

ПЕРСОНАЛЬНАЯ ЭВМ ПК-01 «ЛЬВОВ». РУКОВОДСТВО ПРОГРАММИСТУ

589.0196339 00001-01 35 01

Приложение 3

ПЕРСОНАЛЬНАЯ ЭВМ ПК-01 "ЛЬВОВ"  
РУКОВОДСТВО ПРОГРАММИСТУ

# 1. Архитектура ПК-01

## 1.1 Общие особенности.

Домашняя микро-ЭВМ ПК-01 "Львов" построена на микропроцессоре (МП) КР580ВМ80А (Intel 8080), который работает с тактовой частотой 2.5 МГц, что обеспечивает быстродействие 500 тысяч коротких операций регистр-регистр в секунду. Особенности этого микрокомпьютера являются страничная организация ОЗУ и полное отсутствие системы прерываний (в стандартной конфигурации). Архитектурные особенности ПК-01 обуславливают его несовместимость (без соответствующих эмуляторов) с микро-ЭВМ линии РАДИО-86РК/ОРИОН-128, построенных на том же МП. Объем ПЗУ (ROM) ПК-01, выполненного на 8 УФ-РПЗУ К573РФ2 составляет 16К. Общий объем ОЗУ (RAM), выполненного на МС К565РУ5 составляет 64К, причем 16К из них предназначены для хранения изображения (Video - RAM) и не входят в обычном режиме в основное адресное пространство. Формирование изображения и регенерация динамического ОЗУ осуществляются специальными контроллерами, построенными на ИМС средней степени интеграции серий 155 и 555. Интерфейс с устройствами ввода-вывода, управление палитрой изображения и распределением адресного пространства ОЗУ (подключение Video-RAM к шинам МП) осуществляются с помощью двух БИС параллельных программируемых адаптеров (ППА) КР580ВВ55.

## 1.2 Распределение адресного пространства.

FFFFH	-----	65535
ROM	Драйверы устройств, системные под-	
16K	программы, знакогенератор, BASIC	
C000H	-----	49152
BFFFH	-----	49151
SYSTEM RAM	Системная область	
4K		
B000H	-----	45056
AFFFH	-----	45055
RAM	Буфер символьных переменных BASIC,	
	стек, свободная область	
8000H	-----	32768
7FFFH	-----	32767
RAM/Video	Свободная область/Экран	
16K		
4000H	-----	16384
3FFFH	-----	16383
RAM	Свободная область	
10K		
1723H	-----	05923
1722H	-----	05922
RAM	Интерпретатор BASIC	
6K		
0006H	-----	00006
0005H	-----	00005
RAM	Свободная область	
0000H	-----	00000

## Распределение адресного пространства в системной области ОЗУ

BFFFH	-----	49151
Таблица переходов на драйверы устройств		
BFEBH	-----	49131
BFEAH	-----	49130
Системные переменные		
BE00H	-----	48640
BDFH	-----	48639
Свободная область		
B800H	-----	47104
B7FFH	-----	47103
Знакогенератор: CHR 128-255		
B400H	-----	46080
B3FFH	-----	46079
Знакогенератор: CHR 000-127		
B000H	-----	45056

### 1.3 Перезапуск системы.

При нажатии клавиши [СБР] происходит формирование сигнала RESET для МП и его перезапуск. Дальнейшие действия системы определяются клавишей, нажатой одновременно со [СБР]. Возможны следующие варианты :

- [СУ ]+[СБР] : Выполняется тестовая подпрограмма, размещенная в ПЗУ. При неисправности выдается звуковой сигнал, при нормальном завершении происходит "холодный" рестарт системы ( аналогично [СБР] ).
- [Т/Т]+[СБР] : Производится подсчет КС ПЗУ по отдельным корпусам К573РФ2 и в целом, после чего запуск системы не происходит. При нажатии [РУС],[Ф] выдается информация о разработчике содержимого ПЗУ.  
Вероятные значения КС ПЗУ ( в ПК разного времени выпуска они отличаются ) следующие :  

03348A  
0380DD  
037E80  
03F6A3  
042870  
039C0B  
046594  
038D67  
1DE200
- [С/Ц]+[СБР] : Выполняется "теплый" рестарт системы: обнуляется область системного ОЗУ В000Н-0000Н, перезагружаются из ПЗУ знакогенератор (CHR 000-127), системные переменные, таблица переходов на драйверы устройств ввода-вывода, после чего происходит переход по адресу, размещенному в ячейках памяти (ЯП) BFEC/BFED. После "холодного" старта там находится адрес старта BASIC- системы D7D9, но пользователь может изменить это значение, обеспечив запуск по [С/Ц]+[СБР] любой другой программы.

При "горячем" перезапуске содержимое всех прочих областей ОЗУ не изменяется.

При нажатии [СБР] с любой другой клавишей или в одиночку происходит "холодный" перезапуск системы. Кроме действий "горячего" перезапуска выполняются также восстановление стандартного значения BFEC/BFED, обнуление области 0000H-0000H, перезагрузка BASIC из ПЗУ в ОЗУ, формирование заставки ( при этом используется область 8000H-0000H ) и запуск BASIC- системы.

#### 1.4 Порты ввода-вывода, клавиатура.

Для обеспечения обмена информацией с внешними устройствами в компьютере предусмотрена БИС ППА КР580ВВ55 ( D30 ), которая содержит три 8-разрядных порта ввода-вывода А, В, С. Они выведены на разъем "ВНЕШ - 2" ( распайку см. в "Руководстве по эксплуатации" ). Программно к каждому из портов можно обратиться по адресам :

Порт А ( только на вывод ) - C0H ( 192 )  
Порт В ( ввод или вывод ) - C1H ( 193 )  
Порт С ( ввод и вывод ) - C2H ( 194 )  
Регистр управляющего слова - C3H ( 195 )

Порт А предназначен только для вывода информации ( стандартно- на принтер ) и буферизован. Порт В предназначен для формирования палитры изображения, но может быть использован и для обмена с внешними устройствами ( при этом будет происходить "мелькание" палитры изображения ). Порт С в компьютере задействован для управления принтером, обмена с магнитофоном и управлением картой памяти ( подключение / отключение экранного ОЗУ ). Стандартно порты А, В и С(0-3) запрограммированы на вывод, С(4-7) - на ввод.

Назначение бит порта С (C2H ) :  
-----T---T---T---T---T---T---T---T---T---  
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |  
L---+---T-+---+---T-+---+---T-+---T-+---T-+---  
AC ЧМЛ SC | ЗМЛ  
BUSY STROBE L----- VIDEO  
ГОТОВ ПЕЧАТЬ

Для подключения клавиатуры в ПК также применена БИС КР580ВВ55 ( D1 ). Ее адреса :

Порт А ( только на вывод ) - D0H ( 208 )  
Порт В ( только на ввод ) - D1H ( 209 )  
Порт С ( ввод и вывод ) - D2H ( 210 )  
Регистр управляющего слова - D3H ( 211 )

Подключение портов к клавиатуре ПК-01 показано на рисунке. Принцип опроса клавиатуры заключается в следующем : последовательной подачей логического "0" на каждый из разрядов портов А и С(0-3) разрешается опрос соответствующего полуряда клавиатуры. Результат опроса содержат порты В и С(4-7) для основной и функциональной клавиатуры соответственно.

Основная клавиатура

Порт D0H ( 208 ) - выбор полуряда ( вывод )										Порт D1H ( 209 )		
7	6	5	4	3	2	1	0	BIT		опрос полуряда		
7FH	BFH	DFH	EFH	F7H	FBH	FDH	FEH	HEX		( ввод )		
127	191	223	239	247	251	253	254	DEC	HEX	BIT		
HP	;		СТР	ПРБ	R	G	6	254	FEH	0		
Q	РУС		(G)	В	O	[	7	253	FDH	1		
^	СУ	J	(B)	@	L	]	8	251	FBH	2		
X	Р	N	5	ВР	ЗБ	ВК	ГТ	247	F7H	3		
T	A	E	4	_____	.	ПС	ТАБ	239	EFH	4		
I	W	K	3	ЛАТ	\	:	-	223	DFH	5		
M	Y	U	2	/	V	H	0	191	BFH	6		
S	F	C	1	,	D	Z	9	127	7FH	7		

#### Функциональная клавиатура

Порт D2 ( 210 )				
3	2	1	0	
ПРА	ДИА	П/Д	ПЧ	4
ВРХ	F5	F0	CD	5
ЛЕВ	F4	F1	ДИН	6
НИЗ	F3	F2	(R)	7

Значения, выводимые и вводимые с портов представлены для случая нажатия только одной клавиши. Подачей "0" одновременно на несколько разрядов портов А или С(0-3) можно разрешить опрос нескольких полурядов, но различить полуряды в таком случае невозможно.

При необходимости взаимодействия с большим количеством нестандартных внешних устройств в ПК-01 имеется возможность подключения двух дополнительных ППА КР580ВВ55 по адресам E0/E1/E2/E3 и F0/F1/F2/F3 соответственно. Необходимые для этого сигналы выведены на разъем "ВНЕС - 1".

#### 1.5 Подключение принтера.

В компьютере предусмотрено стандартное подключение EPSON -

совместимого принтера по интерфейсу "ИРПР" или "CENTRONIX" ( ИРПР-М ). Распайка разъема " ВНЕШ - 2 " для подключения принтера "ROBOTRON CM 6329.01M" по "ИРПР" приведена на рисунке. При подключении принтера следует помнить о том, что длина соединительного кабеля для "ИРПР" не должна превышать 15 м. и каждая сигнальная жила кабеля (С) должна быть переплетена со своей "земляной" (З) ( для плоского кабеля - ЗСЗС....СЗСЗ ), причем "земляные" жилы должны быть подключены к 0V с обоих концов кабеля.

ПРИНТЕР			ВНЕШ -2		
-----Т-----			-----Т-----		
1	GND	-----	0V		
2	SC	<----- ПЕЧАТЬ -----	C2	6	
3	AC	----- ГОТОВ ----->	C6	21	
4					
5	D0	-----	A0	39	
6	D1	-----	A1	37	
7	D2	-----	A2	35	
8	D3	-----	A3	33	
9	D4	-----	A4	31	
10	D5	-----	A5	29	
11	D6	-----	A6	27	
12	D7	-----	A7	25	
13					
L-----+			L-----+		

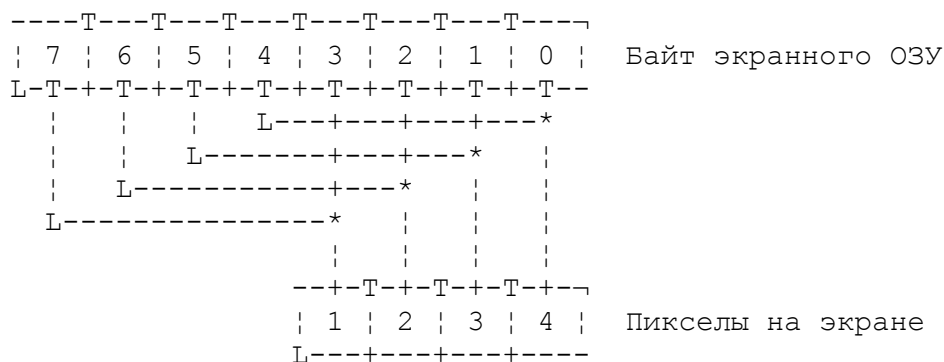
Вывод информации осуществляется следующим образом: компьютер опрашивает сигнал "ГОТОВ" от принтера; если этот сигнал не равен "0", то компьютер ожидает. Как только "ГОТОВ" становится равным "0", компьютер выдает на порт А символ в коде КОИ-7 ( КОИ-8 ) в инверсном ( прямом ) виде, а затем переводит сигнал "ПЕЧАТЬ" в "0", что является для принтера командой к приему и печати символа. Принтер переводит "ГОТОВ" в "1", а компьютер, приняв этот сигнал переводит "ПЕЧАТЬ" в "1", после чего цикл повторяется.

## 1.6 Видеоконтроллер.

Для обеспечения отображения информации на экране видеомонитора ( телевизора ) в компьютере имеется видеоконтроллер, который осуществляет периодическое считывание информации из экранного ОЗУ и формирование видеосигнала.

В обычном режиме экранное ОЗУ отключено от шин МП и программный доступ к нему невозможен. Для работы МП с этой областью ОЗУ она может быть подключена вместо участка основного ОЗУ по адресам 4000H-7FFFH ( 16384-32767 ) с помощью бита 1 порта С2Н ( 194 ) ( 0 - Open Video RAM, 1 - Close Video RAM ). Однако следует помнить, что при таком переключении от шин МП отключается область основного ОЗУ 0000H-7FFFH ( 00000-32767 ), поэтому процедуры, работающие при открытой видеопамати нельзя размещать ниже 8000H ( 32768 ). Все основные подпрограммы ПЗУ, работающие с экраном "не забывают" закрыть видеопамать перед возвратом и с ними такой проблемы не возникает.

Формирование видеосигнала в ПК-01 выполняется так, что один байт экранного ОЗУ кодирует цвета четырех соседних пикселей по горизонтали. Принцип кодировки цветов показан на рисунке :



Наличие "1" в битах 7-4 задает логический цвет 1 для соответствующих пикселей, в битах 3-0 - цвет 2, в паре битов (7-3, 6-2, 5-1, 4-0) - цвет 3, иначе соответствующие пиксели имеют цвет фона ( 0 ). Соответствие логических цветов 0,1,2,3 реальным ( палитра ) определяется состоянием порта C1H ( 193 ).

## 2. Системные подпрограммы ПЗУ.

### 2.1 Таблица переходов

Основная таблица переходов на подпрограммы, реализующие важнейшие системные функции и образующие Базовую Систему Ввода-Вывода ( БСВВ или BIOS или МОНИТОР ) расположена в ПЗУ по адресам F800H-F838H ( 63488-63544 ) и в ОЗУ по адресам BFEBH-BFFDH ( 49131-49149 ) и имеет следующий вид :

----- ROM -----	----- RAM -----	Name
F800H    JMP   BFEBH	BFEBH    JMP   D7D9H	BASIC
F803H    JMP   BFEEH	BFEEH    JMP   E800H	KEY
F806H    JMP   BFF1H	BFF1H    JMP   E843H	KY*
F809H    JMP   BFF4H	BFF4H    JMP   DF68H	TTY
F80CH    JMP   BFF7H	BFF7H    JMP   E236H	PRN
F80FH    JMP   BFFAH	BFFAH    JMP   E222H	LST
F812H    JMP   BFFDH	BFFDH    JMP   E888H	STK
F815H    JMP   FF90H		CHECKSUM
F818H    JMP   F067H		VADDR
F81BH    JMP   DEBFH		BEEP
F81EH    JMP   DE94H		SOUND
F821H    JMP   F048H		PSET
F824H    JMP   F112H		LINE
F827H    JMP   F1A5H		BOX
F82AH    JMP   F1DAH		FIL_BOX
F82DH    JMP   DF02H		LOCATE
F830H    JMP   E774H		PAINT
F833H    JMP   DE3CH		COLOR
F836H    JMP   EBD0H		CLS

Приведенные здесь условные названия системных подпрограмм не следует отождествлять с одноименными операторами BASIC 2.0, хотя операторы BASIC и представляют собой обращения к этим подпрограммам.

Таким образом, используя размещенную в ОЗУ часть таблицы,

пользователь может переопределить основные системные процедуры ввода-вывода, подключив драйверы нестандартных внешних устройств, например драйвер, позволяющий использовать в качестве текстового принтера телетайп или электрическую печатающую машинку ( типа CONSUL ).

Стандартная BASIC - система всегда работает с устройством вывода LST.

Более подробное описание системных подпрограмм ПЗУ приводится ниже. Следует также отметить, что лишь некоторые из системных подпрограмм ПК-01 совпадают по адресам вызовов с аналогичными в микрокомпьютерах линии "РАДИО-86РК/ОРИОН-128".

## 2.2 Ввод-вывод символов.

### 2.2.1 Ввод символа с клавиатуры с обработкой специальных функций ( KEY ).

Точка входа F803H - BFEEH - E800H ( 63491 - 49134 - 59392 ). Подпрограмма ожидает нажатия символьной клавиши и возвращает код символа в КОИ-7 в регистре А МП. Введенный символ на устройства вывода не передается. Используются системные переменные KYB\_MODE ( BE1DH (48669)) и KYB\_BEER ( BE1EH (48670)).

Под системной переменной здесь и далее понимается зарезервированная ячейка ( ячейки ) системного ОЗУ, значение которой является входным, выходным, либо промежуточным параметром при работе одной из системных подпрограмм.

Структура системной переменной KYB\_MODE приведена на рисунке:

```

----T---T---T---T---T---T---T---T---
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
L-T-+---+---+---+---+---+---T-+-T-+-T-+
|                                     |   |   L----- 1 - BP
L 1 - РУС                           |   |   L----- 1 - HP
0 - ЛАТ                             L----- 1 - СУ

```

Признаки BP, HP и СУ не фиксируются, а РУС/ЛАТ -фиксируется. Системная переменная KYB\_BEER определяет наличие звукового сигнала нажатия клавиши ( 00 - откл., FFH (255) - вкл., возможны промежуточные значения). Тональность сигнала зависит от установленных параметров записи на МЛ.

Коды функциональных клавиш, вводимые подпрограммами KEY и KY\* :

00H( 0) - (G)	01H( 1) - (B)	02H( 2) - (R)	03H( 3) - КЛ_ЗВУК
04H( 4) - CD	05H( 5) - ПЧ	06H( 6) - П/Д	07H( 7) - F0
08H( 8) - ДИАГ	09H( 9) - ГТ	0AH(10) - ПС	0BH(11) -
0CH(12) -	0DH(13) - ВК	0EH(14) -	0FH(15) -
10H(16) -	11H(17) -	12H(18) - F2	13H(19) - F5
14H(20) - F4	15H(21) - F1	16H(22) - F3	17H(23) -
18H(24) - ТАБ	19H(25) - ПРАВ	1AH(26) - ЛЕВ	1BH(27) -
1CH(28) - ВЕРХ	1DH(29) - НИЗ	1EH(30) -	1FH(31) -

Остальные коды ( 20H - 7FH ( 32 -127 )) - стандартные для КОИ-7. Следует отметить, что коды, соответствующие функциональным клавишам, при выводе подпрограммами TTY, PRN, LST могут обрабатывать иные функции, чем при вводе. Подпрограмма KEY обрабатывает следующие специальные функции :



СТР - очистка экрана ( обращение к п/п CLS );  
 (G) - то же, но с обнулением рамки ( BORDER );  
      и установкой COLOR 0,0,0;  
 (B) - то же, но COLOR 1,0,6;  
 (R) - то же, но COLOR 0,7,3;  
 КЛ\_ЗВУК - вкл./выкл. звукового сигнала клавиатуры ( инверсия  
           системной переменной KYB\_BEEP );  
 CD - вкл./выкл. постраничного вывода текста на экран  
      ( инверсия системной переменной SCRL\_LOCK );  
 ПЧ - вкл./выкл. вывода текста на принтер подпрограммой  
      LST ( переключение системной переменной PRN\_OUT );  
      Признак активации вывода на принтер - символ "\*"   
      внизу экрана;  
 П/Д - вкл./выкл. вывода текста на экран подпрограммой LST  
      ( инверсия системной переменной DISP\_OUT );  
 СУ+ПЧ - функция COPY ( копирование экрана на EPSON- совме-  
          стимом принтере в графическом режиме ( режим графики  
          SINGLE DENSITY ( ESC "K" ) ).

#### 2.2.2 Ввод символа без отработки специальных функций ( KY\* ).

Точка входа F806H - BFF1H - E843H ( 63494 - 49137 - 59459 ).  
 Работает аналогично подпрограмме KEY, но без отработки специаль-  
 ных функций, описанных выше.

#### 2.2.3 Статус клавиатуры ( STK ).

Точка входа F812H - BFFDH - E888H ( 63506 - 49149 - 59528 ).  
 Подпрограмма определяет, нажата ли какая-либо клавиша и заверша-  
 ется вне зависимости от результата проверки. Результат находится  
 во флагах МП Z(нуль), S(знак), C(перенос) и регистре А.

	Т	Т	Т	Т
Состояние	Рег.			
клавиатуры	А	Z	S	C
Нажата	FFH	0	1	1
Не нажата	00H	1	0	0

#### 2.2.4 Вывод символа на экран ( TTY ).

Точка входа F809H - BFF4H - DF68H ( 63497 - 49140 - 57192 ).  
 Подпрограмма выводит в текущую позицию экрана символ с кодом,  
 равным содержимому регистра С МП. После заполнения 24 строк про-  
 исходит сдвиг экрана вверх в соответствии со значением системной  
 переменной SCRL\_LOCK ( 00H (00) (по умолчанию) - самопроизвольно,  
 FFH (255) - с приостановкой до нажатия любой клавиши ). При ра-  
 боте используются также системные переменные TXT\_COLOR и CURSOR.

Системная переменная TXT\_COLOR определяет цвет выводимого  
 на экран текста:

FFH (255) - COLOR 0;  
 00H (000) - COLOR 1;  
 01H (001) - COLOR 2;  
 02H (002) - COLOR 3 и т.д. циклически.

Вывод символов производится всегда только на "нулевом" фоне.

Системная переменная CURSOR определяет видимость курсора ( 00H (00) (по умолчанию) - видим, FFH (255) - невидим ).

Для пользователя могут представлять интерес также системные переменные COL# и ROW#, в которых после отработки п/п находятся соответственно номер столбца и номер строки следующей текущей позиции экрана. Это выходные параметры п/п TTY и они не могут использоваться для позиционирования курсора.

Существенный интерес представляет также системная переменная TXT ADDR, являющаяся "входной" для п/п TTY. Из нее п/п берет экранный адрес текущей позиции вывода и задавая "принудительно" ее значение перед началом вывода, пользователь может организовать вывод текста в любом "необычном" месте экрана ( и даже вне экрана! Такой "вывод" невидим, но при неосторожном использовании вполне может "запортить" содержимое ОЗУ и привести к непредсказуемым последствиям, поэтому в качестве псевдо-экрана для "невидимой" печати лучше использовать адреса области ПЗУ. ). Следует также отметить, что "неестественные" адреса вывода актуальны только до ближайшего обращения к подпрограмме CLS или LOCATE.

Для кодов 20H-7FH (32-127) выводятся стандартные символы КОИ-7. Символы 80H-FFH (128-255) по умолчанию выводятся как пробелы, т.к. соответствующая область системного ОЗУ заполнена нулями, но их изображения могут быть закодированы пользователем.

Принцип кодировки изображений символов в области знакогенератора иллюстрируется следующим примером:

Ж...Ж.	A	100010XX	128+8	=136
.ЖЖЖ..	A+1	011100XX	64+32+16	=112
ЖЖЖЖЖ.	A+2	111110XX	128+64+32+16+8	=248
Ж.Ж.Ж.	A+3	101010XX	128+32+8	=168
ЖЖЖЖЖ.	A+4	111110XX	128+64+32+16+8	=248
Ж...Ж.	A+5	100010XX	128+8	=136
.ЖЖЖ..	A+6	011100XX	64+32+16	=112
	A+7	XXXXXXXX		

Биты и байт, помеченные символом X, в стандартном знакогенераторе ПК-01 не задействованы и их содержимое не влияет на изображение символа. Таким образом для хранения изображения каждого из символов в ОЗУ знакогенератора отводится по 8 байт и, следовательно, адрес A может быть вычислен по формуле:

$A = 45056 + K * 8$ , где K - код ASCII, соответствующий символу.

Модифицированное пользователем содержимое ОЗУ знакогенератора сохраняется только до ближайшего перезапуска системы (см. выше).

Из специальных символов подпрограммой TTY обрабатываются только 0AH (10), 0DH (13) и 7FH (127) - ПС, ВК и ЗБ соответственно. Все остальные символы с кодами до 1EH (30) включительно отрабатываются так же, как и символы 80H-FFH (128-255). Символ с кодом 1FH (31) - изображение курсора.

????? 2.2.5 Вывод символа на принтер ( PRN ).

Точка входа F80CH - BFF7H - E236H ( 63500 - 49143 - 57910 ). Производится вывод на стандартно ( см. выше ) подключенный принтер символа с кодом, равным содержимому регистра С МП. Режим вывода задается системной переменной 7/8\_BIT ( V8H (184) ( по умолчанию ) - вывод в 7-разрядном коде с инверсией, 00H (00) - вы-

ывод в 8-разрядном коде без инверсии с перекодировкой символов кириллицы ).

#### 2.2.6 Вывод символа на экран и/или принтер ( LST ).

Точка входа F80FH - BFFAH - E222H ( 63503 - 49146 - 57890 ). Производится вывод символа из регистра С МП на экран и/или принтер в зависимости от значения системных переменных PRN\_OUT и DISP\_OUT по следующей схеме:

-----T-----		-----T-----		-----T-----	
PRN_OUT	DISP_OUT	Принтер	Экран		
80H 128	00H 0	ON	ON		
80H 128	FFH 255	ON	OFF		
00H 0	00H 0	OFF	ON		
00H 0	FFH 255	OFF	ON		
L-----		-----		-----	

#### 2.2.7 Вывод на экран текстового сообщения ( TEXT ).

Точка входа E4A4H ( 58532 ). Перед обращением к подпрограмме адрес начала текстового сообщения необходимо занести в регистровую пару HL.

Например: BAS: LXI H,E067  
CALL E4A4  
RET

После обращения к подпрограмме BAS на экране появится текст  
BASIC 2.0 ПК-01 "ЛЪВІВ"

Текст размещается в памяти от младших адресов к старшим в кодах ASCII. Признаком конца текста служит код 00. Эта подпрограмма использует подпрограмму TTY.

#### 2.2.8 Вывод на экран содержимого регистра А в шестнадцатеричном виде ( HEX(A) ).

Точка входа FFD6H ( 65494 ). Использует подпрограмму TTY.

#### 2.2.9 Вывод на экран содержимого регистровой пары HL в шестнадцатеричном виде ( HEX(HL) ).

Точка входа FFD1H ( 65489 ). Использует подпрограмму HEX(A).

#### 2.2.10 Вывод на экран контрольной суммы блока ( CHECKSUM ).

Точка входа F815H - FF90H ( 63509 - 65424 ).  
Входные параметры : HL - начальный адрес блока;  
DE - конечный адрес блока.

В ПК-01 контрольная сумма вычисляется просто как сумма значений всех байт, входящих в блок ( в отличие от ПК линии РАДИО-86РК/ОРИОН-128 ). Вычисленную контрольную сумму подпрог-

рамма выводит на экран в шестнадцатеричной форме. Используются подпрограммы HEX(A), HEX(HL).

### 2.2.11 Позиционирование курсора ( LOCATE ).

Точка входа F82DH - DF02H ( 63533 - 57090 ).

Подпрограмма устанавливает текущую позицию вывода на экран и видимость курсора в соответствии со значениями системных переменных:

LOC\_COL : (00H-1FH) (0-31) - Номер позиции в строке;

LOC\_ROW : (00H-17H) (0-23) - Номер строки;

CURSOR\* : 00H (0) - ON, FFH (255) -OFF.

Содержимое системной переменной CURSOR\* копируется в CURSOR, устанавливаются соответствующие значения системных переменных COL#, ROW# и TXT\_ADDR.

## 2.3 Графика

Все стандартные подпрограммы поддержки графики, описанные в этом разделе рассчитаны на работу в пределах основной области экрана размером 200x225 пикселей, ограниченной рамкой (BORDER) и имеют общую систему координат с началом (0,0) в левом верхнем углу основной области экрана. Значения координат должны находиться в пределах:

X (00H-C7H) (0-199);

Y (00H-E0H) (0-224).

При выходе за левую или верхнюю границу основной области происходит генерация сообщения об ошибке ( обращение к подпрограмме обработки ошибок BASIC 065CH ( 1628 ) ). При выходе за правую ( нижнюю ) границу подпрограммы работают корректно вплоть до достижения позиции (0,0) ( экран как бы "закольцован" в этих направлениях ), после чего генерируется сообщение об ошибке.

В качестве входных параметров подпрограммы графики используются системные переменные X1,Y1,X2,Y2,GRF\_COLOR,BRD\_COLOR,BORDER.

### 2.3.1 Очистка экрана ( CLS ).

Точка входа F836H - EBD0H ( 63542 - 60368 ).

Входной параметр: BORDER.

Подпрограмма заполняет экранное ОЗУ кодом из системной переменной BORDER, затем заполняет основную область экрана кодом 00, а также устанавливает текущие графическую и текстовую позиции вывода в (0,0).

### 2.3.2 Вывод на экран точки ( PSET ).

Точка входа F821H - F048H ( 63521 - 61512 ).

Входные параметры: X1 : Координата X;  
Y1 : Координата Y;  
GRF\_COLOR : (0-3) - Цвет точки.

### 2.3.3 Стирание точки ( PRESET ).

Точка входа F020H ( 61472 ).

Входные параметры: X1 : Координата X;  
Y1 : Координата Y;  
Действие аналогично подпрограмме PSET при GRF\_COLOR=0.

### 2.3.4 Вывод на экран линии ( LINE ).

Точка входа F824H - F112H ( 63524 - 61714 ).

Входные параметры: X1 : X начальной точки;  
Y1 : Y начальной точки;  
X2 : X конечной точки;  
Y2 : Y конечной точки;  
GRF\_COLOR : Цвет линии.

### 2.3.5 Вывод на экран прямоугольника ( LINE .. B ).

Точка входа F827H - F1A5H ( 63527 - 61861 ).

Входные параметры: X1 : X начальной точки;  
Y1 : Y начальной точки;  
X2 : X конечной точки;  
Y2 : Y конечной точки;  
GRF\_COLOR : Цвет.

Выводится прямоугольник со сторонами, параллельными рамке, заданный диагональю (X1,Y1)-(X2,Y2).

### 2.3.6 Вывод закрашенного прямоугольника ( LINE .. BF ).

Точка входа F82AH - F1DAH ( 63530 - 61914 ).

Входные параметры: X1 : X начальной точки;  
Y1 : Y начальной точки;  
X2 : X конечной точки;  
Y2 : Y конечной точки;  
GRF\_COLOR : Цвет.

Выводится закрашенный прямоугольник со сторонами, параллельными рамке, заданный диагональю (X1,Y1)-(X2,Y2).

### 2.3.7 Закраска замкнутой фигуры ( PAINT ).

Точка входа F830H - E774H ( 63536 - 59252 ).

Входные параметры: X1 : X начальной точки;  
Y1 : Y начальной точки;  
GRF\_COLOR : Цвет закрашки;  
BRD\_COLOR : Цвет границы.

Выполняется закрашка замкнутой фигуры цвета BRD\_COLOR цветом GRF\_COLOR. Если фигура не замкнута, будет закрашен весь экран.

### 2.3.8 Установка цветовой палитры ( COLOR ).

Точка входа F833H - DE3CH ( 63539 - 56892 ).

Входные параметры: PALETTE : Палитра (0-6);  
GROUND : Цвет фона (0-7).

Выполняется установка заданной палитры и фона путем вывода

в порт C1 (193) ( Канал В ППА D30 ) соответствующего кода.

### 2.3.9 Копирование экрана на принтере ( COPY ).

Точка входа E627H ( 58919 ).

Производится вывод содержимого экранного ОЗУ на стандартно подключенный EPSON- совместимый принтер в графическом режиме SINGLE DENSITY ( ESC "K" ) с перекодировкой цветного изображения в черно-белое по принципу "все, что не фон, то цвет" и "съеданием" каждой восьмой горизонтальной линии точек (для улучшения пропорциональности изображений на экране и на печати вследствие неквадратности пиксела в ПК-01).

Прервать выполнение этой подпрограммы можно нажатием клавиши "СТРЕЛКА ВНИЗ".

### 2.4 Подпрограммы вывода на МЛ.

В ПК-01 "Львов" ( версия 2.0 ) для кодирования бит аналоговым сигналом при обмене с МЛ применен метод частотной модуляции, известный как метод FSK ( Frequency Shift Keying ) или метод "Kansas City", аналогичный применявшемуся в микро-ЭВМ линии YAMANA-MSX. По этому методу "0" кодируется одним периодом прямоугольного сигнала ( меандра ), а "1" - двумя периодами сигнала удвоенной частоты. Прием такого сигнала сводится к подсчету числа периодов сигнала на интервале времени, соответствующем биту.

Следует отметить, что в ПК-01 версии 1.0 ( мало кто о нем знает, но тем не менее он существовал и был даже РАДИО-86РК-совместимым ) применялся принципиально иной метод кодирования бит ( фазовый ), аналогичный формату РАДИО-подобных ПК. Именно для загрузки файлов в таком формате ( если вы их где-то все же отыщете ) и предназначены оператор BASIC 2.0 SLOAD и подпрограмма XXXXH ( XXXXX ). Подпрограммы записи в формате 1-й версии в ПЗУ версии 2.0 нет.

#### 2.4.1 Передача бита "0" ( WR\_0 ).

Точка входа E2BEN ( 58046 ).

Происходит выдача на порт C2H ( 194 ) ( бит 0 ) длинного периода меандра, задаваемого значением системной переменной WR0PERIOD.

#### 2.4.2 Передача бита "1" ( WR\_1 ).

Точка входа E2C5H ( 58053 ).

Происходит выдача на порт C2H ( 194 ) ( бит 0 ) двух коротких периодов меандра, задаваемого значением системной переменной WR1PERIOD.

#### 2.4.3 Передача периода меандра.

Точка входа E2D5H ( 58069 ).

Происходит выдача на порт C2H ( 194 ) ( бит 0 ) периода меандра с длительностью, задаваемой значением регистровой

пары HL.

#### 2.4.4 Передача пилот-сигнала ( WR\_PILOT ).

Точка входа E42BH ( 58411 ).

Происходит формирование пилот-сигнала с длительностью, задаваемой значением системной переменной PILOT\_DUR. При A=0 формируется пилот обычной, а при A<>0 - четырехкратной длительности. Пилот-сигнал представляет собой меандр с коротким периодом, аналогичный серии битов "1".

#### 2.4.5 Передача байта ( WR\_BYTE ).

Точка входа E437H ( 58423 ).

Выполняется передача на МЛ байта из регистра A. Формат представления байта имеет вид :

```
00 X0 X1 X2 X3 X4 X5 X6 X7 01 01
LT- L----- L-T--
СТАРТ.1бит--- Информационные биты L---- СТОП.2бита
```

Таким образом, каждый байт в ПК-01 передается в старт-стопном режиме ( в отличие от других ПК ), что несколько ухудшает эффективность обмена ( на каждый байт передается не 8, а 11 бит ), но существенно повышает надежность. Если какой-либо байт будет искажен при чтении ( пропуск или ложный прием бита ), то следующие за ним байты сохраняют правильное значение ( при непрерывном же режиме передачи искажены будут и все последующие байты ). Старт-стопный режим записи допускает также некоторые "вольности" при загрузке, например, динамическое изменение адреса загрузки т.к. интервал времени между приемами байт не регламентирован.

Подпрограмма выполняет сохранение и восстановление содержимого используемых регистров.

#### 2.4.6 Передача слова ( WR\_WORD ).

Точка входа DD86H ( 56710 ).

Выполняется передача на МЛ слова из регистровой пары HL.

Подпрограмма выполняет сохранение и восстановление содержимого используемых регистров.

#### 2.4.7 Передача заголовка файла ( WR\_HEAD ).

Точка входа E3E4H ( 58340 ).

Выполняется передача на МЛ заголовок файла, имеющего следующую структуру :

```
Пилот 4x длительности T T T T T T T T T T F L N A M E
L----- L-----L-----
                        10 байт          6 байт
```

где T - идентификатор типа файла : D0 - BSAVE/BLOAD  
( хранится в системной D3 - CSAVE/CLOAD  
переменной FILE\_TYPE ) EA - SAVE/LOAD/MERGE

FLNAME - имя файла ( 6 символов ASCII из системной переменной SAVE\_NAME ).

Входные параметры - SAVE\_NAME, FILE\_TYPE. Состояние регистра A определяет длительность пилота ( см. выше ).

#### 2.4.8 Передача кодового файла ( BSAVE ).

Точка входа DD31H ( 56625 ).

Выполняется передача на МЛ файла в формате BSAVE, имеющего следующую структуру :

```
Заг. Пилот норм.длит. AO AE AS XX XX XX ..... XX XX
L--- L----- L- L- L- L-----
                        26 26 26          Тело файла
```

где Заг. - заголовок файла ( см. выше )  
AO - начальный адрес ( CODFL\_ORG )  
AE - конечный адрес ( CODFL\_END )  
AS - адрес запуска ( CODFL\_RUN )

Входными параметрами подпрограммы являются CODFL\_ORG, CODFL\_END, CODFL\_RUN, SAVE\_NAME. После выполнения передачи на экран с новой строки выдаются адреса файла ( подпрограмма FFD1H ( 65489 ) ) и его KC ( подпрограмма CHECKSUM ) в шестнадцатеричном виде. Прерывание работы подпрограммы BSAVE не предусмотрено.

#### 2.5 Подпрограммы ввода с МЛ.

Внимание ! Во все стандартные подпрограммы ввода с МЛ включена подпрограмма E3DCH ( 58332 ), выполняющая опрос клавиши "СТРЕЛКА ВНИЗ" и устанавливающая флаг переноса C в "1" при ее нажатии. Эта подпрограмма используется для прерывания процедур ввода. При прерывании или аварийном завершении процедур ввода с МЛ выдается сообщение об ошибке и производится переход на E52BH ( 58667 ), откуда управление передается на BEF0H ( 48880 ). После холодного перезапуска здесь размещены коды C3H FDH 02H ( JMP 02FDH ( 00756 ) ), т.е. управление будет передано в BASIC. Изменив значение системной переменной LOAD\_ERR, пользователь может назначить нестандартную процедуру обработки ошибок ввода.

##### 2.5.1 Прием пилот-сигнала ( RD\_PILOT ).

Точка входа E4D0H ( 58576 ).

Выполняется многократная проверка стабильности периода пилот-сигнала. Если длительность периода выходит за допустимые пределы ( меньше 4 или больше DEN ( 222 ) ) или разница между предыдущим и текущим периодом больше 3, то прием пилота начинается заново. Если проверка прошла успешно, то производится 256-кратное измерение периода пилота ( результат накапливается в HL ). На основе измерений вычисляются значения параметров считывания BE85H ( 48773 ) и BE86H ( 48774 ).

Подпрограмма RD\_PILOT использует для измерения периода меандра подпрограмму E3C9H ( 58313 ), а она, в свою очередь использует п/п E3B4H ( 58292 ) для измерения длины одной полуволны ( при E=00H - отрицательной, при E=FFH - положительной ). Ре-



зультат помещается в С.

Подпрограмма выполняет сохранение и восстановление содержимого используемых регистров.

#### 2.5.2 Прием байта ( RD\_BYTE ).

Точка входа E4VEN ( 58558 ).

Выполняется прием с МЛ байта в регистр А в старт-стопном режиме ( см. выше ). В основу считывания бит положено определение количества полупериодов сигнала на заданном интервале времени ( в бите ) оно выполняется подпрограммой E390H ( 58256 ) ( отрицательных или положительных в зависимости от Е ( см. выше ). Результат помещается в А и С. Если количество полупериодов больше 3, фиксируется ошибка и подпрограмма завершается с флагом С=1. Если результат равен 2 или 3, считается, что принят бит "1", а если меньше 2 – бит "0". Накопление байта происходит в D.

Подпрограмма выполняет сохранение и восстановление содержимого используемых регистров.

#### 2.5.3 Прием слова ( RD\_WORD ).

Точка входа DDCAH ( 56778 ).

Выполняется прием с МЛ слова в регистровую пару HL.

Подпрограмма выполняет сохранение и восстановление содержимого используемых регистров.

#### 2.5.4 Прием заголовка файла ( RD\_HEAD ).

Точка входа E443H ( 58435 ).

Выполняется проверка стабильности и вычисление скорости по пилоту ( см. выше ) затем 10 раз считывается и сравнивается с ожидаемым идентификатор типа файла ( при несовпадении процесс приема заголовка начинается заново ), считывается имя файла ( 6 байт ) в системную переменную LOAD\_NAME, выполняется его сравнение с заданным ( SAVE\_NAME ). При несовпадении на экран выдается сообщение "ИМЯ:?", принятое имя и прием заголовка начинается заново. Если же принятое имя совпадает с заданным ( или вместо заданного – только пробелы ), то выдается сообщение "ФАЙЛ:", содержимое LOAD\_NAME и подпрограмма завершается.

Входные параметры – SAVE\_NAME, FILE\_TYPE.

#### 2.5.5 Прием блока ( RD\_BLOCK ).

Точка входа DDBCH ( 56764 ).

Входные параметры: HL : Начальный адрес;

DE : Конечный адрес.

Выполняется прием блока данных ( тела файла ) в указанные адреса ОЗУ. Завершение загрузки – аналогично подпрограмме BLOAD ( см. ниже ).

#### 2.5.6 Прием кодового файла ( BLOAD ).

Точка входа DD94H ( 56724 ).

Выполняется прием заголовка файла ( см. выше ), затем прием второго пилота, адресов начала, конца, запуска файла ( они помещаются в системные переменные CODFL\_ORG, CODFL\_END и CODFL\_RUN соответственно ) и наконец основного тела файла.

Значение системной переменной OFFSET определяет смещение адреса начала загрузки относительно принятого с МЛ адреса начала файла ( CODFL\_ORG ).

После выполнения загрузки на экран с новой строки выдаются адреса файла и его КС ( подпрограмма DD61H ( 56673 )) в шестнадцатеричном виде.

Если была задана загрузка с автозапуском, ( в системной переменной AUTOSTART находится ASCII-код буквы "R" ( без автозапуска - значение 00 )), производится передача управления на адрес, находящийся в CODFL\_RUN.

Входными параметрами подпрограммы являются OFFSET, AUTOSTART, SAVE\_NAME.

## 2.6 Прочие

### 2.6.1 Короткий звуковой сигнал ( BEEP ).

Точка входа F81BH - DEBFH ( 63515 - 57023 ).

Выдается короткий звуковой сигнал, аналогичный сигналу нажатия клавиши.

### 2.6.2 Управляемый звуковой сигнал ( SOUND ).

Точка входа F81EH - DE94H ( 63518 - 56980 ).

Входные параметры: L : Код частоты (период);  
D : Код длительности.

Выдается звуковой сигнал с заданным периодом и длительностью. Кодировку нот см. в "Описании языка BASIC ", табл. 2.

### 2.6.3 Вычисление экранного адреса по координатам ( VADDR ).

Точка входа F818H ( 63512 ).

Входные параметры: C : Координата X ( <C8H ( 200 ));  
A : Координата Y ( <DEH ( 222 ));

Вычисляет абсолютный адрес байта в ОЗУ по экранным координатам. Результат - в регистровой паре HL.

### 2.6.4 Заполнение экранного ОЗУ кодом.

Точка входа EBVCB ( 60348 ).

Входной параметр: BORDER : Код заполнения.

Выполняется заполнение экранного ОЗУ указанным кодом.

#### 2.6.5 Очистка основной области экрана.

Точка входа EBA5H ( 60325 ).

Выполняется заполнение основной области экранного ОЗУ кодом 00.

#### 2.6.6 Курсор в начало экрана.

Точка входа EBE4H ( 60388 ).

Выполняется установка текущей текстовой позиции вывода в (0,0). Видимость курсора определяется системной переменной CURSOR.

Внимание !

Перед обращением к подпрограммам 2.6.4, 2.6.5 и 2.6.6 необходимо подключить экранное ОЗУ к шинам МП ( см. выше ), после обращения, - отключить.

#### 2.6.7 Перемещение массивов ( MOVE ).

Точка входа E11FH ( 57631 ).

Входные параметры: HL : Адрес, куда копировать;

DE : Начальный адрес копируемого массива;

BC : Длина массива.

Выполняется копирование содержимого одной области ОЗУ в другую.

#### 2.6.8 Заполнение области ОЗУ кодом ( FILL ).

Точка входа E12AH ( 57642 ).

Входные параметры: HL : Начальный адрес заполнения;

BC : Длина массива;

E : Код заполнения.

Выполняется заполнение заданной области ОЗУ указанным кодом.

#### 2.6.9 Перевод полубайта из HEX в DEC.

Точка входа FFBDH ( 65469 ).

Входной параметр : A : Полубайт в шестнадцатеричной системе.

Выполняется перевод полубайта из шестнадцатеричной системы счисления в десятичную. Результат - в регистре C.

#### 2.6.10 Вывод на экран полубайта в HEX.

Точка входа FFDFH ( 65503 ).

Входной параметр : A : Полубайт в шестнадцатеричной системе.

Выполняется перевод полубайта из шестнадцатеричной системы счисления в десятичную. Результат выводится на экран.

#### 2.6.11 Умножение DE на A ( сложение A раз ).

Точка входа FF24H ( 65316 ).  
Входные параметры: DE : Слагаемое ( множимое );  
                  A : Количество сложений ( множитель ).  
Выполняется многократное ( A раз ) сложение содержимого регистровой пары DE. Результат - в регистровой паре HL.

#### 2.6.12 Проверка условия DE=HL.

Точка входа E425H ( 58405 ).  
Входные параметры: DE ;  
                  HL .  
Выполняется сравнение содержимого регистровых пар DE и HL.  
Результат - состояние флага Z ( Z=1 - равны, Z=0 - неравны ).

#### 2.6.13 Заполнение пробелами.

Точка входа DB63H ( 56163 ).  
Входные параметры: B : Количество пробелов;  
                  DE : Адрес начала заполнения.  
Выполняется заполнение области ОЗУ с начальным адресом DE и длиной B кодом 20H ( 32 ) - ASCII-кодом символа "пробел".

### 3. Сводная таблица системных подпрограмм.

0107-0207H - Экранный буфер BASIC  
0243/0244H - Адрес начала BASIC-программы ( 1723H )  
0245/0246H - Адрес конца BASIC-программы ( начало обл. переменных )  
0247/0248H - Адрес конца переменных BASIC-программы ( начало обл. массивов )  
0249/024AH - Адрес начала свободной области ОЗУ  
  
AFC1H - Вершина стека BASIC  
AF00H - Вершина стека GOSUB

#### 4. Сводная таблица системных переменных.

##### 4.1 Системные переменные BIOS.

Hex	Dec	Name	Comment
BE10	48656	KEYCODE	код нажатой клавиши
BE14	48660	KEYSTAT	состояние клавиатуры
BE1B	48667	DISP_OUT	вкл./выкл. принтера
BE1D	48669	KYB_MODE	режим клавиатуры
BE1E	48670	KYB_BEEP	вкл./выкл. звук клав.
BE30/BE31	48688/48689	TXT_ADDR	адрес вывода символа
BE32	48690	COL#	X след.поз.выв. текста
BE33	48691	ROW#	Y след.поз.выв. текста
BE36	48694	TXT_COLOR	цвет текста
BE38	48697	BORDER	байт заполнения рамки
BE39	48698	SCRL_LOCK	скроллинг авт./ожид.
BE3C	48700	CURSOR	видимость курсора
BE3D	48701	CURSOR*	уст. видимость курсора
BE3E	48702	LOC_COL	уст. X выв. текста
BE3F	48703	LOC_ROW	уст. Y выв. текста
BE40	48704	PRN_SHIFT	смещение букв кирилл.
BE41	48705	PRN_XOR	выв.на PRN норм./инвер.
BE50	48720	X1	X начальной точки
BE51	48721	Y1	Y начальной точки
BE52	48722	GRF_COLOR	цвет графич. объекта
BE57	48727	X2	X конечной точки
BE58	48728	Y2	Y конечной точки
BE80/BE81	48768/48769	WR0PERIOD	длит.периода сигн."0"
BE82/BE83	48770/48771	WR1PERIOD	длит.периода сигн."1"
BE84	48772	PILOT_DUR	длительность пилота
BE85	48773		- параметры чтения МЛ
BE86	48774		L (опред. автоматич.)
BE87	48775	FILE_TYPE	идентификат.типа файла
BE88/BE89	48776/48777	BASFL_ORG	начало BASIC-файла
BE8A/BE8B	48778/48779	BASFL_END	конец BASIC-файла
BE8C-BE91	48780-48785	SAVE_NAME	имя файла запись/эталон
BE92-BE97	48786-48791	LOAD_NAME	считанное имя файла
BEA3	48803	AUTOSTART	признак автостарта
BEA4/BEA5	48804/48805	CODFL_ORG	начало CODE-файла
BEA6/BEA7	48806/48807	CODFL_END	конец CODE-файла
BEA9/BEAA	48809/48810	CODFL_RUN	старт CODE-файла
BEAB/BEAC	48811/48812	OFFSET	смещение CODE-файла
BEC0	48832	PALETTE	палитра
BEC1	48833	GROUND	цвет фона
BEF1/BEF2	48881/48882	LOAD_ERR	адр.перех. по ОШИБ.В/В
BEF3	48883	PRN_OUT	вкл./выкл. дисплея

L-----+-----+-----+-----			
4.2 Системные переменные интерпретатора BASIC.			
-----T-----T-----T-----			
Hex	Dec	Name	Comment
+-----+-----+-----+-----			
0243/0244	00579/00580	BAS_PROG	адр.нач. BAS-программы
0245/0246	00581/00582	BAS_VARS	адр.нач. BAS-переменн.
0247/0248	00583/00584	BAS_ARRS	адр.нач. BAS-массивов
0249/024A	00585/00586	BAS_FREE	адр.нач. свободн. ОЗУ
L-----+-----+-----+-----			

-----

E6AA - Вывод на принтер ESC+chr\$(C)

E6A3 - Вывод на принтер chr\$(C)