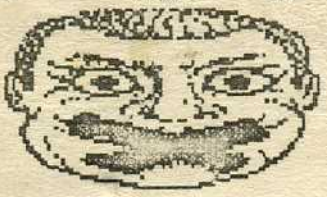
Инструкция по эксплуатации и описание компьютера

ZX Spectrum BOOK 2



Содержание

Введение	1
Командный режим	1
Литерный режим	2
Расширенный командный режим	2
Графический язык	2
Ключевые слова Спектрума	3
Работа на компьютере	11
Загрузка программы	11
Программа TAPER	13
Как выполнить запасную копию программы	13
Программа COPY86/M	13
Программа TFCOPY	14
Программа COPY-COPY	14
Составление и редактирование программ	16
Системные переменные	19
Организация памяти компьютера	23
Специально для Вас	25
Методы введения русского шрифта	25
Исползование графики пользователя	25
Изменение набора знаков	26
Защита программ	27
Исключение возможности остановки программы	27

Отключение команды LIST 28

Как сделать текст программы нечитаемым 28

Операции с дисплеем 29

БОЛЬШИЕ ВОЗМОЖНОСТИ ВАШЕГО "SPECTRUMA"

< В в е д е н и е >

Клавиатура SPECTRUM имеет всего 40 клавиш. В принципе это очень мало, и здесь наглядно проявился результат неуклонной борьбы К.Синклера за миниатюризацию и дешевизну своих изделий. Каждая клавиша может выполнять по 6-7 различных действий в зависимости от того, в сочетании с какими клавишами она нажимается. Рассмотрим некоторые характерные клавиши.

☐ E N T E R

Нажатие этой клавиши обычно завершает ввод в компьютер чего-либо (команды, данных, программной строки). На компьютерах других систем эта клавиша может иметь другое наименование - RETURN, ВК (возврат каретки) и др.

☐ CAPS SHIFT

Переключение регистра на печать прописными буквами (аналогично пишущей машинке).

☐ SYMBOL SHIFT

Переключение регистра для печати символов (+, -, /, и др.) а также некоторых ключевых слов (THEN, TO, OR, AND, STEP и др.)

☐ S P A C E

Пробел. Эта же клавиша выполняет BREAK прерывание исполнения программы если её нажать вместе с CAPS SHIFT. Остальные клавиши имеют буквенное или цифровое обозначение.

Если вы уже подключили компьютер, то на экране должно появится исходное сообщение 1982 SINCLAIR RESEARCH LTD. Это обозначает, что первичные проверки в компьютере прошли нормально и, скорее всего, он исправен и готов к работе.

Нажмите клавишу ENTER. На экране появится чёрный квадрат с буквенным обозначением на нём. Это курсор. Во-первых, он показывает то место на экране, в котором будет помещаться очередной набранный вами символ, а, во-вторых, указывает в каком режиме находится клавиатура. Этих режимов - ЧЕТЫРЕ, и им соответствует ПЯТЬ разных курсоров.

КОМАНДНЫЙ РЕЖИМ

Курсор - "К". Он означает, что сейчас при нажатии клавиши будет введена команда, которая закреплена за этой клавишей, или цифра. Компьютер сам понимает что может начинаться только либо с номера строки, либо с команды, поэтому дает курсор "К". После того, как

команда набрана, и дальше должны пойти её параметры режим автоматически переключится на литерный (курсор "L"). Если вы введёте ":" (двоеточие), то курсор опять переключится на "K", так как двоеточие является разделителем между несколькими командами, если они записываются в одной строке.

ЛИТЕРНЫЙ РЕЖИМ

Курсоры - L,C. Курсор L соответствует печати строчными буквами, а курсор C - прописными. Переключиться на курсор "C" можно командой CAPS LOCK. Это выполняется одновременным нажатием "CAPS SHIFT" и цифры "2". В этом же режиме набираются символы и служебные слова, связанные с клавишей "SYMBOL SHIFT".

РАСШИРЕННЫЙ КОМАНДНЫЙ РЕЖИМ

Курсор "E". В этом режиме набираются команды, записанные над клавишами или под клавишами. Переход в режим "E" выполняется одновременным нажатием "CAPS SHIFT" и "SYMBOL SHIFT". Он действует только на одно нажатие. Если в режиме "E" нажать какую-либо клавишу, то появится слово, записанное над клавишей, а если нажать эту клавишу совместно с "CAPS SHIFT", то слово, записанное под клавишей. Здесь имеется исключение для цифровых клавиш (верхний ряд). Чтобы набрать слово, записанное под клавишей, надо в режиме "E" нажать не CAPS SHIFT, а SYMBOL SHIFT совместно с клавишей. Такая уникальная способность "SPECTRUM" вводить операторы и функции не по буквам, а одним нажатием клавиши, называется токенизированной формой записи ключевых слов. Сначала это выглядит несколько сложновато, но опыт приходит быстро, а с ним и удобство работы. По крайней мере вы не сделаете ошибок в прописании английских слов, если этот язык вам знаком плохо.

ГРАФИЧЕСКИЙ ЯЗЫК

Курсор - "G". В этом режиме набираются символы блочной графики, расположенные на цифровых клавишах, а также символы графики пользователя (UDG - user defined graphics), но о них речь пойдёт позже. Переход в графический режим выполняется одновременным нажатием клавиш "CAPS SHIFT" и "9". Если вы в этом режиме нажмёте, например, клавишу "5", то на экране появится квадрат, левая половина которого черная, а правая - белая. Нажатие клавиши "5" совместно с "CAPS SHIFT" изобразит тот же квадрат в инвертируемом виде, т.е. левая половина будет белая, а правая - черная. Выход из графического режима выполняется так же, как и вход.

☐ ПРИМЕР

Если вы уже набрали что-то произвольное на экране, то скорее всего этого это неправильная строка и при нажатии ENTER на экране появится знак "?". Чтобы начать работу, надо эту строку стереть.Это выполняется командой DELETE (одновременное нажатие CAPS SHIFT и цифры 0). Когда все символы будут уничтожены, курсор станет на исходную позицию и примет вид "K", то есть компьютер ждет от вас команду.

Нажмите клавишу "R". На экране появится команда RUN. Курсор изменится на "L". Еще раз нажмите ту же клавишу - появится буква r. Если клавишу задержать в нажатом положении, то буква будет повторена многократно. Эта удобная функция называется "автоповтор"(кстати он действует и при стирании). Нажатие этой клавиши совместно с CAPS SHIFT даст прописную букву R, а совместно с SYMBOL SHIFT - знак "". Перейдите в расширенный командный режим - CAPS SHIFT "+" SYMBOL SHIFT одновременно.Появится курсор "E".Теперь нажатие этой же клавиши даст оператор INT, снова войдите в режим "E" и нажмите клавишу совместно с CAPS SHIFT. Получите команду VERIFY.

Те, кто работает с компьютерами "СПЕКТРУМ+", "+2" и "+3", могут упростить свою работу благодаря наличию дополнительных клавиш, которые выполняют ряд функций одним нажатием.

DELETE - стирание символа.
GRAPH - переход в графический режим.
EXTENDED MODE - переход в расширенный командный режим.
EDIT - выполнение редактирования строки.
CAPS LOCK - переключение регистра на печать прописными буквами.
INV.VIDEO - включение инверсного режима (печать белым по черному).
TRUE VIDEO - возвращение из инверсного режима.

Кроме того, там имеются дополнительные клавиши для набора знаков препинания: точка, запятая, точка с запятой, кавычки а также четыре клавиши для управления перемещением курсора при редактировании (см. ниже).

КЛЮЧЕВЫЕ СЛОВА СПЕКТРУМА

Условные обозначения:

■ аддр - адрес от 0 до 65535;
A - действительное число;
N,M,X,Y,R - целое число.

КЛЮЧЕВОЕ СЛОВО	ПРИМЕР ПРИМЕНЕНИЯ	НАЗНАЧЕНИЕ

ФУНКЦИИ

ABS	LET A=ABS-3,5	Дает абсолютную величину числа.
ACS	LET A=ACS 0,5	Арккосинус в радианах.
AND		Логическое "И".
ASN	LET A=ASN 0,5	Арксинус в радианах.
ATN	LET A=ATN 1,5	Арктангенс в радианах.
ATTR	PRINT ATTR(10,5)	Дает число,показывающее состояние цветовых атрибутов в заданном знакоместе экрана. Если это число разложить на биты, то бит 7(старший) указывает на признак мигания. 1- есть; 0 - нет.Бит 6 признак яркости.Биты 3...5-цвет фона (от 0 до 7) ; биты 0...2 - цвет символа.
BIN	PRINT BIN 10011001	Перевод числа из двоичной формы в десятичную.
CHR	PRINT CHR N	Дает символ,код которого равен N
CODE	PRINT CODE ABCD	Дает код первого символа строки (стринга).
COS	LET A=COS 0,5	Косинус числа.
EXP	LET A=EXP 0,5	Экспонента числа.
FN	PRINT FN W(X,Y...)	Вызов функции пользователя W,заданной DEF FN W. Аргумент должен быть в скобках. Если аргумента нет, скобки все равно должны быть.
IN	PRINT IN X	Ввод данных с внешнего порта номер X.
INKEY	LET A=INKEY	Читает клавиатуру.Выдает символ, равный тому,клавиша которого была нажата,если ничего не нажато, то выдает пустой стринг.INKEY не ждет нажатия. Это обходят например так: 10 LET A=INKEY 20 IF A="" THEN GO TO 10
INT	LET N=INT A	Выделение целой части числа (округление вниз).
LEN	LET N=LEN ABCD	Дает длину стринга в символах.
LN	PRINT LN 2,345	Натуральный логарифм.
NOT		Логическое отрицание "НЕ"
OR		Логическое "ИЛИ".

- PEEK LET N = PEEK аддр Дает содержимое ячейки памяти по ее адресу.
- PI 3.14159265
- POINT PRINT POINT (x,y) Проверяет включена или выключена точка экрана с координатами X,Y. Если включена -1, если нет -0. X = 0...255; Y = 0...175.
- RND LET N = INT(5*RND) Дает псевдослучайное число между 0 и 1.
- SCREEN PRINT SCREEN (X,Y) Дает символ, равный тому, который находится на экране в знакоместе с координатами X,Y. Если символ не распознаваем, то выдается пустой стринг. X = 0...23; Y = 0...31.
- SGN PRINT SGN A Знак числа. 1 - положительное, -1 отрицательное, 0 - если число равно нулю.
- SIN LET A = SIN 2,345 Синус угла, заданного в радианах.
- SQR PRINT SQR 25 Квадратный корень.
- STR PRINT STR 100 Перевод числа в строковую переменную (стринг), изображающую то же число. STR 100 = "100".
- TAN LET A = TAN 2,345 Тангенс угла, заданного в радианах.
- USR аддр. RANDOMIZE USR 30000 Вызывает подпрограмму в машинном коде и стартует ее с указанного адреса.
- USR A LET аддр. = "A" Дает адрес в области графики пользователя UDB, в котором расположено изображение символа графики пользователя, соответствующего "A".
- VAL LET A = VAL"100" Функция, обратная STR. переводит стринг в соответствующее число.
- VAL PRINT VAL""100"" Рассматривает свой стринговый параметр без ограничительных кавычек как новый стринг.

ОПЕРАТОРЫ

- BEEP BEEP 20.0, 0.5 Воспроизводит в динамике компьютера звуковой сигнал продолжительностью 20 сек. и с высотой на 0,5 полутона выше, чем "до" первой октавы.
- BORDER BORDER N Установка цвета бордюра. N = 0...7.
- BRIGHT BRIGHT 1 Установка яркости. 0 - нормальная. 1 - повышенная.
- CAT Распечатать каталог картриджа микродрайва.
- CIRCLE CIRCLE X,Y,R Вычерчивание окружности с центром X,Y и с радиусом R.

- CLEAR CLEAR ADDR Вычищает все переменные,выполняет RESTORE и CLS,вычищает стек GO SUB. Устанавливает новую верхнюю границу памяти по адресу ADDR.
- CLOSE# Команда ИНТЕРФЕЙСА-1 (закрыть поток).
- CLS Очистка экрана.
- CONTINUE Продолжить исполнение программы после прерывания.
- COPY Команда ZX - принтера.Выполняет копирование экрана на бумагу.
- DATA Данные для оператора READ.
- DEF FN DEF FN W(x,y,...) Имя функции W должно быть однобуквенным, а параметры стоять в скобках. Если параметров нет-скобки все равно должны быть.
- DIM DIM A(N1,N2...NK) Задает к-мерный массив A. Устанавливает исходные значения элементов массива в нуль.Уничтожает все массивы с тем же именем.

DIM A2(N1,N2...NK) Задает символьный к-мерный массив.Исходное значение элементов массива устанавливается в " ". Его можно рассматривать как к-1-мерный массив стрингов фиксированной длины,равной NK.

- DRAW DRAW X,Y Выполняется вычерчивание прямой линии от точки, в которой производилась печать в последний раз до точки, отстоящей от нее по горизонтали на X и по вертикали на Y.

DRAW X,Y,Z То же самое, но между этими двумя точками вычерчивается не прямая, а дуга окружности, охватывающая угол Z, где угол измеряется в радианах.

- ERASE Команда микродрайва.
- FLASH FLASH I Признак мигания. 1 - включено; 0 - выключено.
- FOR FOR I=X TO Y Составная часть оператора FOR... TO...NEXT

FOR I=X TO Y STEP Z Организует цикл с начальным значением параметра цикла X, конечным значением Y и шагом Z. Шаг может быть и отрицательным. В этом случае начальное значение больше, чем конечное.

- FORMAT Команда микродрайва.
- GO SUB GO SUB N Переход к исполнению программы, расположенной начиная со строки N. Подпрограмма должна завершаться командой RETURN.
- GO TO GO TO N Переход к исполнению строки N.

- IF X THEN Y IF A5 THEN GO TO N Условный оператор. Если выражение X справедливо, то выполняется выражение после THEN. В противном случае выполняется переход к следующей строке.

Не к следующему оператору в этой строке, а именно к следующей строке! Типичная ошибка!..

- INK INK N Устанавливает цвет печати. N=0...7
- INPUT INPUT A Служит для ввода с клавиатуры числа во время исполнения программы. Может вводиться и символьный стринг.В этом случае оператор должен иметь форму INPUT LINE.
- INVERSE INVERSE 0 Признак инверсии. 1 - включено (печать белым по черному). 0 - нормальная печать.
- LET LET A=X Оператор присваивания.Переменной A присваивается значение X.
- LIST LIST Выполняется печать листинга (текста) программы на экране.В форме LIST N листинг выдается, начиная со строки N. Если текст занимает более, чем один экран, то после заполнения экрана выдается запрос SCROLL ? Если нажать клавиши BREAK или N, то распечатка прервется. Нажатие любой другой клавиши продолжит листинг.
- LLIST, LLIST N Команда ZX-принтера. То же, что и LIST но распечатка выполняется на ZX - принтере.
- LOAD LOAD "имя" Загрузка БЕЙСИК-программы с магнитофона.Если имя не указано,(LOAD"") то загрузится первая встреченная программа.Обратите внимание: между кавычками не должно быть пробела. LOAD "имя"CODE M,N Команда на загрузку блока в машинных кодах.Загружает N байтов, начиная с адреса M.Если эти параметры не указаны, то блок будет загружаться в те адреса, из которых он был когда-то выгружен.
- MERGE MERGE "имя" Аналогична LOAD, но при загрузке новой программы не уничтожает старую, а производит их слияние. При этом остается старая программа с добавлением таких строк из новой программы, каких в старой программе не было.Для блоков, записанных в машинных кодах, этот оператор неприменим.
- MOVE Команда микродрайва.
- NEW Уничтожает БЕЙСИК-программу, устанавливает в исходное положение системные переменные,очищает память до границы RAMTOP, установленной CLEAR.
- NEXT см.FOR Часть оператора FOR...NEXT .Заканчивает цикл.
- OPEN# Команда интерфейса-1 и микродрайва.

64-23-62

- OUT · OUT M,N Выдает байт N на внешний порт M.
- OVER OVER 1 Управляет условием наложения символов друг на друга при печати.0 - новый символ полностью закрывает старый. 1 - изображение получается со смещением символов.
- PAPER PAPER N (N=0...7) То же, что INK, но задает цвет фона.
- PAUSE PAUSE N Пауза в работе программы на время N/50 секунд или до нажатия какой-либо клавиши. PAUSE 0 - Пауза до нажатия клавиши.
- PLOT PLOT C;M,N Выполняет печать точки на экране с цветом C в координатах M,N. Если параметр C не задан, то текущий цвет.M=0...255, N=0...175. Обычно эта команда предшествует DRAW.
- POKE POKE аддр,N Засылает байт по указанному адресу.
- PRINT PRINT N Печать числа. PRINT X - Печать результата расчета выражения X. PRINT "ABCD..." - Печать текста, стоящего в кавычках. PRINT A - Печать стринга(строковой переменной). Одной командой PRINT может выполняться печать сразу нескольких переменных или чисел.При этом между ними должен стоять разделитель.Таких разделителей три.Это запятая (,),точка с запятой(;) или апостроф(').Если стоит запятая, то печать выполняется на той же строке, но начиная с 16-й колонки,т.е. в правой половине экрана.Точка с запятой означает печать в той же строке без зазора.Апостроф дает переход к новой строке печати. Пример PRINT A,B,C,D.E'F'N. Просто PRINT дает пустую строку. Так PRINT:PRINT:PRINT эквивалентно PRINT... PRINT AT M,N - Печать в заданной позиции (строка,столбец). PRINT TAB N - Печать с табуляцией.N показывает величину табуляции. Оператор PRINT может сопровождаться операторами задания цветовых атрибутов INK, PAPER, FLASH,BRIGHT, INVERSE, OVER. В этом случае они вводятся через точку с запятой. Пример:PRINT INK 6; PAPER 1; "VERY WELL",FLASH 1;" "
- RANDOMIZE N Устанавливает системную переменную SEED - системная переменная используется для работы генератора псевдослучайных чисел (см.RND)
- RANDOMIZE То же, что RANDOMIZE 0
  N=0...65535
- RANDOMIZE USR RANDOMIZE USR аддр. Вызывает и стартует процедуру записанную в машинных кодах,находящуюся по указанному адресу.RANDOMIZE USR 0 инициализирует компьютер,т.е. выполняет эффективный сброс,эквивалентный выключению питания.Ср. с NEW.
- READ READ V1,V2,V3...VN Присваивает переменным значение,

находящиеся в соответствующем операторе DATA.

- REM Пустой оператор.Открывает строку комментариев.Все,что записано за REM(в пределах строки) не исполняется,а только распечатывается по LIST.Это помогает впоследствии разобраться с программой.
- RESTORE N Устанавливает указатель DATA, привязывающий DATA к соответствующему READ, так,чтобы он указывал на DATA, расположенный в строке N или,если такой нет,то к ближайшей старшей.
- RESTORE То же,что и RESTORE 0.
- RETURN Пустой оператор.Завершает подпрограмму, к которой обращались по GO SUB. Передает управление к строке, следующей за вызывающей.
- RUN RUN N Команда на исполнение программы. Все переменные,существовавшие до нее,уничтожаются и начинается исполнение со строки N.Эквивалентна CLEAR : GO TO N RUN - То же, что и RUN 0.
- SAVE . SAVE "имя" Выполняет выгрузку БЕЙСИК-программы на магнитную ленту.Имя давать обязательно.Требует нажатия ENTER. После ENTER появляется указание "START TAPE, THEN PRESS ANY KEY"(Запустите ленту и нажмите любую клавишу). SAVE "имя "LINE N - Записывает программу на ленту так, что она после загрузки будет автостартовать со строки N. SAVE "имя" CODE M,N - Запись на ленту блока машинных кодов,начиная с адреса M,в количестве N байтов. SAVE "имя" DATA ( ) Запись числового массива. SAVE "имя" DATA ( ) Запись на ленту символьного массива. SAVE "имя" SCREEN - Запись экрана.То же самое,что и SAVE "имя" CODE 16384, 6912. Наличие всех параметров в команде обязательно.
- STOP Останавливает исполнение программы.Продолжить можно оператором CONTINUE.
- VERIFY VERIFY"" Аналогична LOAD ,но в отличие от нее не производит загрузку в ОЗУ, а только сверяет то, что записано на ленте с тем, что находится в памяти компьютера и в случае несоответствия выдает сообщение об ошибке. Применяется для проверки качества записи.Имеет все те же записи, что и оператор LOAD.

☐ ПРИМЕЧАНИЕ.

Выше мы не рассматривали ряд операторов, указав, что они имеют отношение к Интерфейсу-1 и к микродрайву.Интерфейс-1 - это устройство,выполняющее одновременно три функции - контролера локальной сети,порта RS232 и контролера микродрайва.Выпускается той же фирмой SINCLAIR RESEARCH LTD.Имеет собственное ПЗУ и служит для подключения компьютера к другим компьютерам(прак-

тически любым),к глобальной сети(через модем) и для объединения класса "СПЕКТРУМОВ" в локальную сеть(до 64-х шт.).Команды Интерфейса-1,строго говоря,уже не являются операторами БЕйСИКа как языка программирования, а командами операционной системы. ZX - микродрайв- устройство,работающее с высокоскоростной закольцованной магнитной лентой,находящейся на сменных картриджах.Оставаясь ленточным носителем,он создает эффект работы с дисководом.Емкость картриджа порядка 100 К. Одновременно можно подключить до восьми микродрайвов. ИНФОРКОМ может предложить инструкцию по работе с Интерфейсом-1 и ZX-микродрайвов. (Как и все наши разработки - на русском языке).Кроме того,для интересующихся мы имеем довольно объемистый перевод книги Я.Секйлса "Путеводитель по периферии Спектрума".

ЗАГРУЗКА ПРОГРАММ С МАГНИТОФОНА

Если Вы недавно приобрели компьютер,то вполне понятно желание начать работу с готовыми фирменными программами,а самостоятельное программирование отложить на несколько более поздний срок. Загрузка и запуск фирменных программ выполняются весьма просто: LOAD" "[ENTER].Помните,что между кавычками не должно быть пробела. Основную проблему на этом этапе представляет плохая загружаемость программ, о чем свидетельствует сообщение об ошибке TAPE LOADING ERROR зависание программы после загрузки или самопроизвольный сброс.Причина не обязательно связана с плохой загрузкой, но если эта лента получена Вами из стороннего источника,то это скорее всего так..Здесь мы приведем некоторые принципы,которыми надо руководствоваться,чтобы улучшить загружаемость программ.

■ 1. По-настоящему надежную загрузку может обеспечить только высококачественная лента,если она записана на том же магнитофоне, на котором и воспроизводится. Поскольку стоимость программ обычно значительно превышает стоимость ленты,то не надо экономить,а понравившиеся вам программы надо откопировать на своем магнитофоне (см. ниже).

■ 2. Для работы с компьютером годится любой бытовой кассетный магнитофон, причем чем он проще, тем легче добиться надежной работы.В то же время желательно, чтобы он имел регуляторы тембров(по-возможности раздельные).Очень хорошо,если есть регулировка скорости движения ленты.Компьютер весьма чувствителен к отклонениям скорости. Монофонические магнитофоны во всех случаях предпочтительнее, чем стерео. Обязательным элементом является счетчик ленты, позволяющий быстро находить нужную программу.

■ 3. Приступая к загрузке программы,установите уровень громкости на магнитофоне примерно на 70%-80% от максимального и, если загрузка не пошла,поэкспериментируйте с ним. Помните, что программы чаще страдают от избыточного уровня, чем от недостаточного.Поиск ведите в сторону уменьшения уровня.Иногда на фирменных компьютерах помогает следующий прием.Загрузочный шнур вставляют не в гнездо EAR компьютера, а в гнездо MIC. На самодельных аппаратах этот метод не сработает, т.к. там эти цепи электрически разорваны.

■ 4. Установка регуляторов тембра также имеет большое значение.Нормально "высокие" надо ставить на 80% от максимума, а "низкие" примерно на 20%.Если программа не загружается, а регулировка громкости ничего не даст, то попробуйте установить 50%-50% и 20% - 80%.

5. Загрузка программ сопровождается перемещением цветных полос по бордюрному полю телевизора.По нему можно также многое сказать о настройке магнитофона.Сначала идут в течение 3-х - 5-ти секунд широкие красные и голубые полосы.Это так называемый пилот-тон. Он нужен для того,чтобы процессор подготовился к загрузке.Ширина красных и голубых полос должна быть одинаковой.Если голубые полосы шире,то значит уровень слишком большой.Если шире красные полосы - слишком низкий.Эти полосы должны перемещаться по экрану.Слишком быстрое их перемещение говорит о несоответствии скоростей вашего магнитофона и того,на котором была сделана запись.Неравномерное перемещение (с ускорением и рывками) свидетельствует о некачественности лентопротяжной системы магнитофона либо вашего,либо того, на котором делалась запись. Бывают случаи, когда широкие красные и голубые полосы "превращаются" в широкие желтые и синие.Это может указывать на то, что следует поменять местами установку регуляторов тембра.

6. Если у вас стереомагнитофон и программа не загружается,попробуйте загружать ее отдельно по левому и по правому каналам.

7. Если все пути регулировки исчерпаны, но программа не загружается и при этом точно известно, что копия работоспособна, по всей видимости причина в настройке головок у писавшего и воспроизводящего магнитофона.Как правило,недорогие монофонические магнитофоны имеют в крышке отверстие для узкой крестовой отвертки,чтобы можно было при воспроизведении записи на слух поворотом подпружиненного винта регулировать азимутальное положение головки.При этом следует добиваться наиболее четного и резкого звучания сигналов.Обычно бывает достаточно поворота в ту или иную сторону на угол 120 градусов.Если у вас только один магнитофон, и вы используете его также и для записи программ,то регулировать головки нежелательно,либо придется сталкиваться с необходимостью юстировки.Самый лучший вариант - использовать для воспроизведения программ,полученных со стороны, отдельный магнитофон, выбрав его из самых простых и дешевых, а для записи программ - наиболее качественный, по возможности специализированный аппарат.

8. Обычно программы состоят из нескольких блоков, которые загружаются последовательно один за другим.При плохой загрузке бывает очень трудно загрузить программу,разные блоки которой требуют разной установки органов управления магнитофона.Для того, чтобы обойти эту проблему, надо проводить загрузку в копирующую программу.Если,предположим,при загрузке второго блока произойдет сбой, то первый блок уже загружен и остается, так что его повторять не надо.К тому же, по окончании длительной трудоемкой загрузки, можно сразу же выполнять качественную копию и больше не иметь с этой программой проблем..

9. Сделать процесс настройки магнитофона наглядным и значительно более простым можно с помощью специальных программ, например с помощью программы TAPER.

ПРОГРАММА TAPER

Программа предназначена для настройки кассетного магнитофона.Она анализирует (раскладывает по частям) сигнал, приходящий на вход компьютера и изображает его на экране в виде спектра.Вертикальная шкала имеет три метки.Против этих меток должны находиться пики сигналов во время загрузки.Самая верхняя метка указывает на пик во время считывания "пилоттона",а две другие - показывают уровень "нулей" и уровень "единиц".При прослушивании цифровой записи через программу TAPER пики в участках отмеченных метками,должны быть ярко выражены и никаких других побочных всплесков не должно быть.

КАК ВЫПОЛНИТЬ ЗАПАСНУЮ КОПИЮ ПРОГРАММЫ

Если вы сами написали какую-либо программу на БЕЙСИКе,то выполнить копию несложно. Для этого достаточно дать команду SAVE "имя" и нажать ENTER.Если вы хотите, чтобы программа автостартовала после загрузки, начиная со строки M,то команда имеет вид: SAVE "имя" LINE M.Если в программе есть блоки,записанные в машинном коде,то дается команда SAVE "имя" CODE M,N , где M - адрес,с которого начинается блок кодов,а N - длина этого блока в байтах.При загрузке программ блоки,записанные на БЕЙСИКЕ индицируются на экране как PROGRAM...,а блоки машинных кодов как BYTES...Если вы выгружаете свою разработку, то вы,конечно,знаете те адреса, в которых располагаются ваши блоки.Однако, если вы имеете дело с чужой(фирменной программой), то загрузив ее, вам не так-то просто ее выгрузить.Во-первых, она автостартует и остановить ее сложно.Во-вторых, вам неизвестны адреса блоков, из которых она состоит. Упростить процесс копирования и автоматизировать его можно с помощью специальных копирующих программ.Эти программы имеют свою нестандартную загрузочную систему.Таких программ очень много.Мы здесь рассмотрим несколько наиболее широко распространенных.

COPY 86/M

Это, по-видимому, наиболее удобный и наглядный копировщик общего применения.В исходном состоянии он имеет объем свободного пространства 45000 байтов. и способен компрессировать данные при загрузке.Компрессирование состоит в том,что например вместо последовательности из N нулевых байтов записывается один "нуль", а следующий байт указывает сколько их (N).Это позволяет загружать в копировщик значительно более 45 К.Особенно много места экономится при загрузке экранов, т.к. в графических изображениях часто встречаются длинные последовательности нулей. Результат компрессии данных отправляется на хранение в область экрана, что изображается появлением на экране полос и точек, напоминающих телеграфный код.

Сводка команд программы COPY 86/M

BREAK - переход в исходное положение.
L - загрузить файлы.
C - выгрузить(скопировать файлы).Выгрузка производится с паузой 1,5 секунд между записями.
M - то же,но с паузой 3 сек.
V -VERIFY - проверка выгруженных записей.
D -DELETE - удаление ненужных записей.
H - перевод в шестнадцатиричную систему.
X - просмотр записей с отбоем ранее поданных команд.
S - то же,но без отбоя команд.
B - просмотр БЕЙСИК-программы.
A -ALL - после этой команды очередная команда будет одновременно распространяться на все записи. Все команды подтверждаются нажатием ENTER. На экране файлы записываются в "окне",имеющем 4 строки.Если записей более 4, то одновременно их вывести на экран нельзя, поэтому и необходимы клавиши S и X для "протягивания" записей через "окно". При использовании программы следует помнить, что она может не работать с некоторой периферией.Так, она работать не будет, если к компьютеру подключен кемпсотон-джойстик. К сожалению, эта программа не работает и со многими самодельными компьютерами.В таком случае как правило пользуются другой компрессирующей программой TF COPY.

TF COPY

После загрузки на экране появляется исходное меню.Нажмите "O",программа стартует и перед вами появится основное меню программы,которое выглядит так:

LOAD SAVE DELETE VERIFY MODE

Нажав первую букву нужной команды,вы входите в нужный вам режим.

LOAD - загрузка файлов.
SAVE - выгрузка файлов.
DELETE - удаление файлов.
VERIFY - проверка выгруженных файлов.
MODE - переключение режима.

В программе имеется три режима работы. Режим 1 - 41984 байта свободной памяти. Режим 2 - 44032 байта свободной памяти. Режим 3 - 44288 байтов свободной памяти. Все команды должны завершаться нажатием ENTER. Обратите внимание, что при переходе на другой режим происходит очистка памяти, т.е. загруженная информация будет уничтожена. После выбора одного из вышеуказанных режимов программа выдает запрос.Например "D" дает режим DELETE и появляется вопрос DELETE FROM...TO... Удалить записи с... по... . В ответ надо дать номера тех записей, которые вам нужны.Например, если вы хотите уничтожить запись с третьей по шестую, то соответственно надо ввести 3 и 6.

COPY - COPY (PIRATE 02)

Эта программа находится в эксплуатации довольно давно и представляет классический образец копировщика.Она не выполняет компрессирование и имеет довольно ограниченный объем свободной памяти (42К), но в ней есть несколько оригинальных вспомогательных режимов,которые в ряде случаев делают ее незаменимой.

СВОДКА КОМАНД

Все команды являются ключевыми словами "СПЕКТРУМА" и потому не набираются по буквам.Они требуют завершения нажатием клавиши ENTER. LOAD - выполнить загрузку очередной записи (очередного блока). LOAD N TO M - выполнить загрузку записей,начиная с номера N до номера M. При этом существующие записи с номером N и выше будут уничтожены.т.е. LOAD I уничтожит все ранее существовавшие записи и подготовит компьютер к загрузке новых. Возможные вариации этой команды: LOAD N TO LOAD TO M SAVE - выполнение выгрузки записи(записей) на ленту.Может иметь форму: SAVE N TO M , SAVE TO M, SAVE N TO , SAVE N TO M STEP K Параметр K после оператора STEP показывает величину паузы в секундах, которую компьютер выдержит между отдельными блоками.VERIFY - проверка выгруженных записей.Имеет все те же формы,что и команда LOAD. LOAD AT ADDR - загрузка блока,начиная с адреса ADDR .Например -LOAD AT 16384 - загрузка экрана. LIST ADDR - выдача на экран содержимого памяти,начиная с указанного адреса. LIST = LIST0 POKE ADDR,N - по указанному адресу поместить байт N. Этой командой часто пользуются совместно с командой LIST .Они позволяют вносить изменения в машинный код программы.В принципе с подобными задачами лучше справляются специализированные программы для отладки машинного кода(их часто называют МОНИТОРЫ), но наличие такой функции и в копировщике делает его весьма удобным. COPY 16384 - команда,которая позволяет копировать блоки длиной 49К. Этот режим делает данную программу незаменимой во многих случаях.Дело в том,что если программа имеет один значительный блок длиной порядка 49К и не поддается компрессии,то откопировать его копировщиками типа COPY 86/M и TF COPY не удастся,т.к. он не помещается в памяти.(Например программы FIRELORD и URIDIUM фирмы NUSON CONSULTANTS и др) В режиме COPY 16384 программа COPY - COPY,удаляет сам у себя из памяти и оставляет только небольшой блок длиной несколько байтов,предназначенный для выгрузки загруженного блока.Выгрузка выполняется нажатием клавиши CAPS SHIFT .Возможна только однократная выгрузка,после этого копировщик необходимо снова загрузить.Разумеется магнитофон должен быть включен на запись до нажатия CAPS SHIFT .

ДОПОЛНИТЕЛЬНЫЕ СВЕДЕНИЯ.

Мы рассмотрели только три наиболее распространенные программы копирующие.На самом деле их очень много, но принципы их работы отличаются не очень значительно.В то же время надо помнить, что с помощью копировщиков можно только копировать програм-

мы, в которых не принято специальных мер для защиты от копирования.Многие формы такие меры применяют.Это,например "спидлок"(ускоренный загрузчик),джеркитон(пилоттон,сопровождающийся рывками),блоки избыточной длины(более 50 К, фальшхэдеры (блоки длиной 17 байтов, которые воспринимаются копировщиком как заголовок очередного блока, в то время как они таковыми не являются),укороченный пилоттон. удлиненный пилоттон, слишком широкий или слишком узкий пилоттон, блоки с замеряемой паузой между ними и многие другие методы.Принцип их действия состоит в том, что первый блок программы является БЕЙСИК-загрузчиком, который подготавливает загрузку второго блока. Второй блок в машинных кодах подготавливает нестандартную загрузку прочих блоков,которые уже не могут быть загружены никуда, где предварительно не отработал второй блок, в том числе и копировщик. Копирование таких программ довольно трудоемкий процесс, но со многими защитами справляется например специальная программа LERM - 7 (ТС -7) Другой подход к таким программам состоит в том, что используются специальные периферийные устройства.Их назначение - остановить ("заморозить") программу в каком-либо месте, а затем сделать полный дамп памяти(полную выгрузку всей памяти).Широко известны устройства типа MULTIFACE позволяющие кроме этого производить выгрузку экрана в любом месте программы, вносить в программу изменения и продолжать исполнение программы с места прерывания. Аналогичными устройствами снабжаются иногда и дисковые системы,например БЕТА-ДИСК ИНТЕРФЕЙС (ТР ДОС) фирмы TECHNOLOGY RESEARCH.Он имеет специальную кнопку,называемую MAGIC BUTTON(волшебная кнопка) для прерывания работы программы и сброса ее на диск.Правда запустить откопированную таким образом программу через LOAD не удается.Для этого интерфейс реализует нестандартный подход и выполняет это по команде GO TO "имя". Можно предложить и такой способ копирования с помощью двух магнитофонов. Один магнитофон включается на вход компьютера,а другой на выход.Компьютер считывает информацию бит за битом и тут же транспортирует ее на выход,при этом он выполняет роль компаратора и выравнивает фронты сигналов.Несложная программа в машинных кодах будет предложена вам в следующей главе.Правда, этот метод не очень надежно работает на фирменных "СПЕКТРУМАХ",т.к. у них вход и выход имеют электрическую связь, но на компьютерах"ИАЙМЕКС" и на большинстве самодельных дает отличный результат.

СОСТАВЛЕНИЕ И РЕДАКТИРОВАНИЕ ПРОГРАММ

Компьютер "СПЕКТРУМ" может работать в двух режимах.Это режим непосредственного исполнения команд и программный режим. В режиме непосредственного исполнения команды выполняются после нажатия клавиши ENTER и после исполнения в памяти не хранит-

ся.Если вам надо повторить команду, ее придется снова набрать с клавиатуры.

Например: PRINT 2*2 [ENTER] 4 O.K. - лишний раз убедитесь,что дважды два - четыре.Сообщение O.K. (о'кэй) говорит о том,что команда выполнена и все в порядке.Команды, подаваемые в режиме непосредственного исполнения, называют прямыми командами.Прямая команда может состоять и из нескольких команд.В этом случае они должны разделяться двоеточием.Например, для создания таблицы квадратов целых чисел от 1 до 20 дайте прямую команду:
FOR i=1 TO 20 : PRINT i, i * i NEXT i

В программном режиме команды записываются в виде строк программы.Каждая строка начинается с номера.Строки в программе размещаются в порядке возрастания номеров,поэтому,если вы в программе после строки номер 10 наберете номер 5, то она встанет в программе до строки 10.Ввод строки в программу завершается нажатием ENTER ,а начать работу программы можно прямой командой RUN или RUN ;где N - номер строки,с которой вы хотите, чтобы программа начала работу.Можно запустить программу с нужной строки и прямой командой GO TO N. Обычно строки программы нумеруются через десяток - 10,20,30,... Это делают для того,чтобы в случае необходимости можно было легко вставить пропущенную строку.Так, если вам понадобилось вставить строку между строкой 1 и строкой 2.Вам пришлось их переделать, а чтобы вставить строку между строкой 10 и строкой 20, вам достаточно присвоить ей номер от 11 до 19,например 15. В программе для "СПЕКТРУМА" вы можете использовать любые номера строк от 1 до 9999.В одной строке могут быть записаны несколько команд или операторов, но они должны разделяться двоеточием. Если вы наберете строку с некоторым номером N,,а строка с таким номером у вас уже есть,то после нажатия ENTER новая строка встанет на свое место в программе, а старая будет уничтожена полностью.Это дает эффективный метод стирания ненужных строк.Для этого достаточно набрать номер строки,подлежащей уничтожению, и нажать ENTER. Программа для печати квадратов целых чисел от 1 до 20 будет выглядеть так:

```
10 FOR i=1 TO 20
20 PRINT I, i* i
30 NEXT i
```

Запускается программа командой RUN.После того как программа отработает,она, в отличие от прямой команды, остается в памяти и может быть повторена сколько угодно раз.Прервать исполнение работающей программы можно командой BREAK. Если вам надо внести изменения в программу, то надо ее остановить,затем командой LIST или LIST N вывести на экран текст программы, а затем выполнить редактирование. СПЕКТРУМ имеет строчный программный редактор.Это означает, что редактировать можно каждую строку отдельно. Для того,чтобы редактировать(переделывать) строку, надо сначала вызвать ее в позицию редактирования,которая расположена в нижней части экрана.Это делается так.Предположим, что вам надо

устранить ошибку в работающей программе.Прервите ее нажатием BREAK.Дайте команду LIST для печати текста на экране.Если программа достаточно длинная и на экране целиком не помещается, то после того, как экран будет заполнен,распечатка прекратится и появится запрос SCROLL ?Если строка,которую вы хотите переделать,есть на экране, нажмите в ответ на запрос N(нет) или BREAK, а если ее нет, то любую другую клавишу и распечатка текста программыб на экране будет продолжена. до получения очередного запроса SCROLL ?

Когда нужная строка найдена и вы увидели ее,вы должны подвести к ней курсор.Этот курсор имеет вид стрелки "" и перемещается курсорными клавишами. CAPS SHIFT 6 -вниз и CAPS SHIFT 7 -вверх.Если программа длинная,то перегонять курсор из одного места в другое довольно утомительно.В этом случае лучше сразу обратиться по номеру строки N командой LIST N; а в ответ на запрос SCROLL? нажать BREAK.После этого курсор "" сразу будет установлен против нужной строки. Теперь надо эту строку вызвать в позицию редактирования.Это делается ключевым словом EDIT (CAPS SHIFT). Строка будет повторена в нижней части экрана.Здесь можно вносить в строку изменения.Чтобы удалить из строки ненужные символы, установите курсор в строке с помощью курсорных клавиш справа от символа, подлежащего удалению и нажмите DELETE (CAPS SHIFT 0).Курсорные клавиши в данном случае CAPS SHIFT 5 - влево; CAPS SHIFT 8 - курсор вправо.Ввод новых символов и ключевых слов производится по месту установки курсора.После того, как редактирование строки закончено, нажмите ENTER и если отредактированная строка не содержит синтаксических ошибок, она займет свое место в программе. Процесс редактирования упрощается несколько за счет того, что если после остановки программы нажать не LIST, а сразу ENTER, то на экране появится текст программы, причем именно в том фрагменте, который последним редактировался или вводился.Это сокращает время на поиск нужной строки. Конечно, стандартный редактор СПЕКТРУМА очень удобным не назовешь,но к нему можно быстро привыкнуть.Кроме того, существуют другие версии БЕЙСИКа, например БЕТА БЕЙСИК (версия 1.0; 1.8; 3.0).Эта программа загружается с кассеты.Она предоставляет неплохие возможности для эффективного редактирования программ.Например, она имеет автонумерацию строк,возможность перенумерации строк, команду EDIT N для немедленного редактирования строки N,возможность быстрой переброски курсора,отладочные операторы,дополнительные операторы организации цикла (как в ПАСКАЛе),возможность работы с "окнами",развитые средства структурного программирования(процедуры) и еще несколько десятков удобных функций и команд.Опыт показывает,что при программировании на БЕЙСИКе это программа сокращает время примерно в два раза на малых программах(за счет средств редактирования) и примерно в три-четыре раза на больших(более 200 строк) программах за счет средств структурного программирования и отладочных операторов.

Программа BETA BASIC занимает определенное место в оперативной памяти,но во-первых это критично только для очень больших программ, а во-вторых именно для них программа и предоставляет наибольшие возможности по экономии памяти за счет структурного программирования.Поэтому занятая память на больших программах с лихвой окупается,а на малых не имеет значения. В заключении этой главы мы предлагаем вам набрать небольшую программу,предназначенную для копирования программ с одного магнитофона на другой через компьютер.

```
10 DIM a(12)
20 FOR i=1 to 12
30 READ a(i)
40 POKE (29999+1), a(i)
50 NEXT i
60 DATA 273,175,219,254,31,31,31,211,254,195,50,117
70 RANDOMIZE USR 30000
```

В принципе эта программа написана в машинном коде.БЕЙСИК служит только для того,чтобы организовать этот код по адресу 30000 и стартовать его.

СИСТЕМНЫЕ ПЕРЕМЕННЫЕ

Компьютер выделяет место в оперативной памяти от адреса 23552 по адрес 23733 для размещения в нем некоторых переменных,которые нужны для нормального функционирования компьютера.Тем не менее, пользователь может использовать информацию,содержащуюся в них для повышения эффективности программирования.Первоначальные значения системных переменных устанавливаются после включения компьютера.Их устанавливают те процедуры,которые содержатся в ПЗУ и стартуют после включения.Вы можете в любое время посмотреть содержание однобайтной системной переменной с помощью команды PRINT PEEK ADDR, где ADDR - это адрес ее, а для двухбайтной PRINT PEEK ADDR + 256 * PEEK(ADDR1) Вы можете и сами заслать какие-либо значения (от 0 до 255) в адрес системной переменной оператором POKE, но в некоторых случаях такое изменение будет иметь лишь кратковременный характер и предыдущее значение будет само восстановлено (такие системные переменные помечены буквой N), а в некоторых случаях изменение приведет к сбросу компьютера (такие переменные помечены буквой X).

ТАБЛИЦА СИСТЕМНЫХ ПЕРЕМЕННЫХ

Длина	Адрес	Имя	Содержание
N 8	23552	KSTATE	Используется при сканировании клавиатуры
N 1	23560	LAST K	Запоминает последнюю нажатую клавишу

- I 23561 REPDEL Время, в течение которого надо задержать нажатую клавишу,чтобы произошел автоповтор.Измеряется в 60-тых долях секунды.Исходная установка - 35,но можете заслать сюда свое значение.
- I 23562 REPPER Интервал между повторениями нажатой и задержанной клавиши.Измеряется в 50-х долях секунды.Исходное значение - 5.
- N 2 23563 DEFADD При расчете функции, заданной пользователем,здесь хранится адрес ее аргументов.В противном случае - 0
- N I 23565 K DATA Второй байт кода управления цветом при вводе с клавиатуры
- N 2 23566 TV DATA Здесь хранится информация о цвете и о параметрах AT и TA для телевизора X38 23568 STRMS Адреса каналов, подключенных к потокам.
- 2 23606 CHARS Эта системная переменная указывает на адрес, который лежит на 256 байтов ниже,чем набор символов(знаков),начинающихся с символа "пробел" и кончающийся символом "копирайт".Обычно этот набор находится в ПЗУ,но вы можете организовать свой набор в ОЗУ и изменить CHARS, чтобы она на него указывала.
- I 23608 RASP Продолжительность предупредительного звукового сигнала
- I 23609 PIP Продолжительность звукового щелчка при нажатии клавиши.
- I 23610 ERR NR На единицу меньше, чем код сообщения об ошибке.В исходном состоянии - 255. XI 23611 FLAGS Различные флаги,необходимые для работы операционной системы.
- XI 23612 TV FLAG Флаги,связанные с работой телевизора.
- X2 23613 ERR SP Адрес параметра на машинном стеке, который используется как адрес возврата после ошибки.
- N2 23615 LIST SP Адрес возврата после автоматического листинга.
- NI 23617 MODE Указывает на тип курсора K,L,C или B.
- 2 23618 NEWPPC Номер строки к которой следует переход.
- I 23620 NSPPC Номер оператора в строке, к которому следует переход.Заслав нужное значение в NEWPPC,а затем в NSPPC.Вы можете вызвать переход к нужному оператору в нужной строке.
- 2 23621 PPC Номер строки, в которой расположен исполняемый в данный момент оператор.
- I 23623 SUBPPC Номер оператора в строке,который в данный момент исполняется.

- 1 23624 BORDCR Цвет бордюра,умноженный на 8.Здесь же содержатся атрибуты нижней части экрана.
- 2. 23625 E PPC Номер текущей строки (в которой находится курсор).
- X2 23627 VARS Адрес программных переменных.
- N2 23629 DEST Адрес переменной при исполнении оператора присвоения.
- X2 23621 CHANS Адрес информации о каналах.
- X2 23633 CURCHL Адрес информации, которая в настоящий момент вводится или выводится. X2 23635 PROG Адрес,с которого начинается программа,написанная на БЕйСИКе.
- X2 23637 NXTLIN Адрес следующей строки программы.
- X2 23639 DATADD Адрес разделителя после последнего введенного параметра из блока DATA.
- X2 23641 E LINE Адрес вводимой команды.
- 2 23643 K CUR Адрес курсора.
- 2 23645 CH ADD Адрес следующего интерпретирующего символа.
- 2 23647 X PTR Адрес символа после маркера "?".
- X2 23649 WORKSP Адрес рабочего пространства.
- X2 23651 STKBOT Адрес основания стека калькулятора.
- X2 23653 STKEND Адрес вершины стека калькулятора.
- NI 23655 BREG Содержимое регистра "B" калькулятора.
- N2 23656 MEM Адрес области,используемой в качестве области памяти калькулятора.Обычно совпадают с MEMBOT, но это не всегда.
- 1 23658 FLAGS2 Флаги.
- XI 23659 DF SZ Количество строк в нижней части экрана (в системном окне).
- 2 23660 S TOP Номер верхней программной строки при автоматическом листинге.
- 2 23662 OLDPPC Номер строки, к которой выполняется переход по CONTINUE.
- 1 23664 OSPCC Номер оператора в строке, к которому выполняется переход по CONTINUE.
- NI 23665 FLAG X Вспомогательные флаги.
- N2 23666 STRLEN Длина строковой переменной(стринга) в операторе присвоения.

- N2 23668 T ADDR Адрес следующего параметра в синтаксической таблице.
- 2 23670 SEED Здесь хранится величина для функции RND.Исходное значение можно установить оператором RANDOMIZE N.
- 3 23672 FRAMES Трехбайтная переменная.(Старший байт-первый). Получает приращение на единицу каждую пятидесятую долю секунды.Может использоваться при организации контроляремени.
- 2 23675 UDO Адрес 1-го символа графики пользователя.Вы можете его изменить.
- I 23677 COORD & Координата X последней помещенной на экран точки.
- I 23678 Координата Y последней помещенной на экран точки.
- I 23679 P POZN Номер колонки позиции,в которой находится принтер.
- I 23680 PR CC Младший байт адреса следующей позиции печати по LPRINT (в буфере принтера).
- I 23681 Не используется.
- 2 23682 ECHD E Номер колонки и номер строки конца буфера клавиатуры.
- 2 23684 DF CC Адрес в экранной области памяти позиции печати.
- 2 23686 DF CCL То же,но для нижней части экрана.
- XI 23688 S POSN Номер колонки позиции печати.
- XI 23689 Номер строки позиции печати.
- X2 23690 SPOSNL То же,что и S POSN, но для нижней части экрана.
- I 23692 SCR CT На единицу больше,чем необходимое количество передвижек строк на экране до остановки с запросом SCROLL? Если вы организуете постоянную подачу после PRINT какого-либо числа большего, чем 1 в эту ячейку,то печать будет идти непрерывно,без вопроса SCROLL ?
- I 23693 ATTR P Текущие цветовые атрибуты (для всего экрана).
- I 23694 MASK P Маска локальных атрибутов.Каждый бит этой переменной,если он равен единице,показывают,что соответствующий цветовой атрибут надо брать не из ATTR P,а из того,что уже есть на экране в позиции,для которой выполняется печать.
- I 23695 ATTR T Временные атрибуты.Их установка справедлива

только для выполнения текущего оператора PRINT , INPUT.

- 1 23696 MASK T То же,что и MASK P,но для временных атрибутов.
- 1 23697 PFLAG Флаги атрибутов.
- 30 23698 MEMBOT Область памяти калькулятора.Используется для хранения чисел,которые неудобно помещать в стек калькулятора.
- 2 23728 Не используется.
- 2 23730 RAMTOP Адрес вершины области, доступной для размещения программы на БЕЙСИКе.
- 2 23732 P-RAMT Адрес физической вершины памяти компьютера

ОРГАНИЗАЦИЯ ПАМЯТИ КОМПЬЮТЕРА

Для возможности работы операционной системы компьютера в его памяти выделено несколько областей.Эти области отделены друг от друга некоторыми границами,которые могут иметь постоянный(фиксированный) характер, а могут изменяться в зависимости от конкретных требований.

КАРТА ПАМЯТИ КОМПЬЮТЕРА.

Фиксированные границы обозначены числом,а переменные - именем системной переменной,в которой они хранятся.

- 0...16383 - ПЗУ.
- 16384...22527 - экранная область. Изображение хранится здесь в черно-белом виде.
- 22528...23295 - область атрибутов. Цвета,признаки яркости и мигания.
- 23296...23551 - область буфера ZX-принтера.При отсутствии такого принтера часто используется для хранения небольших процедур в машинных кодах.Здесь они не повреждаются БЕЙСИКОМ, а, кроме того, размещение их в нижней половине памяти компьютера повышает их быстродействие. В компьютерах 128К в этой области хранятся дополнительные системные переменные для поддержания повышенных возможностей этих машин.Там категорически нельзя ничего засылать в эту область памяти.
- 23552...23733 - область системных переменных.
- 23734...GHANS-1 - карты памяти микродрайва.При работе с микродрайвом здесь хранится информация например об испорченных секторах на ленте и т.п.Если микродрайва нет, то эта область не организуется.
- CHANS...PROG - 2 - здесь размещается информация о каналах.

- PROG...VARS - 1 - текст программы,записанной на БЕЙСИКе.Отметим,что он хранится несколько в иной форме,чем то,что вы видите на экране.
- VARS...ELINE - 2 - область хранения переменных, с которыми оперирует БЕЙСИК-программа.
- ELINE...WORKSP - 1 - буфер редактора.Сюда помещается строка программы, когда вы выполняете ее ввод или редактирование.
- WORKSP...STKBOT - 1 - буфер инструкции INPUT.
- STKBOT...STKEND - 1 - стек калькулятора.К нему обращается интерпретатор БЕЙСИКа.Калькулятор служит для выполнения арифметических операций.Калькулятор может вызываться из машинного кода командой ассемблера RST 40. По окончании операции на вершину стека помещается ее результат.
- STKEND...sp - свободное пространство.Сюда развиваются навстречу друг другу стек калькулятора и машинный стек.sp - это не системная переменная.Это указатель машинного стека.Его можно вызвать только из машинного кода.Из БЕЙСИКа он недоступен. sp...ERR SP - машинный стек.Нужен процессору Z80 - для выполнения обработки данных,адресов и пр. ERR SP + 1...RAMTOP - стек возвратов.(GO SUB - стек).Здесь хранится информация о номере строки, к которой надо вернуться после выполнения подпрограммы по RETURN.
- RAMTOP...UDB - свободная область.Может использована для хранения про- цедур в машинных кодах.
- WOG...PRAMT - область графики пользователя.
- PRAMT - физическая вершина памяти компьютера.Для СПЕКТРУМА 48 - PRAMT= =65535,а для СПЕКТРУМА 16 -это 32767.

МЕТОДЫ ВВЕДЕНИЯ РУССКОГО ШРИФТА

Обеспечение возможности работы компьютера с символами русского алфавита является,наверное,одной из первейших задач,которые возникают перед тем,кто хочет сам писать программы на СПЕКТРУМЕ. Существует два принципиально различных подхода для решения этой задачи.Первый подход состоит в использовании символов графики пользователя,а второй - в создании нового набора символов,размещении его в оперативной памяти и нажатии его в качестве действующего с помощью системной переменной CHARS.

ИСПОЛЬЗОВАНИЕ ГРАФИКИ ПОЛЬЗОВАТЕЛЯ

В графическом режиме вы можете,во-первых,набирать графические символы блочной графики,расположенные на цифровых клавишах,а также можете воспользоваться символами графики пользователя.Их может быть до 21; они берутся нажатием на клавиши от "A" до "U" в графическом режиме(курсор"G").Изображение символов графики пользователя должны быть сформированы в специальной области памяти,на которую указывает системная переменная UDG.Адрес,в котором расположено изображение символа (8 байтов) может быть получен с помощью функции USR.Таким образом,зная,начиная с адреса USR "A", 8 байтов,формирующих изображение,скажем буквы "Я",мы будем иметь возможность нажатием клавиши "A" в графическом режиме печатать на экране букву "Я". Учитывая,что многие буквы русского и латинского алфавита имеют сходное начертание(например O, A, X, P, и др.можно задать прописные буквы,не имеющие аналогов и при этом уложиться в разрешенный 21 символ. Изображение буквы "Я" формируется из точек экрана (пикселей) например так:

```
00000000 0
00xxxxx0 0 128+0 64+1 32+1 16+1 8+1 4+1 2+0 1 = 62
0x0000x0 0 128+1 64+0 32+0 16+0 8+0 4+1 2+0 1 = 66
0x0000x0 0 128+1 64+0 32+0 16+0 8+0 4+1 2+0 1 = 66
00xxxxx0 0 128+0 64+1 32+1 16+1 8+1 4+1 2+0 1 = 62
00x000x0 0 128+0 64+1 32+0 16+0 8+0 4+1 2+0 1 = 34
0x0000x0 0 128+1 64+0 32+0 16+0 8+0 4+1 2+0 1 = 66
00000000 0
```

Таким образом, в результате такого синтеза шаблона буквы "Я",мы можем представить ее последовательностью из 8 байтов,разместив их в соответствующем месте - 0, 62, 66, 62, 34, 66, 0. Точно так же, можно изобразить и остальные желаемые символы.Для тех,кто не хочет глубоко вдаваться в суть процесса,мы предлагаем небольшую готовую программу, которую вы можете набрать и выгрузить на ленту.Всякий раз,когда вы хотите составить какую-либо про

грамму,в которой возможна необходимость в русских буквах,загрузите эту программу и стартуйте ее (RUN).После остановки программы дайте LIST и продолжайте со строки 100 набирать свою программу.

```
10 FOR Z=1 TO 20
20 READ A
30 FOR X=0 TO 7
40 READ Y
50 POKE USR A +X,Y
60 NEXT X
70 NEXT Z
80 DATA "B",0,126,64,124,66,66,124,0
81 DATA "G",0,126,64,64,64,64,64,0
82 DATA "D",0,28,36,36,36,36126,66
83 DATA "J",0,65,73,62,73,73,65,0
84 DATA "I",0,66,70,74,82,98,66,0
85 DATA "L",0,30,34,34,34,34,98,0
86 DATA "P",0,126,66,66,66,66,66,0
87 DATA "O",0,66,66,36,24,16,96,0
88 DATA "C",0,68,68,68,68,68,126,2
89 DATA "H",0,66,66,66,126,2,2,0
90 DATA "N",0,65,73,73,73,73,127,0
91 DATA "M",0,65,73,73,73,73,127,1
92 DATA "E",0,60,66,30,2,66,60,0
93 DATA "U",0,76,82,114,82,82,76,0
94 DATA "A",0,62,66,66,62,34,66,0
95 DATA "S",24,66,70,74,82,98,66,0
96 DATA "R",0,64,64,124,66,66,124,0
97 DATA "T",0,192,64,124,66,66,124,0
98 DATA "F",0,127,73,73,73,127,8,0
99 DATA "Q",0,66,66,114,74,74,114,0
```

В результате работы этой программы устанавливается следующее соответствие между клавишами в графическом режиме и символами русского алфавита:

"Q" = "Ы" "R" = "Ь" "U" = "Ю" "O" = "Ч" "A" = "Я"
"D" = "Д" "." = "Г" "," = "Х" "C" = "Ц" "N" = "Ш"
"E" = "Э" "T" = "Ъ" "I" = "И" "P" = "П" "S" = "И"
"F" = "Ф" "H" = "Н" "L" = "Л" "B" = "Б" "M" = "Щ".

ИЗМЕНЕНИЕ НАБОРА ЗНАКОВ

Изменение знаков компьютеров "СПЕКТРУМ" состоит из 96 символов и занимает 96 8 = 768 байтов.Стандартный набор находится в ПЗУ по адресу 15616.Этот адрес можно определить с помощью системной переменной CHARS. PRINT PEEK 23606+256*PEEK23607+256 [ENTER]15616 О.К. При этом по адресу 23606 находится 0, а по адресу 23607 - 60. Изменение набора знаков состоит в том,чтобы сформировать с помощью какой-либо специализированной программы или вручную желаемый набор знаков,поместить его в любое свободное

место оперативной памяти и изменить содержимое ячеек 23606 и 23607,чтобы оно указывало на адрес места расположения нового набора.Рассмотрим процедуры на конкретном примере.

1. Для форматирования набора знаков очень удобно использовать графический редактор ARTSTUDIO .Эта программа является не только идеальным графическим редактором,позволяющим легко выполнять графические изображения,но и позволяет конструировать шрифты.Выгрузим сформированный шрифт на ленту.

2. Допустим,мы хотим разместить новый шрифт,начиная с адреса 30000.Загрузим его с ленты LOAD "" CODE 30000,768.

3. Системная переменная CHARS должна указывать на адрес,находящийся на 256 байтов ниже,т.е. на 30000-256=29744.

4. Определим старший байт двухбайтной переменной CHARS .B2=INT (29744/256)=116.

5. Определим младший байт: BI=29744-116 256=48.

6. Переключение на печать сформированным вами шрифтом выполняется командой POKE 23606,48: POKE 23607,116.Обратное переключение на латинский шрифт - командой POKE 23606,0: POKE 23607,60.

ЗАЩИТА ПРОГРАММ

Защита программ состоит в исключении возможности просмотра текста программы и внесения в него изменений лицом,не уполномоченным на это автором программы. Методов защиты программ существует очень много, и здесь мы рассмотрим только основные принципы и обозначим подходы к этой проблеме. Можно выделить три основные направления:

1. Исключить возможность остановки (прерывания работы программы).

2. Исключить возможность подачи команды LIST для распечатывания текста программы.

3. Сделать листинг программы нечитаемым.

ИСКЛЮЧЕНИЕ ВОЗМОЖНОСТИ ОСТАНОВКИ ПРОГРАММЫ

Самое первое,что вы должны сделать - это выполнить автостартующий со строки номер N. Это выполняется при выгрузке программы командой SAVE "имя" LINE N. Теперь программа после загрузки будет сама стартовать с указанной строки.Тем не менее, она может быть остановлена командой BREAK или по сообщению об ошибке. Отключить клавишу BREAK можно,если в той строке,с которой происходит автостарт,поместить команду POKE 23613, PEEK 23730 - 5. Если программа вами хорошо отлажена,то остановку по ошибке можно исключить.Надо помнить,что чаще всего при взломе программы

сознательно вносят ошибку во время исполнения оператора INPUT.Так,если программа просит от пользователя ввести какое-либо число,он сознательно нажимает на клавишу с буквой.Программа останавливается с сообщением VARIABLE NOT FOUND.Рекомендуем не использовать оператор INPUT, а организовать опрос клавиатуры на основе функции INKEY. Если в строке автостарта поместить нижеприведенные команды,то при попытке сделать BREAK или при проявлении ошибки,программа будет сбрасывать

LET ERR = 256*PEEK 23614+PEEK 23613: POKE ERR,0: POKE ERR+1,0

Примерно тот же эффект дает помещение в стартовой строке команды POKE 23659,0. Одним из способов остановки автостартующей программы является использование команды MERGE "" вместо LOAD ""при загрузке. Если ваша программа имеет размер больше 7К, то защитить ее от этого приема можно,если перед выгрузкой подать следующие прямые команды:

LET X = 256*PEEK 23636 + PEEK 23635: POKE X,60: POKE X+1,0

Другой способ защиты от ME E состоит в том,чтобы перед выгрузкой вашей программы на ленту,дать ложную информацию о том,что якобы первая строка вашей программы имеет чрезмерно большую длину.Это делается прямой командой: POKE(PEEK23635 + 256*PEEK 23636 + 3),255.

ОТКЛЮЧЕНИЕ КОМАНДЫ LIST

Введите первой строкой программы I REM.Перед выгрузкой программы на ленту дайте прямую команду:

POKE (PEEK 23635 + 256*PEEK 23636),100

Теперь программу можно запускать (RUN), но читать нельзя.

КАК СДЕЛАТЬ ТЕКСТ ПРОГРАММЫ НЕЧИТАЕМЫМ

Простейший прием, применяемый во многих программах состоит в том,что устанавливают одинаковые цвета символов (LINK) и фона (PAPER). Например: 10 INK 7: PAPER 7

В тех же местах,где программа должна сделать вывод на экран в операторе PRINT вставляют в качестве временных правильные цвета:

20 PRINT INK 0; "ZX-Spectrum"

Другой способ состоит в искажении набора символов путем задания "фальшивого" значения системной переменной CHARS. Попробуйте,например:

POKE 23606,8: PRINT "ZX-Spectrum"

Другой способ состоит в искажении набора символов путем задания "фальшивого" значения системной переменной CHARS.Попробуйте, например:

POKE 23606,8: PRINT " ZX-Spectrum ":POKE 23606,0

Если адресовать системную переменную CHARS в те области памяти, где вообще ничего нет, например POKE 23607,200, то все

символы будут выглядеть как пробелы и на экране вообще ничего не будет.Вам же нужно перед всяким оператором PRINT включать истинное значение CHARS и включать его после PRINT. Еще один регулярно используемый прием состоит в организации "нулевой"строки, которая реально используется программой,но не может быть удалена при редактировании.Чтобы первая строка стала нулевой, дайте команду. POKE 23755,0: POKE 23756,0.

ОПЕРАЦИИ С ЭКРАНОМ

Типичной задачей для тех,кто начинает создавать собственные программы на "СПЕКТРУМЕ" является создание красочных изображений в графике высокого разрешения,хранения их в памяти компьютера и вывод на экран. Мы не будем рассматривать графические операции с операторами DRAW, CIRCLE, PLOT, а также печать символов графики пользователя, а рассмотрим операции с экраном, созданным в графическом редакторе.Наиболее удобным в работе является графический редактор ARTSTUDIO, но можно использовать и какой-либо другой доступный. Созданное в редакторе изображение вы можете выгрузить на ленту в виде блока кодов, длина которого 6912 байтов.Если теперь выгрузить его командой LOAD "" CODE 16284, 6912 то изображение сразу будет загружено на экран, но хранить его там, естественно, нельзя.Можно загрузить изображение в произвольное место и вызвать его от туда на экран по мере необходимости.Примем адрес для хранения картинки 30000 ; LOAD "" CODE 30000,6912 Переброску на экран можно организовать в БЕЙСИКе.

```
500 FOR I = 1 TO 6912
510 POKE 16383 + 1, PEEK 29999 + 1
520 NEXT X
530 RETURN
```

Всякий раз,когда вам надо вызвать изображение на экран, вы можете делать это командой GO SUB 500. К сожалению, этот метод очень медленно работает, что связано с ограниченными возможностями БЕЙСИКа.Построение экрана происходит более 20 секунд. Возможность быстрой переброски экрана (менее 1 сек.)предоставляет программирование в машинном коде.Процессор Z-80 для этой цели имеет мощную команду перемещения блоков LDIR (ее код 237,176).Для ее работы необходимо, чтобы в регистре H процессора находился адрес начала блока,подлежащего переброски(в нашем случае - 30000), в регистре BC - длина блока(6912),а в регистре DE - адрес места назначения (16384).Тогда программа в машинном коде будет выглядеть так:

Машинный код	Ассемблер	Комментарий
33 48 117	LD HL,48 117	Загрузить в регистр HL адрес 30000. Обратите внимание на то,что 48 + 117*256 = 30000.

```
1  0  27        LD BC,0 27   Загрузить в регистр BC 6912.
17  0  64       LD DE,0 64   Загрузить в регистр DE 16384.
237  176        LDIR         Выполнить переброску блока в
                             новое место.
201             RET          Возврат в вызывающую программу.
```

Таким образом, программа в маш. коде выглядит так: 33, 48, 117, 1, 0, 27, 17, 0, 64, 237, 176, 201

Поместим ее в произвольное место оперативной памяти, например начиная с адреса 65000 следующей программой написанной на БЕЙСИ-Ке.

```
10 FOR I=1 TO 12
20 READ A
30 POKE 59999+1,A
40 NEXT I
50 DATA 33,48,117,1,0,27,17,0,64,237,176,201
```

Теперь всякий раз, когда вам нужно поместить картинку на экран, вам надо стартовать эту процедуру,что выполняется командой RANDOMIZE USR 65000. Если у вас в памяти компьютера хранится не одно, а несколько графических изображений,то их можно также перебрасывать на экран этой процедурой,надо только изменить адрес,поступающий в регистр процессора HL,что можно сделать командой POKE по адресу 65001 и по адресу 65002. Предположим, что у вас есть вторая картинка, помещенная начиная с адреса 37000.Тогда (37000/256)=144, а 37000-256*144=136.Выполним POKE 65001,136 и POKE 65002,144.Команда RANDOMIZE USR 65000 выполнит переброску второй картинки и т.д. Очевидно, что поскольку каждый экран занимает 6912 байтов, то одновременно в памяти нельзя хранить более 6 изображений, что во многих случаях бывает недостаточно.Если учесть, что каждое изображение имеет длительные последовательности нулей, то имеется возможность компрессировать каждое изображение.Это позволяет во многих случаях вдвое сократить объем памяти,занимаемой графикой,а иногда даже больше. Мы предлагаем вам две процедуры в машинном коде.Первая процедура называется FNS и служит для компрессирования изображения с экрана и размещения его в заданном пользователем адресе.Она имеет два параметра h и l,которые задают адрес для хранения экрана.h- старший байт адреса,а l- младший байт. Сама процедура FNS расположена начиная с адреса 56600.Длина - 60 байтов. Вторая процедура служит для компрессирования изображения и переброски его на экран.Она имеет два тех же параметра .Адрес размещения этой процедуры -56500.Длина - 40 байтов. Чтобы создать процедуру ,вам необходимо набрать и запустить следующую БЕЙСИК-программу.После того,как она отработает, выгрузите созданную процедуру на ленту командой SAVE "" CODE 56600,60 и в дальнейшем можете использовать в своих программах.Точно также используется копия процедуры : SAVE "" CODE 56500,40.

ПРОЦЕДУРА

```
8300 LET B = 56600: LET L=60: LET ZX=0: RESTORE 8310
8301 FOR I = 0 TO L = 1 : :READ a
8302 POKE (B + I),a: LET Z = Z+a
8303 NEXT I
8304 LET Z = INT(((Z/L)-INT(Z/L))*I)
8305 READ A: IF AZ THEN PRINT"???"; STOP
8310 DATA 42,11,92,1,4
8311 DATA 0,9,86,14,8
8312 DATA 9,94,237,83,82
8313 DATA 221,33,0,64,6
8314 DATA 1,126,44,32,8
8315 DATA 36,245,124,254,91
8316 DATA 40,16,241,78,185
8317 DATA 32,4,4,32,238
8318 DATA 5,18,19,120,18
8319 DATA 19,24,227,241,18
8320 DATA 19,120, 18,237,83
8321 DATA 22,221,201,0,0
8322 DATA 56,0,0,0,0
```

После того,как процедура выполнит компрессирование экрана и перебросит его в место,отведенное для дополнительного хранения, вы можете узнать адрес,в который можно помещать следующий экран. PRINT PEEK 23296 + 256 PEEK 23297.Но помните,что делать это можно только НЕМЕДЛЕННО после окончания компрессирования и переброски блока.

ПРОЦЕДУРА FNT

```
8350 LET B=565; LET L=35: LET Z=0; RESTORE 8360
8351 FOR I=- TO L=1; READ A
8352 POKE (B+1),A; LET Z=Z*A
8353 NEXT I 8354 LET Z=INT(((Z/L)-INT(Z/L))*L)
8355 READ A: IF AZ THEN PRINT "???":STOP
8360 DATA 42,11,92,1,4
8361 DATA 0,9,86,14,8
8362 DATA 9,94,33,0,64
8363 DATA 26,245,19,26,19
8364 DATA 71,241,119,35,16
8365 DATA 252,124,254,91,32
8366 DATA 240,201,0,0,0
8367 DATA 28,0,0,0,0
```

☐ПРИМЕР ПРИМЕНЕНИЯ ПРОЦЕДУР КОМПРЕССИИ И ДЕКОМПРЕССИИ

Предположим,что вам надо компрессировать и отправить на хранение три экрана scr1, scr2, scr3, причем первый из них идет в адрес 30000(48,117).

Тогда основная программа может выглядеть так:

```
10 DEF FN s(h,l) =USR 56600 20 DEF FN t(h,l) =USR 56500
100 LET a=23296 : LET b=23297
```

```
110 LET h1 = 117 : LET l1 = 48
120 LET ad = l1 + (h1*256)
130 PRINT ad : LOAD"scr1" CODE 16384,6912
140 RANDOMIZE FNs(h1,l1)
150 LET l2 = PEEK a: LET h2 = PEEK b
155 LET ad = l2 + h2*256
160 PRINT ad : LOAD "scr2" CODE 16384,6912
170 RANDOMIZE FN s(h2,l2)
180 LET 13 = PEEK a:LET h3 = PEEK b
185 LET ad = 13 + 256*h3
190 PRINT ad : LOAD "scr3" CODE 16384,6912
200 RANDOMIZE FN s(h3,l3)
210 PAUSE 50
300 RANDOMIZE FN t(h1,l1)
310 PAUSE 50
320 RANDOMIZE FN t(h2,l2)
330 PAUSE 50
340 RANDOMIZE FN t(h3,l3)
350 PAUSE 50 :GO TO
```