываться в случае возникновения сигналов **IRQ 3-0**. Соответствие между сигналами и кодами приоритета показано в табл. 6

Таблица 6

Внешние сигналы	Код приоритета
IRQ 3	7
IRQ 2	7,6
IRQ 1	7,6,5
IRQ 0	7,6,5,4
EVNT	7,6

Коды условий ветвления содержат информацию о результате последней выполняемой ПРЦ команды.

- N** - отрицательный результат
- Z** - нулевой результат
- V** - арифметическое переполнение
- C** - перенос

Процедуру установки их в соответствующее состояние выполняют все арифметические и логические команды. Установка отдельных разрядов этих кодов выполняется в следующих случаях:

Z = 1, если результат равен 0;

ИДБ. № подл. 90394
 Подп. и дата 10.07.87
 ИДБ. № подл. ИДБ. № докл. ИДБ. № зам. ИДБ. № подл. ИДБ. № дата

$V = I$, если результат отрицательный;

$V = I$, если в результате выполнения операции произошло арифметическое переполнение;

$C = I$, если в результате выполнения операции произошел перенос из самого старшего разряда или, если при сдвиге вправо или влево из самого младшего или из самого старшего разряда была выдвинута единица.

При загрузке нового слова состояния процессора может установиться или очиститься T-разряд. Если он установлен, то по завершении выполнения текущей команды будет вызвано прерывание программ с адресом вектора прерывания - четырнадцать. Использование T-разряда особенно эффективно в отладочных программах, когда исполнение интересующих пользователя команд вызывает прерывание программы и переход на программу связи с оператором. Это дает возможность пользователю предпринять необходимые действия.

7.3. Организация стека

Удобным и эффективным методом работы с данными является использование стека, который представляет собой область памяти для временного хранения данных. В стек можно заносить и из стека можно вынимать слова или байты. Стек работает, используя принцип "последний вошел, первый вышел", то есть различные данные могут быть занесены в стек в последовательном порядке и могут быть взяты из стека в обратном порядке.

7.4. Организация памяти

Память может быть рассмотрена, как последовательность ячеек, каждой из которых соответствует свой адрес. На рис. 26 представлены 8192 слов памяти ПРЦ

ИНВ. № подл. Подп. и дата. Взам. инв. №. Инв. № докум. Подп. и дата. 90394 10.07.87 авч.

ИИЗ.480.167 ТО

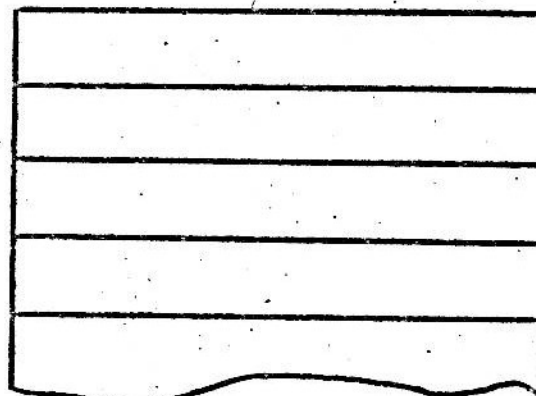
Лист

69

Изм. Лист. Док. Подп. Дата

Восьмеричный
адрес

000000
000001
000002
000003
000004
.
.
.



037774
037775
037776
037777

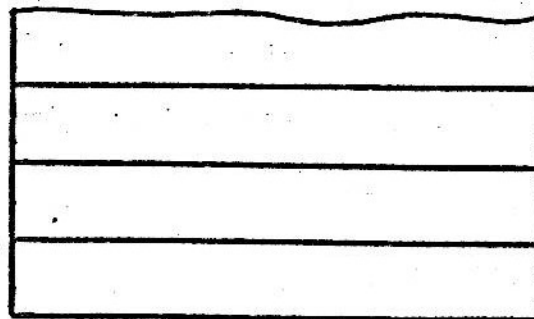


Рис.26. Адресация памяти

Слово делится на младший и старший байты, как показано на рис.27.

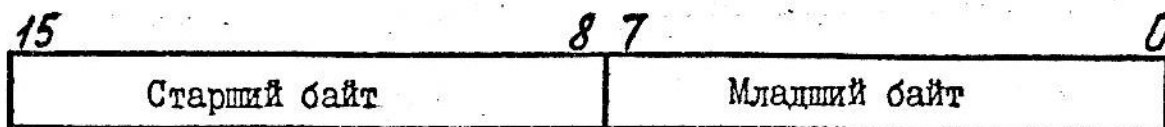


Рис.27. Формат слова процессора

Ячейки, содержащие полные слова, всегда имеют четные адреса. Адреса байтов могут быть как четными, так и нечетными. Младшие байты хранятся в ячейках с четным адресом, старшие - с нечетным (рис.28).

Инв. № инв. 90394
Засл. инв. № 10.07.87
Лист и дата 10.07.87
Лист и дата

Изм.	Лист	№ докум.	Подп.	Дата

ИИЗ.480.167 ТО

Лист

70

Формат А4

байт		байт	байт	
Старший	Младший	000000	Младший	000000
Старший	Младший	000002	Старший	000001
Старший	Младший	000004	Младший	000002
Старший	Младший	000005	Старший	000003
Старший	Младший	000010	Младший	000004
Старший	Младший	017774	Младший	017776
Старший	Младший	017776	Старший	017777

Рис. 28 Организация памяти ЭВМ по словам и байтам для первых 4К адресов

Определенные области памяти зарезервированы для использования при обработке внутренних и внешних прерываний программ, для стека ПРЦ, для интерфейсных регистров.

7.5. Программируемые логические матрицы (ПЛМ)

В тех случаях, когда в процессоре требуется реализовать сложную булеву функцию или систему функций от многих переменных, применяются программируемые логические матрицы. Основным достоинством ПЛМ является регулярность их структур, что определяет применение на этапе проектирования более систематических методов и легкость последующих изменений.

ШИЗ.480.167 Т0

Лист

71

Изм. Лист. Идентиф. Подп. Дата

Формат 5а ГОСТ 2.105-58

Формат А4

Идентиф. Подп. Дата
90394 10.07.8702ч.

8. СИСТЕМА КОМАНД

В ПРЦ используется три типа команд: безадресные, одноадресные и двухадресные. В безадресных командах код команды содержит только код операции. В кодах одноадресных и двухадресных команд обычно содержится информация, которая определяет:

выполняемую функцию (код операции);

регистры общего назначения, используемые при выборке операндов;

метод адресации (способ использования выбранного РОН).

Так как большая часть информации, обрабатываемой ЭВМ, представляет собой данные, сформированные в массивы, списки, потоки символов и т.д. ПРЦ разработан с учетом возможности эффективной обработки сформированных структур данных.

Регистры общего назначения могут быть использованы:

как накопители (обрабатываемые данные хранятся в регистрах);

как указатели адреса (регистр содержит адрес операнда, а не сам операнд);

как указатели адреса, содержимое которых изменяется автоматически с заданным шагом, что позволяет обращаться к последовательно расположенным ячейкам памяти. Автоматическое увеличение содержимого указателя адреса при обращении к последовательным ячейкам памяти носит название автоинкрементной адресации. Автоматическое уменьшение содержимого указателя при обращении к последовательным ячейкам памяти носит название автодекрементной адресации. Эти методы могут быть использованы при обработке табулированных данных;

ШИЗ.480.167 ТО

Лист

72

Имя, № подл. Пост. и дата. Изм. Имя, № подл. Пост. и дата. 90394 10.07.87

Имя, № подл. Пост. и дата. Изм. Имя, № подл. Пост. и дата.

Форма 50 ГОСТ 2.105-68

Формат А4

как индексные регистры, содержимое которых присваивается к индексному слову для вычисления адреса операнда, что позволяет обращаться к различным элементам списка.

Использование автоинкрементной и автодекрементной адресации дает возможность организации стековой памяти. В качестве указателя стека программно может быть использован любой PОН, однако определенные команды (используемые при обслуживании прерываний, обращения и возврата из подпрограмм) автоматически используют PОН R6 как аппаратный указатель стека.

8.1. Формат одноадресных команд

Формат одноадресных команд (таких как очистка, проверка) имеют следующий вид:

КОП	Метод	PОН
15	06 05	03 02 00

Разряды 15-06 содержат код операции, который определяет исполняемую команду. Разряды 05-00 образуют шестизначное поле, именуемое полем адресации операнда - приемника, которое в свою очередь состоит из двух полей:

1) разряды 02-00 определяют один из 8 PОН, который использует данная команда;

2) разряды 05-03 определяют способ использования выбранного регистра (метод адресации). Причем разряд 03 определяет прямую или косвенную адресацию.

8.2. Формат двухадресных команд

Операции над двумя операндами (такие как сложение, пересылка, сравнение) выполняются с помощью команд, в которых имеются два адреса. Задание разрядов в поле адресации первого и

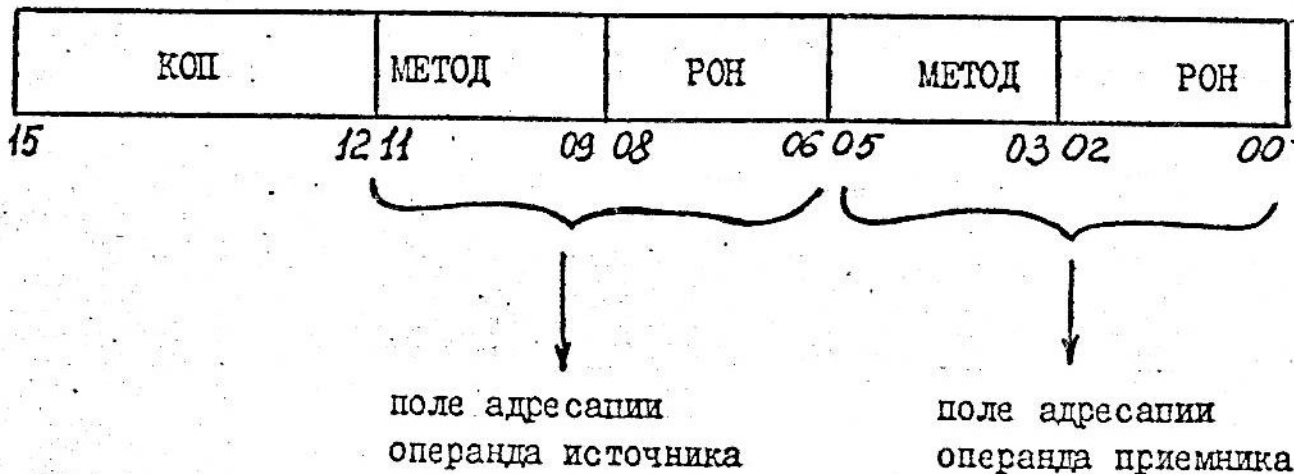
Ш.№: 90394
 Подп. и дата: 10.07.87
 Имя: Ильяшвили
 Подп. и дата:

Изм.	Лист	Всего	Подп.	Дата

ИИЗ.480.167 ТО

Лист
73

второго операндов определяют различные методы адресации и различные регистры общего назначения. Формат двухадресной команды имеет следующий вид:



Поле адресации операнда источника используется для выборки операнда источника или первого операнда. Поле адресации операнда приемника используется для выборки второго операнда (операнда источника) и занесения результата. Например по команде **ADD A, B** складывается содержимое ячейки A (операнд источника) с содержимым ячейки B (операнд приемника). После выполнения операции сложения в ячейке B будет находиться результат операции, а содержимое ячейки A не изменится.

8.3. Методы адресации

В примерах данного подраздела при описании методов адресации используются приведенные в табл.Б следующие команды:

Ш.Н.В. № подл. 90394
 Подп. и дата 10.07.87
 Взам. Ш.Н.В. № 10.07.87
 Подп. и дата

Ш.ИЗ.480.167 ТО

Лист
44

Мнемо-ника	Описание	Восьмеричный код
<i>CLR</i>	Очистка. В указанную ячейку заносятся во все разряды нули	0050 <i>DD</i>
<i>CLRВ</i>	Очистка байта. В старший или младший байт указанной ячейки заносятся нули	1050 <i>DD</i>
<i>INC</i>	К содержимому указанной ячейки прибавляется единица	0052 <i>DD</i>
<i>INCВ</i>	Прибавление единицы к содержимому байта.	1052 <i>DD</i>
<i>COM</i>	Инвертирование. Заменяет операнд его обратным кодом, т.е. все разряды равные 0 устанавливаются в 1, а все разряды равные 1 устанавливаются в 0.	0051 <i>DD</i>
<i>COMВ</i>	Инвертирование байта	1051 <i>DD</i>
<i>ADD</i>	Сложение. Содержимое операнда источника (<i>SS</i>) складывается с содержимым операнда приемника (<i>DD</i>) и заносится по адресу операнда приемника	06 <i>SSDD</i>

8.3.1. Методы прямой адресации

На рис. 29 показаны последовательности операций выполнения команды с каждым из четырех методов прямой адресации. При регистровом методе адресации любой из восьми PОН может быть использован как накопитель. Следовательно, операнд будет нахо-

ЧИБ. № ПОД. ПОДП. И ДОСТА. ВЗОМ. ЧИБ. № ДОК. ПОДП. И ДОСТА.
 90394 10.07.87 08/м

ШИЗ.480.167 ТО

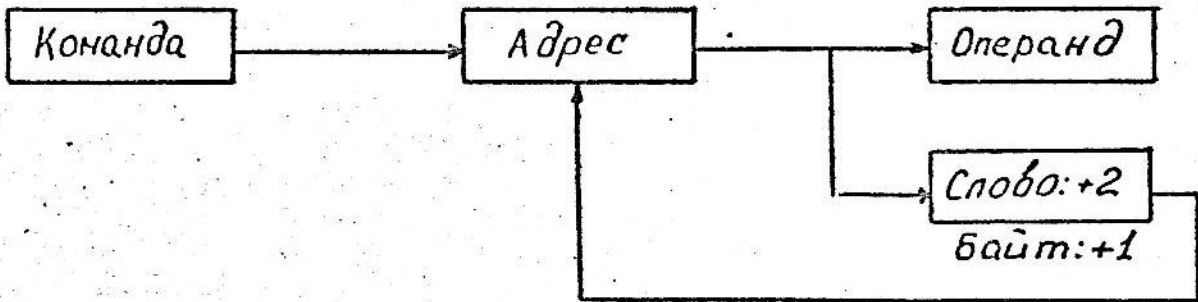
ЛИСТ
75

Методы прямой адресации

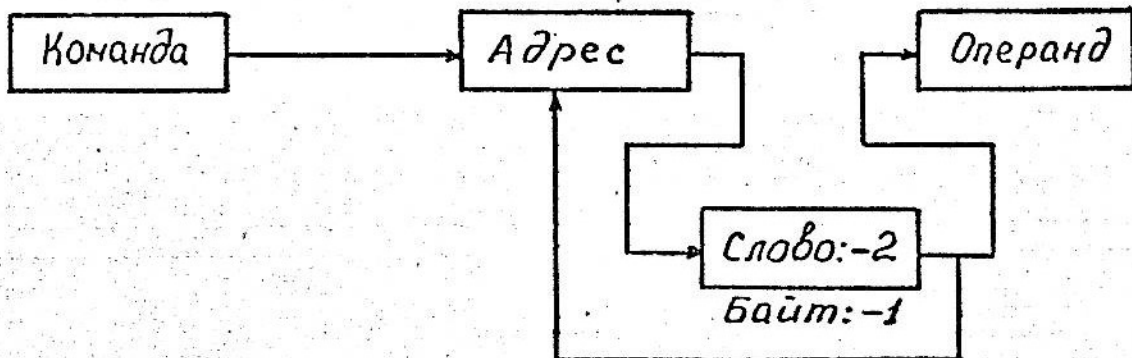
1) Регистровый метод адресации (метод 0)



2) Автоинкрементный метод адресации (метод 2)



3) Автодекрементный метод адресации (метод 4)



4) Индексный метод адресации (метод 6)

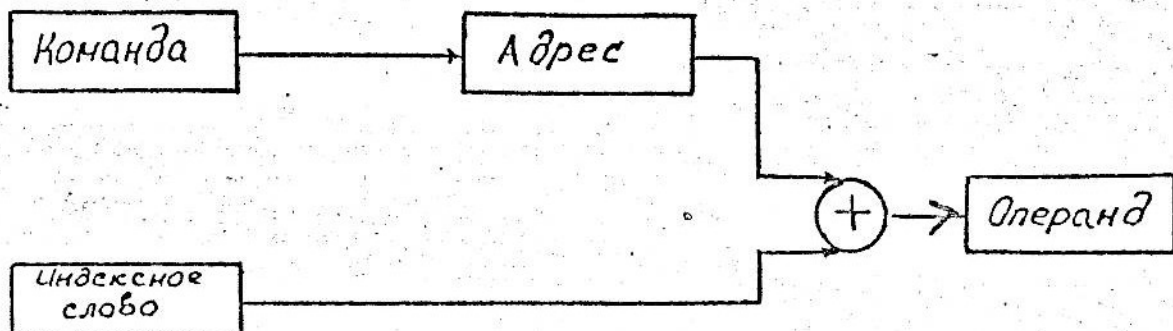


Рис. 29

Инв. № подл.	90394
Подп. и дата	10.07.87 Овс.
Взам. инв. №	
Инв. № дубл.	
Подп. и дата	

Действие: Заменяется содержимое разрядов 07-00 в R4 их обратным кодом (байтовые команды, использующие PОН, оперируют только с младшим байтом информации, содержащимся в PОН).

До выполнения операции

R 4) 022222

После выполнения операции

R 4) 022I55

Автоинкрементный метод адресации используется при автоматическом обращении с заданным шагом к последовательным элементам таблицы операндов. Содержимое выбранного регистра автоматически наращивается для возможности обращения в дальнейшем к следующей ячейке. При байтовых операциях наращивание происходит на единицу, при операциях с полными словами — на 2, а содержимое R6, R7 всегда наращивается на 2. Хотя этот метод наиболее удобен при работе с таблицами, он может быть использован как общий метод для различных целей.

Примеры автоинкрементной адресации (метод 2)

1. CLR (R5)+

005025

Очистка

Действие: Содержимое ячейки, адрес которой содержится в R5, очищается, после чего адрес (содержимое R5) увеличивается на 2.

До выполнения операции

20000) 005025; R5) 030000;

30000) IIIII6

После выполнения операции

20000) 005025; R5) 030002;

30000) 000000

2. CLRВ (R5)+

I05025

Очистка байта

Действие: Очищается выбранный байт, адрес которого содержится в R5, после чего адрес (содержимое R5) увеличивается на единицу

До выполнения операции

20000) I05025; R5) 030000;

30000) IIIII6

После выполнения операции

20000) I05025; R5) 03000I;

30000) IIII00

ИИЗ. № ПОС. ПОСЛ. И ОСТА. ВЗОМ. ИИЗ. ИИЗ. № ДОС. ПОСЛ. И ОСТА.

90394 10.07.8702

ИИЗ. Лист / ФОРМУЛ. ПОСЛ. И ОСТА.

3. ADD(R2)+R4

062204

Сложение

Действие: Операнд, адрес которого содержится в R2, складывается с содержимым R4, а содержимое R2 (адрес источника) наращивается на 2. Результат заносится в R4.

До выполнения операции	После выполнения операции
I00) 062204; R2) 000204;	I00) 062204; R2) 000206;
R4) 001000;	R4) 003000;
204) 002000	204) 002000

Автодекрементный метод адресации также используется для обработки табулированных данных. Однако, в отличие от автоинкрементного метода, адресация к ячейкам массива идет в противоположном направлении. При этом методе адресации содержимое выбранного PОН уменьшается (для байтовых команд на единицу, для команд с полными словами на 2), а затем используется как исполнительный адрес.

Сочетание автоинкрементного и автодекрементного методов адресации может быть эффективно использовано для обработки стека.

Примеры автодекрементной адресации (метод 4)

Символическое обозначение	Восьмеричный код	Наименование команды
1. INC-(R0)	005240	Прибавление единицы

Действие: Содержимое R0 уменьшается на 2 и используется как исполнительный адрес. К содержимому, выбранному из ячейки по этому адресу, прибавляется единица.

До выполнения операции	После выполнения операции	
I00) 005240; R0) 017776;	I00) 005240; R0) 017774;	
I7774) 000000	I7774) 000001	
2. INCB-(R0)	I05240	Прибавление единицы к байту

ШИЗ. № подл. 90394
 Подп. и дата 10.07.87
 ШИЗ. № подл. 90394
 Подп. и дата 10.07.87
 ШИЗ. № подл. 90394
 Подп. и дата 10.07.87

ШИЗ.480.167 Т0

Лист 49

До выполнения операции	После выполнения операции
I020) 005064; R4) 001000;	I020) 005064; R4) 001000;
I022) 000200;	I022) 000200;
I200) 177777	I200) 000000

2. COMB 200 (RI) I0516I Инвертирование байта
000200

Действие: Адрес операнда определяется прибавлением к содержимому регистра RI кода 200, после чего содержимое старшего байта ячейки с адресом на единицу меньше вычисленного адреса заменяется его обратным кодом.

До выполнения операции	После выполнения операции
I020) I0516I; RI) 017777;	I020) I0516I; RI) 017777
I022) 000200;	I022) 000200;
20176) 011000	20176) 166000

3. ADD30 (R2) 20 (R5) 066265 Сложение
000030
000020

Действие: Содержимое ячейки, адрес которой вычисляется сложением кода 30 с содержимым R2 (операнд источника), складывается с содержимым ячейки, адрес которой определяется сложением кода 20 с содержимым R5 (операнд приемника). Результат записывается по адресу операнда приемника.

До выполнения операции	После выполнения операции
I020) 066265; R2) 001100;	I020) 066265; R2) 001100;
I022) 000030; R5) 002000;	I022) 000030; R5) 002000;
I024) 000020;	I024) 000020;
I130) 000001;	I130) 000001;
2020) 000001	2020) 000002

УНБ. № подл. 90394
 Дата 10.07.87
 Подп. и дата
 УНБ. № подл. 90394
 Дата 10.07.87
 Подп. и дата
 УНБ. № подл. 90394
 Дата 10.07.87
 Подп. и дата

ШИЗ.480.167 ТО

Лист
81

8.3.2. Методы косвенной адресации

Четыре основных метода могут быть использованы с косвенной адресацией. Если в регистровом методе операндом является содержимое выбранного регистра, в косвенно-регистровом методе содержимое выбранного регистра является адресом операнда. В трех других косвенных методах содержимое регистра позволяет выбрать адрес операнда, а не сам операнд. Эти методы используются, когда таблица состоит из адресов, а не операндов.

На рис. 30 показаны последовательности операций выполнения команды с каждым из четырех методов косвенной адресации.

Пример косвенно-регистрового метода адресации (метод I).

Символическое обозначение	Восьмеричный код	Наименование команды
<i>CLR @R5</i>	005015	Очистка

Действие: Содержимое ячейки, адрес которой находится в R5, очищается.

До выполнения операции	После выполнения операции
I020) 005015; R5) 001700;	I020) 005015; R5) 001700;
I700) I77777	I700) 000000

Пример косвенно-автоинкрементного метода адресации

(метод 3)

<i>INC@ (R2)+</i>	005232	Прибавление единицы
-------------------	--------	---------------------

Действие: содержимое ячейки, адрес которой содержится в R2, используется как адрес операнда. Операнд увеличивается на единицу, а содержимое R2 на 2.

До выполнения операции	После выполнения операции
I000) 005232; R2) 010300;	I000) 0005232; R2) 010302;
I010) 000000;	I010) 000001;
I0300) 001010	I0300) 0001010

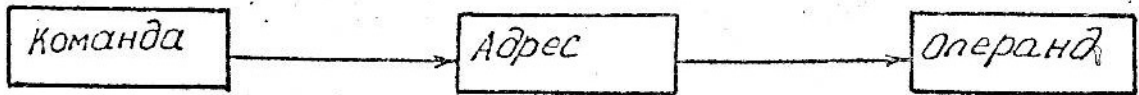
Ш.Б. № 90394
 Дата 10.07.87
 Подп. и дата
 Взам. инв. № 10.07.87
 Инв. № 10.07.87

ИИЗ.480.167 ТО

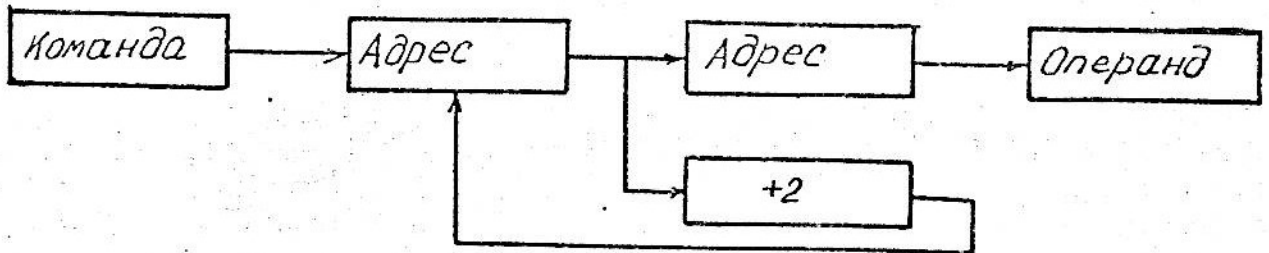
Лист
82

Методы косвенной адресации

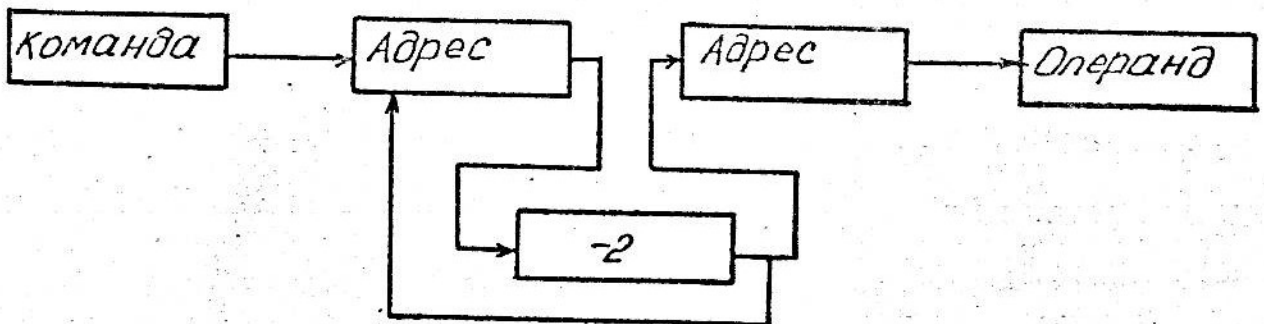
1) Косвенно-регистровый метод адресации (метод 1)



2) Косвенно-автоинкрементный метод адресации (метод 3)



3) Косвенно-автодекрементный метод адресации (метод 5)



4) Косвенно-индексный метод адресации (метод 7)

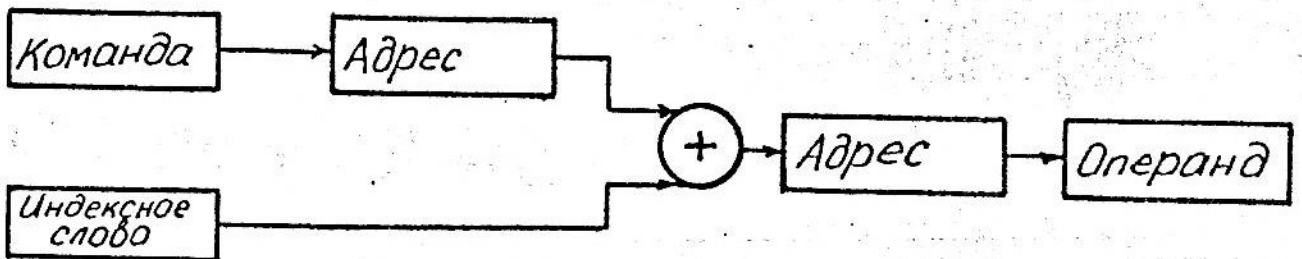


Рис. 30

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № докл.	Подп. и дата
90394	10.07.87			

Пример косвенно-автодекрементной адресации (метод 5)

COM@ - (R0)

005150

Инвертирование

Действие: Содержимое R0 уменьшается на 2 и используется как адрес ячейки, в которой находится адрес операнда. Операнд заменяется его обратным кодом.

До выполнения операции

После выполнения операции

I000) 005150; R0) 010776;

I000) 005150; R0) 010774;

I0100) 012345;

I0100) 165432;

I0774) 010100

I0774) 010100

Пример косвенно-индексной адресации (метод 7)

ADD @1000 (R2), R1 067201

Сложение

001000

Действие: Содержимое ячейки, адрес адреса которой определяется сложением кода I000 с содержимым R2, складывается с операндом, хранящимся в R1. Результат записывается в R1.

До выполнения операции

После выполнения операции

I020) 067201; R1) 001234;

I020) 067201; R1) 001236;

I022) 001000; R2) 000100;

I022) 001000; R2) 000100;

I050) 000002;

I050) 000002;

I100) 001050

I100) 001050

8.3.3. Использование счетчика команд (PC) в качестве регистра общего назначения

Регистр R7, являясь одним из восьми PОН, выполняет в центральном ПРЦ специальную функцию счетчика команд. Когда ПРЦ использует счетчик команд для выборки слова из памяти, его содержимое автоматически увеличивается на 2. Новое содержимое счетчика команд является адресом слова, используемого при выполнении данной команды. Следует отметить, что при работе с байтами содержимое PC также увеличивается на 2.

УНБ. № подл. 90394
 Дата. 19.07.87
 Подп. и дата. Взам. унб. Инв. № подл. Подп. и дата.

ШИЗ.480.167 Т0

Лист

84

Счетчик команд может быть использован во всех методах адресации, применяемых в ПРЦ. Однако, наиболее эффективно он используется только с четырьмя методами адресации. Эти методы адресации получили специальные наименования: непосредственный, абсолютный, относительный и косвенно-относительный. Использование этих методов дает возможность построения программы, работоспособность которой не теряется при перемещении ее в любую область памяти. В табл. 8 приведены методы адресации с использованием R7.

Таблица 8

Двоичный код	Наименование	Функция
010	Непосредственный #n	Операнд выбирается из ячейки, следующей за командным словом
011	Абсолютный @ # A	Из ячейки, следующей за командным словом, выбирается адрес операнда
110	Относительный X (PC) или A	Операнд выбирается из ячейки, адрес которой определяется как сумма содержимого PC и ячейки, следующей за командным словом
111	Косвенно-относительный @ X (PC) или @ A	Из ячейки, адрес которой определяется как сумма содержимого PC и ячейки, следующей за командным словом, выбирается адрес операнда

Методы адресации с использованием PC в значительной мере упрощают обработку данных, несформированных в массивы.

УИВ. №: подл. Подп. и дата. Взам. УИВ. №: подл. Подп. и дата.
 90394 12.07.87 г.

ШИЗ.480.167 ТО

Лист
85

Непосредственный метод адресации имеет символическое обозначение $\#n$. Он эквивалентен автоинкрементному методу адресации через PC.

Этот метод обеспечивает удобство написания программы и экономию времени программиста путем помещения константы в ячейку памяти вслед за командным словом.

Пример непосредственной адресации.

ADD # 10, R0	062700	Сложение
	000010	

Действие: Содержимое R0 складывается с числом 10. Результат записывается в R0.

До выполнения операции	После выполнения операции
I020) 062700; R0) 000020;	I020) 062700; R0) 000030;
I022) 000010	I022) 000010

Примечание. Непосредственно перед выборкой этой команды PC указывает на командное слово.

ПРЦ выбирает командное слово и увеличивает PC на 2. В поле адреса операнда источника записан код 27, следовательно, PC используется как указатель при выборке операнда перед увеличением его содержимого на 2, для указания на следующую команду.

Абсолютный метод адресации имеет символическое обозначение $@\#A$. Он эквивалентен косвенно-автоинкрементной адресации через PC. Этот метод удобен тем, что адрес операнда является его абсолютным адресом (т.е. он остается постоянным независимо от места расположения программы в памяти).

CLR @ # 1100	005037	Очистка
	001100	

Действие: Содержимое ячейки, следующей за командой, используется в качестве адреса операнда (в данном случае исполнительным

ШИЗ.480.167 ТО

Лист
86

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата
90394	10.07.87			

адресом является код II00). Содержимое ячейки с адресом II00 очищается.

До выполнения операции	После выполнения операции
20) 005037;	20) 005037;
22) 00II00;	22) 00II00;
II00) I77777	II00) 000000

Относительный метод адресации имеет символическое обозначение X (PC) или A, где X – исполнительный адрес по отношению к счетчику команд. Этот метод эквивалентен индексной адресации через PC. Индексное слово хранится в следующей за командным словом ячейке и, будучи сложено с содержимым PC, дает адрес операнда. Этот метод полезен при написании программы, которая может располагаться в различных местах памяти, так как адрес операнда фиксируется по отношению к PC. При необходимости перемещения программы в памяти операнд перемещается на то же число ячеек, что и сама команда.

Пример относительной адресации

INCA	005267	Прибавление единицы
	000054	

Действие: К операнду, адрес которого определяется сложением содержимого PC и индексного слова (000054), прибавляется единица.

До выполнения операции	После выполнения операции
I020) 005267;	I020) 005267;
I022) 000054;	I022) 000054
II00) 000000	II00) 000001

Косвенно-относительный метод адресации имеет символическое обозначение @ X(PC) или @ A, где X – адрес ячейки, содержащий исполнительный адрес, по отношению к счетчику команд. Этот метод эквивалентен косвенно-индексной адресации через PC.

Инв. № подл.	90394
Подп. и дата	10.07.87 Вм
Взам. инв. №	
Инв. № дубл.	
Подп. и дата	

Пример косвенно-относительной адресации.

CLR @ A

005077

Очистка

000020

Действие: Очищается содержимое ячейки, адрес которой определяется следующим образом: к содержимому счетчика команд прибавляется индексное слово, определяя тем самым адрес ячейки, в которой хранится исполнительный адрес.

До выполнения операции

После выполнения операции

I020) 005077;

I020) 005077;

I022) 000020;

I022) 000020;

I024)

I024)

I044) 010100;

I044) 010100;

I0100) I23456

I0100) 000000

8.3.4. Использование указателя стека в качестве регистра общего назначения

Регистр R6, являясь одним из PОН, используется в ПРЦ как указатель адреса при обращении к той части памяти, которая отводится под стек. С помощью автодекрементной адресации через R6 данные записываются в стек, а с помощью автоинкрементной адресации производится выборка данных из стека. Индексный метод адресации позволяет производить произвольную выборку элементов стека.

Так как R6 (SP) используется для обслуживания прерываний, то его особенностью является то, что уменьшение и увеличение содержимого SP всегда производится с шагом два. В байтовых операциях содержимое ячеек с нечетными адресами не изменяется.

8.4. Выполнение команд

Описание каждой команды включает: мнемонику, восьмеричный код, формат команды, двоичный код, описание выполнения команды

Инв. № подл.	90394
Подп. и дата	10.07.87 Вм
Взам. инв. №	
Инв. № дубл.	
Подп. и дата	

Инв. № подл.	90394	Подп. и дата	10.07.87 Вм	Взам. инв. №		Инв. № дубл.		Подп. и дата		Илст	88
--------------	-------	--------------	-------------	--------------	--	--------------	--	--------------	--	------	----

ИИЗ.480.167 Т0

и выработки признаков, специальные пояснения и примеры.

При описании команд используются следующие обозначения:

- R - регистр общего назначения;
- PC - счетчик команд (R7);
- SP - указатель стека (R6);
- PSW - регистр состояния процессора;
- PSW - слово состояния процессора;
- SS - поле адресации операнда источника;
- src - источник;
- (src) - операнд источника;
- DD - поле адресации операнда приемника;
- dst - приемник;
- (dst) - операнд приемника;
- XXX - смещение (8 разрядов);
- NN - смещение (6 разрядов);
- () - содержимое ячейки;
- Λ - логическое умножение ("И");
- V - логическое сложение ("ИЛИ");
- ∩ - исключаящее "ИЛИ";
- \bar{A} - отрицание A ("НЕ");
- ← - становиться равным;
- ↓ - запись в стек;
- ↑ - выборка из стека;
- B - байтовая команда;
- temp - временное хранение.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № док.	Подп. и дата
90394	10.07.87	Взм.	90394	10.07.87

ШИЗ.480.167 ТО

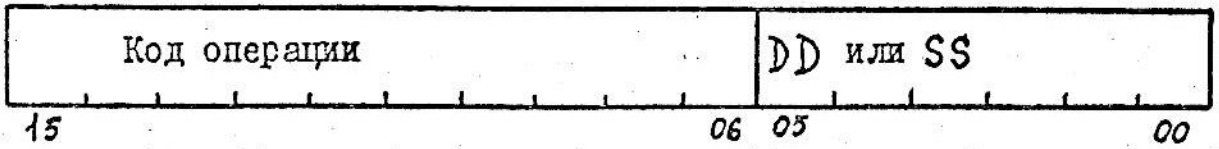
Лист

89

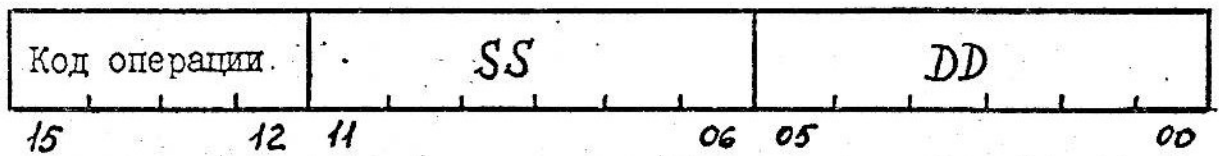
8.4.I. Форматы команд

Ниже приведены форматы команд, реализуемые ППЦ

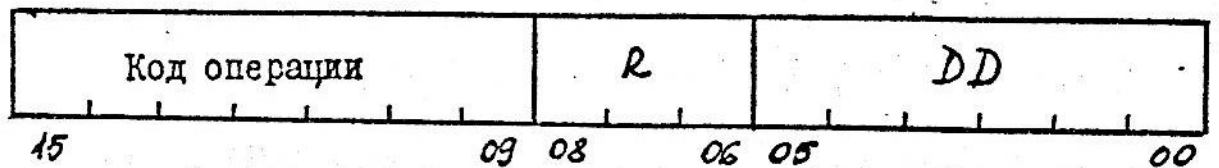
I. Одноадресные команды (CLR, COM, INC, DEC, NEG, ADC, SBC, TST, ROR, ROL, ASR, ASL, JMP, SWAB, MFPS, MTPS, SXT)



2. Двухадресные команды (BIT, BIC, BIS, ADD, SUB, MOV, CMP)

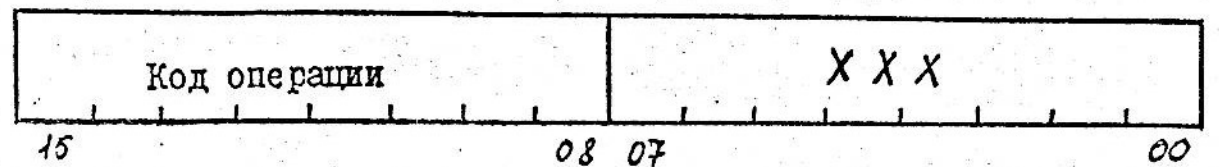


Команда XOR

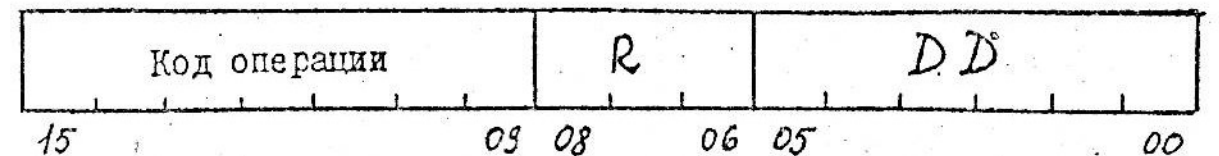


3. Команды управления программой

I) команды ветвления



2) команда JSR

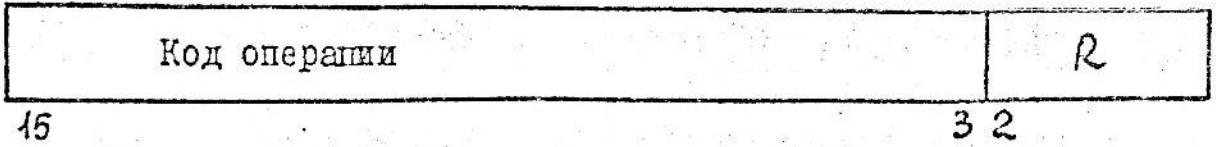


Инв. № подл.	19.07.8702/...
Полн. и дата	19.07.8702/...
Эзот. инв. №	...
Инв. № дубл.	...
Полн. и дата	...

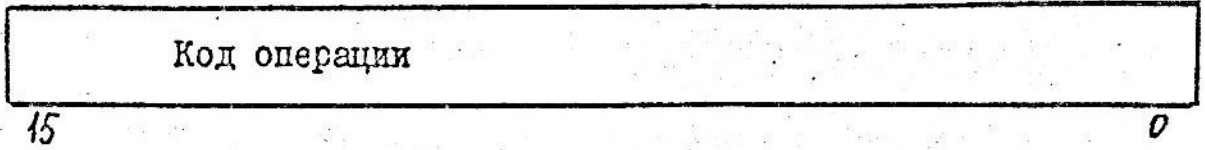
Инв. № подл.	19.07.8702/...	Полн. и дата	19.07.8702/...	Эзот. инв. №	...	Инв. № дубл.	...
--------------	----------------	--------------	----------------	--------------	-----	--------------	-----

ШИЗ.480.167 TO

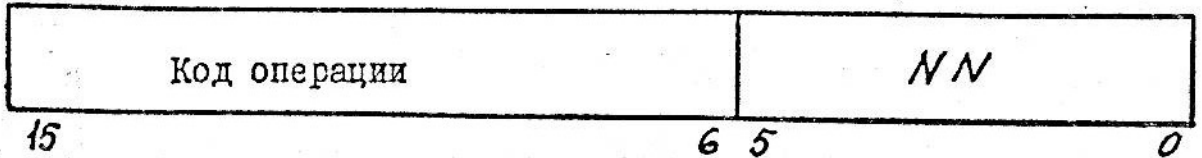
3) команда RTS



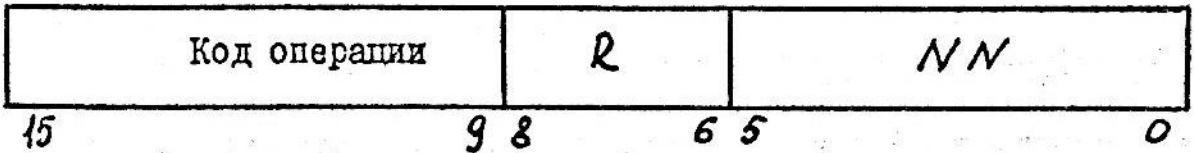
4) команды прерывания (IOT, EMT, TRAP, BPT)



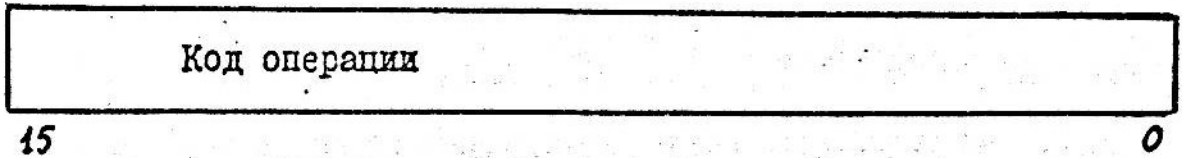
5) команда MARK



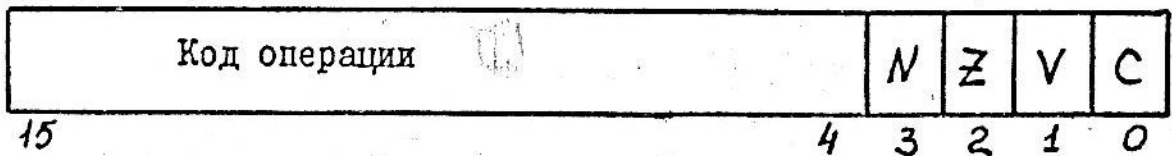
6) команда SOB



4. Специальные команды (HALT, WAIT, RESET, RTI, RTT, NOP)



5. Команды изменения признаков



Инв. № подл.	Подп. и дата
90394	10.07.87 Вил
Инв. № докум.	Подп. и дата
ЭЗСМ. Инв. №	Подп. и дата

Изм	Лист	№ докум.	Подп.	Дата

ИИЗ.480.167 TO

8.4.2. Выполнение байтовых команд

Большинство команд ПРЦ оперируют как с полными словами, так и с байтами. Байтовые команды с автоинкрементным или автодекрементным методами адресации изменяют содержимое указанного регистра на "1" для обращения к следующему байту. Байтовые команды при регистровом методе адресации производят обратку младшего байта выбранного регистра. Если старший разряд командного слова (разряд пятнадцать) установлен, он указывает, что команда байтовая. Если же в разряде пятнадцать командного слова записан "0", команда оперирует с полными словами.

Пример:

CLR	0050 DD	Очистка слова
CLRB	I050 DD	Очистка байта

8.4.3. Одноадресные команды

CLR	0050 DD
CLRB	I050 DD
Очистка	

0	1	0	0	0	1	0	1	0	0	0	d	d	d	d	d	d
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Действие: $(dst) \leftarrow 0$

Признаки:

- N - очищается
- Z - устанавливается
- V - очищается
- C - очищается

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата
90394	10.07.87			

Инв. № подл.	Лист	№ докум.	Подп.	Дата

ИИЗ.480.167 ТО

Лист

92

Описание: В указанную ячейку записываются нули. Для байтовой команды нули записываются в указанный байт.

Обнуление ячейки памяти происходит в цикле — запись.

Пример: CLR RI

До выполнения операции:

(RI) = I77777

NZVC

IIII

COM

COMB

После выполнения операции:

(RI) = 00000

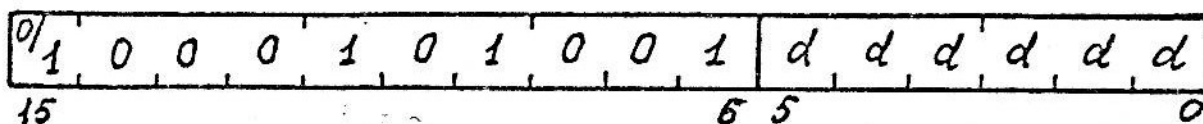
NZVC

OIOO

005IDD

IO5IDD

Инвертирование



Действие: $(dst) \leftarrow \overline{(dst)}$

Признаки: N — устанавливается, если результат < 0, в противном случае очищается;

Z — устанавливается, если результат = 0, в противном случае очищается;

V — очищается;

C — устанавливается.

Описание: Заменяет содержимое указанной ячейки его двоичным обратным кодом (каждый разряд, содержащий "0" устанавливается, а каждый разряд, содержащий "1", очищается). Для байтовой команды операция производится по отношению к указанному байту.

Пример: COM R0

Инв. № подл.	Подп. и дата
90394	10.07.87
Взам. инв. №	Инв. № докл.
Подп. и дата	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата					

ШИЗ.480.167 ТО

Лист

93

До операции:

(R0) = I3333

NZVC

OIIO

INC

INCB

После операции:

(R0) = I64444

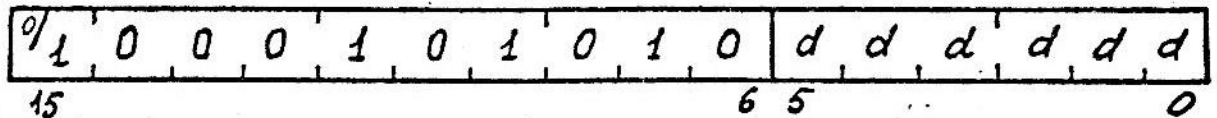
NZVC

I00I

0052 DD

I052 DD

Прибавление единицы



Действие: $(dst) \leftarrow (dst) + 1$

- Признаки:
- N - устанавливается, если результат < 0 , в противном случае очищается;
 - Z - устанавливается, если результат $= 0$, в противном случае очищается;
 - V - устанавливается, если операнд равен 077777, в противном случае очищается;
 - C - не изменяется.

Описание: Прибавляет единицу к содержимому указанной ячейки (или байту, если команда байтовая).

Пример: INC R2

До операции:

(R2) = 000333

NZVC

0000

После операции:

(R2) = 000334

NZVC

0000

ИМЗ.480.167 ТО

Лист

94

Инв. № подл.	90394	Подп. и дата	10.07.87	Изм. инв. №		Инв. № дубл.		Подп. и дата	
--------------	-------	--------------	----------	-------------	--	--------------	--	--------------	--

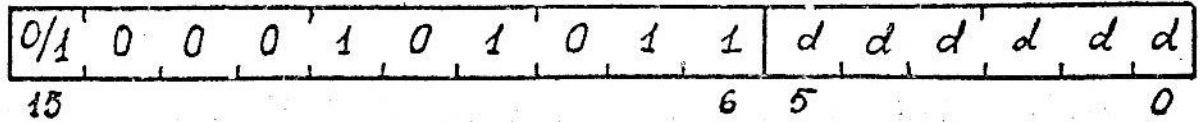
DEC

0053 DD

DECB

1053 DD

Вычитание единицы



Действие: $(dst) \leftarrow (dst) - 1$

- Признаки:
- N** - устанавливается, если результат < 0 , в противном случае очищается;
 - Z** - устанавливается, если результат $= 0$, в противном случае очищается;
 - V** - устанавливается, если операнд был равен 100000, в противном случае очищается;
 - C** - не изменяется.

Описание: Из содержимого указанной ячейки (или указанного байта для байтовых команд) вычитается единица.

Пример: DEC R5

До операции:
 (R5) = 000001

NZVC
 I000

После операции:
 (R5) = 000000

NZVC
 O100

Инв. № подл.	Подл. и дата
90394	10.07.87 Вч
Взам. инв. №	Инв. № дубл.
Подл. и дата	Подл. и дата

Изм.	Лист	№ докум.	Подл.	Дата

ИИЗ.480.167 ТО

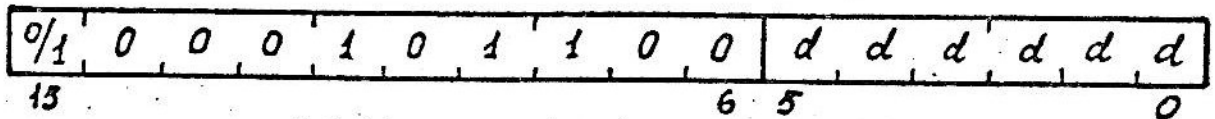
Лист
95

NEG
NEGB

0054 DD

I054 DD

Изменение знака



Действие: $(dst) \leftarrow - (dst)$

- Признаки:
- N** - устанавливается, если результат < 0 , в противном случае очищается;
 - Z** - устанавливается, если результат $= 0$, в противном случае очищается;
 - V** - устанавливается, если результат $= I00000$, в противном случае очищается;
 - C** - очищается, если результат $= 0$, в противном случае устанавливается.

Описание: Содержимое указанной ячейки (или байта для байтовых команд) заменяется двоичным дополнением операнда. Следует заметить, что число I00000 заменяется самим собой, так как не существует соответствующего ему положительного числа.

Пример: **NEGRO**

До операции:

(R0) = 000010

NZVC

0000

TST

TSTB

После операции

(R0) = I77770

NZVC

I001

0057 DD

I057 DD

Инв. № подл.	Подл. и дата
Инв. № докум.	Подл. и дата
Взам. инв. №	Подл. и дата
Инв. № подл.	Подл. и дата

ИИЗ.480.167 TO

Лист
96

мация, сдвинутая за пределы С-разряда, теряется.

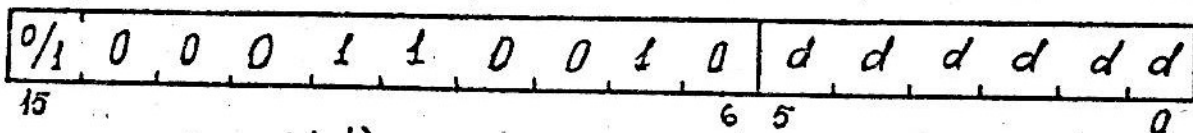
ASR

0062DD

ASRB

1062DD

Арифметический сдвиг вправо



Действие: $(dst) \leftarrow$ сдвинутое на одну позицию вправо (dst)

Признаки: N - устанавливается, если старший разряд результата установлен, в противном случае очищается;

Z - устанавливается, если результат = 0, в противном случае очищается;

V - устанавливается результатом операции: "исключающее ИЛИ" над содержимым N и C разрядов, которые они имеют после операции сдвига;

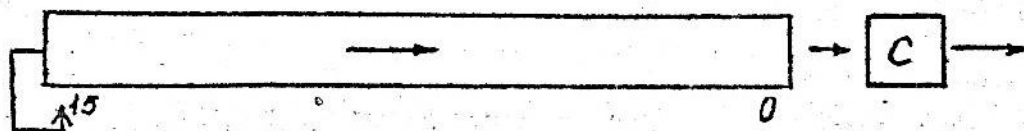
C - загружается содержимым младшего разряда указанной ячейки.

Описание: Сдвигаются все разряды операнда вправо на одну позицию. Содержимое знакового разряда восстанавливается. C - разряд загружается содержимым младшего разряда операнда.

Таким образом, ASR или ASRB выполняет деление числа со знаком на два.

Пример:

Сдвиг полного слова



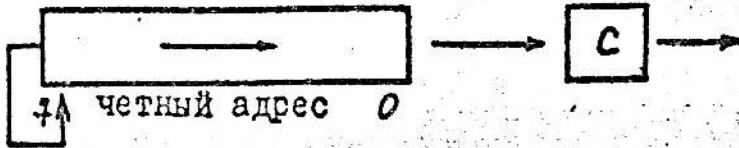
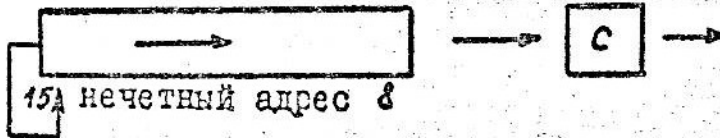
ЛИН. № подл. Подп. и дата. 90394 10.07.870В.

Лист: 98
 Изд. Лист: 98
 Подп. Дата

ИИЗ.480.167 ТО

Лист
98

Сдвиг байта



ASL

0063 DD

ASLB

1063 DD

Арифметический сдвиг влево



Действие: $(dst) \leftarrow$ сдвинутое на одну позицию влево (dst)

Признаки: **N** - устанавливается, если результат < 0 ,
в противном случае очищается;

Z - устанавливается, если результат $= 0$,
в противном случае очищается;

V - загружается результатом операции (значения **N** и **C** после операции сдвига) **N** \wedge **C**

C - загружается содержимым старшего разряда опе-
ранда.

Инв. № подл.	90394	Повл. и дата	10.07.87обм	Взам. инв. №		Инв. № дубл.		Повл. и дата	
--------------	-------	--------------	-------------	--------------	--	--------------	--	--------------	--

Инв. № подл.	Лист	№ докум.	Повл.	Дата
--------------	------	----------	-------	------

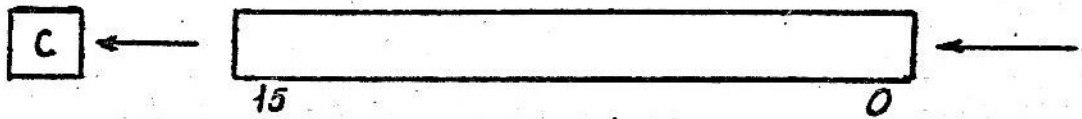
ШИЗ.480.167 ТО

Лист
99

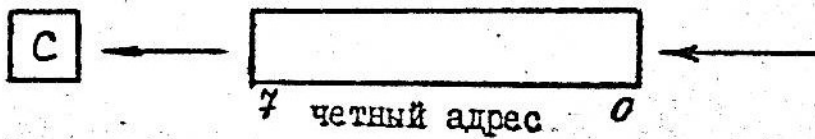
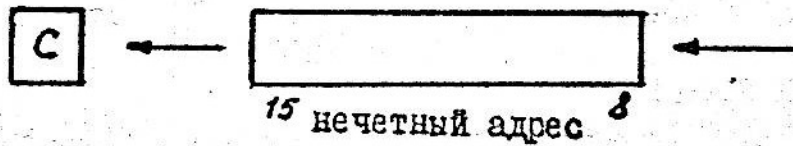
Описание: Сдвигаются все разряды операнда на одну позицию влево. В младший разряд операнда записывается нуль. С - разряд загружается содержимым старшего разряда операнда. Таким образом, A SL или A SLB выполняет умножение числа со знаком на 2.

Пример:

Сдвиг полного слова



Сдвиг байта



Для облегчения последовательной проверки и поразрядной обработки операнда используются команды циклического сдвига. Они оперируют со словом операнда и С -разрядом так, как если бы они образовали семнадцатиразрядный регистр с циклическим переносом.

R O R

R O R B

0060 DD

1060 DD

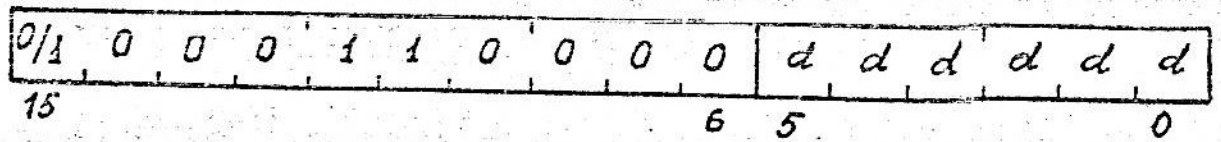
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № докум.	Подп. и дата
90394	10.07.87			

ЩИЗ.480.167 ТО

Лист

100

Циклический сдвиг вправо



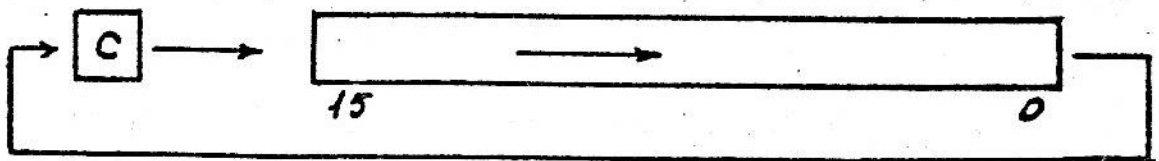
Действие: $(dst) \leftarrow$ циклически сдвинутое на одну позицию вправо (dst)

- Признаки: N - устанавливается, если результат < 0 ,
 в противном случае очищается;
 Z - устанавливается, если результат $= 0$,
 в противном случае очищается;
 V - загружается результатом операции $N \neq C$
 (значения N и C после окончания операции
 сдвига);
 C - загружается младшим разрядом операнда.

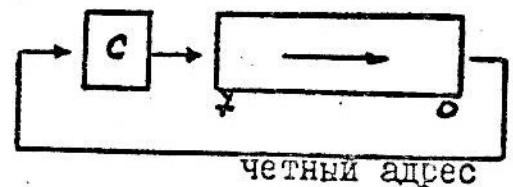
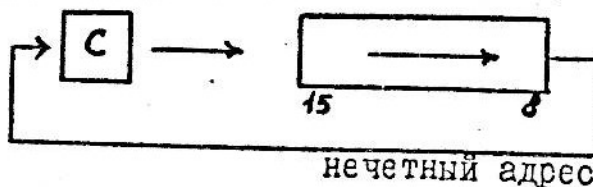
Описание: Циклически сдвигает все разряды операнда на одну позицию вправо. Содержимое младшего разряда загружается в C -разряд, а прежнее содержимое C -разряда загружается в старший разряд операнда.

Пример:

Сдвиг полного слова



Сдвиг байта

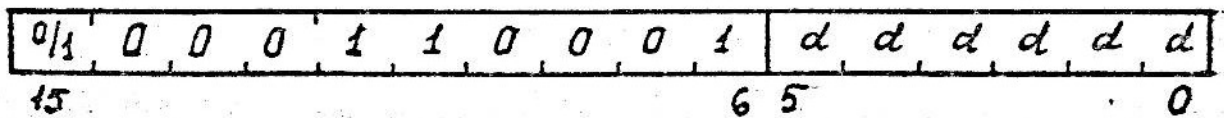


Инв. № подл.	Подл. и дата	Взам. инв. №	Инв. № дубл.	Подл. и дата
90394	10.07.87			

ИИЗ.480.167 ТО

Лист
101

Циклический сдвиг влево



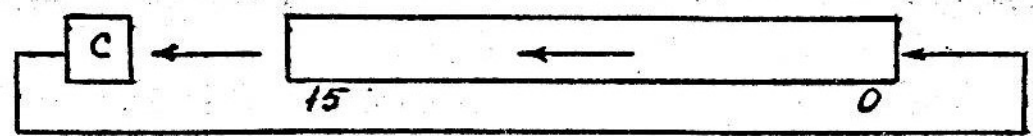
Действие: (dst) - циклически сдвинутое на одну позицию влево
 (dst)

- Признаки: N - устанавливается, если результат < 0 ,
 в противном случае очищается;
- Z - устанавливается, если результат $= 0$,
 в противном случае очищается;
- V - загружается результатом операции $N \vee C$
 (значение N и C после окончания операции сдвига);
- C - загружается старшим разрядом операнда.

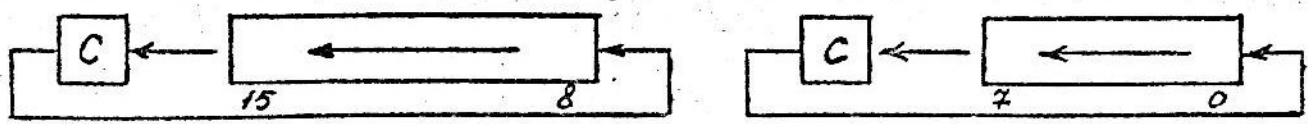
Описание: Циклически сдвигаются все разряды операнда на одну позицию влево. Содержимое старшего разряда загружается в C -разряд, а прежнее содержимое C -разряда загружается в младший разряд операнда.

Пример:

Сдвиг полного слова



Сдвиг байта



При вычислениях с повышенной точностью иногда необходи-

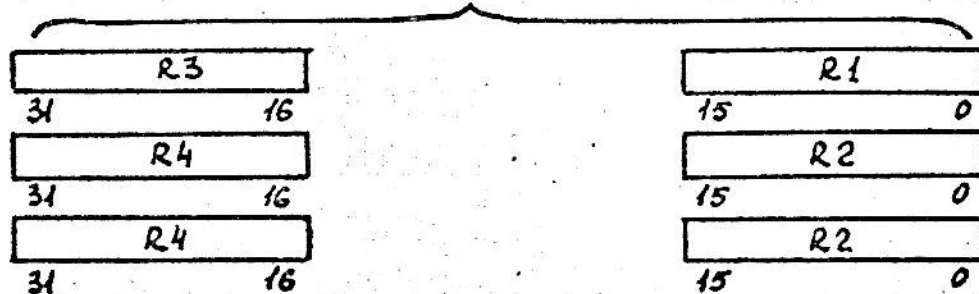
Инв. № подл.	90394
Подп. и дата	10.07.87 Овч.
Взам. инв. №	
Инв. № док.	
Подп. и дата	

Инв. № подл.	№ докум.	Подп.	Дата

ШИЗ.480.167 Т0

мо выполнять арифметические операции над операндами длиной в несколько слов. Для реализации арифметических операций с удвоенной точностью применяются команды прибавления переноса (ADC) и вычитания переноса (SBC). Например, два шестнадцатирядных слова могут быть объединены в тридцатидвухразрядное слово и над ним могут быть выполнены операции сложения или вычитания, как показано ниже:

32-разрядное слово



Пример: Сложение $-I$ с $-I$ может быть выполнено следующим образом:

$$-I = 37777\ 777\ 777$$

$$(R1), (R2), (R3), (R4) = I77777$$

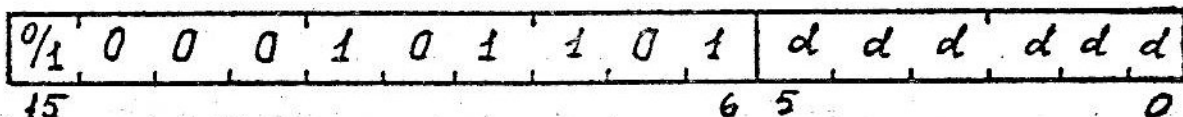
ADD R1, R2
 ADC R3
 ADD R3 R4

1. После сложения (R1) с (R2), в C-разряд загружается I.
2. Команда ADC прибавляет C-разряд к (R3), после чего (R3) = 000000.
3. Складываются (R3) и (R4).
4. Результат = 37777777776 = -2

ADC 0055DD
 ADCB I055DD

Инв. № подл.	Подп. и дата
90394	10.07.87
Взам. инв. №	Инв. № докум.
Подп. и дата	Подп. и дата

Прибавление переноса.



Действие: $(dst) \leftarrow (dst) + (C)$

Признаки: **N** - устанавливается, если результат < 0 ,
в противном случае очищается;

Z - устанавливается, если результат = 0,
в противном случае очищается;

V - устанавливается, если перед выполнением операции $(dst) = 077777$, а $(C) = 1$
в противном случае очищается;

C - устанавливается, если перед выполнением операции $(dst) = 177777$, а $(C) = 1$,
в противном случае очищается.

Описание: Операнд складывается с содержимым C-разряда.

Пример: ADC R3

До операции:

(R3) = 077777

NZVC

0001

SBC

SBCB

После операции:

(R3) = 100000

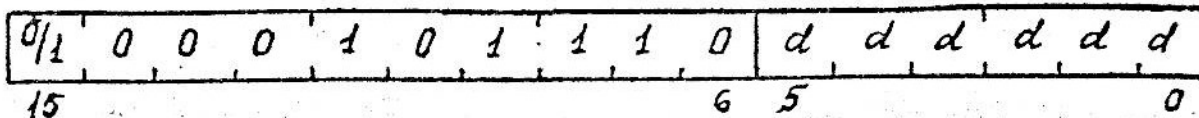
NZVC

1010

0056DD

1056DD

Вычитание переноса



Инв. № подл.	90394
Побл. и дата	10.07.8702
Взам. инв. №	
Инв. № дубл.	
Побл. и дата	

ИИЗ.480.167 Т0

Изм. Иуст. № докум. Подп. Дата

Иуст

104

Действие: $(dst) \leftarrow (dst) - (C)$

Признаки: N - устанавливается, если результат < 0 ,
в противном случае очищается;

Z - устанавливается, если результат $= 0$,
в противном случае очищается;

V - устанавливается, если перед выполнением
команды $(dst) = 100000$ и $(C) = 1$, в
противном случае очищается;

C - устанавливается, если $(dst) = 0$ и
 $(C) = 1$, в противном случае очищается.

Описание: Вычитается содержимое C -разряда из операнда.

Это позволяет вычесть перенос, получившийся при
вычитании двух младших шестнадцатиразрядных слов,
из результата вычитания двух старших шестнадцати-
разрядных слов.

Пример: Вычитание с удвоенной точностью осуществляется сле-
дующей последовательностью команд:

SUB, A0, B0

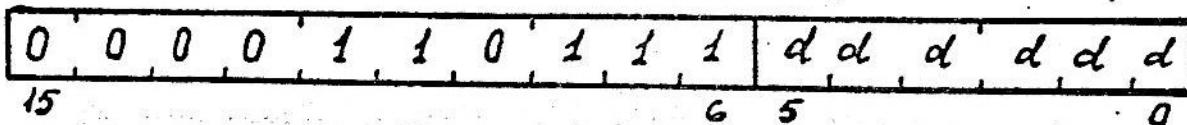
SBC BI

SUB A1, B1

SXT

0067DD

Расширение знака



Действие: $(dst) \leftarrow 0$, если N очищен,
 $(dst) \leftarrow -1$, если N установлен.

Инв. № подл.	90394
Подп. и дата	10.07.87 В.ч.
Узм. инв. №	
Инв. № д.б.	
Подп. и дата	

V - очищается;

C - очищается.

Описание: Меняет местами старший байт с младшим байтом указанной ячейки. Адресация происходит по полному слову.

Пример: SWABR1

До операции:

(RI) = 077777

N ZVC

IIII

После операции:

(RI) = 177577

N ZVC

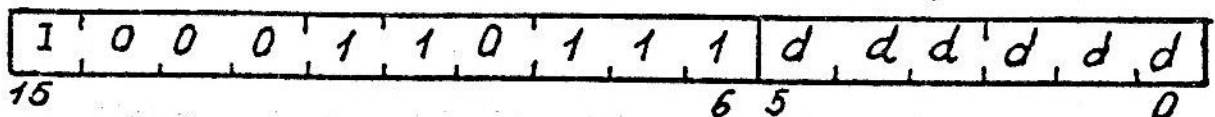
0000

Слово состояния процессора (PSW) хранится в регистре, который не имеет адреса. Возможность программного доступа к PSW обеспечивает команды чтения и записи PSW

MFP S

I067 DD

Чтение PSW



Действие: (dst) ← PSN

Признаки: N - устанавливается, если разряд семь слова состояния процессора равен 1, в противном случае очищается;

Z - устанавливается, если все восемь разрядов PSW (разряды 0-7) = 0, в противном случае очищается;

V - очищается;

C - не изменяется.

Инв. № подл.	90394
Полн. и дата	10.07.870Вч.-
Взам. инв. №	
Инв. № д.б.л.	
Полн. и дата	

ШИЗ.480.167 ТО

Лист

104

Пересылка значения *PSW* в память происходит в цикле "ЗАПИСЬ".

Описание: Восемь разрядов слова состояния процессора *PSW* пересылаются в указанную ячейку.

Если при этом используется регистровый метод адресации, в указанном регистре происходит расширение знака. Адрес операнда приемника воспринимается как адрес байта.

Пример: *MFPS RO*

До операции:

(*RO*) = 000000

PS = 000014

MFPS

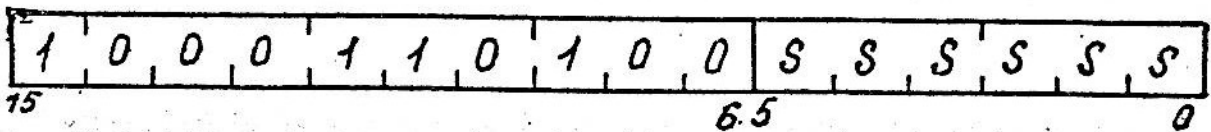
После операции:

(*RO*) = 000014

PS = 000000

I064SS

Запись *PSW*



Действие: $PSW \leftarrow (SZC)$

Признаки: Устанавливаются или очищаются в соответствии с разрядами 0-3 операнда источника (*SZC*).

Описание: Восемь разрядов указанного операнда замещают содержимое слова состояния процессора. Адрес операнда источника воспринимается как адрес байта. Следует заметить, что 7-разряд (разряд 4 *PSW*) не может быть установлен этой командой. Операнд источника (*SZC*) не изменяет своего содержимого. Эта команда может быть использована для изменения приоритетных разрядов *PSW* (разряды 5-7).

Инв. № подл.	Лист 4 дата	Взам. инв. №	Инв. № докл.	Лист 1 дата
90394	12.07.87 08.ч.			

ШИЗ.480.167 Т0

Лист

103

среди байтовых команд) расширяет старший разряд младшего байта (расширение знака). Все разряды старшего байта устанавливаются или сбрасываются в зависимости от того, установлен или сброшен старший (знаковый) разряд младшего байта. В других случаях MOV В оперирует с байтами так, как MOV со словами.

Пример: MOV X, R1 загружает R1 содержимым ячейки памяти
MOV #20, R0 загружает число 20 в регистр R0.

MOV @#20, -(R6) записывает операнд, содержащийся в ячейке с адресом 20, в стек.

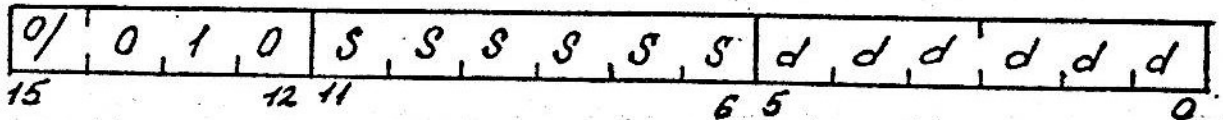
MOV (R6) +, @# I77566 выбирает операнд из стека и записывает его в ячейку I77566.

MOV R1, R3 выполняет межрегистровую передачу.

CMR 02 55DD

CMFB 12 55DD

Сравнение



Действие: (src) - (dst)

Признаки: N - устанавливается, если результат < 0, в противном случае очищается;

Z - устанавливается, если результат = 0, в противном случае очищается;

V - устанавливается, если было арифметическое переполнение. Это может произойти в случае, когда операнды были противоположно-

Подп. и дата	
Инв. № докум.	
ЭЗ см. инв. №	
Подп. и дата	10.07.87
Инв. № подл.	90394

ШИЗ.480.167 ТО

Лист

110

го знака, а знак результата совпадает со знаком операнда приемника (*dst*), в противном случае очищается;

C - очищается, если был перенос из старшего разряда результата, в противном случае устанавливается.

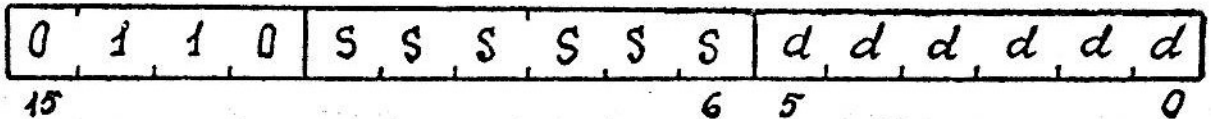
Описание: Сравнивает операнды источника и приемника и изменяет признаки, которые затем могут быть использованы для команд условных переходов. Оба операнда не изменяются. Единственным действием является изменение признаков. За командой сравнения обычно следует команда условного ветвления. Заметим, что в отличие от команды вычитания, порядок действия следующий:

$(src) - (dst)$, а не $(dst) - (src)$

ADD

06SSDD

Сложение



Действие: $(dst) \leftarrow (src) + (dst)$

Признаки: **N** - устанавливается, если результат < 0 , в противном случае очищается;

Z - устанавливается, если результат $= 0$, в противном случае очищается;

V - устанавливается, если в результате операции произошло арифметическое переполнение. Это происходит, когда операнды были одного

Инв. № подл.	90394	Полн. и дата	10.07.87	Инв. № докум.		Полн. и дата		Инв. № докум.	
--------------	-------	--------------	----------	---------------	--	--------------	--	---------------	--

знака, а результат получился противоположного знака, в противном случае очищается.

Описание: Операнд источника (*src*) складывается с операндом приемника (*dst*) и результат записывается по адресу операнда приемника. Первоначальное содержимое теряется. Содержимое *pc* не изменяется. Сложение выполняется в двоичном дополнительном коде.

Пример: ADD R1, R2

До операции:

(R1) 000020

(R2) 000060

После операции:

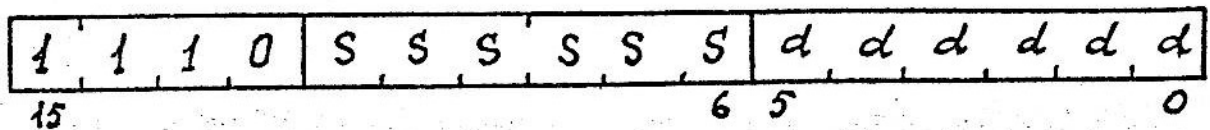
(R1) 000020

(R2) 000100

SUB

16 SSDD

Вычитание:



Действие: $(dst) \leftarrow (dst) - (src)$

Признаки: *N* - устанавливается, если результат < 0 , в противном случае очищается;

Z - устанавливается, если результат = 0, в противном случае очищается;

V - устанавливается, если в результате операции произошло арифметическое переполнение. Это происходит, когда операнды были противоположного знака, а результат получен одного знака с операндом источника, в противном случае очищается;

Инв. № подл.	90394	Подл. и дата	10.07.87	Взам. инв. №		Инв. № докл.		Подл. и дата	
--------------	-------	--------------	----------	--------------	--	--------------	--	--------------	--

Изм.	Лист	№ докум.	Подп.	Дата

ИЗ.480.167 ТО

Лист
112

C - очищается, если был перенос из старшего разряда результата, в противном случае устанавливается.

Описание: Из операнда приемника вычитается операнд источника (*src*) и результат записывается по адресу *dst*. Первоначальное содержимое теряется, а содержимое *src* остается без изменения. При арифметических операциях с удвоенной точностью установка C-разряда означает заем единицы из старшей части вычитаемого.

Пример: **SUB R1, R2**

До операции:

(R1) = 011111

(R2) = 012345

NZVC

1111

После операции:

(R1) = 011111

(R2) = 001234

NZVC

0000

Логические команды

Из четырех логических команд три имеют такой же формат, как и двухадресные арифметические команды. Четвертая команда имеет специфический формат. Логические команды позволяют осуществлять поразрядную обработку данных.

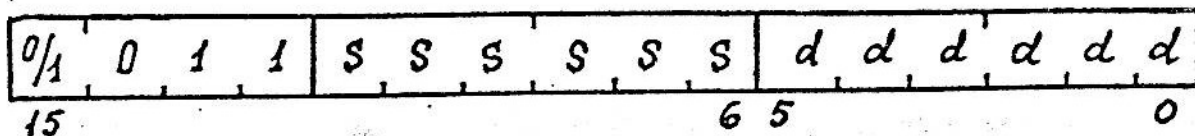
BIT

03 S SDD

BITB

13 S SDD

Проверка разрядов



Действие: $(src) \wedge (dst)$

Признаки: **Z** - устанавливается, если все разряды результата = 0, в противном случае очищается;

Инв. № подл.	90394
Побл. и дата	10.07.87
Взам. инв. №	
Инв. № докл.	
Побл. и дата	

Признаки: *N* - устанавливается, если старший разряд результата установлен, в противном случае очищается;
V - очищается;
C - не изменяется.

Описание: Выполняет логическую функцию "И" над (*src*) и (*dst*), изменяя соответствующим образом признаки. Оба операнда не изменяют своего значения. Команда BIT используется для проверки состояния разрядов операнда (*src*), для которых установлены соответствующие разряды в операнде (*dst*).

Пример: BIT 30, R3

До операции:

(R3) = 0000000000011000

После операции:

(R3) = 0000000000011000

NZVC
 IIII

NZVC
 000I

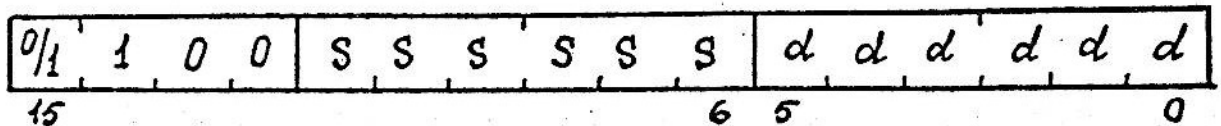
ВІС

04SSDD

ВІСВ

14SSDD

Очистка разрядов



Действие: $(dst) \leftarrow (\overline{src}) \wedge (dst)$

Признаки: *N* - устанавливается, если старший разряд результата установлен, в противном случае очищается;
V - очищается;
C - не изменяется.
Z - устанавливается, если все разряды результата очищены, в противном случае очищается;

ШИЗ.480.167 ТО

Лист

114

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата
90394	12.07.870Вш			

Описание: Очищает каждый разряд операнда (*dst*), соответствующий установленному разряду операнда (*src*).

Первоначальное содержимое *dst* теряется. Содержимое *src* не изменяется.

Пример: BIC R3, R4

До операции:

(R3) = 001234

(R4) = 001111

NZVC

1111

BIS

BISB

После операции:

(R3) = 001234

(R4) = 000101

NZVC

0001

05SSDD

15SSDD

Логическое сложение

0/1	1	0	1	S	S	S	S	S	S	d	d	d	d	d	d
15										6	5				0

Действие: $(dst) \leftarrow (src) \vee (dst)$

Признаки: **N** - устанавливается, если старший разряд результата установлен, в противном случае очищается;

Z - устанавливается, если все разряды результата очищены, в противном случае очищается;

V - очищается;

C - не изменяется.

Описание: Выполняет логическую операцию "ИЛИ" над содержимым *src* и *dst* записывает результат по адресу *dst*. Разряды (*dst*) устанавливаются в "1", если соответствующие им разряды

Инв. № подл.	90394
Подп. и дата	10.07.87 В.В.
Взам. инв. №	
Инв. № докум.	
Подп. и дата	

ШИЗ.480.167 ТО

Лист

115

(*zc*) находятся в "I". Препрежнее содержимое *dst* те-
 ряется, а содержимое *zc* остается неизменным.

Пример: **BIS R0, R1**

До операции:

(R0) = 001234

(R1) = 001111

NZVC

0000

После операции:

(R0) = 001234

(R1) = 001335

NZVC

0000

До операции: (R0) = 0000001010011100

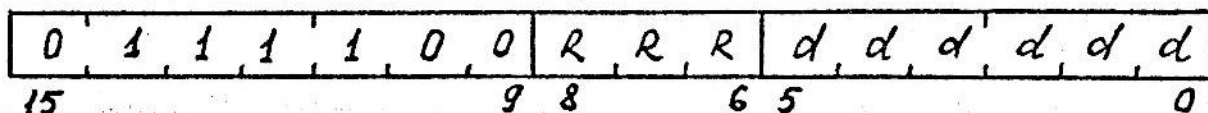
(R1) = 0000001001001001

После операции: (R1) = 0000001011011101

XOR

074 RDD

Исключающее ИЛИ



Действие: $(dst) \leftarrow R \vee (dst)$

Признаки: *N* - устанавливается, если результат < 0 ,
 в противном случае очищается;

Z - устанавливается, если результат = 0,
 в противном случае очищается;

V - очищается;

C - не изменяется.

Описание: Выполняет операцию "исключающее ИЛИ" над содер-
 жимым указанного регистра и содержимым *dst*
 Результат записывается в *dst*. Содержимое
 регистра *R* не изменяется.

Инв. № подл.	Подп. и дата
53 см. инв. №	Инв. № докл.
Подп. и дата	Подп. и дата
90394	12.07.87

Изм.	Лист	№ докум.	Подп.
			Дата

ИИЗ.480.167 ТО

Лист

116

Пример: XOR, R0, R2

До операции:

(R0) = 001234

(R2) = 001111

NZVC

1111

После операции:

(R0) = 001234

(R2) = 000325

NZVC

0001

До операции: (R0) = 0000001010011100

(R2) = 0000001001001001

После операции: (R2) = 0000000011010101

8.4.5. Команды управления программой

К командам управления программой относятся команды ветвлений, перехода к подпрограмме, возврата из подпрограммы, безусловного перехода и др.

Команды ветвления.

Эти команды вызывают ветвления по адресу, являющемуся суммой смещения (умноженного на 2) и текущего содержимого если условие ветвления выполняется.

Смещение показывает, на сколько ячеек нужно перейти относительно текущего содержимого РС в ту или другую сторону. Так как слова имеют четные адреса, то для получения истинного исполнительного адреса смещение необходимо умножить на два перед прибавлением к РС, который всегда указывает на слово. Старший разряд смещения (разряд семь) является знаковым разрядом. Если он установлен, смещение отрицательное, ветвление происходит в сторону уменьшения адреса (в обратном направлении). Если в седьмом разряде содержится 0, смещение положительное, и ветвление происходит в сторону увеличения адресов (прямом направлении).

Инв. № подл.	Подп. и дата
90394	10.07.80
Взам. инв. №	Инв. № док. №
Подп. и дата	Подп. и дата

ШИЗ.480.167 ТО

Лист

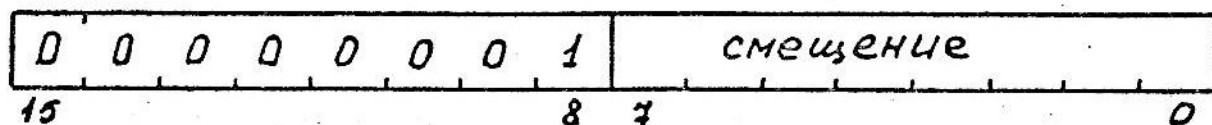
117

Восьмиразрядное смещение позволяет производить ветвление в обратном направлении максимально на 200_8 слов от слова, на которое указывает текущее содержимое РС и на 177_8 слов в прямом направлении.

BR

000400 + XXX

Ветвление безусловное



Действие: $(PC) \leftarrow (PC) + 2 \text{ XXX}$

Признаки: не изменяются.

Описание: обеспечивает способ передачи управления в программе ячейки, адрес которой находится в ограниченной области, с помощью одного слова команды.

Новое содержимое РС = текущее содержимое РС + 2 X (смещение), где текущее содержимое РС = адрес команды ветвления + 2.

Пример:

Адрес	Код команды	Смещение
500	000402	002
502		

В данном примере управление передается ячейке с адресом $PC = 502 + 2 \times 2 = 506$.

Простые условные ветвления

BVE

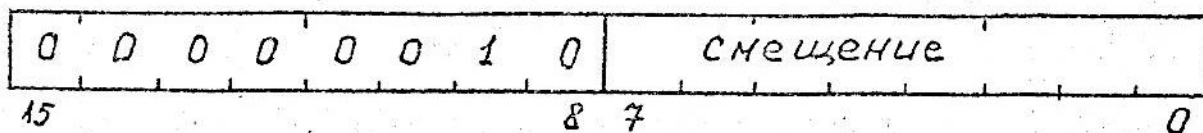
001000 + XXX

Инв. № подл.	46394
Полн. и дата	10.07.87
Взам. инв. №	
Инв. № докум.	
Полн. и дата	

ИИЗ.480.167 Т0

Лист

Ветвление, если не равно (нулю)



Действие: $(PC) \leftarrow (PC) + 2XXX$, если $Z = 0$

Признаки: не изменяются

Описание: Проверяет состояние разряда Z и вызывает ветвление, если он очищен. ВМЕ обратная по действию BEQ . Вместе с командой CMR она используется для проверки того, что установленные разряды операнда источника соответствуют установленным разрядам операнда приемника. В общем случае она используется для проверки неравенства нулю результата предыдущей операции.

Пример: $CMR A, B$ сравниваются A и B

$ВМЕ$ переход, если A не равно B .

Эта последовательность команд будет давать переход на

C , если $A \neq B$, а последовательность:

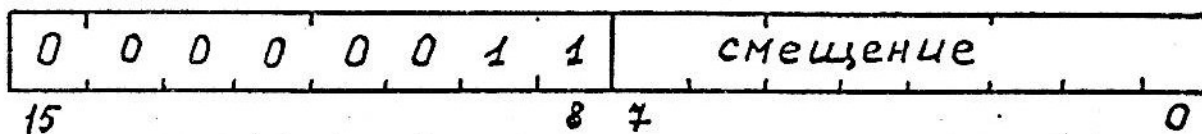
$A \gg B$, B складываются A и B

$ВМЕ C$ переход, если $A + B \neq 0$ будет давать переход, если $A + B \neq 0$.

BEQ

$001400 + XXX$

Ветвление, если равно (нулю)



Действие: $(PC) \leftarrow (PC) + 2XXX$, если $Z = 1$.

Инв. № повл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата
90394	10.07.870Вн.			

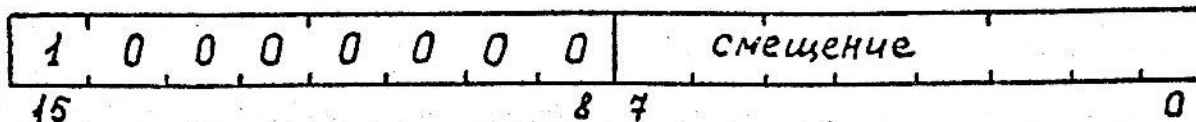
Признаки: не изменяются

Описание: Проверяет состояние разряда Z и вызывает ветвление, если он установлен. Вместе с командой CMP она используется для проверки равенства двух величин. Вместе с командой BIT используется для проверки того, что очищенные разряды операнда источника соответствуют установленным разрядам операнда приемника. В общем случае эта команда используется для проверки равенства нулю результата предыдущей операции.

Пример: CMP A, B сравнивает A и B
BEQ C переход, если они равны
Если $A - B = 0$, то управление передается ячейке C.
ADD A, B складываются A и B
BEQ C переход, если $A + B = 0$
Управление передается ячейке C, если $A + B = 0$

BR4 100000 + XXX

Ветвление, если плюс



Действие: (PC) (PC) + 2XXX, если $N = 0$

Признаки: не изменяются

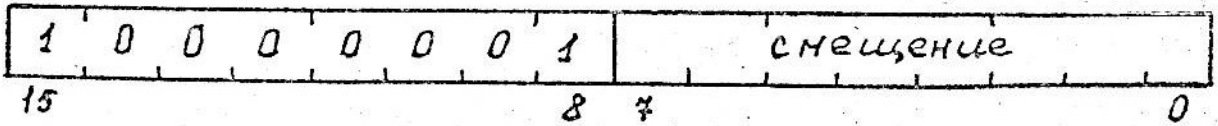
Описание: Проверяет разряд N и вызывает ветвление, если он очищен, BR4 обратна по действию команде BMI.

BMI 100400 + XXX

Инв. № подл.	Подп. и дата	Инв. № док.	Подп. и дата	Взят инв. №
90394	19.07.87			

Инв. № подл.	Лист	№ докум.	Подп.	Дата	ИИЗ.480.167 Т0	Лист
						120

Ветвление, если минус



Действие: $(PC) \leftarrow (PC) + 2XXX$, если $N = 1$

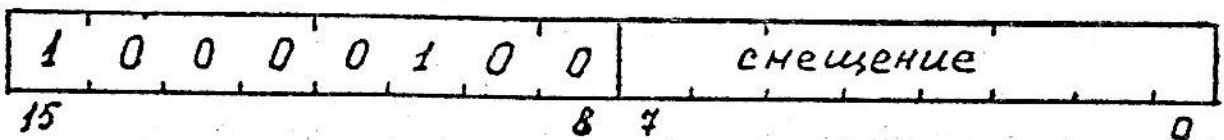
Признаки: не изменяются.

Описание: Проверяет состояние N -разряда и вызывает ветвление, если он установлен. Она используется для проверки знака (старший разряд) результата предыдущей операции.

BVC

I02000 + XXX

Ветвление, если нет арифметического переполнения



Действие: $(PC) \leftarrow (PC) + 2XXX$, если $V = 0$

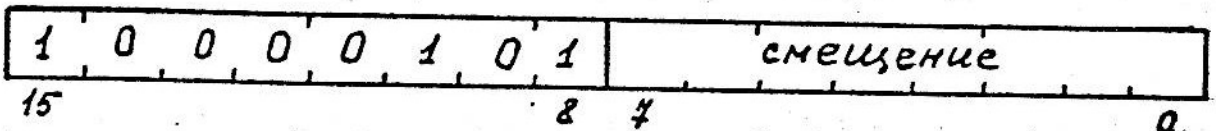
Признаки: не изменяются

Описание: Проверяет состояние разряда V и вызывает ветвление, если он очищен.

BVS

I02400 + XXX

Ветвление, если арифметическое переполнение



Действие: $(PC) \leftarrow (PC) + 2XXX$, если $V = 1$.

Инв. № подл.	Подп. и дата	Инв. № докл.	Подп. и дата
90394	10.07.87	10.07.87	Взм.

ИИЗ.480.167 ТО

Лист
121

КОПИЯ

Признаки: не изменяются.

Описание: Проверяет состояние разряда V и вызывает ветвление, если он установлен.

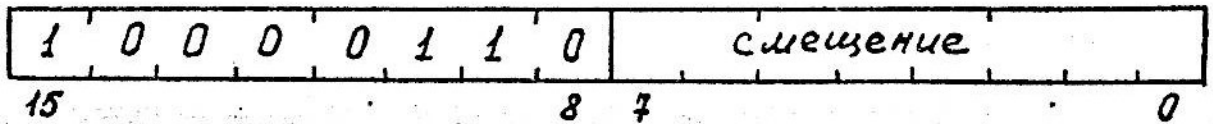
BVS используется для обнаружения арифметического переполнения в результате исполнения предыдущей операции.

BVS обратна по действию команде BVC.

BCC

I03000 + XXX

Ветвление, если нет переноса



Действие: $(PC) \leftarrow (PC) + 2XXX$, если $= 0$

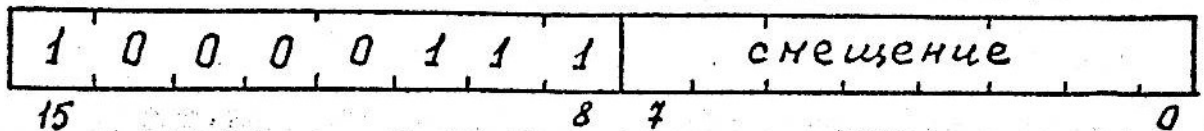
Признаки: не изменяются.

Описание: Проверяет состояние разряда C и вызывает ветвление, если он очищен.

BCS

I03400 + XXX

Ветвление, если перенос



Действие: $(PC) \leftarrow (PC) + 2XXX$, если $C = 1$

Признаки: не изменяются.

Описание: Проверяет разряд C и вызывает ветвление, если он установлен, BCS используется для проверки наличия переноса в результате предыдущей операции.

Инв. № подл.	90394
Полн. и дата	10.07.870Вш-
Взам. инв. №	
Инв. № докум.	
Подп. и дата	

ИИЗ.480.167 Т0

Лист 122

BCS обратна по действию команде BCS.

Условные ветвления по результату операций над числами.

Особые комбинации разрядов признаков проверяются с помощью команд условного ветвления по результату операций над числами. Эти команды используются для проверки результатов команд, в которых операнды рассматриваются как двоичные числа, имеющие знак.

Заметим, что сравнение чисел, имеющих знак, отличается от сравнения чисел без знака тем, что в арифметике, использующей дополнительные коды, последовательность чисел следующая:

наибольшее 077777

077776

.....

.....

положительные

000001

000000

отрицательные

177777

.....

.....

100001

наименьшее

100000,

тогда как для шестнадцатиразрядных чисел, не имеющих знака, последовательность следующая:

наибольшее

177777

.....

.....

000002

000001

наименьшее

000000

Инв. № подл.	90394	Подп. и дата	
Взам. инв. №		Подп. и дата	
Инв. № докл.		Подп. и дата	
Узм.		Подп. и дата	
Лист	10.07.67	Подп. и дата	

ШИЗ.480.167 Т0

Лист

123

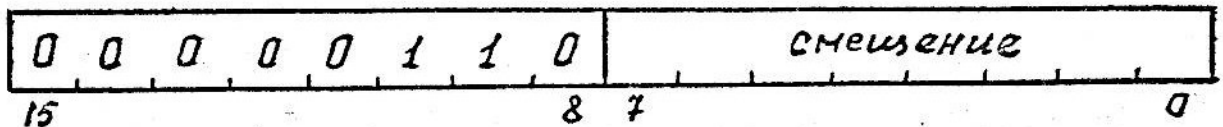
Таким образом, **BGT** будет всегда вызывать ветвление, если она следует за операцией сложения двух отрицательных чисел, даже если происходит переполнение. В частности, **BGT** будет всегда вызывать ветвление, если она следует за командой сравнения отрицательного операнда источника и положительного операнда приемника даже, если произошло переполнение.

BGT никогда не будет вызывать ветвления, если она следует за командой сравнения (**CMR**) положительного операнда источника и отрицательного операнда приемника. **BGT** не будет вызывать ветвления, если результат предыдущей операции равен 0 без переполнения.

BGT

003000 + XXX

Ветвление, если больше (нуля)



Действие: $(PC) \leftarrow (PC) + 2XXX, ZV(N \neq V) = 0$

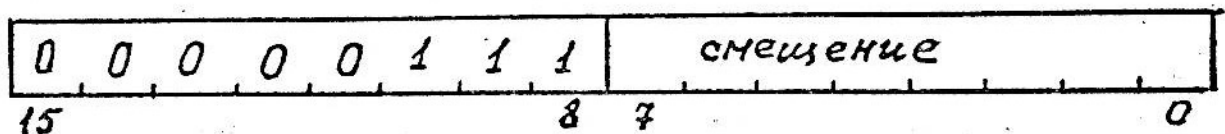
Признаки: не изменяются.

Описание: Команда **BGT** подобна команде **BGE**, за исключением того, что **BGT** не будет вызывать ветвления по нулевому результату.

BLE

003400 + XXX

Ветвление, если меньше или равно (нулю)



Инв. № подл.	Подп. и дата	Инв. № докл.	Взят инв. №	Подп. и дата
	90394			

Действие: $(PC) \leftarrow (PC) + 2XXX$, если $Z \vee (N \vee V) = 1$

Признаки: не изменяются.

Описание: Команда VLE подобна команде VLT , но дополнительно будет вызывать ветвление, если результат предыдущей операции был равен нулю.

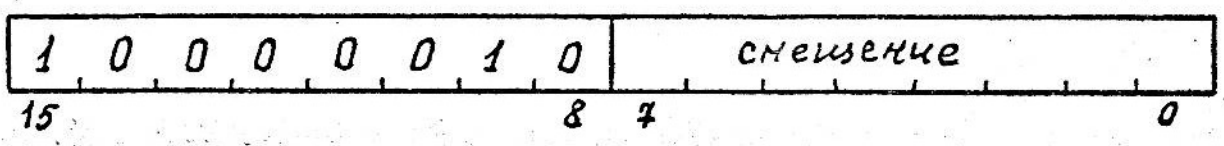
Условные ветвления по результату операции над кодами.

Условные ветвления по результату операций над кодами обеспечивают методы проверки результата операций сравнения операндов, рассматриваемых как величины без знака.

VNI

$101000+XXX$

Ветвление, если больше



Действие: $(PC) \leftarrow (PC) + 2XXX$, если $C \vee Z = 0$

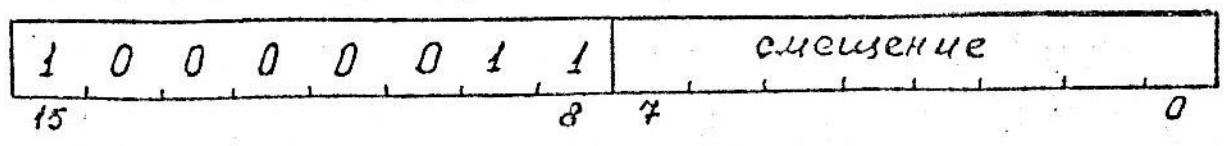
Признаки: не изменяются.

Описание: Вызывает ветвление, если предыдущая операция не вызвала переноса и появления нулевого результата. Это происходит при операциях сравнения (CMP), когда операнд источника больше операнда приемника.

$VLOS$

$101400 + XXX$

Ветвление, если меньше или равно



Инв. № подл.	90394
Дата	10.07.87
Инв. №	
Инв. №	
Подп. и дата	

ЦИЗ.480.167 ТО

Лист
426

Действие: $(PC) \leftarrow (PC) + 2XXX$, если $CVZ = 1$

Признаки: не изменяются.

Описание: Вызывает ветвление, если предыдущая операция вызывает перенос или появление нулевого результата. Команда **BLOS** является обратной по действию команде **BNI**.

Ветвление будет происходить, если операнд источника меньше или равен операнду приемника.

BNIS

I03000 + XXX

Ветвление, если больше или равно.

По своему действию команда **BNIS** идентична команде **BCC**. Эта мнемоника вводится только для удобства.

BLO

I03400 + XXX

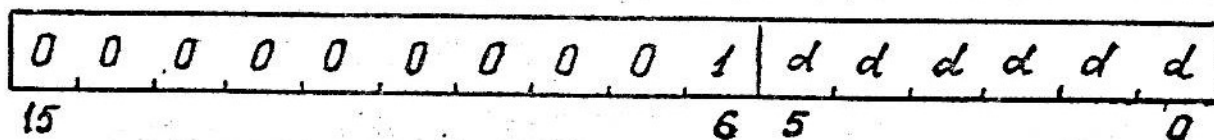
Ветвление, если меньше

По своему действию команда **BLO** идентична команде **BCL**. Эта мнемоника вводится только для удобства.

JMP

000IDD

Безусловный переход



Действие: $(PC) \leftarrow \text{адрес } dst$

Признаки: не изменяются.

Описание: Команда **JMP** обеспечивает возможность перехода программы на любую ячейку памяти с использованием всех методов адресации, за исключением регистрового. Использование регистровой адресации

Инв. № предл.	Подп. и дата	Взм. инв. №	Инв. № докум.	Подп. и дата
90394	19.07.87 Вш			

ИИЗ.480.167 TO

Лист

124

вызывает прерывание программы по условию "запрещенная команда" через адрес вектора IO. Метод косвенной адресации может применяться и вызывает передачу управления программой по адресу, содержащемуся в указанном регистре. Заметим, что команды - это полные слова и, поэтому должны выбираться из ячеек с четным адресом. Косвенно-индексный метод адресации позволяет командой γ MP передать управление по адресу, являющемуся элементом таблицы адресов.

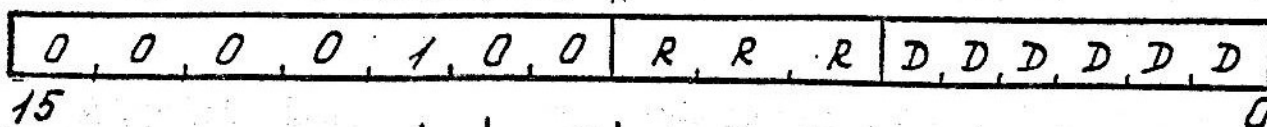
Команды обращения к подпрограмме и выхода из подпрограммы.

Эти команды обеспечивают возможность автоматического вложения программ, выход из подпрограмм, многократный вход в подпрограмму. Подпрограммы могут обращаться к другим подпрограммам (или к самим себе) без специального обеспечения хранения адресов возврата. Процедура обращения к подпрограмме и выхода из нее не изменяет подпрограмму. Это позволяет использовать одну и ту же подпрограмму несколькими процессами, осуществляющими прерывание программы.

γSR

004 RDD

Обращение к подпрограмме



15

0

Действие: $(SP) \leftarrow (R)$;

запись содержимого указанного регистра в стек;

$(R) \leftarrow (PC)$

счетчик команд содержит адрес ячейки, следующей за командой γSR , этот адрес заносится в регистр R.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата
90394	10.07.87 В.с.			

ИИЗ.480.167 ТО

Лист
128

(PC) ← адрес dst ; занесение в счетчик команд нового содержимого, определяющего начальный адрес подпрограммы.

Признаки: не изменяются.

Описание: При выполнении команды *JSR* старое содержимое указанного регистра ("указателя связи") автоматически засылается в стек, и новая связующая информация поступает в регистр. Таким образом, обращение к подпрограммам вложенным в подпрограммы на любую глубину, осуществляются с помощью регистра "указателя связи". Нет необходимости в том, чтобы задавать максимальную глубину обращения к данной подпрограмме или включать команды запоминания и восстановления "указателя связи" в каждую подпрограмму. Так как вся связующая информация сохраняется в стеке, выполнение программы может быть прервано и подпрограмма обслуживания прерывания может обращаться к той же самой прерванной подпрограмме. Выполнение подпрограммы может быть затем возобновлено по окончании обслуживания прерывания. Этот процесс, называемый вложением, может продолжиться до любого уровня.

Обращение к подпрограмме по команде *JSR* может осуществляться с помощью автоинкрементной адресации (если каждый последующий вход в подпрограмму осуществляется через ячейку, адрес которой на 2 больше предыдущего) или индексной адресации (если вход в подпрограмму осуществляется по адресам, расположенным в произвольном порядке), а также с помощью косвенных методов адресации.

Инв. № подл.	Подп. и дата	Узл. инв. №	Инв. № дубл.	Подп. и дата
90394	19.07.87 Вш			

Узм	Лист	№ докум.	Подп.	Дата

ИИЗ.480.167 ТО

Лист

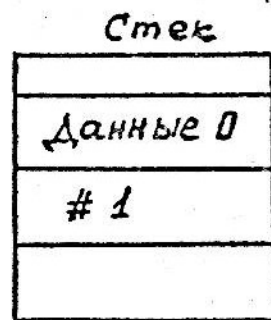
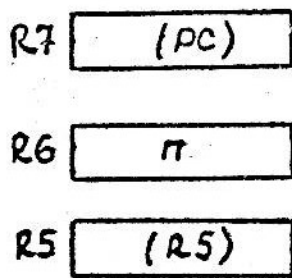
129

Действие: (PC) ← (R)
 (R) ← (SP) + 4

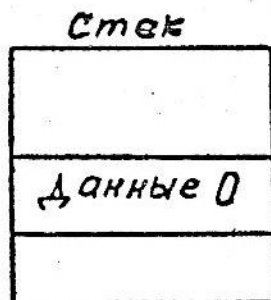
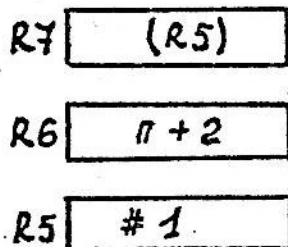
Описание: Загружает содержимое регистра (R) в PC, после чего извлекает верхний элемент стека и засылает его в указанный регистр. Возврат из подпрограммы обычно выполняется через тот же самый регистр, который используется при обращении к ней. Таким образом, выход из подпрограммы, обращение к которой осуществлялось командой *JSR PC, dst* выполняется командой *RTS PC*, а выход из подпрограммы, обращение к которой осуществлялось командой *JSR R5, dst* с использованием любого из методов адресации, выполняется командой *RTS R5*.

Пример: *RTS R5*

До операции:



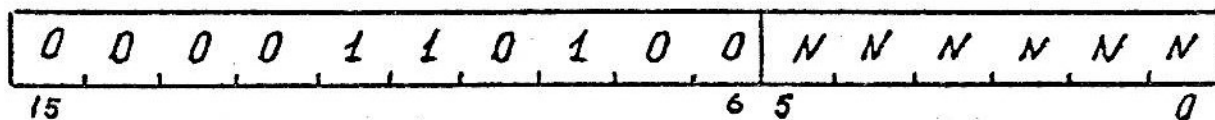
После операции:



MARK

0064 NN

Восстановление SP



ИИЗ.480.167 ТО

Лист
131

ИИЗ. № подл. 90394
 Дата 10.07.87
 ИИЗ. № подл. 90394
 Дата 10.07.87
 ИИЗ. № подл. 90394
 Дата 10.07.87
 ИИЗ. № подл. 90394
 Дата 10.07.87

Действие: $(SP) \leftarrow (PC) + 2 \times NN$
 $(PC) \leftarrow (R5)$
 $(R5) \leftarrow (SP) \uparrow$

Признаки: не изменяются

Описание: Эта команда используется для облегчения выхода из подпрограммы.

При использовании стека для записи в него параметров команда MARK восстанавливает указатель стека (SP) во время выхода из подпрограммы.

Пример: MOV R5, - (SP); предыдущее (R5) заносится в стек
 MOV R1, - (SP); для использования
 MOV R2, - (SP); в программе в стек заносится
 N параметров

MOV RN, - (SP);

MOV MARK, - (SP); команда MARK заносится в стек
 MOV SP, R5; в R5 запоминается адрес команды MARK

JSR PC, dst ; обращение к подпрограмме;

выполнение подпрограммы с начальным адресом, определяемым (dst);

RTS R5 ; возврат из подпрограммы.

После выполнения команды JSR PC, dst стек будет выглядеть следующим образом:

Предыдущее (R5)
R1
R2
RN
MARK
Предыдущее (PC)

ЧИЗ. № подл. Подп. и дата
 50394 10.07.8708
 ЧИЗ. № подл. Подп. и дата
 50394 10.07.8708
 ЧИЗ. № подл. Подп. и дата
 50394 10.07.8708

ЧИЗ. № подл. Подп. и дата
 50394 10.07.8708

ЧИЗ.480.167 Т0

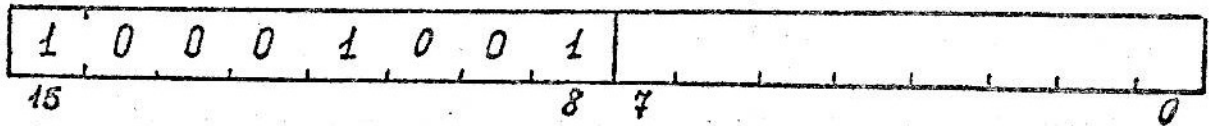
Лист
132

com 32.

TRAP

I04400 - I04777

Командное прерывание



Действие: ↓ (SP) ← (PSW)
 ↓ (SP) ← (PC)
 (PC) ← (34)
 (SP) ← (36)

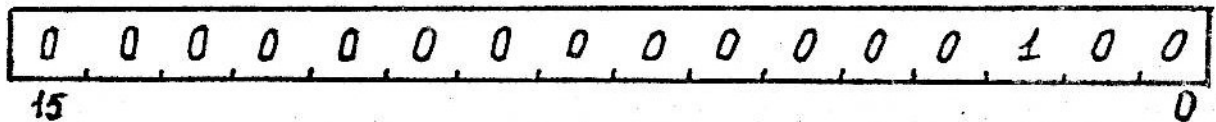
Признаки: Загружаются из вектора прерывания.

Описание: Команды с кодами операций от I04400 до I04777 являются командами TRAP, которые по своему действию идентичны командам EMT, за исключением того, что вектор прерывания команды TRAP имеет адрес 34.

IOT

000004

Командное прерывание для ввода-вывода



Действие: ↓ (SP) ← (PSW)
 ↓ (SP) ← (PC)
 (PC) ← (20)
 (PSW) ← (22)

Признаки: загружаются из вектора прерывания.

Описание: осуществляет прерывание с вектором прерывания, расположенным по адресу 20. Используется для обращения к подпрограмме управления вводом-вы-

Ш. № подл. Подп. и дата. 90394 10.07.87 Зв. Ш. № инв. Ш. № док. Ш. № подл. и дата.

Изм.	Лист	№ докум.	Подп.	Дата

Ш. № 480.167 ТО

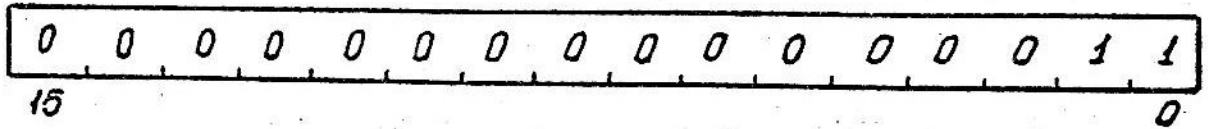
Лист 135

ВОДОМ.

BPT

000003

Командное прерывание для отладки



Действие: $\downarrow (SP) \leftarrow (PSW)$
 $\downarrow (SP) \leftarrow (PC)$
 $(PC) \leftarrow (14)$
 $(PSW) \leftarrow (16)$

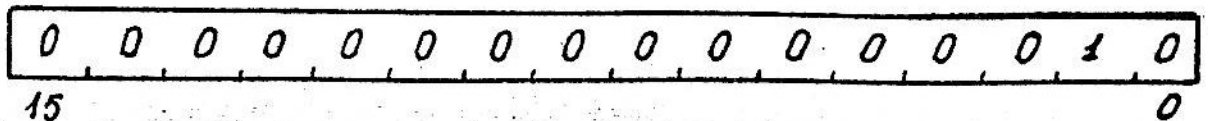
Признаки: загружаются из вектора прерывания.

Описание: выполняется прерывание с вектором расположенным по адресу четырнадцать. Используется для обращения к подпрограммам отладки. Пользователю запрещается употребление кода 000003 в программах, которые выполняются под управлением подпрограмм отладки.

RTI

000002

Возврат из прерывания



Действие: $(PC) \leftarrow (SP) \uparrow$
 $(PSW) \leftarrow (SP) \uparrow$

Признаки: загружаются из стека.

Описание: Используется для выхода из подпрограмм обслуживания внешних и внутренних прерываний. Содержимое PC и PSW восстанавливается с помощью стека. Если при выполнении этой команды будет установлен Т-разряд PSW, то следующая после RTI команда выполняться не будет.

ИИЗ.480.167 ТО

Лист

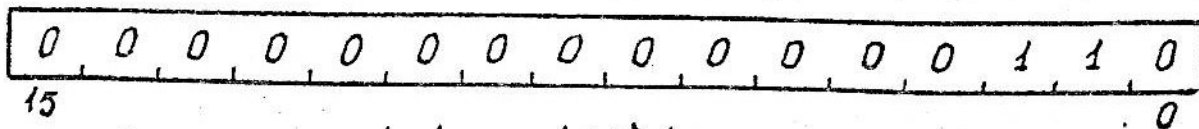
136

Ш.№ № подл. 90394
 Ш.№ № подл. 10.07.8708
 Ш.№ № подл. 10.07.8708
 Ш.№ № подл. 10.07.8708
 Ш.№ № подл. 10.07.8708

RTT

000006

Возврат из прерывания



Действие: (PC) ← (SP) ↑
 (PSW) ← (SP) ↑

Признаки: загружаются из стека.

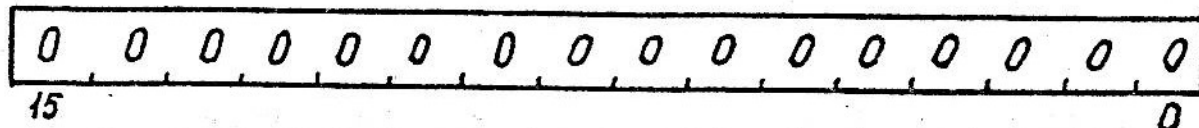
Описание: эта команда по своему действию идентична команде RTI за исключением того, что при установке T-разряда PSW прерывание будет иметь место после того, как выполнится первая команда, следующая за RTI.

8.4.7. Специальные команды

HAZT

000000

Останов



Действие: ↓ (HSP) ← PSW
 ↓ (HSP) ← PC
 0 → PC
 340 → PSW

ИИЗ.480.167 TO

ЛИСТ

137

ЦИФ. № ПОД. ПОДП. И ДАТА. ВЗЛОМ. ЦИФ. № ДОКУМ. ПОДП. И ДАТА.
 90394 10.07.87

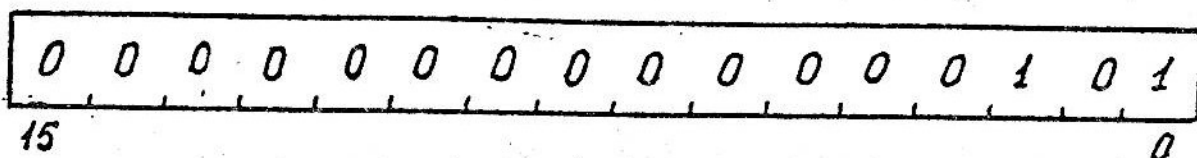
ИЗМ. ЛИСТ. № ДОКУМ. ПОДП. ДАТА

ющей за командой WAIT. Таким образом, когда по прерыванию вызывается передача содержимого PC и PSW в стек, адрес команды, следующий за командой WAIT, сохраняется. Выход из подпрограммы, обслуживающей прерывание (т.е. выполнение команды RTI или RTT) вызовет возобновление прерванного процесса с командой, следующей за командой WAIT.

RESET

000005

Сброс внешних устройств



Признаки: не изменяются.

Описание: по этой команде на выводе микропроцессора вырабатывается импульс длительностью в 1545 периодов тактовой частоты СЛС . После импульса микропроцессор возобновит свою работу через время ожидания, равное 1545 периодам тактовой частоты СЛС .

Примечание: в моде пользователя команда выполняется как NOP :

ШИЗ. № 1000 / Подп. и дата: 10.07.87 г. / ШИЗ. № 0901 / ШИЗ. № 0901 / Подп. и дата:

Изм.	Лист	№ докум.	Подп.	Дата

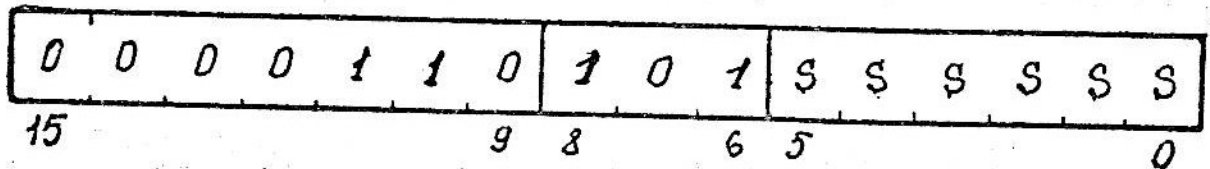
ШИЗ.480.167 ТО

Лист
139

MFPI

0065 SS

Засылка инструкции в стек текущей моды по адресу предыдущей моды



Действие: $(temp) \leftarrow (pc)$
 $\downarrow (sp) \leftarrow (temp)$

Признаки: **N** - устанавливается, если $(pc) < 0$, в противном случае очищается;
Z - устанавливается, если $(pc) = 0$, в противном случае очищается;
V - очищается;
C - не изменяется.

Описание: Эта команда засылает слово в стек текущей моды по адресу, вычисленному в предыдущей моде. Виртуальный адрес источника вычисляется с помощью текущих регистров ДП.

MFPD

1065 SS

Засылка данных в стек текущей моды по адресу предыдущей моды

Эта команда выполняется так же как и команда MFPI

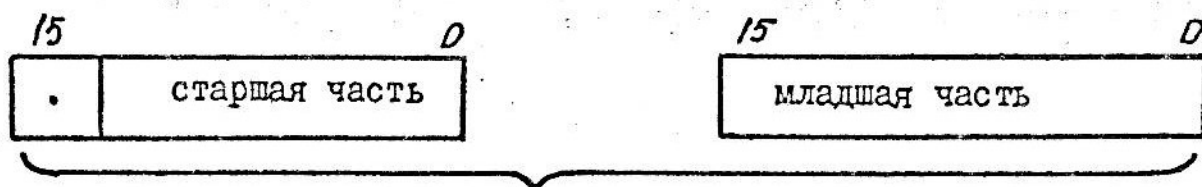
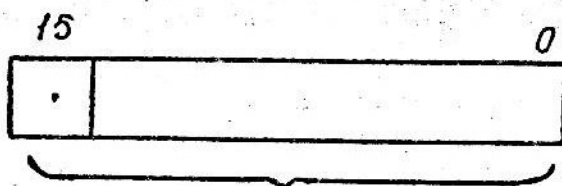
ЦИФ. № ПОД. ПОДП. И ДАТА. ВЗЛОМ ЦИФ. № ДОК. ПОДП. И ДАТА.
 90394 10.07.8708м.

Комбинации указанных выше операций очистки или установки, соединенных по схеме "ИЛИ", могут образовывать комбинированные команды.

8.4.9. Команды расширенной арифметики

MUL команда расширенной арифметики (РА) — умножение
 Прием операнда — источника происходит в цикле "чтение".

Формат чисел, используемых при выполнении команд расширенной арифметики, следующий:



тридцатидвухразрядное слово

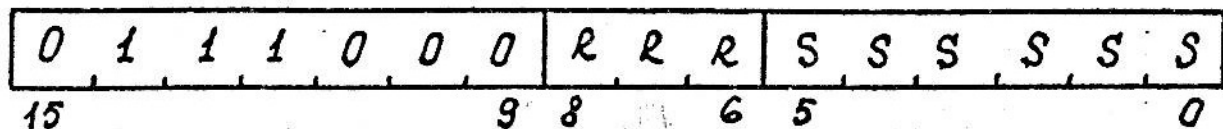
Разряд пятнадцати старшей части тридцатидвухразрядного слова или разряд пятнадцати обычного шестнадцатиразрядного слова являются знаковыми разрядами.

Положительным числам соответствует 0 в знаковом разряде, отрицательным — 1

MUL

070RSS

Умножение



Ш. № подл. Подп. и дата. Взам. инв. № докум. Подп. и дата.
 90394 10.07.870 В.

ШИЗ.480.167 ТО

ЛИСТ
443

Действие: $R, R \vee I \leftarrow R \times (RC)$

Признаки: N - устанавливается, если результат < 0 ,
в противном случае очищается;

Z - устанавливается, если результат $= 0$,
в противном случае очищается;

V - очищается;

C - устанавливается, если результат меньше
чем -2^{15} или больше $2^{15} - 1$.

Описание: Перемножаются операнды источника и приемника,
взяты в двоичнодополнительном коде. Результат
помещается в регистр, используемый в качестве
приемника, и в следующий за ним регистр, если
регистр приемника имеет четный номер. Если же
регистр приемника имеет нечетный номер, сохра-
няется только младшая часть результата.

Мнемоническая запись команды умножения: MUL, RC, R

Пример: $MUL \# 10, RI$

До операции

I000)070I27

I002)000010

После операции:

RI) 000400

I000)070I27 RI) 004000

I002)000010

DIV

07IRSS

Деление

0	1	1	1	0	0	1	R	R	R	S	S	S	S	S	S
15						9	8	6	5						0

ШИЗ.480.167 ТО

Лист

144

ИЗМ. Лист № докум. Подп. Дата

ФОРМА 50, ГОСТ 2.106-68

СОДМСТАЧ

ИЗМ. № ПОДП. И ДАТА. ВЗОМ. ИЗМ. И ЧИВ. № ДОКУМ. ПОДП. И ДАТА.
90394 12.07.87 В.В.

Действие: $R, RVI \leftarrow R, RVI / (src)$

Признаки: N — устанавливается, если частное < 0 , в противном случае очищается;

Z — устанавливается, если частное $= 0$, в противном случае очищается;

V — устанавливается, если $(src) = 0$, или, если $(dst) > (src)$ по абсолютной величине. (В этом случае выполнение команды прекращается, т.к. частное будет превышать пятнадцать разрядов);

C — устанавливается, если делитель $= 0$.

Описание: Тридцатидвухразрядное число в двоично-дополнительном коде, находящееся в регистрах R и RVI , делится на операнд источника. Частное заносится в R , а остаток — в RVI . После выполнения операции деления знак остатка будет таким же, как и у делимого. Следует заметить что номер регистра R должен быть четным.

Мнемоническая запись команды деления: $DIV src, R$

Пример: $DIV \neq 2, R0$

До операции:

2000)071027 R0)000000

2002)000002 R1)020001

После операции:

2000)071027 R0)010000

2002)000002 R1)000001

в R0 находится частное
а в R1 остаток.

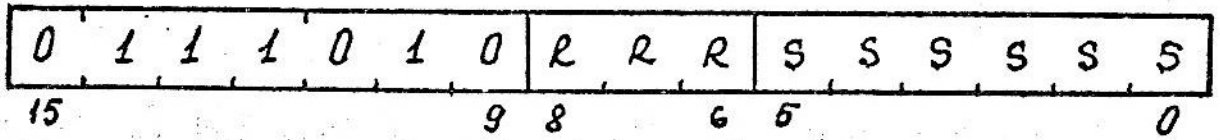
Инв. № подл.	Подл. и дата	Узлом. инв. №	Инв. № дубл.	Подл. и дата
90394	10.07.87авм			

Узм	Лист	№ докум.	Подл.	Дата

ШИЗ.480.167 Т0

Лист
145

Арифметический сдвиг



Действие: $R \leftarrow R$ сдвинутое на NN позиций влево или вправо, где NN - 6 младших разрядов (zrc).

Признаки: N - устанавливается, если результат < 0 , в противном случае очищается;

Z - устанавливается, если результат $= 0$, в противном случае очищается;

V - устанавливается, если после выполнения операции сдвига операнд изменил знак, в противном случае очищается;

C - загружается содержимым последнего разряда, выдвинутого из регистра.

Описание: Содержимое указанного регистра сдвигается влево или вправо на количество позиций, определяемое счетчиком сдвига. Функцию счетчика сдвига выполняют шесть младших разрядов операнда источника. Его значение может изменяться в пределах от минус тридцатидвух до плюс тридцатиодного. Отрицательные значения счетчика определяет сдвиг вправо, положительное - влево.

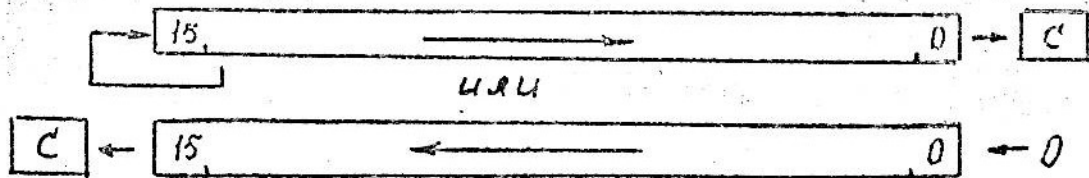
ИНБ. № подл.	ПОДП. И ДАТА
90394	10.07.87024
ИНБ. № докум.	ПОДП. И ДАТА
1000000	

ЦЕН. ЛУС. № док. ум.	ПОДП.	ДАТА	

ИИЗ.480.167 Т0

ЛУС
146

Выполнение команды ASH иллюстрируется ниже.



Счетчик сдвига
(6 младших разрядов
операнда источника)

Выполняемая функция

01111

Сдвиг R на 31 позицию влево

000001

Сдвиг R на 1 позицию влево

11111

Сдвиг R на 1 позицию вправо

100000

Сдвиг R на 32 позиции вправо

Мнемоническая запись команды арифметического сдвига:

ASH *rc*, R

Пример: ASH R3, R0 (072003)

До операции

После операции

R0) 001234

R0) 012340

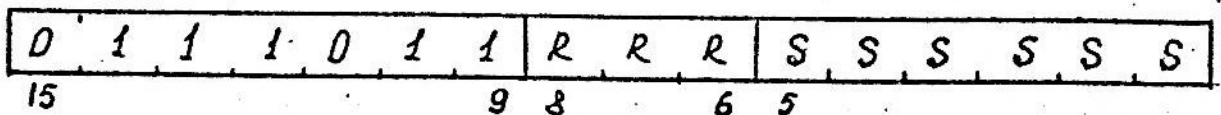
R3) 000003

R3) 000003

ASHC

073RSS

Арифметический сдвиг двойного слова



Действие: R, RvI \leftarrow R, RvI. Сдвинутое на *NN* позиций вправо или влево, где *NN* - 6 младших разрядов (*rc*).

Признаки: *N* - устанавливается, если результат < 0 , в противном случае очищается;

ЧИВ. № подл. 90394
 ПОСП. И ДАТА. 10.07.87
 ВЗМ. И ДАТА. 10.07.87
 ИМЬ. И ДАТА. ПОСП. И ДАТА.

ШИЗ.480.167 ТО

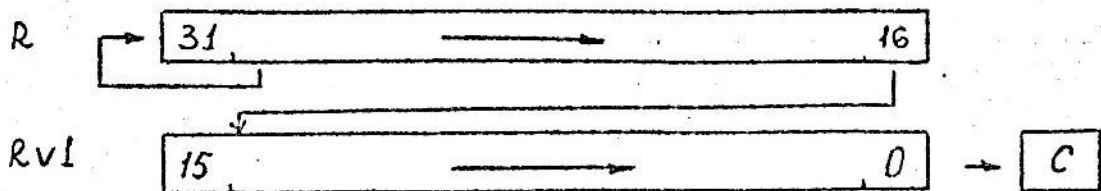
Лист

147

- Z - устанавливается, если результат = 0, в противном случае очищается;
- V - устанавливается, если после выполнения операции сдвига операнд изменил знак, в противном случае очищается;
- C - загружается содержимым последнего выдвинутого разряда тридцатидвухразрядного слова;

Описание: Содержимое регистров R и RVI интерпретируется как одно тридцатидвухразрядное слово. Причем, младшая часть слова (разряды 00-15) содержится в RVI, а старшая часть (разряды 16-32) - в R. Тридцатидвухразрядное слово сдвигается вправо или влево на количество позиций, определяемом счетчиком сдвига. Функцию счетчика сдвига выполняют шесть младших разрядов операнда источника. Его значение может изменяться в пределах от минус тридцатидвух до плюс тридцатиодного. Отрицательное значение счетчика определяет сдвиг вправо, положительное - влево. Если выбранный регистр имеет нечетный номер, то R и RVI являются одним и тем же регистром. В этом случае сдвиг вправо будет выполняться циклически (шестнадцатиразрядное слово сдвигается циклически на количество позиций, определяемое счетчиком сдвига).

Выполнение команды A SHC иллюстрируется ниже.



ИИЗ.480.167 ТО

Лист

148

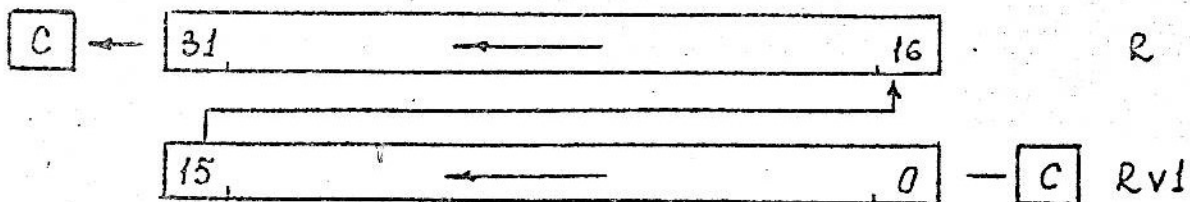
ЧИЗ. № подл. 90394
 Подп. и дата 10.07.87
 Подп. и дата 10.07.87
 Подп. и дата 10.07.87
 Подп. и дата 10.07.87

Изм. Лист Подком. Подп. Дата

форма 5а по ГОСТ 2.106-68

Формат А4

или



Мнемоническая запись команды арифметического сдвига
двойного слова: ASHC *nc, R*

Пример: ASHC # 75, R2

До операции:

4000)073227 R2) 012345

4002)000075 R3) 000000

После операции:

4000)073227 R2)001234

4002)000075 R3)120000

ИНВ. № подл.	ПОДП. И ДАТА	ВЗАМ. ИНВ. ИЛИ КОУП.	ПОДП. И ДАТА
90394	10.07.87		

ИИЗ.480.167 ТО

Лист
149

8.5. Время выполнения некоторых команд.

Время выполнения команд в общем случае определяется как сумма

$$t_{\text{вып.}} = t$$

$$t_{\text{вып.}} = t_{\text{осн}} + t_{\text{ss}} + t_{\text{DD}}, \text{ где}$$

$t_{\text{осн}}$ - время выборки команды из памяти, ее декодирование и выполнение операции;

t_{ss} - время выборки операнда-источника;

t_{DD} - время выборки операнда-приемника.

Время выполнения некоторых команд включает не все эти слагаемые. Так как выборка команды, ее декодирование и выполнение операции в операционном блоке выполняются параллельно во времени, то при последовательном выполнении программы время этих операций оказывается "скрытым". Не влияет на время выполнения команды и время работы диспетчера памяти, независимо от того, включен он или выключен. Таким образом, время $t_{\text{осн}}$ определяется обычно как время работы операционного блока необходимое для выполнения операции, указанной в команде. Только в случае изменения счетчика команд в командах условных переходов, прерываниях и т.п. время $t_{\text{осн}}$ увеличивается на время выборки команды из памяти и ее декодирование, равное $10T + t_{\text{пам}}$ ($t_{\text{пам}}$ - задержка памяти, $T=1/f_{\text{цкл}}$).

Времена выполнения некоторых наиболее часто встречающихся команд приведены в табл. 8. Все времена даны при $t_{\text{пам}} = 0$ в циклах как чтения, так и записи (время $t_{\text{пам}}$ считается равным нулю, если задержка сигнала $RPLY$ относительно сигналов DLN или $DOVT$ не превышает 20нс). Также считается, что сигнал $SSYNC$ установлен равным "0".

Инв. № подл.	Подп. и дата	Инв. № подл.	Подп. и дата
90394	10.07.87	Инв. № подл.	Подп. и дата
		Инв. № подл.	Подп. и дата
		Инв. № подл.	Подп. и дата

СИЗ.480.167 Т0

Мода адресации	Команда	Время(Т)	Команда	Время (Т)
0	0I0000	3	0I0000	3
I	0II000	II	0I00I0	II
2	0I2000	I6	0I0020	II
3	0I3000	I9	0I0030	24
4	0I4000	I6	0I0040	I4
5	0I5000	I9	0I0050	2I
6	0I6000	II	0I0060	I2
7	0I7000	I9	0I0070	I9
27	0I2700	IO	0I0027	I2
37	0I3700	II	0I0037	I2
67	0I6700	I4	0I0067	I2
77	0I7700	I9	0I0077	I9
0	II0000	9	II0000	3
I	III000	I7	II00I0	II
2	II2000	22	II0020	II
3	II3000	25	II0030	2I
4	II4000	22	II0040	II
5	II5000	25	II0050	I8
6	II6000	22	II0060	I2
7	II7000	27	II0070	I8
27	II2700	I5	II0027	I2
37	II3700	20	II0037	I2
67	II6700	20	II0067	I2
77	II7700	25	II0077	I9

Инв. № подл. 90394. Подп. и дата 10.07.87 ОВМ.
 Взам. инв. № Инв. № дубл. Подп. и дата

ИИЗ.480.167 ТО

Лист 151

Мода адресации	Команда	Время (T)	Команда	Время (T)
0	060000	3	060000	3
1	061000	8	060010	13
2	062000	16	060020	18
3	063000	19	060030	24
4	064000	16	060040	15
5	065000	19	060050	25
6	066000	14	060060	21
7	067000	21	060070	23
27	062700	10	060027	16
37	063700	10	060037	16
67	066700	14	060067	16
77	067700	24	060077	23
0	005100	3	I05100	3
1	005110	8	I05110	8
2	005120	10	I05120	10
3	005130	26	I05130	26
4	005140	15	I05140	15
5	005150	21	I05150	21
6	005160	11	I05160	16
7	005170	18	I05170	23
27	005127	16	I05127	16
37	005137	16	I05137	16
67	005167	11	I05167	11
77	005177	18	I05177	18

ШИЗ. № подл. 90394
 Дата 10.07.87
 ШИЗ. № докл. ШИЗ. № докл.
 ШИЗ. № докл. ШИЗ. № докл.
 ШИЗ. № докл. ШИЗ. № докл.

ШИЗ.480.167 TO

Лист 152

Мода адресации	Команда	Время (Т)	Команда	Время (Т)
0	000300	9		
1	000310	19		
2	000320	21		
3	000330	27		
4	000340	22		
5	000350	27		
6	000360	22		
7	000370	27		
27	000327	22		
37	000337	22		
67	000367	22		
77	000377	29		

Ниже представлены времена выполнения некоторых других команд для нулевой моды:

Команда	Время (Т)
MFPS	18
MTPS	27
MFPI (D)	12
MTRI (D)	13
MDL	60
DIV	93
Команды ветвления	3

Инв. № подл.	90394
Подп. и дата	10.07.87 В.ч.
Взам. инв. №	
Инв. № дубл.	
Подп. и дата	

ШИЗ.480.167 ТО

Команда	Время (Т)
RTS	32
MARK	25
Команды изменения признаков	3
HALT	44
WAIT	7
RESET	I8T+I545T+I545T
SOB без ветвления	2I
SOB с ветвлением	9

Инв. № подл.	Подп. и дата	Инв. № докл.	Подп. и дата
90394	10.07.870Вн.		
Взам. инв. №			

Инв. № подл.	Подп. и дата	Инв. № докл.	Подп. и дата	Изм. лист.	№ докум.	Подп.	Дата	ИЗ.480.167 TO	Лист
									154

9. ДИСПЕТЧЕР ПАМЯТИ

9.1. Общее описание

Диспетчер памяти обеспечивает:

- 1) расширение емкости адресуемой памяти с 64К до 2М слов;
- 2) перераспределение виртуального адреса в физический и защиту памяти в системах с разделением времени;
- 3) использование различных областей адресов для режима пользователя и внутреннего режима работы процессора.

9.1.1. Расширение емкости адресуемой памяти

Диспетчер памяти позволяет адресовать память емкостью до 2М слов (в зависимости от режима работы).

Шестнадцатиразрядная длина слова позволяет адресовать 32К слов. Из них 4К слов резервируется для регистров периферийных устройств. Таким образом, общая емкость адресуемой памяти расширяется с 28К до 2М слов. Это осуществляется преобразованием шестнадцатиразрядного виртуального адреса в восемнадцать и двадцатидвухразрядный физический адрес. Преобразование осуществляется с помощью набора регистров перераспределения.

9.1.2. Перераспределение виртуального адреса в физический

Диспетчер памяти перераспределяет все адреса автоматически, поэтому можно считать, что ОБ работает в области виртуальных адресов. Это значит, что не имеет значения, где в физической памяти размещена программа.

9.1.3. Деление памяти.

Область виртуального адреса делится на восемь отдельных страниц, каждая 4К слов. Каждая страница перераспределяется

ШИЗ.480.167 ТО

Лист

155

Ш.И.В. № подл. 90394
Подп. и дата 10.07.87
Взам. инв. и дата 10.07.87
Инв. № инв. Подп. и дата

Изм.	Лист	Идокум.	Подп.	Дата

отдельно, что является важной особенностью в системах с разделением времени. Это позволяет размещать большие программы в дискретные блоки физической памяти.

Диспетчер памяти обеспечивает перераспределение страниц длиной в тридцать два слова. Это позволяет занимать под небольшие массивы данных необходимый объем памяти.

9.1.4. Защита памяти

Каждая виртуальная страница имеет свой код защиты. Есть три вида защиты памяти:

- 1) разрешены запись и считывание;
- 2) разрешено только считывание;
- 3) запрещен любой доступ.

Попытка нарушения любого из этих видов защиты предотвращается диспетчером памяти. Например, при попытке запрещенного чтения (попытке чтения из страницы с кодом защиты - запрещение любого доступа) информация из ячейки не считывается. При попытке запрещенной записи, содержимое ячейки не изменяется. Все попытки запрещенного доступа вызывает немедленное прерывание (отказ) через область внутреннего режима. Диспетчер памяти запоминает информацию об этих прерываниях с тем, чтобы пользователь мог распознать причину прерывания.

9.1.5. Управление режимами работы

В системах с разделением времени программе пользователя запрещается вмешиваться в операционную систему или в программы других пользователей. Диспетчер памяти обеспечивает два режима работы процессора: внутренний режим и режим пользователя, что является основой системы с разделением времени. Для каждого режима работы существует набор регистров: регистры адреса страницы/ регистры признака страницы (PAR/PDR). При каждом обращении к памяти, ДП анализирует регистр состояния процессора (PSW)

ИЧБ. № подл. 90394
ИЗМ. № подл. 10.07.8702
ВЗАМ. ИЧБ. № подл. 10.07.8702
ПОДП. И ЧАТА

ИИЗ.480.167 ТО

Лист
156

ИЗМ. Лист / ДОКУМ. Подп. Дата

для выбора соответствующего набора PAR/PDR. Например, программа пользователя не может работать в области внутреннего режима.

9.1.6. Управление памятью

В системах с разделением времени память должна использоваться как можно более эффективно с тем, чтобы позволить работу максимально возможного количества пользователей с минимальными задержками. Логика ДП содержит разряд для каждой страницы, указывающий, что в данную страницу производилась запись. Программа управления памятью может опросить каждый P_{SW} с тем, чтобы определить, использовалась ли данная страница. Если производилась запись в текущую активную страницу, то об этом необходимо информировать операционную систему управления памятью, с тем, чтобы измененная программа могла быть записана в промежуточную память перед перераспределением в данную страницу.

9.2. Работа диспетчера памяти

9.2.1. Перераспределение памяти

Основными функциями ДП является перераспределение памяти и расширение емкости адресуемой памяти для систем, использующих более чем 28К физической памяти. ДП использует два набора регистров адреса страницы для перераспределения виртуального адреса в физический адрес памяти. Эти наборы используются как аппаратные регистры перераспределения, которые позволяют одновременно располагать в физической памяти нескольких пользователей, которые все начинаются с нулевого виртуального адреса.

9.2.2. Перераспределение программы

Регистры адреса страницы используются для задания адресов каждой перераспределенной программы в физической памяти. На рис. 31 приведен пример перераспределения программы. В этом примере нулевой начальный адрес программы А перераспределяется с помощью константы перераспределения для получения физического

Уч. № подл. 90394
Подп. и дата 12.07.87
Взам. инв. инв. № подл. Подп. и дата

Изм.	Лист	Итого	Подп.	Дата

ИИЗ.480.167 ТО

Лист
157

Схема перераспределения программы.

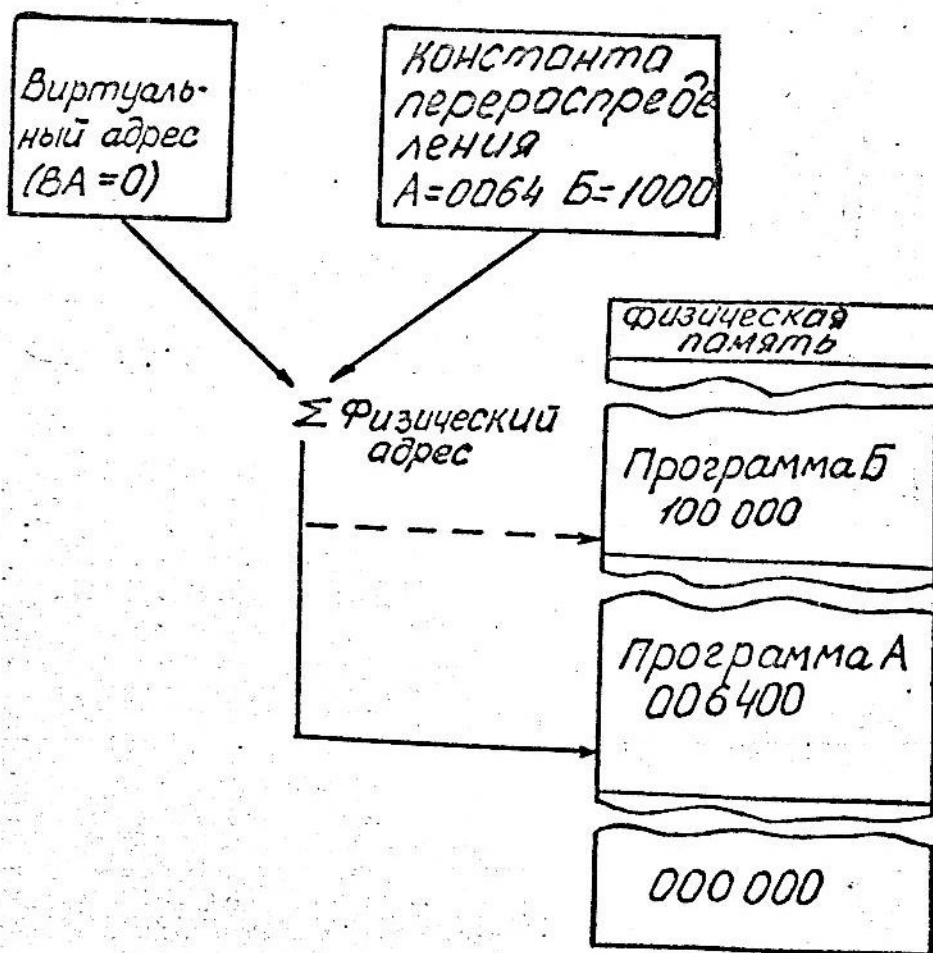


Рис. 31

Инв. № позн.	Позн. и дата	Взам. инв. №	Инв. № дубл.	Позн. и дата
90394	10.07.87	Взм.		

Лист	№ докум.	Позн.	Дата

ИИЗ.480.167 ТО

Лист
158

деляется (хотя константа перераспределения не равна нулю);

3) все страницы, показанные в примере на рис. 32, начинаются в областях с интервалом в тридцать два слова;

4) каждая страница перераспределяется отдельно. Несколько страниц могут быть перераспределены в одну и ту же область физической памяти. Использование нескольких регистров адреса страницы в наборе, для доступа к одной и той же области физической памяти обеспечивает обращение различных частей программы к одним и тем же данным. В примере на рис. 32 страницы четыре и шесть перераспределяются в одну и ту же область физической памяти.

9.2.3. Расширение емкости адресуемой памяти

При работе ДП адрес, вырабатываемый ОБ, не является непосредственным физическим адресом устройства или ячейки памяти. Этот адрес является шестнадцатиразрядным виртуальным адресом, который используется диспетчером памяти для построения восемнадцати и двадцатидвухразрядного физического адреса.

На рис. 33 показано построение восемнадцатиразрядного физического адреса. Разряды виртуального адреса VA (15...13) рассматриваются как поле активной страницы (ПАС), которое используется для выбора одного из восьми активных регистров страницы, включает в себя регистр адреса страницы и регистр признака страницы.

Разряды виртуального адреса, VA (12...06), указывают номер блока (от 0 до 127_{10}) внутри страницы. VA (05...00) указывают смещение внутри каждого блока из 32 слов.

PAR содержит базовый адрес страницы, который записывается в PAR программно.

VA (12...06) складывается с базовым адресом для получения двенадцати старших разрядов физического адреса. Младшие

№ п/п
90394
Дата
10.07.87
Подп. и дата
Взам. инв. и инв. № докум.
Подп. и дата

ИДЗ.480.167 ТО

Лист

160

шесть разрядов \sqrt{A} (05...00) не изменяются при построении физического адреса. Таким образом формируется восемнадцатиразрядный физический адрес. Формирование двадцатидвухразрядного физического адреса будет показано далее.

9.2.4. Команды диспетчера памяти

ДП обеспечивает возможность связи между двумя областями (внутренней области и областью пользователя), задаваемыми регистром состояния процессора, разрядами (I5...I2). Эта возможность осуществляется с помощью команд:

MPRI - пересылка из текущей в предыдущую область (0066DD);

MPPI - пересылка из предыдущей в текущую область (0065SS);

Эти команды с аппаратной точки зрения являются модифицированной командой MOV.

При рассмотрении команд ДП необходимо учитывать следующее:

1) есть два режима работы:

- внутренний режим;
- режим пользователя;

2) выбор режимов осуществляется разрядами PSW (I5...I2)

следующим образом:

PSW (I5...I2)		Текущий режим	Предыдущий режим
00	00	Внутренний	Внутренний
II	II	Пользователя	Пользователя
II	00	Пользователя	Внутренний
00	II	Внутренний	Пользователя

3) команды MPRI и MPPI используются для работы в текущем и внутреннем режиме при предыдущем режиме пользователя;

ШИЗ.480.167 Т0

Лист

164

Ш.№ подл. 90394 Подп. и дата 10.07.87 Ю.И.

Изм. Лист Проверка Подп. Дата

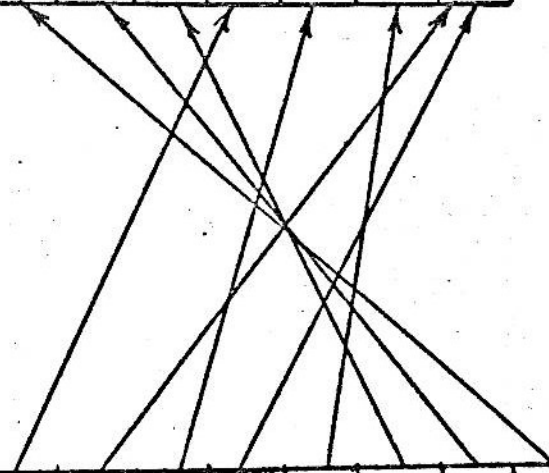
№ докум. по подл.	Подп. и дата	Зам. инв. №	Инв. № докл.	Подп. и дата
90394	10.07.87	Вч.		

Перераспределение 32К виртуальных адресов
в физической памяти из 124К слов.

Область виртуального адреса
160000-177776
140000-157776
120000-137776
100000-117776
060000-077776
040000-057776
020000-037776
000000-017776

№ мер. стр.	Константа перераспределения
7	1500XX
6	0200XX
5	1000XX
4	0200XX
3	0600XX
2	2500XX
1	3200XX
0	4000XX

Область физической памяти
400000-417776
320000-337776
250000-267776
150000-167776
100000-117776
060000-077776
020000-037776



Примечание. xx - адрес внутри блока.

Рис. 32

Построение Восемнадцатиразрядного физического адреса.

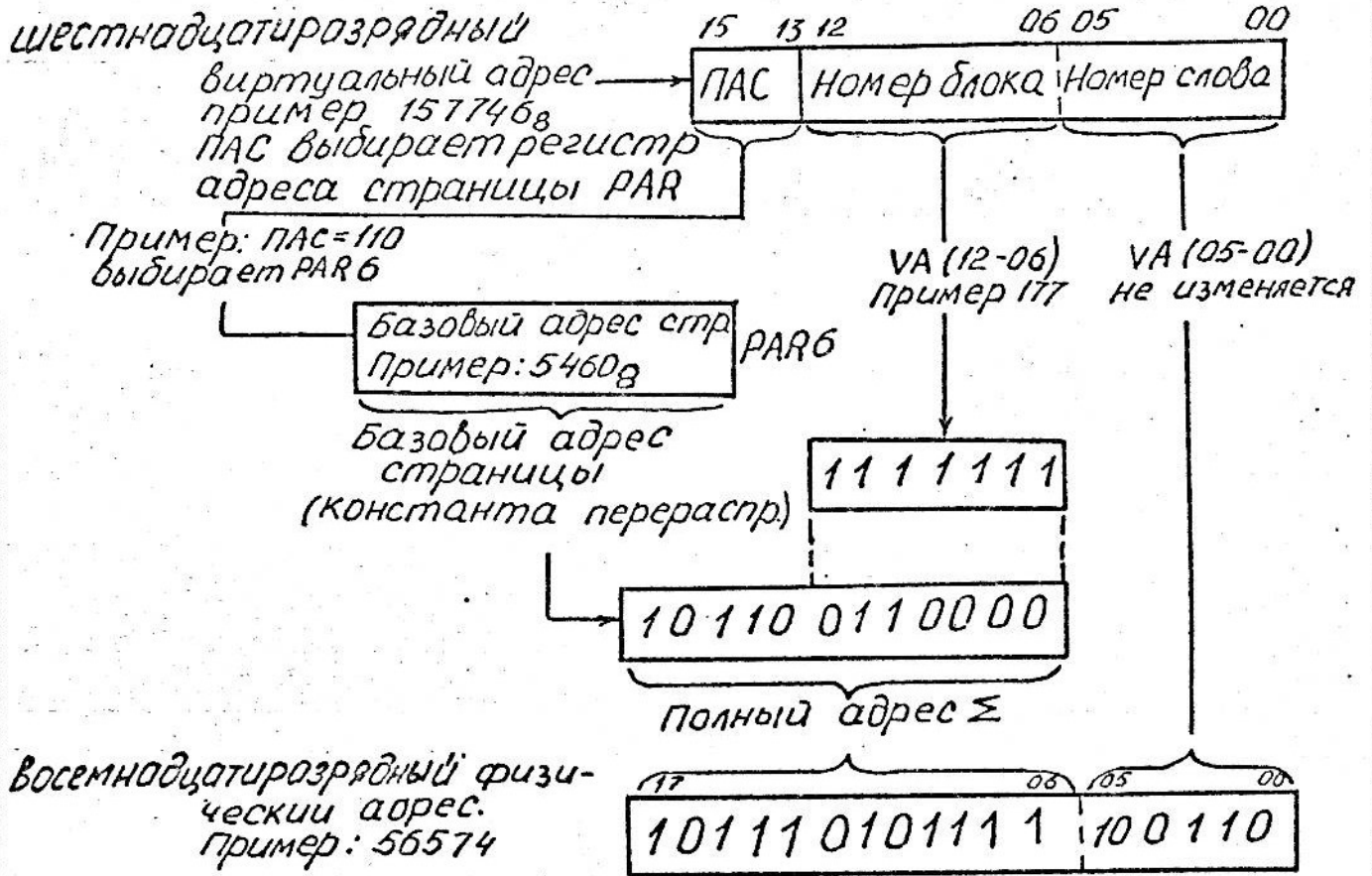


рис. 33

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата
90394	10.07.87 Вч.			

Изм.	Лист	№ док. ум.	Подп.	Дата

ШИЗ.480.167 Т0

Лист
463

4) текущий режим задает набор активных регистров страницы для перераспределения виртуального адреса в физический.

При выборке команды MFPI она преобразуется следующим образом:

MFPI \$S → MOV \$S, → (SP)

Преобразованная команда выполняется ОБ совместно с ДП.

Вычисление адреса источника производится в текущей области. Это значит, что индексные слова и косвенные адреса выбираются с использованием регистров адреса страницы, задаваемых разрядами текущего режима в PSW. Окончательная выборка операнда источника производится в предыдущей области, т.е. с использованием регистров адреса страницы. Необходимо отметить, что, если метод адресации источника ноль, а регистр шесть, то выбор указателя стека (SP) осуществляется в предыдущем режиме.

Но если метод адресации источника отличен от нуля, а регистр шесть то выбор SP осуществляется в текущем режиме, т.к. в этом случае содержимое регистра используется при вычислении адреса, а не как конечный операнд. Операнд источника затем засылается в стек текущего режима.

При выборке команды MTRI, она преобразуется следующим образом:

MTRI DD → MOV (SP) + , DD

Вычисление адреса приемника производится в текущей области. Это значит, что индексные слова и косвенные адреса

ШАН.№: ПОСЛ. ПОСЛ. И ДАТА ВЗАМ. ШАН. И Ш. И ДУБЛ. ПОСЛ. И ДАТА
 90394 19.07.87 ЗВМ

Исх.	Лист	Максимум	Посл.	Дата

ШИЗ.480.167 Т0

Лист
164

выбираются с использованием регистров адреса страницы, задаваемых разрядами текущего режима в PSW . Окончательная запись операнда в приемник осуществляется в предыдущем режиме. Необходимо отметить, что если метод адресации приемника 0, а регистр 6, то выбор SR осуществляется в предыдущем режиме. Но если метод адресации приемника отличен от 0, а регистр 6, то выборка SR осуществляется в текущем режиме, т.к. в этом случае содержимое регистра используется при вычислении адреса, а не как окончательный операнд.

Эта команда извлекает слова из текущего стека, определяемого PSW (I5, I4) и записывает это слово по адресу приемника в предыдущей области, определяемой PSW (I3, I2).

Таким образом, эти команды используются для обмена данными между внутренней областью и областью пользователя.

Пользователь не всегда может непосредственно управлять своими устройствами ввода и вывода, т.е. область адресов устройств ввода и вывода не всегда доступна пользователю (это определяется содержимым PAR пользователя, которое устанавливается программно). Пользователь вырабатывает запрос обращения к вводу и выводу с помощью прерывания, например, командой EMT. Перед прерыванием пользователь засылает в свой стек (R16) определенные параметры.

При прерывании PSW (I5...I2) устанавливается таким образом, что текущим является внутренний режим, а предыдущим - режим пользователя. После этого командой MFPI параметры пользователя извлекаются из стека пользователя и засылаются во внутренний стек.

ЧИВ. № подл. Подп. и дата. Взам. инв. №. Инв. № подл. Подп. и дата.
 90394 10.07.87

Изм. Лист. Номер. Подп. Дата.

ШИЗ.480.167 ТО

Лист
165

9.2.5. Защита памяти

В системах с разделением времени в оперативной памяти одновременно могут находиться несколько программ, которые выполняются последовательно. Доступ к этим программам и к области памяти, которые они занимают, необходимо четко определять и управлять ими. ДП обеспечивает следующие виды защиты памяти:

1) программа пользователя не должна выходить за пределы отведенной для нее области памяти, если это не предусмотрено операционной системой;

2) программа пользователя не должна изменять общие подпрограммы и алгоритмы, которые используются всеми пользователями;

3) программе пользователя запрещается брать управление на себя или изменять операционную систему.

9.2.5.1. Код защиты памяти.

Для каждой страницы есть двухразрядное поле защиты памяти. Код защиты памяти устанавливается программно. Когда код равен 0, страница является нерезидентной. Любая попытка обращения программы пользователя к нерезидентной странице вызывает немедленный отказ. Для всех страниц, связанных с неиспользуемыми в данный момент программами, код защиты устанавливается в 0, а для страниц, связанных с текущей выполняемой программой, код защиты устанавливается для разрешения к ним доступа.

ИНВ. № 7008 ПОСП. И ОБСТА. ВЗЛОМ ИНВ. № 0908. ПОСП. И ОБСТА.
90394 10078702

ИЗМ. ЛУСТ. № 000КУМ. ПОСП. И ОБСТА.

ИИЗ.480.167 ТО

ЛУСТ
166

Код защиты для страницы может быть установлен равным двум, что позволяет обращение к странице для считывания, но вызывает немедленный отказ при попытке записи. Такой вид защиты может быть установлен для страниц, которые содержат общие для всех программ данные, программы и алгоритмы.

Если код защиты установлен равным шести, то разрешено любое обращение к данной странице, т.е. разрешены запись и считывание.

Регистры адреса страницы в каждом наборе (внутренний и пользователя) могут быть установлены для обращения к одной и той же физической странице памяти, но каждый с разным кодом защиты. Например, код защиты пользователя может быть равен двум (только считывание), а внутренний код защиты может быть равен шести (запись и считывание).

9.2.5.2. Наборы PAR/PDR

Для каждого режима используется свой набор PAR/PDR.

Разряды выбора режима в PSW (I5, I4 - текущий режим, I3, I2 - предыдущий режим) задают конкретный набор, используемый при обращении к памяти. Обычно набор задается разрядами текущего режима, но в командах MTP1 и MPPI используются также разряды предыдущего режима. В зависимости от значений этих разрядов выбор наборов PAR/PDR осуществляется следующим образом:

Разряды PSW		Набор PAR/PDR
I5 (I3)	I4 (I2)	
0	0	внутренний режим
0	1	запрещенный (любое обращение вызывает отказ)
1	0	
1	1	режим пользователя

ШИЗ.480.167 ТО

Лист
167

ЧИВ. № подл. Подп. и дата. Взам. инв. и дата. Инв. и подл. Подп. и дата. 90394 10.07.87 08.

Изм. Лист. И док. И подл. Дата

9.2.5.3. Режим работы процессора

Программы внутреннего режима не накладывают никаких ограничений на работу процессора.

При выполнении программ в режиме пользователя накладываются следующие ограничения:

1) попытка выполнения команды HALT вызывает прерывание по резервной команде с адресом вектора прерывания IO;

2) команда RESET выполняется как команда NOP;

3) при прерываниях и возвратах из прерываний содержимое PSW изменяется следующим образом:

Содержимое PSW	RTI, RTT	Прерывания
Признаки PSW (03...00)	Загружаются из стека	Загружаются из вектора
T-бит PSW 04	Загружается из стека	Загружается из вектора
Приоритет PSW (07...05)	Не изменяется	Загружается из вектора
Предыдущий режим PSW (I3, I2)	Не изменяется	Загружается содержимым PSW (I5, I4)
Текущий режим PSW (I5, I4)	Не изменяется	Загружается из вектора

4) защита от переполнения стека осуществляется с помощью средств защиты памяти.

В каждом режиме в качестве регистра указателя стека используются разные регистры:

ЩИЗ.480.167 ТО

Лист

168

ЧИЗ. № подл. 00394
 Подп. и дата 12.07.87
 ЧИЗ. № подл. 00394
 Подп. и дата 12.07.87
 ЧИЗ. № подл. 00394
 Подп. и дата 12.07.87

внутренний режим - R6, R [SP] ;

режим пользователя - R16, R [SP ПОЛЬЗ] .

При прерываниях вектор всегда находится во внутренней области, а PC и SP засылаются в стек, задаваемый разрядами текущего режима в PSW, которые были загружены из вектора.

Программы пользователя и внутренние программы перераспределяются с помощью своих выборов PAR/PDR. Если активные регистры страницы установлены правильно, то программа, выполняемая в этом режиме, не может обращаться к областям, предназначенным для другого режима. Например, программа пользователя не может обращаться к внутренней области.

9.2.6. Описание PAR/PDR

ДП содержит два набора из восьми активных регистров страницы каждый. Каждый активный регистр страницы включает в себя пару PAR/PDR. PAR/PDR используются всегда совместно и содержат информацию, необходимую для перераспределения и описания текущей активной страницы для каждого режима работы.

Один набор используется во внутреннем режиме, а другой в режиме пользователя. Разряды текущего режима (а в некоторых случаях разряды предыдущего режима) в PSW задают конкретный набор, используемый при обращении к памяти.

Программа, выполняемая в одном режиме, не может использовать для обращения к памяти PAR/PDR другого режима. Такое разделение используется для защиты памяти в системах с разделением времени. Каждый регистр имеет свой адрес канала.

В табл. 9 приведены номер и адрес каждого регистра в обоих наборах.

Необходимо отметить, что периферийные устройства не имеют доступа к PAR/PDR .

ШИЗ.480.167 ТО

Лист

169

Ш.№.№.подл. 90394
Ш.№.№.подл. 10.07.8708
ВЗАМ.Ш.№.№.подл. 10.07.8708
Ш.№.№.подл. 10.07.8708

Изд. Лист 1/докум. Подп. Дата

Формат 50 по ГОСТ 2.106-68

Формат А4

$SRO(6,5)$ – режим процессора (пользователя или ОС), при обращении в котором произошла ошибка ДП.

$SRO(3-1)$ – номер страницы, при обращении к которой произошла ошибка ДП.

EM – бит включения ДП. Когда $EM = 1$ выполняется релокация адресов и защита памяти. После возникновения ошибки доступа содержимое SRO "замораживается". Биты M и EM устанавливаются программно. Запись информации в другие биты SRO лишена смысла: интерес представляет лишь аппаратная установка этих битов. Так, например, программная запись единицы в разряды $SRO(13-15)$ не вызывает прерывания. Однако, после обработки прерывания необходимо сбрасывать эти биты в нуль, чтобы ДП отслеживал новые ошибки.

Прерывание по ошибке ДП при обращении в память с использованием в качестве адреса РС возникает лишь при попытке приема соответствующего командного слова через входной регистр RD операционного блока. В случае предварительной выборки с ошибкой ДП первого слова команды чтение этого слова повторяется заново через RVA .

Если при обращении в память в качестве адреса используется RVA и возникла ошибка ДП, то она вызывает прерывание сразу же после начала выдачи адреса в системную магистраль. В случае, если в этом же обращении возникла ошибка нечетного адреса, прерывание от ДП не возникает. Адрес $SRO - 17777572$.

В регистре SRO хранится виртуальный адрес первого слова выполняемой команды. В случае ошибки при чтении команды содержимое $SR2$ не обновляется. Регистр $SR2$ доступен лишь по чтению, попытка записи не изменяет его содержимого. При возникновении ошибки ДП и установки в SRO битов $15,14$

Идентификационная таблица:
Идентификационный номер докум. № 0001. Подп. и дата.
90394 19.07.87034

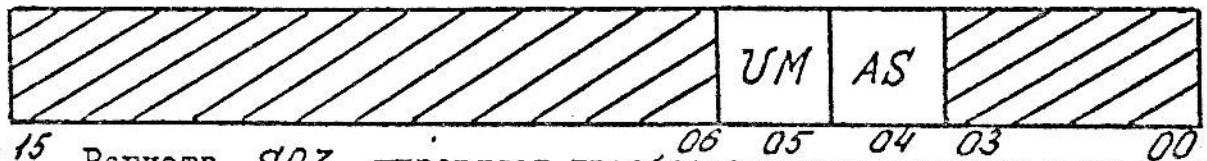
Изм. лист № докум. Подп. Дата

ИИЗ.480.167 ТО

Лист 143

или I3 содержимое *SR2* "замораживается" до тех пор, пока эти биты не будут обнулены. Адрес *SR2* - $I7777576_8$.

Кроме этих двух регистров в ДП существует двухразрядный регистр *SR3* с адресом - $I7772543_8$



Регистр *SR3* управляет преобразованием виртуального адреса в двадцатидвух или восемнадцатиразрядный, а также преобразованием адресов *UNIBUS*. Если *AS* равен "0" и разрешена работа ДП (нулевой бит *SRO* равен "1"), процессор использует восемнадцатиразрядное преобразование. Если *AS* равен "1", и бит 0 регистра *SRO* равен "1", то процессор использует двадцатидвухразрядное преобразование. Если *UM* регистра *SR3* равен "1" релокация адресов *UNIBUS* разрешена, если равен "0" - запрещена.

9.2.8. Определение физического адреса

Шестнадцатиразрядный виртуальный адрес от 0 до $I77776_8$. Три старших разряда виртуального адреса задают номер страницы, т.е. конкретную пару *PAR/PDR*, с которой необходимо обращаться при перераспределении адресов страницы. В табл. 10 приведены области виртуального адреса, соответствующие им номера страниц и *PAR/PDR*.

Таблица 10

Область виртуального адреса	Номер страницы и <i>PAR/PDR</i>
000000-017776	0
020000-037776	1
040000-057776	2

Инв. № подл.	90394
Взам. инв. №	
Инв. № докум.	
Подп. и дата	10.07.87
Подп. и дата	

10. ПЕРЕРЫВАНИЕ ПРЦ

При возникновении условий прерывания из-за наличия внешних прерываний, ошибок магистрали, ошибок диспетчера памяти, а также при выполнении инструкций прерывания ПРЦ переходит к микропрограмме обработки прерывания, включающей сохранение *PSW*, счетчика команд и загрузки вектора прерывающего процесса. Дальнейшая обработка ведется на программном уровне.

В ПЛМ формирования адреса вектора прерывания вырабатывается адрес, соответствующий условию прерывания и из блока констант микропрограммно читается сам вектор. ПРЦ помещает в стек РС и *PSW*, считывает новое значение РС из ячейки, определяемой вектором прерывания. Новое *PSW* берется из ячейки, следующей за ячейкой, где находился вектор.

Условиями, вызывающими прерывание ПРЦ могут быть следующие условия:

1) ошибки магистрали;

Ошибка магистрали указывает на то, что ПРЦ сделал попытку обратиться по несуществующему адресу памяти или по нечетному адресу (при небайтовом обмене) или сигнал *RP4Y* не был получен в течение *I28T*

2) переполнение стека;

Переполнение стека вызывается любой попыткой процесса декрементировать содержимое указателя стека (*RI6*) до значения менее 400. Это не относится к указателю стека пользователя (*RI6*). В *RI6* переполнения стека не бывает.

3) выключение питания;

Временная диаграмма сигналов ПРЦ и питающих напряжений приведена на рис. 14

ИИЗ.480.167 ТО

Лист

146

ИИЗ.480.167 ТО

Формат 5а ГОСТ 2.106-68

СОРМАТ ИИ

ИИЗ. № 1000 ПОСП. И ДАТА
90394 10.07.80
ИИЗ. № 1000 ПОСП. И ДАТА
90394 10.07.80
ИИЗ. № 1000 ПОСП. И ДАТА
90394 10.07.80

При снижении питания за пределы допустимого уровня, аппаратура блока питания системы, в которую входит ПРЦ, должна обеспечить приведенную последовательность возникновения сигналов АСЛО и DСЛО. При переходе сигнала АСЛО их высокого уровня в низкий ПРЦ, после выполнения очередной команды, прерывается и переходит к микропрограмме обработки прерывания и затем, к программе обработки сбоя питания. Работе этой программы отведено время до появления низкого уровня сигнала DСЛО

4) прерывание по T-биту;

Это прерывание, программно управляемое пользователем, позволяет ему вставлять интерактивные подпрограммы в основную программу. Используется для отладки программ. Установка и очистка T-бита возможна лишь при выполнении команд RTT, RTI, при загрузке вектора прерывания. Если T-бит устанавливается при загрузке вектора прерывания, то прерывание по T-биту произойдет до выполнения первой команды нового процесса. Команда с T-битом равным "1" (прослеживаемая команда) будет выполняться до конца и затем произойдет прерывание. Далее описаны случаи прерывания по T-биту в зависимости от прослеживаемой команды.

- Прослеживаемая команда RTT очищает T-бит. После выполнения команды произойдет прерывание по T-биту;
- Прослеживаемая команда RTI считает T-бит. Прерывание не произойдет;
- Прослеживаемая команда RTT устанавливает T-бит. Прерывание произойдет после выполнения следующей за RTT командой;
- Прослеживаемая команда RTI устанавливает T-бит. После выполнения команды произойдет прерывание по T-биту;
- Прослеживаемая команда - команда прерывания - очищает T-бит. Прерывание не произойдет.

№ докум. 90394
 Дата 10.07.87
 Подп. и дата 46306

№ докум. 90394
 Дата 10.07.87
 Подп. и дата 46306

ШИЗ.480.167 ТО

Лист 437

— Прослеживаемая команда - команда прерывания - устанавливает T-бит. После выполнения команды произойдет прерывание по T-биту.

— Прослеживаемая команда - WAIT. Произойдет прерывание по T-биту.

5) резервные команды, запрещенные команды, команды прерывания - BMT, TRAP, BPT, IOT.

Резервные команды - команды, коды которых в данном ПРЦ не используются. Запрещенные команды - команды *JMP* и *JSR* с регистровой адресацией.

6) ошибки диспетчера памяти;

7) внешние прерывания.

Этими прерываниями могут быть прерывания, вызванные возникновением сигналов HALT, FPPTRAP, IRQ (3:0), *EVNT*

Прерывание текущего процесса в ПРЦ обычно допускается в конце выполнения очередной команды. Лишь ошибки магистрали, диспетчера памяти и ошибка ШЗ - процессора могут прерывать работу ПРЦ в любой момент.

ШЗ № 000. Подп. и дата. 330 М. ШЗ № 000. Подп. и дата. 10.07.8703. 90304

ШЗ.480.167 ТО

Лист 148

ШЗ № 000. Подп. и дата. 330 М. ШЗ № 000. Подп. и дата.

II. РАЗЛИЧИЯ В РАБОТЕ МИКРОСХЕМ КМ1801ВМ2 И КМ1801ВМ3

Ситуация	КМ1801ВМ2	КМ1801ВМ3
1. Загружен T-бит равный "1" в цикле ожидания команды	Ожидает внешнего прерывания	Возникает прерывание по T-биту
2. При выполнении команды <i>RTT</i> устанавливается T-бит, равный "1" и присутствует незамаскированное внешнее прерывание	Следующая за <i>RTT</i> команда выполняется	Возникает прерывание
3. Возможность доступа к <i>PSW</i> по адресу <i>I77776</i>	Нет	Есть
4. Число уровней прерывания	Один	Четыре
5. Переполнение стека	Нет	Возникает прерывание (граница 400)
6. Ошибка нечетного адреса при адресации к слову памяти	Нет	Есть
7. Регистр (в том числе R7) содержит адрес несуществующей памяти и произошла ошибка магистрали при обращении с использованием этого регистра в моде 2	Значение регистра инкрементировано	Значение регистра не изменяется
8. Диспетчер памяти	Отсутствует	Есть

Инв. № подл.	Подл. и дата	Взам. инв. №	Инв. № докл.	Подл. и дата
90394	10.07.8702			

Узм. лист	№ докум.	Подл.	Дата

ИИЗ. 480.167 Т0

Ситуация	KM1801BM2	KM1801BM3
9. Обращение к PSW в командах MTPS, MPFS	Без использования адресации PSW по адресу I77776	С использованием адресации PSW по адресу I77776
10. Разряды PSW (I5-I2)	Не используются	Хранятся текущая и предыдущая моды работы ДП
11. Отсутствие сигнала <u>SACK</u> в течение I28T после возникновения сигнала <u>DMG</u>	Продолжает ожидание сигнала <u>SACK</u>	Вызывает сброс сигнала <u>DMG</u>
12. Отсутствие сигнала <u>SSYNC</u> в течение I28T после начала выдачи адреса на системную магистраль	Продолжает ожидание сигнала <u>SSYNC</u>	Вызывает окончание выдачи адреса и продолжение этого цикла обмена

ШИЗ. № 703. Подп. и дата. 90394 10.07.87гв.
 Возм. ШИЗ. № 703. Подп. и дата.
 ШИЗ. № 703. Подп. и дата.

ШИЗ. № 703. Подп. и дата. 90394 10.07.87гв.
 ШИЗ. № 703. Подп. и дата.

ШИЗ.480.167 ТО

ШИЗ
 180

Сокращения

Сокращение, условное обозначение	Полное наименование
АВП	Адрес вектора прерывания
АЛУ	Арифметико-логическое устройство
БМУ	Блок микропрограммного управления
БК	Блок констант
БПР	Блок прерываний
БС	Блок синхронизации
БУРЕГ	Блок управления внутренними регистрами
ВхР	Входной регистр
ВыхР	Выходной регистр
ДП	Диспетчер памяти
КПДП	Контроллер прямого доступа в память
КОП	Код операции
КСМ	Контроллер системной магистрали
МК	Микрокоманда
ОБ	Операционный блок
ОЗУ	Оперативное запоминающее устройство
ОС	Операционная система
ПЛМ	Программируемая логическая матрица
ПЛМ ОСН	Основная ПЛМ
ПЛМ ПРД	ПЛМ предварительного разбора
ПЛМ АС	ПЛМ анализа состояния

ШИЗ № подл. Подп. и дата. Взам. инв. № дубл. Подп. и дата.
 90394 10.07.87 г.

ШИЗ.480.167 ТО

Лист
181

Сокращение, условное обозначение	Полное наименование
VA	Виртуальный адрес
RD	Регистр данных
RVA	Регистр виртуального адреса
SP	Указатель стека
SRO-SR3	Регистры состояния ДП
N	Признак отрицательного результата
Z	Признак равенства нулю
V	Признак арифметического переполнения
C	Признак расширения

ЦИБ. № подл. Подп. и дата. Взам. инв. №. Инв. № докл. Подп. и дата.
 90394 10.07.87 в.м.

ЦИБ. Лист. № докум. Подп. Дата.

ИИЗ.480.167 ТО

Лист

183

СИСТЕМА КОМАНД

КОМАНДА		Наименование команды
МНЕМОНИКА	КОД	
1. HALT	000000	Останов
2. WAIT	000001	Ожидание
3. RTT	000002	Возврат из прерывания
4. BPT	000003	Командное прерывание для отладки
5. IOT	000004	Командное прерывание для ввода/вывода
6. RESET	000005	Сброс внешних устройств
7. RTT	000006	Возврат из прерывания
8. JMP	0001DD	Безусловный переход
9. RTS	00020R	Возврат из подпрограммы
10. JSR	004RBD	Обращение к подпрограмме
11. EMT	I04000 I04377	Командное прерывание для системных программ
12. TRAP	I04400 I04777	Командное прерывание
13. NOP	000240	Нет операции
14. CLC	000241	Очистка C
15. CLV	000242	Очистка V
16. CLZ	000244	Очистка Z
17. CLN	000250	Очистка N
18. SEC	000261	Установка C
19. SEV	000262	Установка V

УИВ № 7003 Подп. и дата. 10.07.87
 УИВ № 7003 Подп. и дата. 10.07.87
 УИВ № 7003 Подп. и дата. 10.07.87

УИВ № 7003 Подп. и дата. 10.07.87

ШИЗ.480.167 Т0

ИУСТ
184

КОМАНДА		Наименование команды
МНЕМОНИКА	КОД	
20. SEZ	000264	Установка Z
21. SEN	000270	Установка N
22. SCC	000277	Установка всех разрядов (N, Z, V, C)
23. CCC	000257	Очистка всех разрядов (N, Z, V, C)
24. SWAB	0003DD	Перестановка байтов
25. CLR(B)	*050DD	Очистка
26. COM(B)	*051DD	Инвертирование
27. INC(B)	*052DD	Прибавление единицы
28. DEC(B)	*053DD	Вычитание единицы
29. NEG(B)	*054DD	Изменение знака
30. ADC(B)	*055DD	Прибавление переноса
31. SBC(B)	*056DD	Вычитание переноса
32. TST(B)	*057DD	Проверка
33. ROR(B)	*060DD	Циклический сдвиг вправо
34. ROL(B)	*061DD	Циклический сдвиг влево
35. ASR(B)	*062DD	Арифметический сдвиг вправо
36. ASL(B)	*063DD	Арифметический сдвиг влево
37. MAPK	0064NN	Восстановление SP
38. SXT	0067DD	Расширение знака
39. MTPS	I064SS	Запись PSW
40. MFPS	I067DD	Чтение PSW
41. MOV(B)	*I SSDD	Пересылка
42. CMP(B)	*2 SSDD	Сравнение
43. BIT(B)	*3 SSDD	Проверка разрядов
44. BIC(B)	*4 SSDD	Очистка разрядов

Инв. № подл.	Подп. и дата	Взят. инв. №	Инв. № дубл.	Подп. и дата
90394	19.07.87	8708		

ШИЗ.480.167 TO

Лист

185

КОМАНДА		Наименование команды
МНЕМОНИКА	КОД	
45. <i>BIS (B)</i>	ж5 SDD	Логическое сложение
46. <i>XOR</i>	074RDD	Исключающее ИЛИ
47. <i>ADD</i>	06SSDD	Сложение
48. <i>SUB</i>	I6SSDD	Вычитание
49. <i>BR</i>	0004XX	Ветвление безусловное
50. <i>BNE</i>	0010XX	Ветвление, если не равно (нулю)
51. <i>BEQ</i>	0014XX	Ветвление, если равно (нулю)
52. <i>BGE</i>	0020XX	Ветвление, если больше или равно (нулю)
53. <i>BGT</i>	0024XX	Ветвление, если меньше (нуля)
54. <i>BGT</i>	0030XX	Ветвление, если больше (нуля)
55. <i>BLE</i>	0034XX	Ветвление, если меньше или равно (нулю)
56. <i>SOB</i>	077RNN	Вычитание единицы и ветвление
57. <i>BPL</i>	I000XX	Ветвление, если плюс
58. <i>BMI</i>	I004XX	Ветвление, если минус
59. <i>BHI</i>	I010XX	Ветвление, если больше
60. <i>BLOS</i>	I014XX	Ветвление, если меньше или равно
61. <i>BVC</i>	I020XX	Ветвление, если нет арифметического переполнения
62. <i>BVS</i>	I024XX	Ветвление, если арифметическое переполнение
63. <i>BHIS, BCC</i>	I030XX	Ветвление, если больше или равно
64. <i>BLO, BCS</i>	I034XX	Ветвление, если меньше
65. <i>MUL</i>	070RSS	Умножение

№ п. по подл. 92394
 Дата 10.07.87
 Изм. № 1
 № докум. 1007870
 Дата
 Подл. и дата

ИИЗ.480.167 ТО

Лист 186

КОМАНДА		Наименование команды
МНЕМОНИКА	КОД	
66. DIV	071RSS	Деление
67. ASH	072RSS	Сдвиг на " N " разрядов одного слова
68. ASHC	073RSS	Сдвиг на " N " разрядов двойного слова
69. MFPI	1065SS	Засылка слова (D - данные, I - ин - струкция) в стек текущей моды по адре - су предварительной моды
70. MFPI	0065SS	
71. MTPD	1066 DD	Засылка слова из стека текущей моды по адресу предварительной моды
72. MTPD	0066 DD	

Учб. № подл.	Подп. и дата	Узач. Учб. №	Учб. № докум.	Подп.	Дата
90394	10.07.87 е.в.м.				

ИИС.480.167 ТО

Лист
104

Лист регистрации изменений

№ п/п	Номера листов (страниц)				Всего листов (страниц) в докум.	№ докум	Входящий № сопроводительного документа и дата	Подп.	Дата
	Измнен. лист	Заменен лист	новы	изъяты					
1	Т.Л. Л. 2	3, 4, 5, 6, 7, 11, 12, 13, 14, 15	Л. 5а, 6а, 7а			ЦУИ 1757-87		Кув	9.10.88
2	9, 10, 16, 17, 18, 19, 20, 25, 181	12, 3, 4, 5, 5а, 6, 6а, 7, 7а, 8, 11, 12, 13, 14, 15	68			ЦУИ 502-88		Син	28.9.88

№ п/п
 Подп. и дата
 90394
 10.07.87
 10.07.87
 10.07.87

ИИЗ.480.167 ТО

Лист
188