

ЭЛЕКТРОНИКА

**КОНСТРУКТОР
РАДИОЛЮБИТЕЛЯ**

КРО2



**РУКОВОДСТВО
ПО
ЭКСПЛУАТАЦИИ**

Десять (десять и одиннадцать)
 (десять и одиннадцать)
 (десять и одиннадцать)

0100H-01FFH	9B47	37E3
0200H-02FFH	602E	26F1
0300H-03FFH	35FD	B570
0400H-04FFH		1242

РАДИОКОНСТРУКТОР

ЭЛЕКТРОНИКА

КР-02

РУКОВОДСТВО
ПО
ЭКСПЛУАТАЦИИ

1989

СОДЕРЖАНИЕ

1. Общие указания	стр. 2
2. Комплектность	2
3. Основные технические характеристики	3
4. Указания по технике безопасности	5
5. Краткое описание радиоконструктора	5
6. Описание аппаратной части	9
7. Описание программного обеспечения	15
8. Подготовка к работе и порядок работы	59
9. Техническое обслуживание изделия	65
10. Гарантии изготовителя	66
11. Перечень принятых сокращений	68
12. Список дополнительной литературы	69

Приложения: 1. Описание игровых программ

2. Схема электрическая принципиальная
3. Схема электрическая принципиальная блока процессора
4. Схема электрическая принципиальная блока питания
5. Схема электрическая принципиальная блока клавиатуры
6. Блок клавиатуры "Электроника МС7007"

I. ОБЩИЕ УКАЗАНИЯ

При покупке радиоконструктора "Электроника КР-02", (в дальнейшем радиоконструктор), проверьте его комплектность. Убедитесь, что в гарантийном талоне на радиоконструктор проставлены: штамп магазина, разборчивая подпись или штамп продавца и дата продажи.

ВНИМАНИЕ! ПРЕЖДЕ, ЧЕМ ПРИСТУПИТЬ К СБОРКЕ И ЭКСПЛУАТАЦИИ РАДИОКОНСТРУКТОРА, НЕОБХОДИМО ОЗНАКОМИТЬСЯ С НАСТОЯЩИМ РУКОВОДСТВОМ.

Радиоконструктор предназначен для эксплуатации при температуре воздуха от 10° до 35° , влажности до 80%. Не допускается наличие в воздухе паров кислот, щелочей и других агрессивных примесей.

2. КОМПЛЕКТНОСТЬ

В комплект поставки входят:

системный блок	1 шт.
блок питания	1 шт.
межблочный соединительный жгут	1 шт.
магнитофонная кассета МК-60	
с записями программ	1 шт.
руководство по эксплуатации	1 экз.
упаковка	1 шт.
запасная вставка плавкая на 1 А.	1 шт.

3. ОСНОВНЫЕ ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ

- 3.1. Система команд - для микропроцессора серии KP560.
- 3.2. Количество команд - 76.
- 3.3. Тактовая частота - 1,77 МГц.
- 3.4. Разрядность шины данных - 8 бит.
- 3.5. Разрядность шины адреса - 16 бит.
- 3.6. Объем оперативной памяти - 16 Кбайт.
- 3.7. Количество восьмиразрядных портов для связи с внешними устройствами - 3.
- 3.8. Амплитуда сигнала на видеовыходе - 1В.
- 3.9. В качестве внешнего накопителя используется бытовой магнитофон.
- 3.10. Амплитуда сигнала для записи на магнитофон - 1В.
- 3.11. Чувствительность магнитофонного входа - 0,25В.
- 3.12. Формат изображения при выводе на экран телевизионного приемника:
 - алфавитно-цифровой 64x25 знаков
 - псевдографический 126x50 точек
- 3.14. Питание радиоконструктора осуществляется от сети 220В 50 Гц.
- 3.15. Потребляемая мощность не более 40 Вт.

3.16. Масса и габариты блоков приведены в табл. I.

Таблица I

Наименование блока	Габариты, не более, мм	Масса, не более, г
Системный блок	420x242x52	2300
Блок питания	286x100x115	3000
Жгут		260

3.17. Содержание драгоценных металлов:

золото - 0,0261975 г.

серебро - 0,2916956 г.

4. УКАЗАНИЯ ПО ТЕХНИКЕ БЕЗОПАСНОСТИ

В радиоконструкторе используется опасное для жизни сетевое напряжение 220В, поэтому при сборке и эксплуатации радиоконструктора необходимо соблюдать осторожность. Собирать радиоконструктор можно при выключенном напряжении питания. Запрещается эксплуатировать блок питания со снятым кожухом. Запрещается использовать в блоке питания самодельные предохранители, рассчитанные на другие токи. Во избежание перегрева блока питания запрещается вставлять жалон в блок питания посторонними предметами.

Телевизор и магнитофон необходимо эксплуатировать в строгом соответствии с их инструкцией по эксплуатации.

5. КРАТКОЕ ОПИСАНИЕ РАДИОКОНСТРУКТОРА

5.1. Введение

Радиоконструктор предназначен для подготовленных радиолюбителей, имеющих навыки работы с цифровыми схемами. Радиоконструктор служит для сборки в бытовых условиях радиолюбительского персонального компьютера. Радиоконструктор ориентирован на пользователей, не имеющих навыков работы с IBM. Радиоконструктор может применяться как в домашних условиях, так и в сферах производства и обучения. Схема радиоконструктора разработана коллективом авторов МИЭМ под руководством Г. Зеленио (см. статьи в журнале РАДИО № 4-12 за 1986 г.).

В радиоконструкторе не полностью использованы возможности микропроцессора КР580ВМ80А, поэтому радиоконструктор может быть усовершенствован силами пользователя.

5.2. Структурная схема радиоконструктора

Структурная схема радиоконструктора представлена на рис. 1. "Сердцем" ее является микропроцессор КР580ВМ80А. Обмен информацией между процессором и другими устройствами ведется по шине данных.

СТРУКТУРАЛЬНАЯ СХЕМА РАБОЧЕГО УСТРОЙСТВА

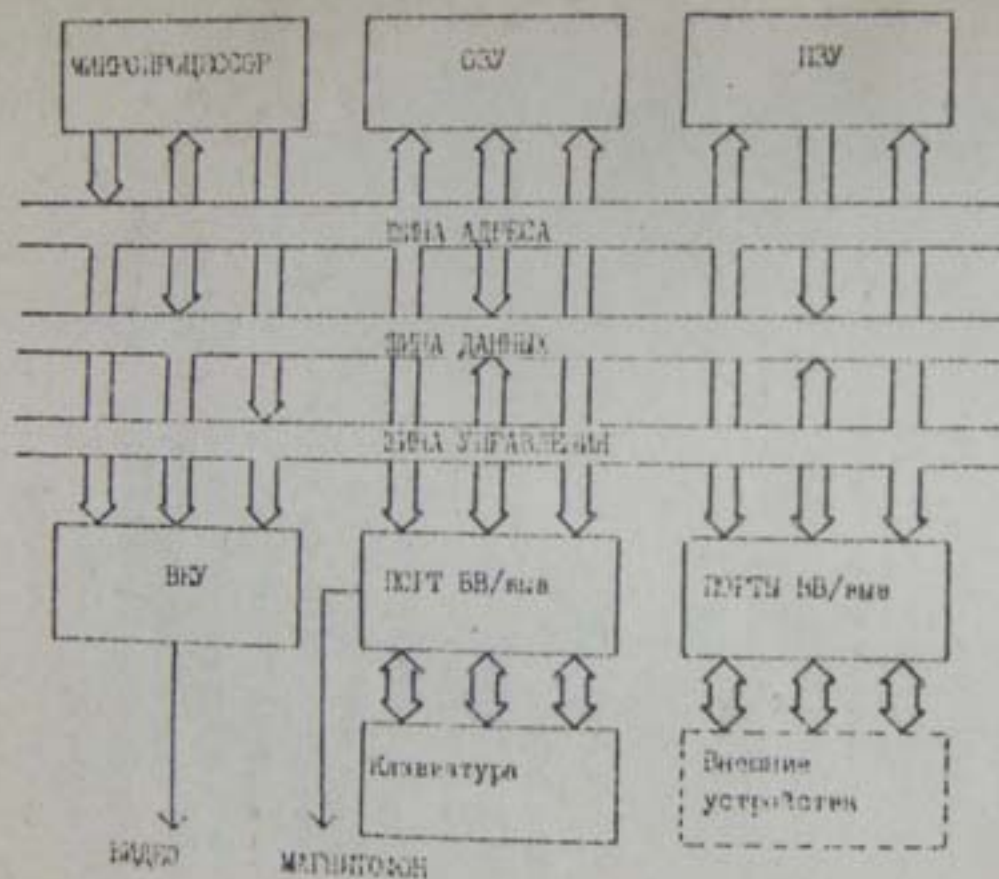


Рис. 1

Информация на шине управления определяет режим и направление передачи данных. Информация на шине адреса определяет устройство, с которым ведется обмен данными.

Оперативное запоминающее устройство предназначено для хранения данных и программ.

В постоянном запоминающем устройстве хранится управляющая программа МОНИТОР. В эту программу входит набор стандартных подпрограмм и интерпретатор приказов.

Программа МОНИТОР осуществляет на первый запуск компьютера после включения питания. Набор приказов МОНИТОРА позволяет производить ввод-вывод программ, записанных на магнитной ленте, запуск программ с указанного адреса, отладку и корректировку программ.

Контроллер дисплея осуществляет вывод информации на экран телевизионного приемника.

Порты ввода-вывода предназначены для обмена информацией с внешними устройствами. Таковыми внешними устройствами являются клавиатура и магнитофон. Кроме того, радиоконструктор имеет три свободных восьмиразрядных шины ввода-вывода-свободные порты. С помощью этих шин можно подключить внешние устройства с цифровым управлением, например, устройство печати или внешнее ЦКУ.

Работу устройства по структурной схеме можно описать следующим образом: микропроцессор выставляет на шине адреса адрес ячейки, в котором находится очередная команда (ОЗУ и ЦКУ). Каждому устройству в микропроцессорной системе соответствует свой адрес или совокупность адресов. Совокупность всех адресов называется адресным пространством. Шестнадцатиразрядная шина адреса дает возможность адресовать $64K (K=1024)$ ячеек. Распределение адресного пространства микропроцессора приведено в табл. 2. Одновременно с адресом на шине управления появляется информация о том, что

ТАБЛИЦА 2

РАСПРЕДЕЛЕНИЕ АДРЕСНОГО ПРОСТРАНСТВА КОМПЬЮТЕРА

	FFFFH
ПЗУ МОНИТОР (только чтение)	F800H
Регистры КР580ИК57 (только запись)	E000H
Регистры КР580ВГ75	C000H
Регистры КР580ИК55 (Д 14)	A000H
Регистры КР580ИК55 (Д 20)	8000H
НЕ ИСПОЛЬЗУЕТСЯ	4000H
Экранная область ОЗУ	3600H
Рабочие ячейки МОНИТОРА	3600H
ОЗУ Пользователя	0000H

производится ввод данных в микропроцессор. После этого информация (код команды), которая находится по указанному адресу, выставляется на шине данных. Микропроцессор анализирует команду и выполняет её. Аналогичным образом ведется обмен информацией с портами ввода - вывода, так как микропроцессор обращается к ним так же, как к ячейке памяти.

Для того, чтобы изображение постоянно присутствовало на экране телевизора, контроллер дисплея периодически прерывает работу процессора, переводя его в режим ожидания, считывает информацию из экранной области ОЗУ и выводит её на экран. Такой способ обмена информацией, минуя процессор, называется прямым доступом к памяти.

Шина адреса, шина данных и шина управления являются обязательными в любом устройстве, выполненном в микропроцессоре КР580ВМ204 и аналогичных ему. Все внешние устройства включаются в систему с помощью этих трех шин.

6. ОПИСАНИЕ АППАРАТНОЙ ЧАСТИ

6.1. Общие сведения

Радиоконструктор состоит из трех функционально законченных блоков: блок процессора, блок клавиатуры и блок питания, соединенных между собой шнуром (см. приложение 2).

6.2. Блок процессора

Принципиальная электрическая схема блока приведена в приложении 3. Центральным элементом в ней является микропроцессор КР580ВМ204, который управляет работой всех остальных периферийных БИС. Тактовый генератор микро3ВМ выполнен на микросхеме ДД1 (КР580Г424), которая предназначена специально для работы с микропроцессором.

Частота тактовых импульсов определяется кварцевым резонатором. Тактовый генератор формирует импульсы для синхронизации микропроцессора на выходах $\Phi 1$ и $\Phi 2$ амплитудой 12В. На выходе $\Phi 2$ TTL формируются сигналы для синхронизации периферийных БИС.

Вход $\overline{RES\bar{IN}}$ тактового генератора используется для формирования импульса $\overline{CEP\bar{OC}}$. Элементы R1, R2, C1, C2, VD1 используются в схеме ограничения длительности импульса $\overline{CEP\bar{OC}}$. Это необходимо из-за того, что в схеме применена память динамического типа, которая требует периодической регенерации. Во время действия сигнала $\overline{CEP\bar{OC}}$ процесс регенерации прекращается и, если не принять мер по ограничению длительности импульса до 2 мс, то информация, хранящаяся в ОЗУ, будет утрачена.

Как уже говорилось в предыдущем разделе, каждому устройству в микроЭВМ соответствует свой адрес или группа адресов. Для того, чтобы выделить устройство, с которым в данный момент работает процессор, служит дешифратор адресов DD11(K555ИД7).

На входы дешифратора подаются четыре старших разряда шины адреса, а на одном из выходов появляется сигнал логического "0", позволяющий выделить БИС, к которой производится обращение. Таким образом, все адресное пространство разбито на 8 блоков по 8 Кбайт каждый. На выходах элементов DD6.3(KI55ЛБ5) и DD10.4(KI55ЛАЗ) формируется сигнал "логическая 1" при обращении к ячейкам ОЗУ с адресами в диапазоне 0-3FFFH и 4000H-7FFFH соответственно. Сигналы с выхода 4-7 микросхем DD11(K555ИД7) используются для выбора периферийных БИС.

Оперативное запоминающее устройство выполнено на микросхемах памяти динамического типа DD21-DD28. Особенностью этих микросхем является временное мультиплексирование адресов. Код адреса заносится в адресный регистр микросхемы через входы A₆ - A₀ сначала семи младших адресов, а затем семи старших адресов. Выборка

младших и старших разрядов производится сигналами RAS и CAS соответственно. Коммутация адресов осуществляется мультиплексорами DD 18, DD 19 (K555KIII1). Временные соотношения, необходимые для нормальной работы микросхем ОЗУ, обеспечиваются микросхемой DD 16 (K555PI1). Объем ОЗУ может быть расширен до 32 Кбайт. Для этого к схеме необходимо ввести 8 дополнительных микросхем памяти. Выводы этих микросхем должны быть соединены с одноименными выводами микросхем DD 21- DD 28. Исключение составляют выводы CA S дополнительных микросхем, которые необходимо соединить между собой и подключить к выходу элемента DD 10.3 (K155ЛА3). Наиболее просто напаять дополнительные корпуса ОЗУ сверху на микросхемы памяти, установленные на печатной плате. Такое изменение требует изменения программы МОНИТОР, записанной в ПЗУ DD 17 (K573P45). Информацию об изменении можно найти в табл. 3. Необходимо заметить, что при установке дополнительных микросхем ОЗУ, возможно, снизится устойчивость работы компьютера. Для устранения этого явления необходимо будет установить дополнительные развязывающие конденсаторы на шинах питания, а также проложить дополнительные провода параллельно шинам питания. Следует заметить, что отыскание причин сбоев в компьютере - задача сложная, поэтому мы не рекомендуем заниматься расширением памяти радиолюбителям, не имеющим достаточного опыта работы с цифровыми схемами.

ВНИМАНИЕ! НА РАДИОКОНСТРУКТОРЫ С ВНЕСЕННЫМИ ИЗМЕНЕНИЯМИ ГАРАНТИЙНЫЕ ОБЯЗАТЕЛЬСТВА ЗАВОДА НЕ РАСПРОСТРАНЯЮТСЯ!

Формирование видеосигнала для телевизора осуществляется контроллером дисплея, построенном на основе ИМС DD 8 (KPS800BT75). Отображаемая информация передается из экранной области ОЗУ методом прямого доступа к памяти (ПДП). Изображение на экране телевизора формируется путем засветки отдельных точек телевизионного раstra.

Именения для 32Кбайтной версии МОНИТОРА

Во всех перечисленных ячейках код 36H необходимо
заменить на код 76H.

F83D	F92E	FA89	FCC7	FE70
F843	F931	FA90	FCCB	FE74
F846	F939	FAF1	FOCE	FF3C
F84E	F93F	F834	FCEC	FF54
F851	F942	F8A0	F0B7	FF50
F860	F94B	F8D6	F0BB	FFA4
F866	F94E	F0FB	FE03	FFA9
F86B	F955	FC0B	FE09	FFAE
F86E	FA4B	FC15	FE16	FFB5
F885	FA4C	FC2B	FE1C	FFB9
F897	FA5C	FC2F	FE20	FFBE
F89C	FA64	FC36	FE34	FFC5
F8A0	FA67	FC64	FE41	FFCE
F8F0	FA7D	FC74	FE4A	FFDB
F91E	FA82	FC8C	FE6D	

В следующих ячейках следует изменить:

F85D	:	35H	на	75H
F051	:	37H	на	77H
F054	:	38H	на	78H
F0A5	:	3FH	на	7FH
F0B7	:	37H	на	77H

Информация о засвеченных точках записана в ИЗУ знакогенератора DD 12 (K573P41). Модуляция луча кинескопа осуществляется по сигналам с выхода сдвигового регистра DD 15, (K155ИР13), куда заносится в параллельном коде с микросхемы DD 12.

Контроллер дисплея вырабатывает также строчные и кадровые синхронизирующие импульсы, которые смещаются на элементе DD 5.2 (K155ЛН5) и подаются на модулятор, собранный на транзисторе VT1. С резистора R19 снимается полный видеосигнал. Номинал резистора R19 можно изменять в пределах 27+100 Ом для получения наилучшего качества изображения.

Многоразрядный буферный регистр DD 7 (K589ИР12) работает совместно с контроллером ПДП и предназначен для хранения восьми старших разрядов кода адреса. Это необходимо потому, что в контроллере ПДП выходы D 0-D7 используются в мультиплексорном режиме как для приёма информации с шины данных в режиме настройки, так и для выдачи на адресную шину старших разрядов кода адреса в режиме ПДП.

Микросхемы DD 14 и DD 20 (K560BB55) являются портами ввода-вывода, причем микросхема DD 14 предназначена для связи с устройствами пользователя, микросхема DD 20 обслуживает клавиатуру и магнитофон.

Управляющая программа МОНИТОР записана в микросхеме ИЗУ DD 17 (K573P45). Программа начинает работать после нажатия кнопки СБРОС. Так как после сигнала СБРОС процессор считывает команду, находящуюся по адресу 0000H, а программа МОНИТОР расположена по адресу F800H, в схеме предусмотрен узел начального запуска, собранный на элементах DD 10.1 (K155ЛЛ3), DD 9.2 (K155ЛН1) и DD 13.2 (K155ТМ2).

Для четкого распознавания сигналов с выхода магнитофона в схеме предусмотрен компаратор D АЦКР140УД8).

6.3. Блок клавиатуры

Схема блока клавиатуры приведена в приложении 5, расположение клавиш в приложении 6.

Клавиатура подключается к плате процессора гибким плоским шлейфом.

Основные клавиши составляют матрицу 8×11 нормально разомкнутых контактов, замыкающихся при нажатии на кнопки клавиатуры и подключенных к портам А и В микросхемы DD 20.

Расположение клавиш соответствует принятому в большинстве промышленных дисплеев. Опрос матрицы и устранениедребезга контактов производится программным путём. Кроме основной матрицы клавиатура содержит клавишу СЕРОС, подключенную к микросхеме DD 1.

6.4. Блок питания (Тр-р: ТПП 260-220-50)

Схема блока питания радиоконструктора приведена в приложении 4. Блок питания обеспечивает стабилизированные напряжения питания $(5 \pm 0,15)В$ при токе $1,2А$; $(12 \pm 1)В$ при токе $0,25А$ и минус $(5 \pm 0,25)В$ при токе $0,12А$. Блок питания состоит из сетевого трансформатора, трех независимых стабилизаторов напряжения и узла защиты.

Стабилизаторы собраны на основе интегральных стабилизаторов DA 1 - DA 3. Точные значения напряжений устанавливаются резисторами R20, R23, R26.

Токи срабатывания защиты устанавливают резисторами R10, R12, R14.

В блоке питания переменен способ контроля питающих напряжений непосредственно на плате процессора. Для этого в плату введен дополнительно три провода обратной связи.

В блоке питания предусмотрен узел защиты, собранный на эле-

ментах VT5, DD1, VT6, который снимает все питающие напряжения при срабатывании защиты в любом из плеч стабилизатора. Защита срабатывает также при обрыве любого из проводов питания.

Для повторного пуска блока питания необходимо нажать на кнопку ПУСК. При этом на выходах блока питания одновременно появятся питающие напряжения.

Блок питания поставляется в собранном, отлаженном виде. Блок опломбирован. В случае нарушения сохранности пломбы претензии по качеству не принимаются.

6.5. Жгут

Схема жгута приведена в приложении 2

7. ОПИСАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

7.1. Общие сведения

В комплект поставляемого программного обеспечения входят программы "СИСТЕМНЫЙ МОНИТОР", "ИНТЕРПРЕТАТОР ЯЗЫКА БЕЙСИК", ТЕСТ-программа, игровые программы.

7.2. Краткие сведения о системе команд KP580BM90A

Система команд микропроцессора определяется его архитектурой. Структурная схема микропроцессора представлена на рис.2. Основным регистром в этой схеме является аккумулятор (A-регистр), с которым, в основном, производятся операции.

Кроме аккумулятора, микропроцессор имеет три пары регистров общего назначения (POH): BC, DE, HL. Эти регистры могут использоваться как раздельно, так и в паре. Кроме того, имеются счетчик команд PC, указатель стека SP и регистр признаков F. В регистре PC при выполнении программ находится адрес ячейки памяти, из которой должен быть считан код команды. Указатель стека SP используется для организации памяти магазинного типа. При выполнении операции записи или считывания из стека содержимое указате-

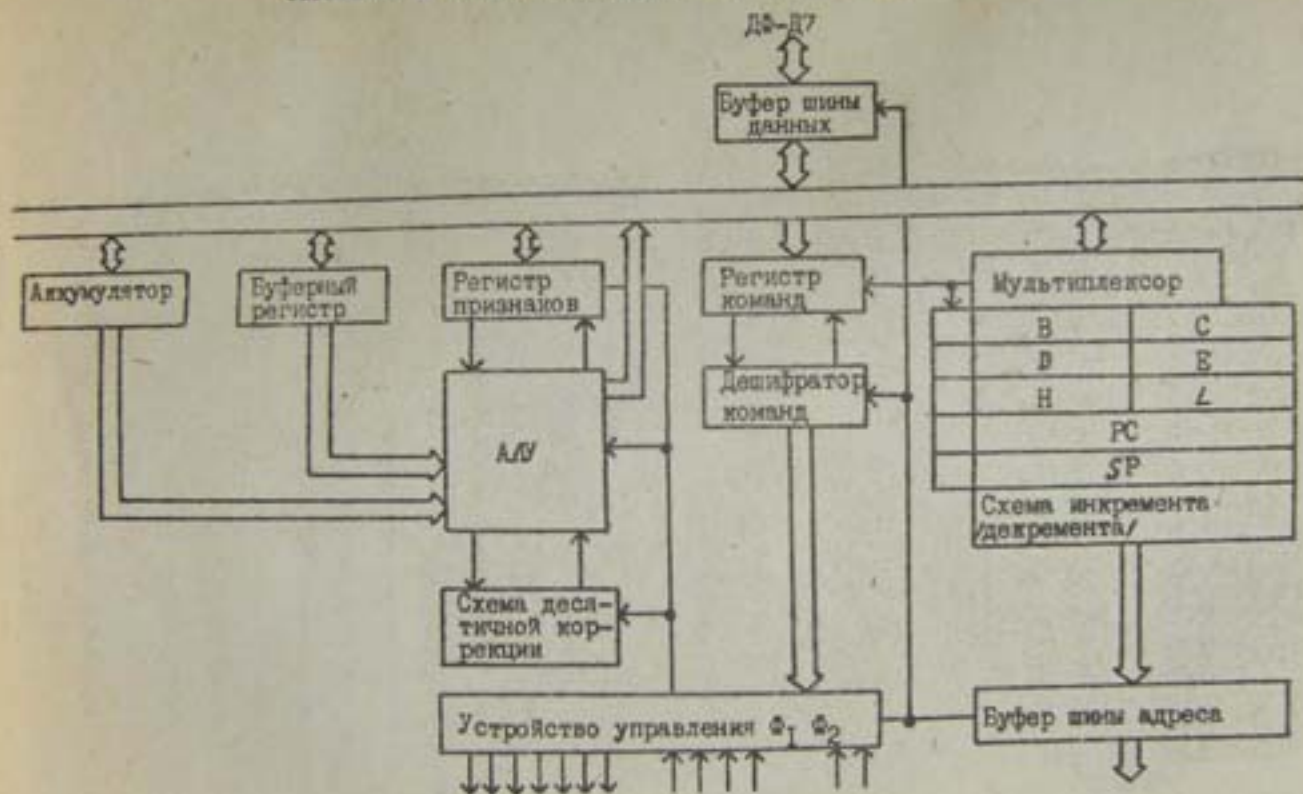


Рис. 2

ля стека S^P автоматически увеличивается или уменьшается на единицу.

Регистр признаков имеет формат, приведенный в табл. 4.

Таблица 4

Разряд	Обозначение	Признак	Примечание
0	C	перенос	$C=1$, если в результате операции возникает перенос из старшего разряда
1	I	-	Всегда 1
2	P	четность	$P=1$, если количество единиц в двоичном коде четное
3	0	-	Всегда 0
4	AC	вспомогательный перенос	$AC=1$, если перенос возникает из 4-го разряда
5	0	-	Всегда 0
6	Z	нулевой результат	$Z=0$, если результат равен 0
7	S	флаг знака	$S=1$, если результат операций отрицательный

Анализ состояния регистра признаков дает возможность выполнять условные операторы.

Все команды микропроцессора можно разделить на пять групп: команда пересылки;

арифметические и логические команды;

команды сдвига содержимого аккумулятора;

команды передачи управления;

специальные команды.

Все команды микропроцессора КР580ВМ80А сведены в табл. 5.

Хнемоника и коды команд представлены в табл. 6.

СИСТЕМА КОМАНД МИКРОПРОЦЕССОРА КР580ВМ50А

ОДНОБАЙТОВЫЕ ПЕРЕСЫЛКИ		ДВУХБАЙТОВЫЕ ПЕРЕСЫЛКИ	
MOV R1, R	: R → R1	LXI YZ, D16	: D16 → YZ
MVI R, DB	: DB → R	SHLD ADR	: HL → M(ADR) M(ADR+1)
STAX YZ*	: A → M(YZ)	LHLD ADR	: M(ADR), M(ADR+1) → HL
LDAX YZ*	: M(YZ) → A	PUSH YZ**	: YZ → M(SP-1) M(SP-2),
STA ADR	: A → M(ADR)		: SP-2 → SP
LDA ADR	: M(ADR) → A	POP YZ**	: M(SP) M(SP+1) → YZ,
SPHL	: HL → SP	(POP PSW)	: SP+2 → SP
ОБМЕН БАЙТАМИ			
XCHG	: HL ↔ DE	XTHL	: HL ↔ M(SP+1), L ↔ M(SP)
АРИФМЕТИЧЕСКИЕ И ЛОГИЧЕСКИЕ ОПЕРАЦИИ С ОДНИМ ОПЕРАНДОМ			
CNC**	: C → C	INR**	: R → R+1
STC**	: 1 → C	DEC**	: R → R-1
CMA	: A → A	INX	: YZ → YZ+1
DAA*	: DEC.KOFF	DCX	: YZ → YZ-1
АРИФМЕТИЧЕСКИЕ И ЛОГИЧЕСКИЕ ОПЕРАЦИИ С ДВУМЯ ОПЕРАНДАМИ			
8-БИТОВЫЕ ОПЕРАЦИИ			
ADD* R	: A + R → A	ADI* DB	: A + DB → A
ADC* R	: A + R + C → A	ACI* DB	: A + DB + C → A
SUB* R	: A - R → A	SUI* DB	: A - DB → A
SBB* R	: A - R - C → A	SBI* DB	: A - DB - C → A
ANA* R	: A & R → A	ANI* DB	: A & DB → A
ORA* R	: A R → A	ORI* DB	: A DB → A
XRA* R	: A ⊕ R → A	XRI* DB	: A ⊕ DB → A
CP*	: R - A		
CM*	: DB - A		
16-БИТОВЫЕ ОПЕРАЦИИ			
DAD*	: YZ → HL + YZ		
КОМАНДЫ СБИВКИ СОДЕРЖАК АККУМУЛЯТОРА			
RLC*	: СБИВКА ВЛЕВО		
RAL*	: СБИВКА ВЛЕВО ЧЕРЕЗ БИТ ПРИЗНАКА C		
RRC*	: СБИВКА ВПРАВО		
RAR*	: СБИВКА ВПРАВО ЧЕРЕЗ БИТ ПРИЗНАКА C		
КОМАНДЫ ПЕРЕДАЧИ УПРАВЛЕНИЯ			
PCHL	: HL → PC		
JMP ADR	: ADR → PC		
J-CON ADR	: ADR → PC		
КОМАНДЫ ВЫЗОВА И ВОЗВРАТА ИЗ ПОДПРОГРАММ			
CALL ADR	: PC → M(SP-1) M(SP-2), ADR → PC		
C-CON ADR	: PC → M(SP-1) M(SP-2), ADR → PC		
RST XD	: PC → M(SP-1) M(SP-2), ADD → PC		
	: ADD = 00H, 08H, 10H, 18H, 20H, 28H, 30H, 38H		
	: ДЛЯ X = 0, 1, ..., 7 СООТВЕТСТВЕННО		
RET	: M(SP) M(SP+1) → PC, SP+2 → SP		
R-CON	: M(SP) M(SP+1) → PC, SP+2 → SP		

ПРОДОЛЖЕНИЕ ТАБЛ. 4

СПЕЦИАЛЬНЫЕ КОМАНДЫ		ФОРМАТ РЕГИСТРА F
EI	РАЗРЕШЕНИЕ ПРЕРЫВАНИЯ	D7 D6 D5 D4 D3 D2 D1 D0
DI	ЗАПРЕЩЕНИЕ ПРЕРЫВАНИЯ	S Z 0 AC 0 P 1 C
HLT	ОСТАНОВ	
NOP	ХОЛОСТАЯ ОПЕРАЦИЯ	

УСЛОВНЫЕ ОБОЗНАЧЕНИЯ

- * - КОМАНДА ОКАЗЫВАЕТ ВОЗДЕЙСТВИЕ НА ВСЕ ПРИЗНАКИ
- ** - КОМАНДА ОКАЗЫВАЕТ ВОЗДЕЙСТВИЕ НА ПРИЗНАК C
- *** - КОМАНДА ОКАЗЫВАЕТ ВОЗДЕЙСТВИЕ НА ВСЕ ПРИЗНАКИ, КРОМЕ C
- R, R1 - СОДЕРЖИМОЕ РЕГИСТРОВ R, B, C, D, E, H, L, ИЛИ ЯЧЕЙКИ ПАМЯТИ M(HL)
- YZ - СОДЕРЖИМОЕ РЕГИСТРОВОЙ ПАРЫ BC, DE, HL ИЛИ SP
- YZ* - СОДЕРЖИМОЕ РЕГИСТРОВОЙ ПАРЫ BC ИЛИ DE
- YZ** - СОДЕРЖИМОЕ РЕГИСТРОВОЙ ПАРЫ BC, DE, HL ИЛИ PSW
- SP - СОДЕРЖИМОЕ УКАЗАТЕЛЯ СТЕКА ПЕРЕД ВЫПОЛНЕНИЕМ КОМАНДЫ
- DB - 8-РАЗРЯДНЫЙ ОПЕРАНД
- D16 - 16-РАЗРЯДНЫЙ ОПЕРАНД
- ADR - 16-РАЗРЯДНЫЙ АДРЕС
- M() - СОДЕРЖИМОЕ ЯЧЕЙКИ ПАМЯТИ (АДРЕС УКАЗАН В СКОБКАХ)
- CON - ЧАСТЬ ИМЕННОЙ КОМАНДЫ, ОПРЕДЕЛЯЮЩАЯ УСЛОВИЕ ПЕРЕДАЧИ УПРАВЛЕНИЯ, ВЫЗОВА И ВОЗВРАТА ИЗ ПОДПРОГРАММЫ (CON-В ИМЕННОЙ ЗАМЕНЯЕТСЯ НА NZ, Z, NC, C, PO, PE, P, ИЛИ M)

КОДЫ КОМАНД МИКРОПРОЦЕССОРА КР580ВМ80А

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
0	NOF	LXI	STAX	INX	INR	DCR	MVI	RLC	-	DAD	LDAX	DCX	INR	DCR	MVI	RRC	1
1	B&B	B	B	B	B	B	B	B	-	DAD	LDAX	DCX	INP	DCR	MVI	RAR	2
2	-	LXI	STAX	INX	INR	DCR	MVI	RAL	-	DAD	LDAX	DCX	INP	DCR	MVI	RAR	3
3	0.4	D	D	D	D	D	D	D	-	DAD	LDAX	DCX	INR	DCR	MVI	CMA	4
4	-	LXI	SKLD	INX	INR	DCR	MVI	DAR	-	DAD	LHLD	DCX	INR	DCR	MVI	CMA	5
5	H.4	*	H	H	H	H	H	H	-	DAD	LHLD	DCX	INR	DCR	MVI	CMA	6
6	-	LXI	STA	INX	INR	DCR	MVI	STC	-	DAD	LDA	DCX	INR	DCR	MVI	CMA	7
7	SP.4	*	SP	H	H	H	H	H	-	DAD	LDA	DCX	INR	DCR	MVI	CMA	8
8	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	9
9	B.4	B.4	B.4	B.4	B.4	B.4	B.4	B.4	B.4	B.4	B.4	B.4	B.4	B.4	B.4	B.4	10
A	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	11
B	0.4	D.4	D.4	D.4	D.4	D.4	D.4	D.4	D.4	D.4	D.4	D.4	D.4	D.4	D.4	D.4	12
C	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	13
D	H.4	H.4	H.4	H.4	H.4	H.4	H.4	H.4	H.4	H.4	H.4	H.4	H.4	H.4	H.4	H.4	14
E	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	MOV	15
F	M.4	M.4	M.4	M.4	M.4	M.4	M.4	M.4	M.4	M.4	M.4	M.4	M.4	M.4	M.4	M.4	16
0	ADD	ADD	ADD	ADD	ADD	ADD	ADD	ADD	ADD	ADD	ADD	ADD	ADD	ADD	ADD	ADD	17
1	B	C	D	E	H	L	M	A	B	C	D	E	H	L	M	A	18
2	SUB	SUB	SUB	SUB	SUB	SUB	SUB	SUB	SUB	SUB	SUB	SUB	SUB	SUB	SUB	SUB	19
3	B	C	D	E	H	L	M	A	B	C	D	E	H	L	M	A	20
4	ANA	ANA	ANA	ANA	ANA	ANA	ANA	ANA	XRA	XRA	XRA	XRA	XRA	XRA	XRA	XRA	21
5	B	C	D	E	H	L	M	A	B	C	D	E	H	L	M	A	22
6	ORA	ORA	ORA	ORA	ORA	ORA	ORA	ORA	CMP	CMP	CMP	CMP	CMP	CMP	CMP	CMP	23
7	B	C	D	E	H	L	M	A	B	C	D	E	H	L	M	A	24
8	RNZ	POP	JNZ	JMP	CHZ	PUSH	ADI	RST	RZ	RET	JZ	-	CZ	CALL	ACI	RST	25
9	B	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	26
A	RNC	POP	JNC	OUT	CNC	PUSH	SUI	RST	RC	-	JC	IN	CC	-	SBI	RST	27
B	D	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	28
C	RPO	POP	JPO	XTHL	CPO	PUSH	RNI	RST	RPE	PCHL	JPE	XGHL	CPE	-	XRI	RST	29
D	H	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	30
E	RP	POP	JP	DI	CP	PUSH	ORI	RST	RM	SPHL	JM	ET	CM	-	CPI	RST	31
F	PSW	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	32

0 - АДРЕС КОМАНДЫ ОПЕРАЦИИ - D16
 1 - АДРЕС КОМАНДЫ ОПЕРАЦИИ - ADP
 2 - КОМАНДА ОПЕРАЦИИ - D16

3 ПРИМЕР:
 4 КОМАНДА HLTH - АДРЕС КОМАНДЫ ОПЕРАЦИИ - D16

Код команды JZ* - CA

7.2.1. Группа команд пересылок

Внутренние однобайтовые пересылки выполняются с помощью группы команд **MOV RI, R**. Содержимое регистра **R** пересылается в регистр **RI**. Содержание регистра **R** сохраняется. В качестве **R** или **RI** может быть любой регистр микропроцессора: **A, B, C, R, E, H, L**, а также ячейка памяти, адрес которой предварительно записан в регистровую пару **HL**. Для пересылок между аккумулятором и ячейкой памяти могут использоваться регистровые пары **BC** или **DE**. Тогда для записи в ячейку памяти используются команды **STAB** или **STAD** или, для обратной пересылки, **L DAB** и **L DAD**.

Для пересылки содержимого аккумулятора по адресу, указанному в команде, используются команды **STA ADP** и **LDA ADP**. Команда **MVI R, DB** помещает операнд **DB** в регистр **R**. Кроме команд однобайтовых пересылок существуют команды двухбайтовых пересылок.

Команда **LXI YZ, D16** - помещает двухбайтовый операнд **D16** в регистровые пары **BC, DE** или **HL**.

Команда **SHL** помещает содержимое **HL** в указатель стека **SP**.

Остальные команды этой группы выполняют пересылку с адресацией по указателю стека.

PUSH YZ - засылка в стек содержимого регистровой пары **BC, DE** или **HL**.

PUSH RSW - засылка в стек содержимого аккумулятора и регистра признаков.

Команды **POP YZ** осуществляет пересылку, обратную командам **PUSH**.

7.2.2. Арифметические и логические команды

Команды с одним операндом:

CMC - инверсия бита переноса в регистре признаков;

STC - установка бита переноса в регистре признаков.

CMA - инверсия содержимого аккумулятора;

DAA - двоично-десятичная коррекция;

INR R - увеличение содержимого внутреннего регистра процессора или ячейки памяти на 1;

INX Y - увеличение содержимого регистровой пары на 1;

DCR R - уменьшение содержимого внутреннего регистра процессора или ячейки памяти на 1;

DCX YZ - уменьшение содержимого регистровой пары на 1.

Команды с двумя операндами. Эта группа команд в качестве одного операнда использует содержимое аккумулятора. Вторым операндом может быть в любом регистре микропроцессора или ячейки памяти, адресуемой по содержимому HL. Имеются следующие команды с двумя операндами:

ADD R, ADI DB - сложение двух операндов;

ADD R, ACII DB - сложение с учетом бита переноса;

SUB R, SUI DB - вычитание;

SBB R, SBI DB - вычитание с учетом бита переноса;

ANA R - поразрядное логическое умножение;

ANI DB - поразрядное логическое умножение аккумулятора и операнда DB;

ORA R - поразрядное логическое сложение содержимого аккумулятора и регистра;

ORI DB - поразрядное логическое сложение содержимого аккумулятора и операнда DB;

XRA R - поразрядное ИСКЛЮЧАЮЩЕЕ ИЛИ содержимого аккумулятора и любого регистра R;

XRI DB - поразрядное ИСКЛЮЧАЮЩЕЕ ИЛИ содержимого аккумулятора и операнда DB;

CMP R, CPI D8

- сравнения двух операндов путем вычитания из содержимого аккумулятора второго операнда;
- содержимое не запоминается и используется для установления битов регистра признаков;

DAD B,

DAD D, DAD H

DAD SP

- позволяют сложить два шестнадцатиразрядных числа, одно из которых HL, а другое BC, DE, HL, или SP. Результат операции помещается в HL.

7.2.3. Группа команд сдвига содержимого аккумулятора

На рисунке 3 схематично показано, как происходит сдвиг содержимого аккумулятора влево и вправо RAL и RAR и командами циклического сдвига RLC и RRC. В операциях участвует бит переноса регистра признаков C. Каждая команда производит сдвиг только на 1 бит.

7.2.4. Группа команд передачи управления

Имеются следующие команды передачи управления:

JMP ADR

PCHL

CALL ADR

RET

JNZ ADR

- безусловный переход по адресу ADR;
- передача управления по адресу, хранящемуся в регистре HL;
- вызов подпрограммы, управление передается по адресу VADR;
- возврат программы на адрес, следующий после команды CALL, вызвавшего подпрограмму;
- переход по ненулевому результату ($Z \neq 0$);

КОМАНДЫ ЦИФРОВОГО СДВИГА

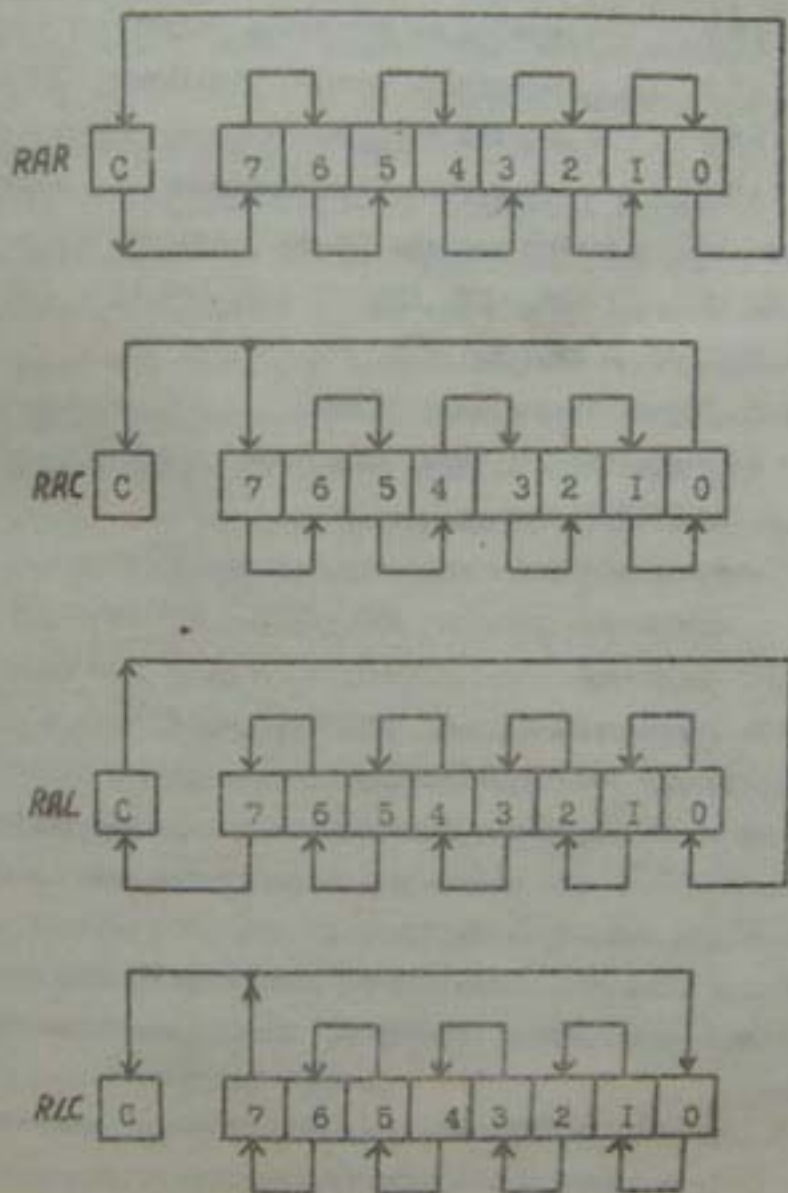


Рис. 3

CZ ADR	- переход по нулевому результату ($Z=1$);
JNC ADR	- переход по отсутствию переноса ($C=0$);
JC ADR	- переход по переносу ($C=1$);
JPO ADR	- переход по нечетному результату ($P=0$);
JPE ADR	- переход по четному результату ($P=1$);
JP ADR	- переход по положительному результату ($S=0$);
JM ADR	- переход по отрицательному результату ($S=1$).

Аналогично работают команды вызова подпрограмм и возврата из них **CNZ ADR**, **CZ ADR**, **CNC ADR**, **CC ADR**, **CPO ADR**, **CPE ADR**, **CP ADR**, **BNZ, RZ**, **BNC, RC**, **RPO, RPE, RP**.

7.2.5. Группа специальных команд

NOP - отсутствие операции. Используется для задания временных интервалов.

HLT - останов. Запуск процессора после команды **HLT** возможен только сигналом **RES (СБРОС)** или сигналом запроса прерывания.

7.3. Описание программы СИСТЕМНЫЙ МОНИТОР

Программа **СИСТЕМНЫЙ МОНИТОР** (в дальнейшем просто **МОНИТОР**) является минимумом программного обеспечения, который необходим пользователю для работы с радиоконструктором. Программа выполняет следующие функции:

1) при включении питания и нажатии кнопки **СБРОС** программа осуществляет начальную настройку БИС блока процессора радиоконструктора,

2) программа **МОНИТОР** обеспечивает обмен информацией с базовыми устройствами ввода-вывода (магнитофоном, видеоконтрольным устройством, клавиатурой) путем обращения к подпрограммам ввода-вывода,

3) **МОНИТОР** предоставляет пользователю возможность просмотр-

ра, сравнения, изменения содержимого памяти и регистров микропроцессора.

Объем оперативной памяти, необходимой для работы программы МОНИТОР составляет около 2 Кбайт, куда входит экранная область ОЗУ (1,6 Кбайт) и область стека. Программа МОНИТОР хранится в ПЗУ радиоконструктора и не требует загрузки.

После запуска МОНИТОРА экран телевизора очищается, в левом верхнем углу появляется надпись РАДИО-86РК и под ней стрелка, уведомляющая пользователя о том, что МОНИТОР готов к вводу очередной директивы. Неверно набранные символы стирают нажатием на клавишу 36 (забой) или ← (курсор влево). Для выполнения директивы нажимают на клавишу возврата каретки BK. Если директива задана правильно, то начнется её выполнение, если нет - на экране появится знак вопроса-признак того, что МОНИТОР "не понял" введенной директивы.

Коды всех директив состоят из одной латинской буквы, непосредственно за которой могут следовать не более трех параметров, представляющих собой шестнадцатеричные числа. Один параметр от другого отделяют запятой. Её ставят и в тех случаях, когда один из параметров отсутствует.

Директивы МОНИТОРА (см. табл. 7) можно поделить на 3 группы:

- 1) директивы работы с памятью
- 2) директивы запуска и отладки программ
- 3) директивы ввода - вывода.

Выполнение директив P, L, S, может быть прервано.

Для этого нужно нажать одновременно клавиши UC+C.

Знак "+" означает, что клавиши нажимаются одновременно.

Директива МОНИТОРА	Назначение
D (начальный адрес), (конечный адрес),	Распечатка на экране в шестнадцатиречном виде области памяти
L (начальный адрес), (конечный адрес)	Распечатка на экране в символьном виде области памяти
F (начальный адрес), (конечный адрес), (записываемый код)	Запись кода во все ячейки области памяти
M (адрес)	Просмотр и изменение содержимого ячеек памяти
T (начальный адрес), (конечный адрес), (адрес области пересмаки)	Копирование содержимого одной области памяти в другую
C (начальный адрес), (конечный адрес), (адрес области сравнения)	Сравнение содержимого двух областей памяти
S (начальный адрес), (конечный адрес), (искомый код)	Поиск кода в заданной области памяти
G (адрес запуска), /адрес останова/, X	Запуск программы
	Просмотр и изменение содержимого внутренних регистров микропроцессора.
O (начальный адрес), (конечный адрес), /константа скорости/	Вывод содержимого памяти на магнитофон

Директива МОНИТОРА	Назначение
I /сведение/, /константа скорости/	Ввод информации с магнитофона
R (начальный адрес ПЗУ), (конечный адрес ПЗУ), (адрес загрузки)	Ввод информации с внешнего ПЗУ

ПРИМЕЧАНИЕ: Обязательные параметры выделены круглыми скобками: при наборе их с клавиатуры незначащие нули (слева) могут быть опущены. Необязательные параметры выделены наклонными чертами. При наборе директив они могут быть опущены, при этом МОНИТОР принимает их значения по "умолчанию", то есть равными константам МОНИТОРА.

7.3.1. Директивы работы с памятью

Для вывода содержимого области памяти на экран в шестнадцатиричном виде служит директива D МОНИТОРА. При этом содержимое будет выведено в виде строк, состоящих из шестнадцати колонок.

Слева от каждой строки выводится шестнадцатиричный адрес первой в строке ячейки.

Для просмотра содержимого памяти в символьном виде служит директива L. Если при выполнении этой директивы встретятся коды, не соответствующие ни одному алфавитно-цифровому символу, то они отображаются точкой.

Для просмотра и изменения одной или нескольких ячеек памяти служит директива M МОНИТОРА. После её ввода на экране высвечивается адрес ячейки и её содержимое, курсор останавливается от этого значения и МОНИТОР "ожидает" ввода с клавиатуры. Если необходимо изменить содержимое ячейки, следует набрать новое значение и нажать клавишу BK, если изменений не требуется, не нажимают сразу же. При каждом нажатии на клавишу BK значение адреса автоматически увеличивается на 1. Выполнение директивы продолжается до тех пор, пока не будет нажата клавиша <·> (точка).

Для записи одного кода во все ячейки области памяти применяется директива F.

Директива T предназначена для копирования (пересылки) содержимого одной области памяти в другую.

Копирование осуществляется побайтно, начиная с младшего адреса.

Для сравнения двух областей памяти необходимо пользоваться директивой C. Если содержимое соответствующих ячеек памяти не совпадает, на экран выводится адрес ячейки из первой области, её содержимое и содержимое ячейки из второй области.

МОНИТОР представляет пользователю возможности поиска кода в заданной области памяти по директиве S. При обнаружении искомого кода на экране появляются адреса ячеек, в которых он был обнаружен.

7.3.2. Директивы запуска и отладки программы

Для запуска программ служит директива G. Вторым параметром этой директивы, задающий адрес останова, используется только при отладке программ и может быть опущен.

При задании адреса останова необходимо иметь в виду, что при этом изменяется содержимое ячеек памяти 0030H, 0031H, 0032H, поэтому использовать ячейки в нашей программе не рекомендуется. Кроме того, пользователь может сам назначать в программе адреса останова, записав по этим адресам код команды RST6(0F7H). Если при выполнении программы встретится эта команда, управление будет передано МОНИТОРУ, который сообщит адрес, в котором было прервано выполнение программы, после чего пользователь может проконтролировать ход выполнения программы с помощью директив МОНИТОРА.

Просмотреть и изменить содержимое внутренних регистров микропроцессора можно с помощью директивы X. В результате её выполнения на экран выводятся имена и содержимое регистров, которые можно изменять так же, как и содержимое ячеек памяти по директиве M. Регистр признаков результата операции обозначен буквой F, обозначение других внутренних регистров стандартно.

7.3.3. Директивы ввода - вывода

Для работы с кассетным магнитофоном в МОНИТОРЕ предусмотрены директивы I и O (латинские буквы).

По директиве O производится вывод информации на магнитофонную ленту. Первые два параметра этой директивы задают область памяти, которая подложит выводу, третий - заставляющий код.

определяющий скорость вывода. Если параметр, задающий скорость, не указывать, то будет использовано либо значение, заданное в предыдущей директиве вывода, либо стандартное $10H$ (около 1200 бит/с), записываемое (при нажатии на кнопку СБРОС) в рабочую ячейку МОНИТОРА $0363FH$ (использовать константу вывода менее $10H$ недопустимо). Рекомендуется пользоваться стандартной скоростью, т.к. она выбрана с учетом использования магнитофона и магнитной ленты невысокого качества. Кроме того, применение стандартной скорости облегчит обмен программами. После завершения вывода на экране отобразятся начальный и конечный адреса и четырехзначная контрольная сумма выведенной информации.

Ввод с ленты осуществляется по директиве I, которая может иметь два параметра. Первым параметром, необязательным, задает смещение. Если он есть, вводимая информация будет загружена по адресу, являющемуся суммой указанного в записи на ленте и смещения. Вторым параметр определяет скорость ввода. Этот параметр тоже может отсутствовать, но в этом случае будет взято значение параметра, использовавшееся в предыдущей команде ввода (если вы не нажимали кнопку СБРОС) или установленное по умолчанию при начальной настройке рабочих ячеек МОНИТОРА стандартное значение, записанное по адресу $0362FH - 2AH$.

После окончания ввода МОНИТОР сообщит начальный и конечный адреса загрузки и контрольную сумму, подсчитанную при вводе информации. Если она не совпадает с введенной с ленты контрольной суммой, то на следующей строке будет выведено значение, записанное на ленте. Этот факт свидетельствует об ошибках при чтении информации с магнитной ленты. Прервать программу ввода с магнитной ленты можно либо выключением магнитофона, либо вводом с клавиатуры кода $UC + C$. Директива R предназначена для считывания инфор-

мации из внешнего ОЗУ (см. РАДИО, 1987 № 3 "ОЗУ для ВЕЙСИКА").

7.3.4. Стандартные подпрограммы МОНИТОРА

Для обслуживания других программ МОНИТОР имеет набор стандартных подпрограмм, которые могут быть использованы Вами в своих программах. Адреса вызова подпрограмм и выходные параметры подпрограмм приведены в табл. 9.

Следует заметить, что при использовании подпрограмм ввода и вывода байта на ленту возможно нарушение процесса отображения на экране телевизора, поэтому после завершения работы с этими подпрограммами рекомендуется вызвать стандартную подпрограмму запуска индикации на экране телевизора. Кроме того, при написании программ с использованием подпрограмм ввода-вывода необходимо учитывать, что для надёжной работы время между обращениями к этим подпрограммам должно быть около 55 мкс (100 тактов работы микропроцессора).

В набор стандартных подпрограмм входят также подпрограммы ввода с магнитной ленты и вывода из неё блоков памяти, а также подпрограмма контрольной суммы блока. При использовании этих подпрограмм запуск изображения на экране производится автоматически, так же, как и при вводе - выводе по директивам 1 и 0.

При выводе на ленту необходимо указать в определенных регистрах (табл. 9) адреса начала и конца блока, а также его контрольную сумму, подсчитанную стандартной подпрограммой монитора или собственной подпрограммой пользователя.

При вводе блока с ленты можно указать смещение, с которым блок данных должен быть загружен в ОЗУ. Подпрограмма чтения блока информации возвращает (в регистрах микропроцессора) адреса загрузки и считанную с ленты контрольную сумму, которая затем может быть проверена пользователем.

СТАНДАРТНЫЕ ПОДПРОГРАММЫ МОНИТОРА

ТАБЛИЦА В

ВВОД СИМВОЛА С КЛАВИАТУРЫ	BF803H -2045D	ВХОДНЫЕ : ВЫХОДНЫЕ : A - ВВЕДЕННЫЙ КОД
ВВОД БАЙТА С МАГНИТОФОНА	BF806H -2042D	ВХОДНЫЕ : A=BFH - С ПОИСКОМ СИНХРОБАЙТА A=BFH - БЕЗ ПОИСКА СИНХРОБАЙТА ВЫХОДНЫЕ : A - ВВЕДЕННЫЙ БАЙТ
ВЫВОД СИМВОЛА НА ЭКРАН	BF809H -2039D	ВХОДНЫЕ : C - ВЫВОДИМЫЙ КОД ВЫХОДНЫЕ : ---
ЗАПИСЬ БАЙТА НА МАГНИТОФОН	BF80CH -2036D	ВХОДНЫЕ : C - ВЫВОДИМЫЙ КОД ВЫХОДНЫЕ : ---
ОПРОС СОСТОЯНИЯ КЛАВИАТУРЫ	BF812H -2030D	ВХОДНЫЕ : --- ВЫХОДНЫЕ : A=00 - НЕ НАЖАТА A=BFH - НАЖАТА
РАСПЕЧАТКА БАЙТА НА ЭКРАНЕ В ШЕСТЬ- НАЦИТИРНОМ ВИДЕ	BF813H -2027D	ВХОДНЫЕ : A - ВЫВОДИМЫЙ КОД ВЫХОДНЫЕ : ---
ВЫВОД НА ЭКРАН СООБЩЕНИЯ	BF818H -2024D	ВХОДНЫЕ : HL - АДРЕС НАЧАЛА ВЫХОДНЫЕ : ---
ПРОС КОДА ПОДЪЕМОМ КЛАВИШИ	BF81DH -2021D	ВХОДНЫЕ : --- ВЫХОДНЫЕ : A=BFH - НЕ НАЖАТА A=BFH - ПОСЛЕД. ИНАЧЕ - КОД КЛАВИШИ
ЗАПРОС ПОЛОЖЕНИЯ КУРСОРА	BF81EH -2018D	ВХОДНЫЕ : --- ВЫХОДНЫЕ : H - НОМЕР СТРОКИ
ЗАПРОС БАЙТА ИЗ ЭКРАННОГО БУФЕРА	BF81FH -2015D	ВХОДНЫЕ : L - НОМЕР ПОЗИЦИИ ВЫХОДНЫЕ : A - КОД ИЗ БУФЕРА
ВВОД БУДКА С МАГНИТОФОНА	BF824H -2012D	ВХОДНЫЕ : HL - СМЕЩЕНИЕ ВЫХОДНЫЕ : HL - АДРЕС НАЧАЛА DE - АДРЕС КОНЦА BC - КОНТР. СУММА

ВЫВОД БЛОКА НА МАГНИТОФОН	0F827H - 2009D	ВХОДНЫЕ : HL - АДРЕС НАЧАЛА BC - АДРЕС КОНЦА BC - КОНТР. СУММА
ПОДСЧЕТ КОНТ- РОЛЬНОЙ СУММЫ БЛОКА	0F82AH - 2006D	ВХОДНЫЕ : HL - АДРЕС НАЧАЛА DE - АДРЕС КОНЦА ВЫХОДНЫЕ : BC - КОНТР. СУММА
ЗАПУСК ИНДИКАЦИИ НА ЭКРАНЕ	0F82DH - 2003H	ВХОДНЫЕ : --- ВЫХОДНЫЕ : ---
ПЕРЕДАЧА АДРЕСА ВЕРХНЕЙ ГРАНИЦЫ СВОБОДНОЙ ПАМЯТИ ПРОГРАММЕ ПОЛЬ- ЗОВАТЕЛЯ	0F830H - 2000D	ВХОДНЫЕ : --- ВЫХОДНЫЕ : HL - АДРЕС ГРАНИЦЫ
УСТАНОВКА АДРЕСА ВЕРХНЕЙ ГРАНИЦЫ СВОБОДНОЙ ПАМЯТИ ПРОГРАММЫ ПОЛЬ- ЗОВАТЕЛЯ	0F833H - 1997D	ВХОДНЫЕ : HL - АДРЕС ГРАНИЦЫ ВЫХОДНЫЕ : ---

Подпрограмма опроса кода нажатой клавиши позволяет ускорить опрос клавиатуры, особенно в программах, работающих в реальном масштабе времени.

С помощью подпрограммы запроса положения курсора можно определить его местонахождение на экране телевизора, а с помощью подпрограммы запроса экранного байта узнать код символа из экранной области памяти, находящегося в позиции курсора.

В MONITORе предусмотрены также две подпрограммы для определения верхней границы свободной памяти. Первая из них возвращает программе пользователя в регистровой паре HL установленный адрес верхней границ доступного ОЗУ (по умолчанию - \$35FFH), вторая позволяет установить новую границу. Недопустимо устанавливать верхнюю границу выше её положения по умолчанию, т.е. это может привести к неправильной работе нашей программы и искажению данных в рабочих ячейках MONITORа.

7.4. Управление и алфавитно-цифровые коды дисплея

Управление и алфавитно-цифровые коды дисплея приведены в табл. 9.

Управляющие коды дисплея не отображаются на экране, а управляют отображением информации на экране дисплея.

Функция прямой адресации курсора позволяет установить курсор в любое место экрана.

Для этого необходимо выдать последовательность кодов $1B(AR2), 59(Y), a_y, a_x, [1BH, 59H, a_y, a_x]$ где a_y - позиция строки, отсчитываемая с левого верхнего угла экрана +2BH (в шестнадцатеричной системе счисления)

a_x - позиция знака/символа в строке +2BH.

Псевдографические изображения формируются из 16 различных

ТАБЛИЦА 9

УПРАВЛЯЮЩИЕ И АЛФАВИТНО-ЦИФРОВЫЕ КОДЫ АИСПДЕЯ.

	0	1	2	3	4	5	6	7
0	K1		ПРОБЕЛ	0	@ D	P80D	89LD	П 1LD
1	K2			1	A65D	Q	Я	Я
2	K3			2	В	R	Б	Р
3	K4			3	С	S	Ц	С
4	K5		Я	4	D	T	А	Т
5			Х	5	E	U	Е	У
6			Б	6	F	V	Ф	В
7				7	G	М	Г	В
8	← K1 ← K2 ←		(8	Н	X	Х	Ь
9	ТАБ ↑ K2)	9	I	Y	И	Ы
A	ТС ↓ K2		*	:	J	Z	И	З
B		AP2	+	;	K	С	К	Э
C	←		,	<	L	\	Л	Э
D	ВК	-	-	И =	M]	М	Ш
E	ГРАФ КОН(СТ)		>	Н	~	Н	Ч	
F	РУС	СТР	/	?	07D	95D	0111	35 127D

ПРИМЕЧАНИЕ: КОД СИМВОЛА ОБРАЗУЕТСЯ ИЗ НОМЕРА СТОЛБЦА И НОМЕРА СТРОКИ, НАПРИМЕР КОД СИМВОЛА "Н" = 4Е

символов. Эти символы и их шестнадцатичные коды представлены табл. 10.

Алфавитно-цифровые, управляющие и псевдографические символы можно выдать как с клавиатуры, так и программно, используя встроенные подпрограммы МОНИТОРА "печать символа" и "печать пепочерка символов".

7.5. Описание программы интерпретатора языка БЕЙСИК

Программа - интерпретатор языка БЕЙСИК предназначена для работы с программами, написанными на языке высокого уровня БЕЙСИК. Описанная версия языка БЕЙСИК отличается от минимальной наличием графических операторов, а также встроенными функциями обработки текстов.

Интерпретатор может работать в режиме непосредственного исполнения команд, подобно тому, как работает микрокалькулятор.

Загрузки интерпретатора в память производится директивой I МОНИТОРА. Интерпретатор размещается в области памяти 0000H - 1977H.

Программа в БЕЙСИКЕ состоит из последовательности пронумерованных строк. Для удобства отладки программ рекомендуется присваивать номера с шагом 10. Это дает возможность дополнительно вставить несколько строк в программу. Номера могут быть любыми от 0 до 65529. Каждая строка программы может состоять из одного или нескольких операторов, которые отделяются друг от друга двоеточиями. При вводе программы с клавиатуры каждую строку необходимо заканчивать нажатием клавиши BK, при этом происходит запись текста строки в оперативную память микроЗВМ.

7.5.1. Набор символов языка БЕЙСИК

В описываемой версии языка БЕЙСИК используются 93 символа:

26 букв латинского алфавита:

ТАБЛИЦА 10

КОДЫ ПСЕВДОГРАФИЧЕСКИХ СИМВОЛОВ

Код	Символ	Код	Символ
00		10	
01		11	
02		12	
03		13	
04		14	
05		15	
06		16	
07		17	

ТБХ, 17F1H - адрес памяти опр. буфера (левый верхний угол).
 Для 034 32КБАЙТ нужно ввести 77C2H (нижний, левый).
 Нажатие «СБРС» активирует установку значения верхней границы доступной памяти 3BFFH (75FFH для 32Кбайт) и восстановит состояние экрана ввода/вывода на н/факт. При вводе
 * 30 * «СТР» ждем из информации.
 При вводе символической инф-ии экран INPUT сигнализирует нажатие на клавишу ФУНКЦ. клавишу ищем в верхнем инт-е и вводим информацию.
 Работа микрокалькулятора идентифицируется нажатием «РЭС/МАТ» (до отсчета)
 * 40 * * 0 * - выключает экран PRINT - на экран выводится только информация. Выключение осуществляется нажатием * 40 * или * 00 *.
 При вводе информации микрокалькулятора в память, данные.

31 буква русского алфавита (за исключением буквы Ё),

10 десятичных цифр:

знаки арифметических операций:

"+" - знак сложения; "-" знак вычитания;

"*" - знак умножения;

"/" - знак деления;

"^" - знак возведения в степень;

знаки операций отношения:

"<" - меньше;

">" - больше;

"=" - знак равенства;

Синтаксические знаки:

("), [] - скобки;

"," - запятая;

"." - точка;

" " - кавычки;

' ' - апостроф;

":" - точка с запятой;

":" - двоеточие;

Специальные знаки:

"\$" - признак символьных переменных;

"%" - знак процента;

"!" - восклицательный знак;

"?" - вопросительный знак;

"&" - амперсанд;

Некоторые комбинации символов имеют следующие значения:

или:

"<>" - не равно;

"<=" - меньше или равно;

$> =$ - больше или равно.

Употребление букв русского алфавита ограничено. Как правило, буквы русского алфавита используются как элементы символьных констант.

7.5.2. Числовые и символьные константы и переменные

В описываемой версии языка БЕЙСИК используются числовые и символьные константы и переменные.

Константа - это число, значение которого в программе не изменяется. Символьная константа - это цепочка символов, заключенных в кавычки. Например "компьютер". Числовые константы представляются десятичными цифрами в диапазоне минус 10 ± 37 до 10 ± 37 . Числовые константы могут содержать не более шести значащих цифр. При вводе большего числа значащих цифр, число округляется до шести значащих цифр. Вещественные константы могут быть представлены в естественном формате или в показательной форме, например:

1.23456E-6. Символьная константа может содержать до 80 символов.

Переменная - это величина, значение которой может изменяться в процессе выполнения программы.

Например, в выражении $A=X/1987$ A и X - переменные, 1987 - константа.

Как и константы, переменные могут быть числовыми и символьными. Для того, чтобы отличить символьную переменную от числовой, символьная переменная должна после имени содержать знак $\$$. Имена переменных могут состоять из двух букв или буквы и цифры.

Например:

B - числовая переменная;

M\$ - символьная переменная;

B2 - числовая переменная;

2B - неверный синтаксис (начинается с цифры).

Для удобства написания программ можно давать имена длиной больше, чем два символа, но при этом необходимо иметь в виду, что интерпретатор отличает переменные друг от друга только по первым двум знакам имени.

7.5.3. Массивы

Группе переменных одного типа может быть присвоено общее имя, в этом случае их называют переменными с индексами или массивами. Индексы указываются в круглых скобках после имени массива. Массивы могут быть как числовыми, так и символьными.

Например:

$A(5)$ - пятый элемент одномерного массива A ;

$XM(9)$ - девятый элемент одномерного массива символьных переменных.

$CON(5,K)$ - элемент, находящийся на пересечении 5-го столбца и K -ой строки двумерного массива CON . Индексы отделяются друг от друга запятыми.

Размерность массива не ограничена, но при использовании многомерных массивов необходимо резервировать под них определенное количество ячеек памяти оператором `DIM`, иначе интерпретатор ответит под массив количество ячеек, соответствующее размерности массива.

Например, для трехмерного массива будет выделено 1000 ячеек памяти. Такое описание массива называется описанием по умолчанию. Массив должен быть описан до его использования, иначе он будет описан по умолчанию.

7.5.4. Выражения

Переменные и константы образуют выражения языка БЕЯСИК. Кроме них, в выражения входят знаки операций, скобки и имена функций.

дий. Выражения можно разделить на 4 типа:

арифметические, например: $744/18 \neq 78$;

выражение отношения, например: $a > b$; $7 = K$; $M \leq K5$;

логические: обозначаемые *AND*; *OR*; *NOT*;

слияния: "BS" + "OB", "ПАРО" + "ХОД".

Числовые переменные могут участвовать в операциях любого типа.

Символьные переменные могут участвовать только в выражениях отношения и слияния, обозначаемого знаком + .

Порядок выполнения операций определяется приоритетом операции. Таблица II показывает приоритет операций. Чем выше в таблице знак операции, тем выше приоритет. Для изменения порядка вычислений можно воспользоваться круглыми или квадратными скобками.

7.5.5. Директивы языка БЕЖИК

Директивами называются команды, которые управляют работой интерпретатора.

После ввода директив необходимо нажать ВК .

Директива *NEW* стирает находящуюся в памяти программу и подготавливает интерпретатор ко вводу новой программы.

Директива *RUN N* служит для запуска программ со строки с номером *N* . Если номер строки отсутствует, то выполнение программы начнется со строки с наименьшим номером.

Директиву *RUN* можно набрать иначе - для этого необходимо нажать клавишу КЗ.

Директива *LIST N* распечатывает текст программ со строки с номером *N* , на экране телевизора. Если номер строки отсутствует, то распечатка начнется со строки с наименьшим номером. Нажатие кнопки К2 тоже приводит к выполнению директивы.

Директива **STOP** - прерывает выполнение программы, в момент вывода на экран какой-либо информации. Эта директива выполняется и при нажатии клавиши K4.

При использовании для задания директив K1, K2, K3, K4 выполнение соответствующих директив начинается сразу после нажатия.

Директива **CONT** позволяет продолжить выполнение программы после её останова оператором **STOP**. Её можно выполнить, нажав клавишу AP2, а затем клавишу K3.

Директива **MSAVE** выводит текст программы на магнитную ленту. Эту директиву можно выполнить, нажав клавишу K1.

Директива **MLoad** загружает программу на языке БЕЙСИК с магнитной ленты в память компьютера. Директиву можно выполнить, нажав клавишу AP2, а затем K1.

7.5.6. Операторы языка БЕЙСИК

Оператор **REM** - служит для вставки комментариев в текст программы. Этот оператор не влияет на выполнение программы, однако занимает место в памяти. Поэтому комментарии должны быть по возможности краткими.

Оператор присваивания. Для него не требуется ключевого слова. Оператор выглядит следующим образом:

$A = C$, где A - переменная, а C - выражение, например:

$KB = 18/5 * 84$.

Оператор **DIM** предназначен для описания массивов.

Этот оператор резервирует определенное количество ячеек памяти под массив. Одним оператором **DIM** можно описать несколько массивов. Например: $30 \text{ DIM } A(10), B(8,6), C(5, 3, 7, 4)$.

Оператор **DATA** - предназначен для описания данных непосредственно в программе. Значения данных присваиваются переменным с помощью оператора **READ**. Программа может содержать любое число операторов **DATA**. Располагаться они могут в любом месте, неза-

зависимо от оператора READ. Оператором DATA могут быть описаны любые данные, как числовые, так и символьные.

Например: 1 DATA 18.5354, 342, 56, - 17, "МИНУС".

Оператор READ предназначен для считывания данных, заданных оператором DATA. Данные будут считываться последовательно, начиная с первого.

Например: 2 READ A, B, C, D.

Оператор RESTORE служит для перемещения указателя данных в начальную позицию. После этого оператора данные из блока будут считываться повторно, начиная с первого.

Оператор INPUT предназначен для ввода данных с клавиатуры.

Значение введенных данных присваивают переменным, имена которых следуют за оператором INPUT; это могут быть как числовые, так и символьные переменные. После оператора INPUT может стоять строка символов, заключенных в кавчки. В этом случае при выполнении оператора на экране дисплея будет выведено это сообщение, а затем знак вопроса, означающий, что компьютер готов к вводу данных.

Например: 10 INPUT "ИМЯ, ОТЧЕСТВО, ФАМИЛИЯ": A, B, C.

Если нужно ввести несколько данных, то они вводятся через запятую. Ввод данных заканчивается после нажатия клавиши BK. Если после появления знака сразу нажать клавишу BK, то компьютер переходит из программного режима в непосредственный.

Оператор PRINT предназначен для вывода на экран дисплея значений переменных, сообщений, и результатов вычислений.

При вводе программы с клавиатуры вместо слова PRINT можно набирать символ <?>. Операндов, стоящих вслед за оператором PRINT, может быть несколько, и тогда их отделяет друг от друга разделителями <,> или <:>, причем при использовании первого из

них каждому выводимому значению отводится 14 позиций в строке, а при использовании второго разделителя — столько, сколько необходимо. В качестве операндов могут быть использованы выражения.

Оператор `CUR X, Y` служит для перемещения курсора в позицию с координатой X по горизонтали и координатой Y по вертикали. X может принимать значения $0 + 63$, а $Y - 0 + 24$. Если после оператора `CUR` следует оператор `PRINT`, то вывод информации на экран начнется с позиции с координатами X, Y .

Оператор `GOTO N` — передает управление на строку с номером N , например:

```
15      GOTO      1600.
```

Оператор `GOSUB N` — оператор вызова подпрограммы, начинающейся со строки с номером N .

Подпрограмма должна заканчиваться оператором `RETURN`.

После выполнения подпрограммы управление будет передано на строку, следующую за строкой `GOSUB`

Например:

```
100 GOSUB 900
```

```
110 REM вывод результата
```

```
120 PRINT A,B
```

```
900 REM ПЕЧАТЬ текущего A
```

```
910 K=51N(B)-U
```

```
950 RETURN
```

Оператор `ON X GOTO I, 2, 3` реализует условную передачу управления на одну из строк $I, 2, 3 \dots$, в зависимости от значения переменной X . При выполнении оператора сначала берется целая часть от X , которая указывает на номер строк. Если $X = 1$, то управление передается на строку I , если $X = 2$, то на строку 2 и т.д..

Если $X < I$, или X больше, чем количество операторов в списке, то выполняется оператор, идущий за оператором `ON X GOTO 1, 2, ...`

Оператор `STOP` останавливает работу программы и переводит интерпретатор в непосредственный режим работы. Этот оператор применяется при отладке программы. Выполнение программы можно продолжить, выдав интерпретатору директиву.

Оператор `FOR X TO Y STEP Z` - оператор организации цикла, где X - выражение, задающее имя переменной и присваивающее ей начальное значение.

Y - выражение, определяющее конечное значение переменной

Z - шаг цикла

Если шаг цикла равен 1, то выражение `STEP` можно опустить.

Цикл должен заканчиваться строкой `NEXT X`.

Все, что находится между операторами `FOR X TO Y STEP Z` и `NEXT X` называется телом цикла. Допускается многократная вложенность циклов. Шаг цикла может принимать и отрицательные значения.

Пример выполнения цикла:

```
80 FOR I=12 TO 5
```

```
90 K=315/I * I:
```

```
100 PRINT I, K
```

```
130 NEXT I
```

Оператор `IF X THEN Y` работает следующим образом:

Если выражение X - истинно, то выполняется операторы, стоящие в строке после слова `THEN`.

Если выражение X ложно, то управление будет передано следующей строке программы.

Пример:

```
20 IF A=34 THEN C=45: PRINT "РЕЗУЛЬТАТ", C:GO TO 100
```

Оператор `CLS` предназначен для полного стирания информации с экрана.

В псевдографическом режиме на экране возможно отображение 128 точек по горизонтали и 50 точек по вертикали. Точки адресуются по горизонтали и вертикали. Начало отсчета - левый нижний угол экрана.

Оператор PLOT X, Y, Z - высвечивает или гасит точку с координатами $0 \leq X \leq 127$ по горизонтали и $0 \leq Y \leq 49$ по вертикали. Если $Z = 1$, то соответствующая точка засвечивается, если $Z = 0$, то гаснет.

Например, следующая программа нарисует прямую горизонтальную линию с начальными координатами 0, 20 и конечными 100, 20

```
10 CLS
20 FOR K=0 TO 100
30 PLOT K, 20, 1
40 NEXT K
```

Однако, удобнее линии на экране задавать оператором LINE X, Y . Операнды X, Y задают координаты конечной точки линии. Начальные координаты, а также вид построения (засветка или гашение) задает оператор PLOT.

Пример построения той же линии, что и в предыдущем примере с помощью оператора LINE.

```
10 CLS
20 PLOT 0, 20, 1
30 LINE 100, 20.
```

Используя графические операторы, можно сформировать различные изображения, в том числе и динамически меняющиеся.

Оператор POKE X, Y позволяет записать в ячейку памяти с адресом X , значение выражения Y . Адрес X должен быть задан в десятичной системе счисления. Если адрес превышает значение 32767

(FFFFH). то он должен указываться в виде отрицательного числа. При этом адресу FFFFH будет соответствовать (-1) и т.д..

Оператор CLEARX предназначен для очистки памяти от переменных. Если параметр X не указан, то после выполнения оператора всем числовым переменным присваивается значение "0", а всем символьным - пустая строка.

Если же параметр указан, то в памяти выделяется область размером X байт, предназначенная для хранения символьных переменных. По умолчанию, размер этой области равен 50 байт.

7.5.7. Функции на языке БЕЗСИК

В описываемом интерпретаторе языка БЕЗСИК есть ряд встроенных функций. В таблице 12 перечислены математические функции языка БЕЗСИК.

Функция RND (X) требует пояснения. X может быть в диапазоне от 0 до 9.

Результатом работы функции RND (X) является случайное число в диапазоне от 0 до 1.

Если необходимо случайное число в другом диапазоне, например: от 0 до 50, то нужно полученное случайное число умножить на пятьдесят:

$A = \text{RND} (1) * 50$, для того, чтобы сделать его целым:

$A = \text{INT} (\text{RND}(1) * 50)$.

Кроме встроенных математических функций интерпретатор содержит ряд функций для обработки текстов.

Функция LEN (X\$). Результатом работы является число, равное количеству символов в переменной X.

Функция LEFT (X\$, Y) выводит на экран строку символов длиной Y, начиная с крайнего левого символа.

Функция RIGHT (X\$, Y), то же, но начиная с крайнего

АРИФМЕТИЧЕСКИЕ ФУНКЦИИ В ЯЗЫКЕ БЕЙСИК

ФУНКЦИЯ	РЕЗУЛЬТАТ РАБОТЫ
SQR(X)	КОРЕНЬ КВАДРАТНЫЙ ИЗ X
EXP(X)	ЭКСПОНЕНЦИАЛЬНАЯ ФУНКЦИЯ E
LOG(X)	НАТУРАЛЬНЫЙ ЛОГАРИФМ ОТ X
ABS(X)	АБСОЛЮТНАЯ ВЕЛИЧИНА X $ABS(X) = \begin{cases} X, & \text{ЕСЛИ } X \geq 0 \\ 0, & \text{ЕСЛИ } X = 0 \\ -X, & \text{ЕСЛИ } X < 0 \end{cases}$
SGN(X)	ЗНАК X $SGN(X) = \begin{cases} 1, & \text{ЕСЛИ } X > 0 \\ 0, & \text{ЕСЛИ } X = 0 \\ -1, & \text{ЕСЛИ } X < 0 \end{cases}$
SIN(X)	СИНУС ОТ X В РАДИАНАХ
COS(X)	КОСИНУС ОТ X В РАДИАНАХ
TAN(X)	ТАНГЕНС ОТ X (X В РАДИАНАХ)
ATAN(X)	АРКТАНГЕНС ОТ X, РЕЗУЛЬТАТ В РАДИАНАХ
INT(X)	ЦЕЛАЯ ЧАСТЬ ОТ X
RND(X)	СЛУЧАЙНОЕ ЧИСЛО В ДИАПАЗОНЕ ОТ 0 ДО 1

Информация OUT X,Y и f-но INP(X) в Бейсике содержится, но лучше или не использовать, а использовать PEEK X,Y и PEEK(X), соотв. Форматы A,B,C и формат унитарного-целого слова и/или D14, соотв. по адресу: -24576, -24575, -24574 и -24573.

Из Бейсика можно собирать и интерпретировать. Меморизация в базе данных клавишного доз. редактора. Внимательно изучите информацию и используйте f-но VSR(-2021). Р-р. оператор A=VSR(-2021) управляет переменными A заданные в базе клавишного клавишного или 255, если не адрес ч. или не клавиша, оператор ON VSR(-2021)-48 GOTO 1000, 2000 или 3000 информации циф.-но строчкам 1000, 2000, 3000, если клавиша «1», «2» или «3», 3 циф. 3-и разряда. Метод др. или не клавиша клавиша - выключенный код. Информация: PRINT CHR X (7).

правого символа.

Функция MID (X, Y, Z) выводит строку символов длиной Z, начиная с позиции Y.

Функция STR (X) преобразует числовые величины в символьные.

Например: если X=3765 от STR = "3765"

Функция VAL (X) предназначена для обратного преобразования, начиная с крайнего левого символа. Если в строке встречается недопустимый символ, то преобразование прекращается.

Функция ASC (X) переводит код первого символа X в десятичный вид.

Функция CHR (X) позволяет вывести на экран символ, код которого равен X. Аргумент X не должен превышать 255. Эту функцию удобно использовать при выводе на экран управляющих символов.

Функция POS (I) определяет номер позиции последнего отпечатанного символа в строке, например:

```
LOCPRINT "МАРТ", TO POS (I) - 4
```

Функция SPC (X) позволяет ввести в выводимую строку X пробелов, например:

```
20 PR INT "МАРТ"; SPC (I=); "ФЕВРАЛЬ".
```

Функция TAB (X) - функция горизонтальной табуляции, помещает курсор в заданную позицию в строке.

Функция FRE (I) - определяет количество свободных байт памяти.

Функция PEEK (X) - десятичное число, равное содержимому ячейки памяти, адрес которой определен аргументом X.

Аргумент задается по тем же правилам, что и в операторе POKE.

Функция USR (X) вызывает обращение к подпрограмме в кодах.

записанной по адресу X. В конце подпрограммы обязательно должна стоять команда RET.

7.5.8. Работа в командном режиме

Кроме работы в программном режиме, интерпретатор может выполнять и арифметические действия, т.е. работать как обычный микрокалькулятор. Для этого необходимо набрать знак вопроса (или оператор PRINT), а затем необходимые действия. Например:

123 * 456 ^ 7/8 BK

6.30334E + 19

7.5.9. Сообщения об ошибках

При обнаружении в программе или директиве ошибок, интерпретатор выдает сообщение об ошибке:

XX - ошибка в N

XX - код ошибки

N - номер строки, в которой была обнаружена ошибка.

Коды ошибок приведены в табл. 13.

7.6. Программа АССЕМБЛЕР (редактор текста)

Эта программа предназначена для составления программ и последующих трансляций.

Распределение оперативной памяти при работе программы показано в табл. 14.

Программа вводится с магнитной ленты по директиве I монитора и запускается директивой монитора G BK .

При этом на экране появится надпись ED* МИКРОН.

Для того, чтобы "набить" текст программы необходимо нажать на клавишу AP2 и затем на клавишу /, при этом в левом верхнем углу появится надпись NEW?; после этого необходимо нажать клавишу < Y > и на экране появится стрелка →, обознача-

СООБЩЕНИЯ ОБ ОШИБКАХ ИНТЕРПРЕТАТОРА ЯЗЫКА БЕЯСИК

КОД	КОММЕНТАРИЙ
01	В ПРОГРАММЕ ВСТРЕТИЛСЯ ОПЕРАТОР NEXT, ДЛЯ КОТОРОГО НЕ БЫЛ ВЫПОЛНЕН ОПЕРАТОР FOR
02	НЕВЕРНЫЙ СИНТАКСИС
03	В ПРОГРАММЕ ВСТРЕТИЛСЯ ОПЕРАТОР RETURN, БЕЗ ПРЕДВАРИТЕЛЬНОГО ВЫПОЛНЕНИЯ ОПЕРАТОРА GOSUB
04	ПРИ ВЫПОЛНЕНИИ ПРОГРАММЫ НЕ ХВАТАЕТ ДАННЫХ ДЛЯ ОПЕРАТОРА READ, Т.Е. ДАННЫХ, ОПИСАННЫХ ОПЕРАТОРАМИ DATA МЕНЬШЕ, ЧЕМ ПЕРЕМЕННЫХ В ОПЕРАТОРАХ READ
05	АРГУМЕНТ ФУНКЦИИ НЕ СООТВЕТСТВУЕТ ОБЛАСТИ ЕЕ ОПРЕДЕЛЕНИЯ
06	ПЕРЕПОЛНЕНИЕ ПРИ ВЫПОЛНЕНИИ АРИФМЕТИЧЕСКИХ ОПЕРАЦИЙ
07	НЕДОСТАТОЧЕН ОБЪЕМ ПАМЯТИ
08	НЕТ СТРОКИ С ДАННЫМ НОМЕРОМ
09	ИНДЕКС НЕ СООТВЕТСТВУЕТ РАЗМЕРНОСТИ МАССИВА
10	ПОВТОРНОЕ ВЫПОЛНЕНИЕ ОПЕРАТОРОВ DIM ИЛИ DEF, ОПИСЫВАЮЩИХ МАССИВ ИЛИ ФУНКЦИЮ, КОТОРЫЕ УЖЕ БЫЛИ ОПИСАНЫ РАНЕЕ
11	ДЕЛЕНИЕ НА НОЛЬ
12	ПОПЫТКА ВЫПОЛНИТЬ ОПЕРАТОРЫ INPUT ИЛИ DEF В НЕПОСРЕДСТВЕННОМ РЕЖИМЕ
13	НЕСООТВЕТСТВИЕ ТИПОВ ДАННЫХ ВОЗНИКАЕТ ПРИ ПОПЫТКЕ ПРИСВОИТЬ СИМВОЛЬНОЙ ПЕРЕМЕННОЙ ЧИСЛОВОЕ ЗНАЧЕНИЕ И НАОБОРОТ
14	ПЕРЕПОЛНЕНИЕ БУФЕРНОЙ ОБЛАСТИ ПАМЯТИ, ОТВЕДЕННОЙ ДЛЯ ХРАНЕНИЯ СИМВОЛЬНЫХ ПЕРЕМЕННЫХ. ДЛЯ РАСШИРЕНИЯ ОБЪЕМА БУФЕРА СЛУЖИТ ДИРЕКТИВА CLEAR
15	ДЛИНА СИМВОЛЬНОЙ ПЕРЕМЕННОЙ ПРЕВЫШАЕТ 255 СИМВОЛОВ
16	ВЫРАЖЕНИЕ, СОДЕРЖАЩЕЕ СИМВОЛЬНЫЕ ПЕРЕМЕННЫЕ СЛИШКОМ СЛОЖНО ДЛЯ ИНТЕРПРЕТАТОРА
17	НЕВОЗМОЖНОСТЬ ПРОДОЛЖЕНИЯ ПРОГРАММЫ ПО ДИРЕКТИВЕ CONT
18	ОБРАЩЕНИЕ К ФУНКЦИИ, НЕ ОПИСАННОЙ ОПЕРАТОРОМ DEF

ТАБЛИЦА 14

РАСПРЕДЕЛЕНИЕ ПАМЯТИ ПРИ РАБОТЕ ПРОГРАММЫ
"РЕДАКТОР-АССЕМБЛЕР"

ОБЛАСТЬ ПАМЯТИ	НАЧЕОА
КОДЫ ПРОГРАММЫ	0000H-0FFFH
БУФЕР	1000H-10FFFH
ОБЛАСТЬ ОТТРАНСЛИРОВАН- НЫХ ПРОГРАММ	1100H-18FFFH
БУФЕР ИСХОДНЫХ ТЕКСТОВ	1900H-35FFFH

чаемая конец строки. После этого можно приступить к набору текста. При вводе текста для перемещения курсора по горизонтали можно пользоваться клавишей TAB. Для выхода в режим редактора после окончания ввода нужно нажать клавишу CTR.

Для исправления ошибок в набранном тексте курсор подводится под нужное место и набирается исправленный текст. При желании можно выключить автоматическую раздвижку, нажав на клавишу AP2, а затем K2. Чтобы снова включить, нужно нажать AP2, а затем K4.

Для сдвижки и раздвижки строк можно пользоваться клавишами K2 и K4.

Для добавления новых строк в тексте необходимо поставить курсор в начале строки, перед которой нужно вставить фрагмент текста, а затем нажать клавишу AP2, затем A, при этом на экране стирается текст от места установки курсора до последней строки.

После набора фрагмента строк нужно нажать на клавишу CTR.

Для уничтожения фрагмента строк необходимо поставить курсор в начале первой стираемой строки и нажать клавиши AP2 и D, затем поместить курсор в начало первой оставаемой строки и снова нажать AP2 и D. Если требуется - нажать "CTR". Фрагмент останется.

Для просмотра текста можно пользоваться следующими комбина-

циями клавиш:

- | | |
|---------|------------------------------|
| AP2 + B | - печать первой страницы |
| AP2 + + | - печать следующей страницы |
| AP2 + - | - печать предыдущей страницы |
| AP2 + E | - печать последней страницы. |

Для поиска нужной последовательности символов необходимо нажать клавиши AP2 и I, а затем набрать нужную последовательность и нажать BK.

На экран будет выведена страница, начинающаяся строкой с заданной папкой символов.

Для продолжения поисков можно воспользоваться директивой AP2. R .

Директивами поиска цепочки символов удобно пользоваться при использовании программы - редактора в качестве справочника, каталога, телефонной книги.

Для вывода программы на магнитную ленту используется директива AP2. O при этом программа - редактор спрашивает имя выводимого файла. После ввода имени файла начнется вывод текста на магнитную ленту.

Для ввода информации с магнитной ленты используется директива AP2. I (лат.) при этом программа спрашивает имя файла и после ввода имени можно включить магнитофон на воспроизведение и ждать ввода программы. Если ввод произошел без сбоев, то на экран выведется I-я страница.

Для проверки правильности ввода можно использовать директиву AP2. V . ввести имя файла и повторить ввод. Если различий в тексте не следует, то на экран выведется I-я страница текста.

Работа программы в режиме ассемблирования. (С 800H - ЦИМОНТОРА)

Для перевода программы из режима редактирования и режим ассемблирования и обратно можно пользоваться клавишей STR .

Команды ассемблера:

- 1 - трансляция с протоколом;
- 2 - трансляция без протокола;
- 3 - составление таблицы меток;
- 4 - выход "МОНИТОР".

Оттранслированная программа находится в области памяти начинающейся с адреса 111H, конечный адрес указан в протоколе ассемблирования.

При выводе протокола ассемблирования перед полем адреса ассемблерной строки могут выдаваться сообщения об ошибках.

- 21 - двойное определение метки
- 22 - использование неопределенной метки
- 24 - использование несуществующей мнемоники
- 26 - неверно определенный операнд.

Если в одной строке встречаются 2 и более ошибок, то на экран выводится сумма кодов сообщений об ошибках. После окончания трансляции, на экран выводится сообщение:

конец ошибок 28

XXXX /YYYY /

где XXXX - конечный абсолютный адрес программы;

YYYY - конечный адрес программы, которая находится в области оттранслированных программ.

В случае, если начальный адрес программы совпадает с началом области оттранслированных программ, то XXXX будет равен YYYY, в других случаях после трансляции программы понадобится переслать программы из области оттранслированных программ по директиве T монитора.

7.7. Тест-программы

7.7.1. Программа ТЕСТ ОЗУ предназначена для отыскания неисправных ячеек в ОЗУ. Программа запускается с нулевого адреса. Если программа обнаружит неисправную ячейку, то на экране появится сообщение:

неисправна ячейка 4B 3B1F, где 4B - код, который не записался в ячейку, а 3B1F - адрес ячейки. По коду можно определить неисправный корпус микросхем ОЗУ (см. табл. 15).

Таблица 15

Код	Микросхема
01	DD 23
02	DD 24
04	DD 21
08	DD 22
10	DD 27
20	DD 28
40	DD 25
80	DD 26

7.7.2. Программа СКОРОСТЬ. Программа облегчает обмен программами между владельцами компьютеров, аналогичных "Электроника КР-02". Программа определяет оптимальные константы ввода и вывода на магнитофон. Программа загружается директивой I МОНИТОРА. После загрузки программы, на магнитофон необходимо поставить кассету с записями, константу ввода для которых необходимо определить. Затем нужно набрать с клавиатуры 6 3000, нажать кнопку ПУСК магнитофона и когда пойдет тональный сигнал перед записью, нажать клавишу BK на клавиатуре компьютера. При этом на экране появятся оптимальные значения констнт ввода и вывода.

7.8. Описание игровых программ

Описание игровых программ дано в приложении I.

8. ПОДГОТОВКА К РАБОТЕ И ПОРЯДОК РАБОТЫ

8.1. Сборка радиоконструктора

Для сборки радиоконструктора необходимо составные части радиоконструктора соединить между собой жгутом, как показано на рис. 4. Блок клавиатуры соединяется с блоком процессора гибким шлейфом следующим образом: поднять планку колодок, вставить в них гибкий шлейф, опустить планку, зафиксировать шлейф. Рекомендуется соединение колодок со шлейфом не более 10 раз.

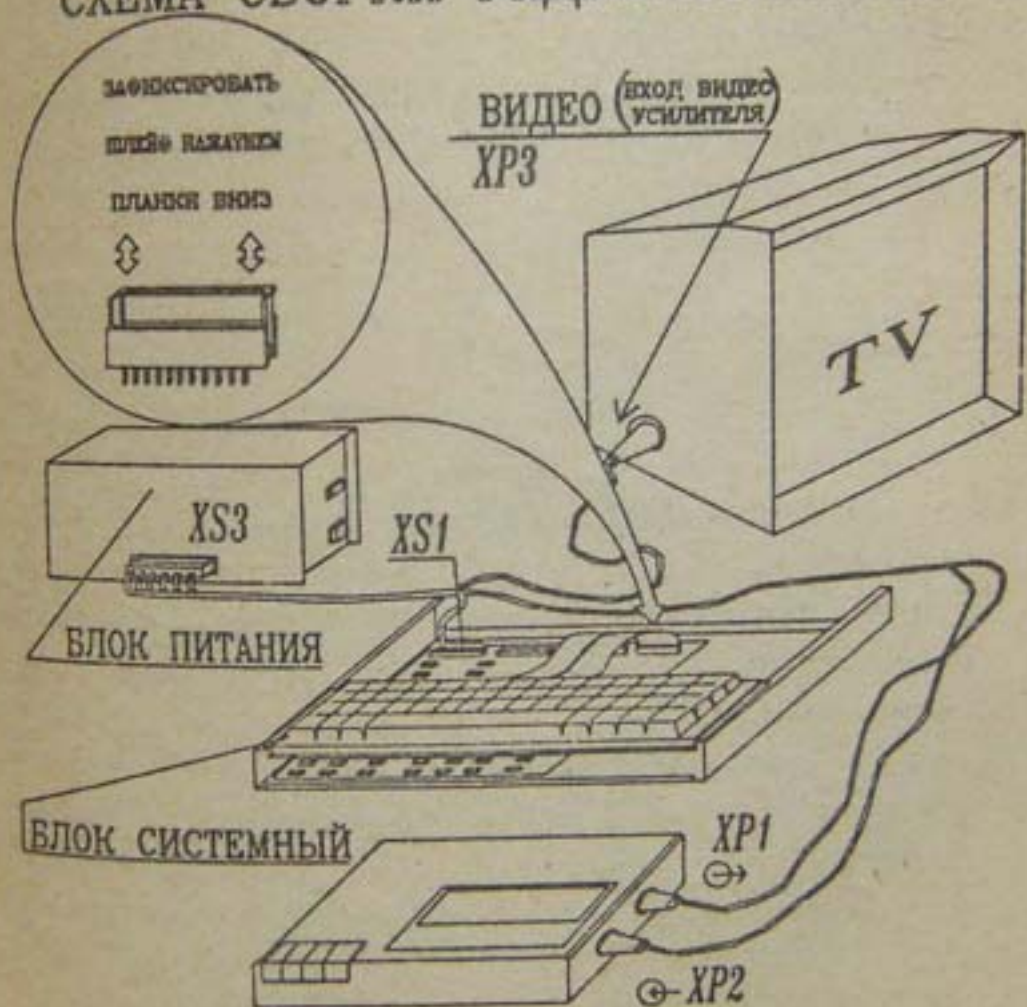
ВНИМАНИЕ! ЗАПРЕЩАЕТСЯ ВКЛЮЧАТЬ РАЗЪЕМ ХРЗ "ВИДЕО" В АНТЕННОЕ ГНЕЗДО ТЕЛЕВИЗОРА. НЕОБХОДИМО ИСПОЛЬЗОВАТЬ СПЕЦИАЛЬНОЕ ГНЕЗДО "ВИДЕОВХОД", а в тех телевизорах, где нет этого гнезда, необходимо произвести доработку телевизора, подключив разъем к входу видеосигнала телевизора, предусмотрев при этом отключение предшествующих точке подключения каскадов. НЕДОПУСТИМО ПОДКЛЮЧЕНИЕ РАЗЪЕМА ХС 1 К ХР2 (ВМЕСТО ХР1) — ЭТО ПРИВОДИТ К ВЫХОДУ ИЗ СТРОЯ МИКРОСХЕМЫ ДР 14.

ВНИМАНИЕ! ПРИ СБОРКЕ РАДИОКОНСТРУКТОРА НЕОБХОДИМО ПРИНЯТЬ МЕРЫ ПО ЗАЩИТЕ ОТ СТАТИЧЕСКОГО ЭЛЕКТРИЧЕСТВА. ДЛЯ ЭТОГО ОБЯЗАТЕЛЬНО ПРИМЕНЕНИЕ МЕТАЛЛИЧЕСКОГО ЕРАСЛЕТА, ЗАЗЕМЛЕННОГО ЧЕРЕЗ РЕЗИСТОР ОКОЛО 1 МОМ. ПРИ НАЛИЧИИ ЗНАЧИТЕЛЬНЫХ ИМПУЛЬСНЫХ ПОМЕХ В ПИТАЮЩЕЙ СЕТИ ВОЗМОЖНО СРАБАТЫВАНИЕ ЗАЩИТЫ В БЛОКЕ ПИТАНИЯ, ЧТО НЕ ЯВЛЯЕТСЯ ОСНОВАНИЕМ ДЛЯ ЗАБРАШОВЫВАНИЯ РАДИОКОНСТРУКТОРА. НЕ РЕКОМЕНДУЕТСЯ ПОДКЛЮЧАТЬ В ОДНУ РОЗЕТКУ РАДИОКОНСТРУКТОР И БЫТОВЫЕ ЭЛЕКТРИЧЕСКИЕ ПРИБОРЫ.

8.2. Включение радиоконструктора

Для включения радиоконструктора необходимо вставить вилок блока питания в сеть 220 В 50 Гц. После этого необходимо нажать

СХЕМА СБОРКИ РАДИОКОНСТРУКТОРА



кнопку СЕТЬ в блоке питания. Затем последовательно необходимо нажать кнопку ПУСК в блоке питания и кнопку СЕРОС на блоке клавиатуры. При этом на экране в левом верхнем углу должна появиться надпись "РАДИО-86РК" под ней стрелки \rightarrow и мигающий курсор. При нажатии на клавиши на экране должны появляться соответствующие знаки. Допускается появление надписи "РАДИО-86РК" без нажатия на кнопки ПУСК и СЕРОС.

8.3. Ввод символов с клавиатуры

Для ввода символов русского и латинского алфавита необходимо нажать клавишу выбора алфавита РУС или LAT. При нажатии на клавишу РУС светодиод РУС должен загореться. Для перехода в латинский регистр необходимо нажать клавишу LAT.

Клавиши \uparrow и \downarrow служат для переключения цифрового и символического регистров. Другими словами для того, чтобы ввести символы $! : " ; \& \% \square \circ - () \backslash _ _ ? > < :$ необходимо нажать клавишу \downarrow . Вернуться к верхнему регистру можно, нажав клавишу \uparrow .

Для ввода псевдографических символов необходимо нажать и отпустить клавишу UPR. При этом можно вводить псевдографические символы согласно табл. 16.

Цифры, а также знаки $+$, $-$, $:$, $..$ можно ввести с группы клавиш, расположенных в правой части клавиатуры.

Ввод строки обычно завершается нажатием клавиши BK.

Клавиша RC - перевод строки

Клавиша TAB в некоторых программах (например в редакторе текстов) предназначена для горизонтального перемещения курсора на несколько позиций.

Клавиши \uparrow , \downarrow , \leftarrow , \rightarrow служат для управления курсором.

Клавиша \curvearrowright предназначена для перемещения курсора в левый верхний угол экрана.


ТАБЛИЦА 16


Клавиша	Отображаемый символ	Клавиша	Отображаемый символ	Клавиша	Отображаемый символ
Ц - С		А - А		Э - /	
К - К		П - Р		Я - Q	
Е - Е		Р - R		Ч - 7	
Н - N		О - O		С - S	
Щ - J		У - U		И - I	
Ф - F		Д - D		Т - T	
В - W		Е - V		В - B	

Клавиша CTR предназначена для стирания информации с экрана.

Функциональные клавиши K1, K2, K3, K4, K5 используются в программах по усмотрению разработчика программ (см. например интерпретатор языка Бейсик).

Следует заметить, что после нажатия клавиши СБРОС компьютер работает под управлением программы МОНИТОР. Эта программа имеет длину буфера строки 32 символа, поэтому при вводе 33-его символа программа печатает знак вопроса, а МОНИТОР переходит в режим ввода новой строки.

Клавиша BK (возврат каретки) в клавиатуре "Электроника MS 7001" обозначена символом .

Клавиша 3B (забой) обозначена символом .

5.4. Загрузка и запуск программ

После нажатия кнопки СБРОС компьютер работает по программе МОНИТОР, при этом на экране появляется надпись "РАДИО-86PK", под ней стрелки и мерцающий курсор. Это говорит о том, что компьютер готов к вводу директивы МОНИТОРА. Для ввода программы с магнитной ленты необходимо воспользоваться директивой I МОНИТОРА. Для этого необходимо подготовить магнитофон к работе, поставить в него кассету с записями программы и нажать клавишу "Воспроизведение". После этого необходимо набрать директиву I на клавиатуре, и дождавшись начала программы, нажать клавишу BK. Начало программы можно определить по тональному сигналу, который идет перед каждой программой.

Во время ввода программы изображение на экране телевизора пропадает, а по окончании ввода появляется вновь. При этом на экране появляется начальный и конечный адрес введенной программы.

и контрольная сумма. Если контрольная сумма не совпадает с указанной на ленте, то на экране появляется знак вопроса, значение подсчитанной контрольной суммы, а под ней значение контрольной суммы, введенной с ленты. Для запуска программы нужно набрать директиву G МОНИТОРА и нажать клавишу ВК, если программа расположена с нулевого адреса. Если же программа находится в памяти, начиная с другого адреса, то нужно после директивы набрать этот адрес. Например: G 1100 ВК. Если программа не вводится с магнитофона в течение трех раз, то следует искать причину в магнитофоне (повышенный уровень детонации, слишком слабый или слишком большой уровень записи, неисправность кассеты, перекус головки, загрязнение головки). Разобраться в причинах нечеткого считывания программ вам поможет программа, приведенная в журнале "РАДИО" за 1987 № 4. Для определения константы для записей, сделанных с другого компьютера, служит программа, описанная в разделе 7.7.2. После определения константы ввода, её следует записать в ячейку 0362FH и снова попытаться ввести программу. Следует заметить, что после нажатия кнопки СБРОС в этой ячейке восстанавливается значение константы равное 2AH.

Вывод программы производится по директиве O МОНИТОРА.

При выводе программ необходимо указать начальный и конечный адреса программ, например:

O 100, 110

Затем подготовить магнитофон к записи, причем уровень сигнала нужно установить на 5-10 дБ выше номинального. После этого выключить магнитофон и нажать кнопку ВК. После вывода программы на экране появится подсчитанная контрольная сумма, например:

→ O 100, 110

0100

0110

XXXX - контрольная сумма

8.5. Загрузка и запуск программ на БЕЙСИКЕ

После запуска интерпретатора языка БЕЙСИК директивой **G-MONITORA** в левом верхнем углу появляется надпись "**РАДИО-86РК**" **BASIC**, а под ней стрелка \rightarrow , указывающая на то, что интерпретатор готов к вводу директивы. После этого можно набирать текст программы на языке БЕЙСИК или загрузить программу с магнитной ленты. Для этого необходимо набрать директиву **MCLOAD**, подготовить магнитофон, включить воспроизведение и во время тонального сигнала нажать клавишу **BK**. После ввода программы её можно запустить директивой **RUN BK**.

Просмотреть текст программы можно по директиве **LIST**. Если необходимо изменить текст строки, то нужно, набрав номер строки, заново ввести текст строки. Если необходимо стереть всю строку, то нужно набрать номер строки и нажать клавишу **BK**. Для вывода программы на магнитную ленту служит директива **MSAVE**. Для записи программы на магнитную ленту необходимо подготовить магнитофон к записи, набрав директиву **MSAVE**, затем нажать кнопку **ПУСК** магнитофона и клавишу **BK**.

9. ТЕХНИЧЕСКОЕ ОБСЛУЖИВАНИЕ ИЗДЕЛИЯ

Исправный радиоконструктор технического обслуживания не требует. Исключение составляет смена предохранителя, который находится на передней панели блока питания.

10. ГАРАНТИИ ИЗГОТОВИТЕЛЯ

Радиоконструктор "Электроника КР-02" соответствует утвержденному образцу.

Изготовитель гарантирует соответствие изделия требованиям ТУ 0.303.000 при соблюдении владельцем правил эксплуатации, изложенных в руководстве по эксплуатации.

Гарантийный срок эксплуатации радиоконструктора "Электроника КР-02" 12 месяцев со дня продажи через розничную торговую сеть.

При отсутствии даты продажи и штампа магазина в гарантийном талоне гарантийный срок исчисляется со дня выпуска изделия предприятием-изготовителем.

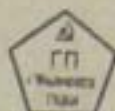
Адрес для предъявления претензий по качеству:

433008, г. Ульяновск, а/я 214.

Без предъявления гарантийного талона, при нарушении сохранности пломб, при внесении изменений в монтажную схему радиоконструктора претензии к качеству работы не принимаются.

Завод не высылает дополнительные микросхемы ОЗУ, ПЗУ и кассеты с программами.

Изготовитель оставляет за собой право вносить изменения в изделия, не ухудшающие его качества.



Цена 395 руб.

Прейскурант №

ГАРАНТИЙНЫЙ ТАЛОН

Заполняет завод-изготовитель

Радиоконструктор "Электроника КР-02"

зав. № 94669

Блок питания

зав. № 9005027

Блок процессора

зав. № 9004559

Блок клавиатуры

зав. № 94669

Агуд

зав. № 9004185

Кассета с записью

зав. № 9004559

Дата выпуска 27.05.90г

представитель ОТК завода - изготовителя

штамп ОТК

Радиоконструктор выпускается под контролем Госприёмки.

Адрес для предъявления претензий по качеству:

432008, г. Ульяновск, а/я 214

Заполняет торговое предприятие

Дата продажи 06.10.90г
число ЭЛЕКТРОНИКА

Продавец

Склад № 4

Москва, Ленинский проспект, дом 60
подпись или штамп

Штамп магазина

и.п. Петров

II. ПЕРЕЧЕНЬ ПРИНЯТЫХ СОКРАЩЕНИЙ

- ОЗУ - оперативное запоминающее устройство
- ПЗУ - постоянное запоминающее устройство
- ВКУ - видеоконтрольное устройство
- порт ВВ - порт ввода - вывод
- БИС - большая интегральная схема
- ПДП - прямой доступ к памяти
- АЛУ - арифметико-логическое устройство

9. СПИСОК ДОПОЛНИТЕЛЬНОЙ ЛИТЕРАТУРЫ

Для более детального ознакомления с устройством микроЗЕМ и основами программирования рекомендуем ознакомиться со следующими публикациями журнала "Радио".

Напоминаем, что описанный в журнале компьютер "Радио-86РК" является аналогом компьютеров "Электроника КР-01", "Электроника КР-02" и "Электроника КР-03".

Устройство и принципы работы компьютера подробно освещены в следующих статьях:

Д. Горшков, Г. Зеленко, П. Озеров, С. Попов

"Персональный радиолобительский компьютер "Радио -46РК".

Радио 1986, № 4, с. 24-26

Радио, 1966, № 5, с. 31-34

Радио, 1966, № 5, с. 25-28

Радио, 1986, № 7, с. 26-28

Радио, 1985, № 8 с. 23-26

Радио, 1966, № 9, с. 27-28

Радио, 1966, № 12, с. 19

Полезно также ознакомиться со следующими публикациями, в которых описаны различные усовершенствования компьютера.

Д. Лукьянов, Радио о "Радио-БФР" с. 32, 13-ая строчка сверху вместо
не сбалансировать K1651H4. На крышке схемы преобразователя с. 34
Радио. 1966. № 10, с. 32-35. Вместо SA1 заменить SA1 ферритом от ради-
отелера с катушкой A10, SA2-A9, ..., SA11-AO.

А. Крылов. Блок питания компьютера "Радио-86РК"

Радио, 1986. № 11, с. 26-29.

Радио, 1986, № 12, с. 17-18.

А. Сергеев. Еще о замене микросхем в "Радио - обРК"

Радио, 1987, в 6, с. 34-35.

Д. Лукьянов, А. Богдан, "Радио -66-РК" - программатор ИЗУ.

Радио, 1987, № 6 с. 21-23.

Радио, 1987, № 9, с. 24-26, 56-57.

Радио, 1988, № 2, с. 24-28

И. Крылсва. Таймер КР580ВИ53 в "Радио-86-РК".

Радио, 1987, № II, с. 35-39.

А. Долгий. Если нет КР580ВГ75 ...

Радио, 1987, № 5, с. 22-24

Радио, 1987, № 6, с. 33-35

А. Соргеев. Динамическое питание ПЗУ.

Радио, 1987, № 12, с. 26-27.

С. Шпов. ПЗУ для БЕЙСИКА.

Радио, 1987, № 3, с. 32

А. Андреев. Программный "Синтезатор речи для "Радио-86РК".

Радио, 1987, № 12, с. 27.

Радио, 1988, № 2, с. 29-30.

Работа отдельных микросхем компьютера будет более понятна, если Вы ознакомитесь с серией статей, в которых описаны устройство и работа компьютера "Микро-80".

Авторы серии - Г. Зеленко, В. Панов, С. Полов.

Первый шаг

Радио, 1982, № 9, с. 33-36.

Процессорный модуль микро-ЭВМ.

Радио, 1983, № 2, с. 40-43.

Модуль памяти.

Радио, 1983, № 3, с. 30-34.

Дисплейный модуль.

Радио, 1983, № 7, с. 23-27

Радио, 1983, № 8, с. 26-27.

Модуль сопряжения.

Радио, 1983, № 9, с. 32-36

Модуль динамического ОЗУ.

Радио, 1983, № 10, с. 28-31.

Справочные данные по микропроцессорным БИС серии КР580 помещены в следующих журналах:

Радио, 1984, № 9, с. 59-60.

Радио, 1984, № 10, с. 59-60.

Радио, 1984, № 11, с. 59-60.

Радио, 1984, № 12, с. 55-56.

Радио, 1985, № 4, с. 59-60.

Подробная методика наладки и ремонта компьютера изложена в статье Д. Горшкова, Г. Зеленко, Ю. Озерова

"Ещё раз о наладке "Радио - 86РК".

Радио, 1988, № 7, с. 29-32.

Если Вы желаете научиться программировать, а также познакомиться с текстами программ, то Вам помогут следующие публикации:

Радио, 1979, № 5, с. 29-32.

Радио, 1979, № 6, с. 26-28.

Радио, 1979, с. 24-26.

Радио, 1979, № 8, с. 26-28.

В вышеуказанной серии статей В. Кальнина содержатся кроме основ программирования также интересные сведения о работе некоторых логических узлов ЗВМ.

С программированием в своих публикациях знакомит Г. Зеленко, В. Панов, С. Попов.

Система команд микропроцессора КР580ИК80.

Радио, 1982, № 10, с. 24-28.

Знакомство с программированием.

Радио, 1982, № 11, с. 38-41.

Радио, 1982, № 12, с. 31-34.

- В. Наугадов. "БЕЙСИК-Сервис" для "Радио-86РК".
Радио, 1988, № 1, с. 22-25.
- В. Барчуков, Е. Фадеев.
Дизассемблер для "Радио-86РК".
Радио, 1988, № 3, с. 27-31.
- Д. Лукьянов. Перемещающий загрузчик.
Радио, 1988, № 3, с. 32-33.
- А. Долгий "Радио-86РК" + программа = мультиметр.
Радио, 1988, № 4, с. 24-27.
"Радио-86РК". Справочные таблицы.
Радио, 1988, № 4, с. 27
- А. Пекин. Программа обработки текстов на БЕЙСИКЕ.
Радио, 1988, № 4, с. 28-29.
- А. Пекин, Ю. Солнцев.
Играем в "Ради".
Радио, 1988, № 5, с. 27-28.
- Радио, 1988, № 6, с. 26-27.
- Д. Горшков, Г. Зеленко.
О переносимости программ.
Радио, 1988, № 5, с. 29-30.
- Что такое "контрольная сумма"?
Радио, 1988, № 7, с. 33-34.
- А. Сорокин. Компьютер помогает настроить телевизор.
Радио, 1988, № 7, с. 33-34.
- В. Барчуков, Е. Фадеев.
БЕЙСИК "Микроп"
Радио, 1988, № 8, с. 37-43.

Программное обеспечение микро-ЭВМ.

Радио, 1983, № II, с. 31-34

Директивы запуска и отладки программ.

Радио, 1983, № 12, с. 24-27.

БЕЙСИК для "Микро-80".

Радио, 1985, № I, с. 33-34,

Радио, 1985, № 2, с. 39-42.

Радио, 1985, № 3, с. 42-45.

Программирование на БЕЙСИКЕ.

Радио, 1986, № 2, с. 34-38.

Радио, 1986, № 3, с. 50-52.

А. Долгий. БЕЙСИК для "Радио-86РК".

Радио, 1987, № I, с. 31.

Радио, 1987, № 8, с. 57.

А. Долгий. Компьютерные игры.

Радио, 1987, № 2, с. 23-26, 38

Радио, 1987, № 3, с. 30-32.

Д. Горшков, Г. Зеленко. Немного о программировании.

Радио, 1987, № 4, с. 17-21.

А. Долгий. О вводе данных с магнитной ленты.

Радио, 1987, № 4, с. 22-24.

В. Барчуков, Г. Зеленко, Е. Фадеев.

Редактор и Ассемблер для "Радио - 86РК".

Радио, 1987, № 7, с. 22-26

В. Барчуков. Е. Фадеев.

Программа - модификатор.

Радио, 1987, № 8, с. 24.

А. Сорокин. Вечный календарь.

Радио, 1987, № 12, с. 28.

В заключение сообщаем, что в журнале "Радио" постоянно публикуется материал, который будет Вам полезен в дальнейшем для изучения как устройства компьютера, так и его программного обеспечения.

В новой версии "редактора":

$AP2 + LX = Y$, x - замещающий символ (символ)
 y - замещаемое слово

Дифференциал $AP2 + D$ - начало указанного
 $AP2 + E$ - конец, окончание

$AP2 + S$ - замещение в программе.
 Дифференциал (кэфф-т)
 на начало стр. 2; $AP2 + S$; дифф-т
 на стр. 2, стр. 2-е же замещено
 словом; $AP2 + E$.

$AP2 + T$ - замена дифференциала (кэфф-т)
 стр. 2-е же замещено словом; стр. 2-е же
 слово, стр. 2-е же замещено словом; стр. 2-е же

Расстояние стр. 2-е же замещено словом; стр. 2-е же
 слово, стр. 2-е же замещено словом; стр. 2-е же

Символ: то же <BK>

В начале ввода слова дифференциал
 стр. 2-е же замещено словом; стр. 2-е же

Конец тестового файла.

0D0D FF 49 8A

~~0000~~
0775 - КРОНИКО
~~0000~~
008D - ТЕТРИС

О Родакторе

При входе с и/фамилией можно вычислить адрес
из тех фрагментов, к-е. Входит директива AP2+M
Модуль diff-бу памяти «меморизация» можно от-
менить, поставив «СТР»

Возмож в ASSM * МИКРОН: "СТР" - из функции фидерной,
"СТР"; "СТР" - из функции входы.

Возмож в МОНИТОР: УС + Е (адресация)

Записыватель

D - записыватель
M - восстановление
A - запись адресов

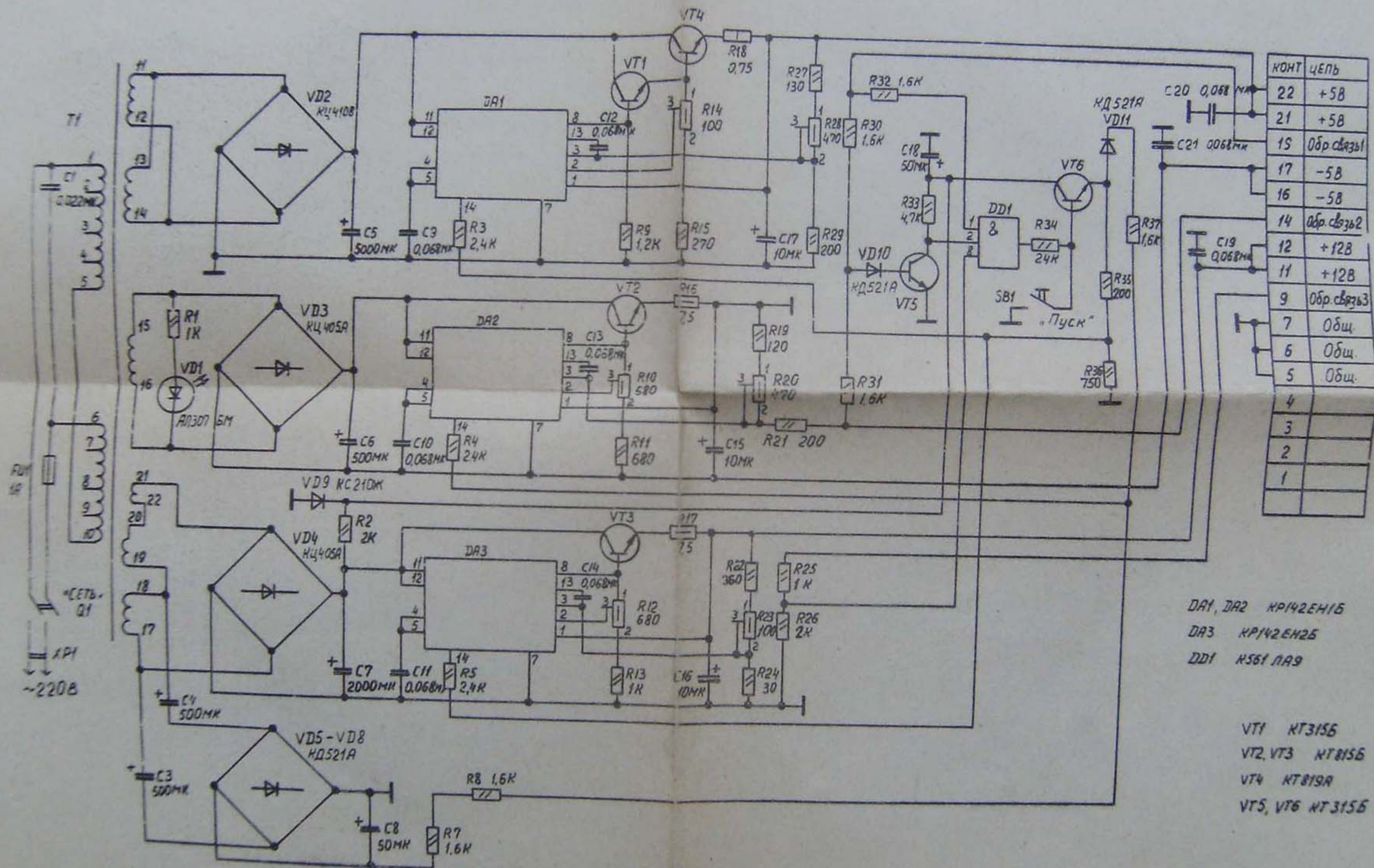
(Замечание - записыватель "ассиметрический" и "фазовый")

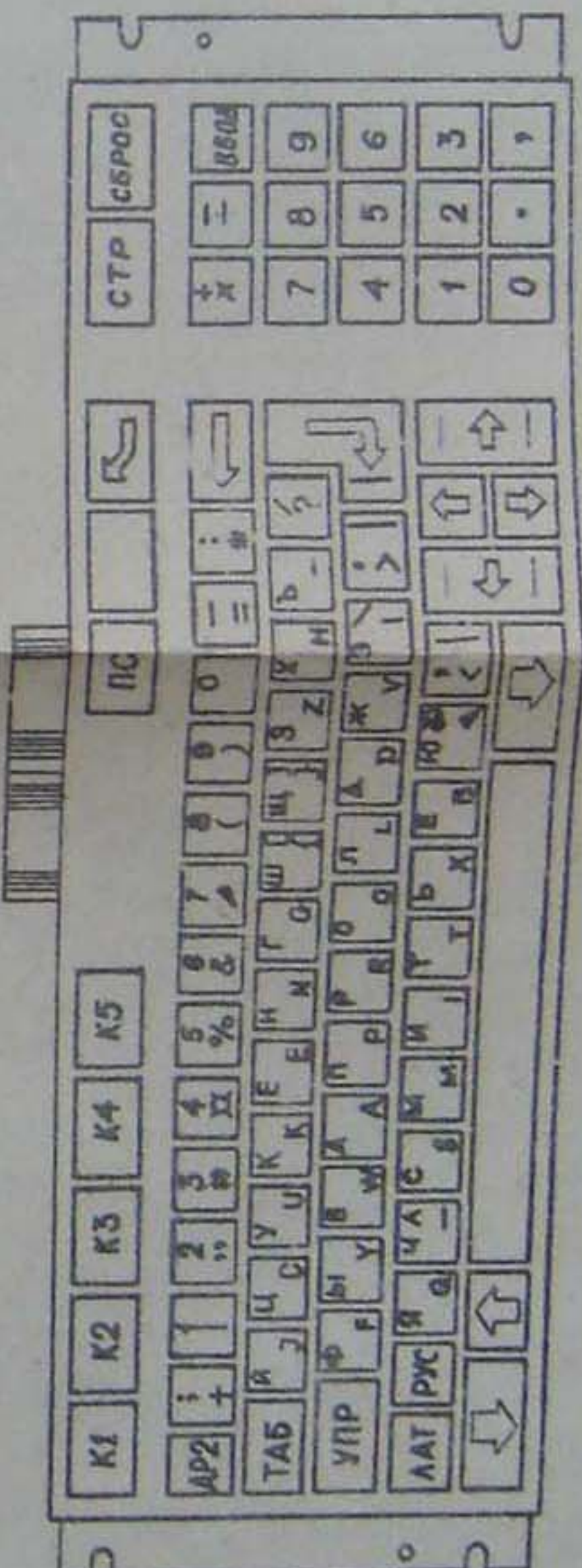
COMPUTER

КРО2



■ ЭЛЕКТРОНИКА





9-значный код

6002M-терминал "Битовый"

Приложение 5

Схема электрическая принципиальная блока клавиатуры



Вопрос В № DD20

