

«ИНФОРКОМ»

**ПЕРСОНАЛЬНЫЙ КОМПЬЮТЕР
ZX-SPECTRUM**
Сборник статей



ZX-ФОРУМ №3



МОСКВА-1995

**PDF version by Deny (Денисенко Д.А.)
e-mail: DenyDA@mail.ru
2008**



Сборник содержит программные и литературные разработки пользователей персонального компьютера "ZX-Spectrum". Книга рассчитана на тех, кто самостоятельно работает над освоением программирования и увлекается компьютерными играми.

СОДЕРЖАНИЕ

СЕКРЕТЫ TR-DOS

Полная адаптация программ к диску без
использования системных переменных
TR-DOS

Станислав Шилов

5

ЧИТАТЕЛЬ-ЧИТАТЕЛЮ

Краткое описание LASER GENIUS

ASSEMBLER/MONITOR

Камертон и музыкальный сопроцессор

Поиск "вечных жизней"

Построение окружности и круга

Процедура построения окружности и круга

Нестандартная загрузка

Илья Фомин

Рюмик С.М.

Сергей Бородавкин

Михаил Усачёв

Герасим Троеглазов

Павел Старков

20

42

55

66

66

76

ADVENTURE GAMES

Алексей Федяев

Евгений Федяев

81

СОВЕТЫ ЭКСПЕРТА

F-19 STEALTH FIGHTER

MARSPORT

Дмитрий Усманов

Станислав Шилов

92

110

POKES

126

ПАРОЛИ И СОВЕТЫ

142

КОМПЬЮТЕРНАЯ НОВЕЛЛА

Узник шахты

SOS или "Русалочка" в беде

Андрей Андреев

148

ПРИЛОЖЕНИЕ

PRESENTATION MASTER

Сергей Ханцис

157

СЕКРЕТЫ TR-DOS

Полная адаптация программ к диску без использования системных переменных TR-DOS (На примере игры DUNDARACH фирмы GARGOYLE GAMES)

© Станислав Шилов, г. Пермь, 1995.

В последнее время на дисках появилось множество адвентюрных программ с одним недостатком - у них нет выгрузки на диск. На диск программу перевести не так уж сложно, а вот сделать приличную процедуру загрузки/выгрузки положения игры с диска, куда сложнее. Зачастую стандартными приемами TR-DOS здесь не обойдешься, т.к. большинство игр с выгрузками расположены сразу после системных переменных Бейсика или вообще занимают всю память, так что для системных переменных TR-DOS, тем более для буфера TR-DOS совершенно нет места. Как же поступают в таком случае? Самое простое, что можно сделать - это сделать программу записывающей выгрузки саму в себя. Все кодовые блоки такой программы, обычно соединяют вместе в один бейсик-блок и оставляют несколько секторов после всех кодовых блоков для выгрузки. Для записи и загрузки положения можно воспользоваться двумя подпрограммами в ПЗУ TR-DOS: #2D73 - запись сектора, #2F1B - чтение сектора. Для чтения/записи блока секторов придется писать собственную программку (уже готовая есть в SPECTROFON'e № 3). Но у этого способа адаптации есть несколько недостатков. Перечислю их. Во-первых, если попытаться записать положение игры на защищенный от записи диск, то на экране появится сообщение "READ ONLY...", после чего можете спокойно сбрасывать игру и перезагружать снова. Аналогичная ситуация возникает при сбойном секторе, только возникнет сообщение "DISK ERROR...". Например, если в дисковод поставить IS-DOS диск и попробовать загрузить положение игры, то на экране возникнет вышеупомянутое сообщение. Во-вторых, очень сильно пользователя ограничивает только одна выгрузка в игре, тем более сложной.

Как же избежать этих недостатков? В приведенных в этой статье программах устранены все перечисленные ошибки, но есть один недостаток, т.к. в DUNDARACH совершенно не было места для процедур чтения/записи, поэтому можно поступить следующим образом. Подгружать все программы чтения/записи файлов с диска, а в памяти оставить только программу чтения секторов с диска. Программу, которая загружает сектора, и постоянно находится в памяти, будет в дальнейшем называться - "считывающей", а программа, которая загружается в память с диска - "основной".

"Основная" программа подгружается в экранную область с диска с помощью "считывающей" программы, которая находится в памяти вместо кассетных подпрограмм. После чего запускается "основная" программа и выполняет запись или загрузку файла с выгрузкой. Сначала рассмотрим, так называемую "считывающую" программу. При вызове опции 2 (SAVE GAME) или 3 (RESTORE GAME) меню OPTIONS, программа переходит соответственно на метки JRSAVE и JRLOAD. Затем очищается экран (строка 170), после чего инициализируется дисковод методом прямого обращения к ВГ-93 (об этом методе я расскажу позже). Затем читаем несколько секторов в экранную область с адреса #4B00 (переменная DATPLC) с "основной" программой (строки 240-660, 810-940). После этого проверяем считанную информацию, если первые два байта равны #A7, #1F, то значит, в дисководе стоит нужный диск, иначе программа просто выйдет в меню OPTIONS. После этого запускается "основная" программа. Теперь объясню, как всё работает в "основной" программе. Сначала запрашиваем версию букву выгрузки (подпрограмма ENTNAM), затем считываем каталог диска (подпрограмма RDCAT) с адреса #5000 (пер. CAT). После чего проверяется длина каждого файла, если длина файла соответствует длине выгрузке, то данные этого файла переносятся на адрес #5000, затем #5010 и т.д. Затем сравнивается каждое имя прошедшего проверку файла с данными нужного нам имени с адреса #AF2C-FNAME (подпрограмма FINDAF), если файл соответствует нужной

выгрузке, то процедура LOAD загружает данные файла сначала в буфер с адреса #4000, а затем перекидывает нужное количество байтов в память программы командой LDIR. Если процедура SAVE находит файл, соответствующий выгрузке, то сразу переходит на процедуру записи данных, т.е. попросту перезаписывает его, иначе создает новый файл в каталоге диска (подпрограмма NEWF). При создании нового файла с адреса #4F00-CATDAT загружаются данные диска, вносятся требуемые изменения, после чего они записываются на диск. Затем загружается сектор с последним файлом и в этот сектор заносится имя нового файла, после чего подпрограмма SVE записывает данные в файл. Ну, вот в принципе и вся работа "основной" программы.

Перечислю ошибки, которые могут возникнуть при работе с диском. Ошибки, на которые не выдается сообщения:

1. Попытка записи на защищенный диск;
2. Попытка записи на не отформатированный диск;
3. Попытка чтения выгрузки с диска со сбойными секторами или с не TR-DOS диска.

В случае этих ошибок программа будет ждать пока Вы не вставите другой диск или не нажмете клавишу SPACE для выхода в меню OPTIONS. Если в дисковом нет диска, то программа будет ждать до тех пор, пока Вы его не вставите. Ошибки, которые сопровождаются сообщением:

1. Нет выгрузок на диске (NO FILE(S) ON DISK);
2. Файл не найден (FILE ABSENT);
3. Нет места на диске (NO SPACE ON DISK).

Вот и все возможные ошибки, которые могут возникнуть у Вас при работе с диском.

Третьей отдельной программой, которая входит в дисковый DUNDARACH - это программа START. Компилируется она вместе со "считывающей" программой, т.к. вносит изменения именно в эту программу. Она настраивает программу на текущий дисковод (A...D), на тип дисковода (односторонний, двухсторонний) и дает "считывающей" программе координаты "основной" программы (дорожка, сектор). Ну, вот я и объяснил все что мог, остальное ищите в комментариях к программам.

Теперь объясню метод обращения напрямую к ВГ-93. Приведенные в этой статье программы пользуются только тремя подпрограммами TR-DOS.

Занесение в регистры ВГ-93 команд.

```
#20B8  OUT      (C),D      ;В регистре С номер порта, в регистр D - число
      DJNZ     #20B1      ;В регистре В обязательно должна быть 1.
      RET                      ;Выход.
```

Жалко, что разработчики TR-DOS не догадались написать аналогичную подпрограммку для чтения портов. Если бы она была, то можно было читать регистр состояния ВГ-93.

Запись данных на диск.

```
#3FCA  IN       A, (#FF)   ;Регистр состояния ВГ-93.
      AND      #C0        ;Ожидание
      JR       Z, #3FCA   ;готовности ВГ-93.
      RET      M          ;Выход, если ВГ-93 окончил запись.
      OUTI                     ;Вывод байта на диск.
      JR       #3FCA      ;Переход на запись следующего байта.
```

Чтение данных с диска.

```
#3FE5  IN       A, (#FF)   ;Регистр состояния ВГ-93.
      AND      #C0        ;Ожидание
      JR       Z, #3FE5   ;готовности ВГ-93.
      RET      M          ;Выход, если ВГ-93 окончил загрузку.
      INI                     ;Ввод байта с диска.
      JR       #3FE5      ;Переход на загрузку следующего байта.
```


Все остальное, относящееся к ВГ-93 Вы можете прочитать в ZX-РЕВЮ № 4, 1994, стр. 10 и в ZX-РЕВЮ № 5, 1994, стр. 19. Для читателей, которые хотят применять "основную" программу для собственных, даю подпрограмму PRINT, без которой ничего работать не будет отдельно от игры.

```
PRINT  POP    HL      ;адрес возврата из подпрограммы в регистре HL.
        CALL   PRN     ;Вызываем подпрограмму печати.
        JP     (HL)    ;Выходим на следующий после текста оператор.
PRN     LD     A, (HL)  ;В регистре A символ.
        INC    HL      ;HL=HL+1
        CP     "%"     ;Конец текста?
        RET    Z       ;Если да, то выход.
        PUSH   HL      ;
        RST    #10     ;Печатаем символ.
        POP    HL      ;
        JR     PRN     ;Переход на следующий символ.
```

Ещё хочу сказать, что с адреса на которую указывает переменная FNAME, должны быть все данные о файле. Т.е. если Вы хотите использовать программу в своих целях, то вместо команды FNAME EQU #AF2C вставьте:

```
FNAME  DEFM    "dun"   ;имя файла (8 байтов).
LETTER  DEFB    "."     ;Версионная буква.
        DEFB    #20, #20
        DEFB    #20, #20
TYPE    DEFB    "C"     ;тип файла "C" - CODE.
LSTR    DEFB    00      ;стартовый адрес файла.
HSTR    DEFB    00      ;
LLEN    DEFB    AC      ;длина файла.
HLEN    DEFB    0A      ;
SLEN    DEFB    0B      ;длина в секторах.
SSECT   DEFB    00      ;Начальный сектор.
STRK    DEFB    00      ;Начальная дорожка.
```

Если Вы используете другие данные в описании файла, то не забудьте в "основной" программе исправить строки 1440, 1470, 1490.

Считывающая программа.

```
10      ORG     #AFE5    ;Адрес компиляции программы.
20 SAVE  EQU     #4C9B    ;Адрес подпрограммы записи.
30 LOAD  EQU     #4C14    ;Адрес подпрограммы загрузки.
40 DATPLC EQU    #4B00    ;Адрес загрузки основных подпрограмм.
50 PRINT  EQU    #AD1A    ;Адрес подпрограммы печати.
60 RDCAT  EQU    #4B96    ;Адрес подпрограммы чтения каталога.
70 MENU   EQU    #AF3C    ;Адрес выхода в меню.
80 BLKLNG EQU    #0300    ;Длина в секторах загрузочного блока.
90 CLS    LD     HL, #5800 ;Подпрограмма очистки экрана.
100      LD     DE, #5801  ;
110      LD     BC, #02FF  ;
120      LD     (HL), L    ;
130      LDIR             ;
140      RET              ;
150 JRSAVE LD     HL, SAVE  ;Сюда осуществляется переход
160      JR     LOADAT     ;из опции 2 (SAVE GAME),
170 JRLOAD LD     HL, LOAD  ;из опции 3 (RESTORE GAME)
180 LOADAT LD     (COMAND+1), HL ;Установка перехода.
190      CALL   CLS       ;Очищаем экран.
```

```

200      CALL BG93      ;Инициализируем дисковод.
210      DEFB #1F      ;
220 RES   DEFB #0C      ;
230      CALL RESET    ;Очистка ВГ-93.
240      CALL BG93      ;Устанавливаем головку дисковода на
250      DEFB #7F      ;нужную дорожку в регистр данных.
260 TRACK DEFB #00      ;Номер дорожки.
270      CALL BG93      ;Устанавливаем системный регистр
280      DEFB #FF      ;(#3C-сторона 1, #2C-сторона
290 SIDE1 DEFB #3C      ;0 для дисковода А).
300      CALL BG93      ;Команда ВГ-93 поиска
310      DEFB #1F      ;дорожки и стороны.
320      DEFB #1C      ;#1C=000111000b.
330      CALL RESET    ;Очистка ВГ-93.
340      LD HL,DATPLC   ;Адрес загрузки в памяти.
350 SECTOR LD BC,BLKLN  ;Регистр В-длина, регистр С-сектор.
360 LOOP  PUSH BC       ;Сохранение BC.
370      PUSH HL        ;Сохранение адр. начала загрузки.
380      LD A,C          ;Устанавливаем
390      LD (SECTPL),A   ;сектор из
400      CALL BG93      ;регистра С
410      DEFB #5F      ;в порт #5F
420 SECTPL DEFB #01     ;TR-DOS.
430      CALL BG93      ;Команда ВГ-93 -
440      DEFB #1F      ;- чтение сектора.
450      DEFB #80      ;#80=10000000b.
460      POP HL         ;В HL - адрес
470      PUSH HL        ;начала загрузки.
480      CALL R_LOAD    ;Вызов подпрограммы чтения сектора.
490      POP DE         ;Проверка
500      PUSH DE        ;на ошибку
510      INC D          ;чтения,
520      AND A          ;если HL
530      SBC HL,DE      ;не
540      POP HL         ;равен 0,
550      POP BC         ;то
560      JR NZ,EXIT     ;выход в меню.
570      LD A,#7F      ;Если нажата
580      IN A,(#FE)     ;клавиша
590      AND #01        ;SPACE, то
600      JR Z,EXIT      ;выход в меню.
610      INC H          ;Адрес=адрес+#0100.
620      INC C          ;Увеличение номера сектора.
630      LD A,C         ;Если номер=#11 (конец дорожки),
640      CP #11         ;то переход на следующую
650      JR Z,NEXTRK    ;дорожку или сторону.
660 DJN   DJNZ LOOP     ;Чтение следующего сектора.
670      LD HL,DATPLC   ;Если первые
680      LD A,#A7        ;два байта
690      CP (HL)         ;прочитанного
700      JR NZ,EXIT     ;блока
710      LD A,#1F      ;не равны #A7,#1F,
720      INC HL         ;то
730      CP (HL)        ;выход
740      JR NZ,EXIT     ;в меню.

```



```

750      CALL NAST      ;Настройка номера дисководов.
760      EI            ;
770      CALL RDCAT     ;Чтение каталога.
780 COMAND CALL #0000   ;вызов подпрограмм SAVE или LOAD.
790 EXIT  EI           ;Выход
800      JP  MENU       ;в меню OPTIONS.
810 NEXTRK             ;Подпрограмма перехода
820      PUSH BC        ;на следующую дорожку.
830      PUSH HL        ;
840 NOFORW CALL BG93    ;Команда сдвига головки
850      DEFB #1F       ;дисковод на 1 шаг, если
860 FORWRD DEFB #5C     ;сектор находится между 0 и
870      CALL RESET     ;1 дорожками, то с метки NOFORW
880 JMP  CALL BG93      ;при старте программы будет
890      DEFB #FF       ;поставлена команда JR JMP.
900 SIDE0 DEFB #2C      ;Установка стороны 1 или 0.
910      POP HL         ;
920      POP BC         ;
930      LD  C,1        ;Установка номера сектора=1.
940      JR  DJN        ;Переход на чтение следующий сектор
950 BG93  EX  (SP),HL    ;Берем после CALL
960      LD  C,(HL)      ;в регистр C - порт TR-DOS,
970      INC HL          ;а в регистр D - число,
980      LD  D,(HL)      ;посылаемое в этот порт.
990      INC HL          ;
1000     EX  (SP),HL     ;На стеке - адрес возврата+2.
1010     PUSH HL         ;Посылаем HL на стек.
1020     LD  HL,#20B8    ;В HL - адрес в ПЗУ TR-DOS
1030     LD  B,#01      ;подпрограмма записи числа
1040     JR  TRDOS       ;в порт TR-DOS.
1050 RESET LD  HL,#0000  ;Подпрограмма очистки ВГ-93.
1060 R_LOAD LD  DE,#0000 ;Подпрограмма чтения сектора, в HL -
1070      LD  C,#7F      ;адрес начала загрузки.
1080      PUSH HL        ;Сохраняем HL.
1090      LD  HL,#3FE5   ;В HL-адрес подпрограммы чтения сектора
1100 TRDOS DI           ;
1110     EX  (SP),HL     ;Меняем содержимое стека и HL.
1120     JP  #3D2F       ;Переход в ПЗУ TR-DOS.
1130 NAST  LD  A,(SIDE1) ;Настройка
1140      OR  #10         ;загруженной
1150      LD  (#4906),A   ;подпрограммы
1160      AND #EF         ;на номер
1170      LD  (#490A),A   ;дисковод (0...3).
1180      RET            ;
1185 ;-----
1190 START             ;Подпрограмма запуска.
1200     LD  A, (#5CF4)   ;В регистр A - текущий сектор.
1210     INC A           ;увеличиваем на 1.
1220     LD  (SECTOR+1),A ;сохраняем в программе.
1230     LD  A, (#5CF6)   ;В регистре A - номер дисководов.
1240     LD  E,A         ;Сохраняем в регистре E.
1250     ADD A,#C8        ;В регистре
1260     LD  L,A         ;HL - адрес кода, определяющего
1270     LD  H,#5C        ;режим работы дисководов.
1280     LD  A,#3C       ;В регистре A - код системного

```

```

1290      ADD    A,E          ;регистра ВГ-93, для стороны 1.
1300      LD     E,A          ;
1310      LD     D,A          ;
1320  _SIDE1 BIT    1,(HL)     ;Проверка типа дисководов.
1330  SSIDE LD     A,(#5CF5)   ;В А - номер текущего трека.
1340      JR     Z,SSIDE      ;Если односторонний, то переход.
1350      AND    A            ;Проверка
1360      RRA              ;текущей стороны,
1370      JR     C,_SIDE1     ;если флаг C, то сторона 0.
1380      RES    4,D          ;В D системного регистра для стороны 0.
1390      LD     HL,#0618     ;Установка перехода в подпрограмму
1400      LD     (NOFORW),HL   ;сдвига головки.
1410      JR     SSIDE        ;
1420      RES    4,E          ;
1430      LD     (TRACK),A     ;Установка дорожки.
1440      LD     A,D          ;
1450      LD     (SIDE0),A     ;Установка текущей стороны.
1460      LD     A,E          ;
1470      LD     (SIDE1),A     ;Установка следующей стороны.
      .....                ;Старт программы.

```

Основная программа.

```

10      ORG     #4B00        ;Адрес компиляции программы.
20  WYGR EQU     #B0D6        ;WYGR - начало выгрузки.
30  CATDAT EQU   #4F00        ;CATDAT-адрес, с которого будут загружены
                               ;данные диска (дорожка 0, сектор 9).
40  BUFFER EQU   #4000        ;Начало буфера выгрузки.
50  LENGHT EQU   #0AAD        ;Длина выгрузки в DUN DARACH.
60  CAT EQU      #5000        ;Адрес загрузки каталога.
70  FNAME EQU    #AF2C        ;Начало имени файла выгрузки.
80  PRINT EQU    #AD1A        ;Адрес процедуры печати.
90  CLS EQU      #AFE5        ;Адрес процедуры очистки экрана.
100 RDSECT EQU   #0B00        ;\
110 RDS1 EQU     #000B        ;Количество секторов в выгрузке.
120 RDS2 EQU     #0B          ;/
130 LETTER EQU   #AF2F        ;Буква в имени файла выгрузки.
140 GO_TR AND    A            ;GO_TR-Подпрограмма поиска дорожки.
      ;В регистре А должна быть установлена искомая дорожка.
150 SDRIVE RRA          ;Проверка стороны.
160      LD     (TRACK),A     ;Сохранение дорожки деленной на 2.
170      LD     A,#3C        ;#3C-код 1 стороны для диска А.
180      JR     NC,SIDE1     ;Если флаг NC, то переход,
190      LD     A,#2C        ;иначе установка стороны 0.
200 SIDE1 LD     (SYSREG),A   ;Сохранение определенной стороны.
210      CALL  BG93          ;Установка стороны диска
220      DEFB   #FF          ;в порту ВГ-93.
230 SYSREG DEFB   #3C        ;Сторона дисковода.
240      CALL  BG93          ;Установка регистра дорожки.
250      DEFB   #7F          ;Адрес порта дорожки-#7F.
260 TRACK DEFB   #00        ;Номер дорожки.
270      CALL  BG93          ;
280      DEFB   #1F          ;Порт команд-состояния ВГ93.
290      DEFB   #18          ;#18=00011000-поиск дорожки.
300      CALL  RESET        ;Очистка ВГ-93.
310      RET                ;

```

```

320 CATSAV LD HL,CATDAT ;ONDISK-Процедура записи сектора на диск.
;Параметры: HL-начало записываемого блока в памяти, после CALL должен
;стоять номер сектора.
330 ONDISK LD A,#A4 ;Подпрограмма записи данных каталога.
;#A4=10100100b-запись сектора.
340 LD (S_OR_L),A ;Установка переменных записи.
350 LD DE,R_SAVE ;В DE адрес подпрограммы вывода на диск.
360 JR JMP1 ;
370 LOAD_C LD HL,CATDAT ;Подпрограммы загрузки каталога.
380 FROM_D LD A,#80 ;#80-команда чтения сектора.
390 LD (S_OR_L),A ;Установка переменных загрузки.
400 LD DE,R_LOAD ;В DE - адрес подпрограммы чтения.
410 JMP1 LD (ROUTIN+1),DE ;Установка перехода.
420 EX (SP),HL ;Обмен содержимым стека с HL.
430 LD A,(HL) ;В регистр A номер сектора.
440 INC HL ;увеличить адрес возврата на 1.
450 EX (SP),HL ;Обратный обмен HL и стека.
460 LD (SECTOR),A ;Установка сектора.
470 LOOP1 PUSH HL ;Сохранение HL.
480 CALL BG93 ;Номер сектора - в регистр.
490 DEFB #5F ;#5F-регистр сектора.
500 SECTOR DEFB #01 ;Номер сектора.
510 CALL BG93 ;Команда записи или загрузки,
520 DEFB #1F ;в зависимости от установленных
530 S_OR_L DEFB #80 ;раннее переменных.
540 POP HL ;Восстановление первоначаль-
550 PUSH HL ;ного значения HL.
560 ROUTIN CALL R_LOAD ;Исполнение SAVE или LOAD.
570 LD A,#7F ;Проверка нажатия клавиши
580 IN A,(#FE) ;SPACE, если клавиша нажата,
590 AND #01 ;то выход из подпрограммы.
600 JR Z,BREAK ;
610 LD DE,#0100 ;Проверка считанного или записанного
620 AND A ;сектора, если он
630 SBC HL,DE ;не равняется 256 байтам, то
640 POP DE ;повтор записи или считывания
650 PUSH DE ;
660 SBC HL,DE ;
670 BREAK POP HL ;
680 JR NZ,LOOP1 ;Повтор чтения/записи при ошибке.
690 RET ;
700 BG93 EX (SP),HL ;Берем после CALL BG93
710 LD C,(HL) ;в регистр C-порт TR-DOS,
720 INC HL ;в регистр D-число, которое
730 LD D,(HL) ;нужно занести в этот порт.
740 INC HL ;
750 EX (SP),HL ;Восстановить адрес возврата+2.
760 PUSH HL ;
770 LD HL,#20B8 ;В HL-адрес подпрограммы записи
780 LD B,01 ;в порты TR-DOS.
790 JR TRDOS ;
800 RESET ;
810 LD HL,#0000 ;Очистка БГ-93.
820 R_LOAD LD DE,#0000 ;Подпрограмма чтения сектора.
830 LD C,#7F ;

```

```

840      PUSH HL      ;
850      LD   HL,#3FE5 ;
860 TRDOS  DI         ;
870      EX   (SP),HL ;Меняем регистры HL со стеком.
880      JP   #3D2F    ;Вход в ПЗУ TR-DOS.
890 R_SAVE LD   DE,#0000 ;Подпрограмма вывода на диск.
900      LD   C,#7F    ;
910      PUSH HL      ;
920      LD   HL,#3FCA ;
930      EX   (SP),HL ;
940      JP   #3D2F    ;Вход в ПЗУ TRDOS.
950 RDCAT  CALL ENTNAM ;RDCAT-Подпрограмма чтения каталога
      ;и сортировки имен. Установка буквы в имени файла.
960      CALL CLS      ;Очистка экрана.
970      XOR   A       ;Установка головки на
980      CALL GO_TR     ;нулевую дорожку.
990      CALL LOAD_C    ;Загрузка каталога в память.
1000     DEFB #09      ;Сектор 09, дорожка 0.
1010     LD   A,(CATDAT+#E3) ;Проверка количества секторов на
1020     BIT   3,A      ;одной дорожке, если 8, то
1030     JR    Z,DDRIVE ;происходит смена
1040     XOR   A       ;переменных для
1050     LD   (SDRIVE),A ;одностороннего
1060     LD   A,#08     ;дисковогода.
1070     LD   (SSECT+1),A ;
1080 DDRIVE LD   HL,CAT ;Считывание имен файлов с
1090     LD   C,#08     ;диска в память с адреса #5000.
1100 LOOP2 PUSH BC    ;Счетчик на чтение первых
1110     LD   A,#09     ;8 секторов каталога.
1120     SUB   C        ;A=A-C
1130     LD   (SECT1),A ;установка текущего сектора.
1140     CALL FROM_D    ;считывание сектора.
1150 SECT1 DEFB #09   ;
1160     POP   BC       ;Восстановление BC.
1170     LD   DE,#0010  ;Проверка имен файлов.
1180     LD   B,E       ;Установка счётчика на 16 имён.
1190 LOOP3 LD   A,(HL) ;Проверка первой буквы
1200     AND   A        ;имени, если она
1210     JR    Z,C_FILE ;равна 0, то выход,
1220     ADD   HL,DE     ;иначе проверка
1230     DJNZ  LOOP3    ;следующего имени файла.
1240     DEC   C        ;Переход на считывание
1250     JR    NZ,LOOP2 ;следующего сектора.
1260 C_FILE ;Отбор файлов по длине.
1270     LD   DE,CAT    ;DE - на начало каталога.
1280     PUSH  DE       ;
1290     POP   HL       ;HL=DE
1300     XOR   A        ;Обнуление счетчика файлов.
1310     LD   (NFILES),A ;Сохранение.
1320 LOOP4 LD   BC,#0010 ;Кол-во байт в заголовке.
1330     LD   A,(HL)    ;В A - первый байт имени файла,
1340     AND   A        ;если он равен 0 (конец
1350     JR    Z,ENDCAT ;каталога), то выход из подпрограммы.
1360     CP   #01      ;Если=1 (стертый файл),
1370     JR    Z,A_TYPE ;то проверка следующего файла.

```

```

1380      PUSH HL      ;
1390      POP  IX      ;IX=HL
1400      LD   A,(IX+8) ;Тип файла: BASIC, CODE или
1410      TYPE NOP      ;другой тип. Здесь можно
1420      NOP           ;вставить XOR [ASCII CODE].
1430      LD   A,(IX+11) ;Младший байт длины. После
1440 LLEN  XOR   #AC     ;его проверки можно отсеять
1450      JR   NZ,A_TYPE ;остальные файлы.
1460      LD   A,(IX+12) ;Проверка младшего байта.
1470 HLEN  XOR   #0A     ;Проверка старшего
1480      JR   NZ,A_TYPE ;байта длины.
1490 SLEN  LD   A,RDS2    ;Проверка на количество секторов.
1500      SUB  (IX+13)    ;Если не равно переменной
1510      JR   NZ,A_TYPE ;RDS2, то это не выгрузка.
1520      LDIR           ;Перенос проверенного имени.
1530      LD   A,(NFILES) ;Увеличение счетчика
1540      INC  A          ;отобранных файлов.
1550      LD   (NFILES),A ;Сохранение счётчика.
1560 A_TYPE ADD  HL,BC    ;Проверка следующего файла.
1570      JR   LOOP4     ;
1580 ENDCAT XOR  A        ;Занесение после последнего
1590      LD   (DE),A     ;отобранного файла 0.
1600      RET            ;Выход из подпрограммы.
1610 NFILES DEFB #00     ;Количество отобранных файлов.
1620 LOAD           ;Подпрограмма загрузки положения игры.
1630      LD   A,(NFILES) ;Проверка количества выгрузок,
1640      CP   #00        ;если ни одной, то выход с
1650      JP   Z,ERR1     ;сообщением об их отсутствии.
1660      CALL FINDAF     ;Поиск имени файла, если файла
1670      JP   C,ERR2     ;нет то выход с ошибкой.
1680      LD   HL,FROM_D  ;Настройка переменных чтения.
1690      LD   (CDLORS+1),HL ;
1700      CALL SORL       ;Вызов подпрограммы загрузки.
1710      JR   Z,ENDLD    ;Выход из подпрограммы при ошибке.
1720      LD   HL,BUFFER  ;Перенос выгрузки из буфера
1730      LD   DE,WYGR     ;в программу.
1740      LD   BC,LENGHT  ;
1750      LDIR           ;
1760 ENDLD  RET          ;

1770 SORL  LD   A,(IX+15) ;Подпрограмма чтения или записи из
      ;буфера, в зависимости от переменных. IX-данные файла.
1780      LD   (FINDTR+1),A ;Поиск начальной дорожки.
1790      CALL FINDTR      ;
1800      LD   A,(IX+14)   ;В A - начальный сектор.
1810      LD   HL,BUFFER   ;В HL начало буфера.
1820      LD   BC,RDSECT   ;BC количество секторов в буфере.
1830 LOOP5 PUSH  BC       ;Сохранение BC.
1840      INC  A           ;Увеличение номера сектора.
1850      PUSH HL          ;
1860      PUSH AF          ;
1870      LD   (SECT2),A   ;Установка сектора.

```

```

1880 CDLORS CALL FROM_D      ;Чтение/запись сектора.
1890 SECT2  DEFB #00         ;
1900        POP AF           ;
1910        POP HL           ;
1920        INC H             ;Увеличение адреса на 256.
1930        CP #10           ;проверка сектора, если не
1940        JR NZ,LAB        ;16, то переход на дальнейшее
1950        PUSH HL           ;чтение/запись, иначе
1960        LD A,(FINDTR+1)   ;сдвиг головки
1970        INC A             ;вперед на 1 трек.
1980        LD (FINDTR+1),A   ;
1990        CALL GO_TR        ;
2000        POP HL           ;
2010        XOR A             ;Обнуление сектора.
2020 LAB      ;
2030        POP BC           ;Восстановление BC.
2040        DJNZ LOOP5       ;Дальнейшее чтение.
2050        RET              ;
2060 FINDTR LD A,01          ;Поиск дорожки.
2070        CALL GO_TR        ;
2080        RET              ;

2090 FINDAF LD HL,CAT        ;Подпрограмма поиска имени файла среди
;отобранных по длине. HL-начало имени первого файла.
2100 L1      LD A,(HL)        ;Проверка конца каталога.
2110        AND A             ;Если первый байт имени
2120        JR Z,END          ;файла=0, то выход.
2130        PUSH HL           ;
2140        LD DE,FNAME       ;В DE - имя искомого файла.
2150        LD BC,#0800       ;Количество байт в имени 8.
2160 L2      LD A,(DE)        ;Провер.первого байта имени.
2170        INC DE            ;DE=DE+1
2180        AND A             ;проверка A на ноль.Если да,
2190        JR Z,L3           ;то проверка следующего имени.
2200        CP (HL)           ;сравнение байтов имен.
2210        INC HL            ;увеличение адреса следующей
2220        JR NZ,L3          ;буквы. Если не равны, то
;проверка следующего имени.
2230        DJNZ L2           ;иначе проверка следующей буквы.
2240        POP IX            ;IX-адрес найденного имени.
2250        SCF               ;выключение флага переноса.
2260        CCF               ;
2270        RET              ;
2280 L3      POP HL           ;
2290        LD DE,#0010       ;Адрес следующего имени:
2300        ADD HL,DE         ;равен HL=HL+16
2310        JR L1             ;Проверка, если файл
2320 END     SCF              ;не найден, то включение
2330        RET              ;флага переноса и выход.
2340 SAVE    ;Подпрограмма записи положения игры.
2350        LD HL,WYGR        ;Перенос выгрузки из
2360        LD DE,BUFFER       ;программы в буфер.
2370        LD BC,LENGHT      ;

```

```

2380      LDIR      ;
2390      CALL FINDAF      ;Поиск имени.
2400      JP       C,NEWF   ;Если файл не найден,
                           ;создание нового файла.
2410 SVE      LD      HL,ONDISK      ;Иначе перезапись старого.
2420      LD      (CDLORS+1),HL      ;Настройка переменных.
2430      LD      A,(IX+15)      ;Дорожка начала файла.
2440      LD      (FINDTR+1),A      ;
2450      CALL FINDTR      ;Поиск дорожки.
2460      LD      A,(IX+14)      ;Сектор начала файла.
2470      LD      HL,BUFFER      ;В HL - начало буфера.
2480      LD      BC,RDSECT      ;В BC - количество секторов.
2490      CALL SORL      ;Запись блока секторов.
2500      RET      ;
2510 NEWF      CALL BG93      ;Создание нового файла.
2520      DEFB #1F      ;Команда восстановления.
2530      DEFB #0C      ;#1F-регистр команд.
                           ;#0C=00001100-восстановление.
2540      CALL RESET      ;RESET BG93.
2550      CALL LOAD_C      ;Загрузка данных каталога.
2560      DEFB #09      ;
2570      LD      HL,(CATDAT+#E5) ;проверка оставшихся сект.
2580      LD      DE,RDS1      ;на диске, если меньше RDS1,
2590      SBC      HL,DE      ;то выход с сообщением об
2600      JP       C,ERR3      ;ошибке (no space on disk!).
2610      LD      (CATDAT+#E5),HL ;
2620      LD      HL,CATDAT+#E4 ;Проверка количества файлов на
2630      INC      (HL)      ;диске, если 128, то выход с
2640      BIT      7,(HL)      ;сообщением об ошибке.
2650      JP       NZ,ERR3      ;
2660      LD      HL,(CATDAT+#E1) ;Изменение данных последнего
2670      LD      E,L      ;трека/сектора.
2680      LD      D,H      ;
2690      PUSH      DE      ;
2700      LD      A,RDS2      ;В A количество секторов.
2710 SSECT      LD      B,#10      ;В - количество секторов в треке.
2720      ADD      A,E      ;
2730 L5         CCF      ;
2740      SUB      B      ;
2750      JR       C,L6      ;
2760      INC      D      ;
2770      JR       L5      ;
2780 L6         ADD      A,B      ;
2790      LD      E,A      ;
2800      EX      DE,HL      ;
2810      LD      (CATDAT+#E1),HL ;Сохранение последнего трека/сектора.
2820      CALL CATSAV      ;Запись измененных данных.
2830      DEFB #09      ;
2840      LD      A,(CATDAT+#E4) ;A - количество файлов на диске.
2850      DEC      A      ;Первоначальное количество файлов.
2860      RRCA      ;В результате этих манипуляций
2870      RRCA      ;получаем в A номер сектора
2880      RRCA      ;(от 1 до 8), в котором
2890      RRCA      ;находится имя последнего
2900      PUSH      AF      ;файла на диске. На стеке

```



```

2910      AND    #07          ;имеем номер файла в
2920      INC    A            ;секторе (1-16).
2930      LD     (SECT3),A     ;Ввод переменных в программу.
2940      LD     (SECT4),A     ;
2950      CALL   LOAD_C        ;Читаем сектор.
2960 SECT3  DEFB   #00        ;
2970      POP    AF            ;
2980      AND    #F0          ;Создаём в памяти имя файла.
2990      LD     E,A           ;
3000      LD     D,#00        ;
3010      ADD    HL,DE         ;
3020      PUSH   HL            ;
3030      POP    IX            ;
3040      EX     DE,HL         ;Копируем имя.
3050      LD     HL,FNAME      ;
3060      LD     BC,#000E      ;
3070      LDIR             ;
3080      EX     DE,HL         ;
3090      POP    DE            ;Вводим начальный
3100      LD     (HL),E        ;сектор
3110      INC    HL            ;и
3120      LD     (HL),D        ;трек.
3130      CALL   CATSAV        ;Записываем на диск
3140 SECT4  DEFB   #00        ;измененный сектор.
3150      JP     SVE           ;На запись данных файла.
3160 ERR1   CALL   PRINT      ;Ошибка 1.
3170      DEFB   #16          ;Если на диске нет
3180      DEFB   #10          ;выгрузок игры,
3190      DEFB   #03          ;то выдается
3200      DEFB   #10          ;это сообщение.
3210      DEFB   #07          ;
3220      DEFB   #11          ;
3230      DEFB   #00          ;
3240      DEFM   "no file(s) of restore game!%"
3250      CALL   ANYKEY        ;
3260      JP     PAUSE         ;
3270 ERR2   CALL   PRINT      ;Ошибка 2.
3280      DEFB   #16          ;Файл не найден.
3290      DEFB   #10          ;
3300      DEFB   #09          ;
3310      DEFB   #10          ;
3320      DEFB   #07          ;
3330      DEFB   #11          ;
3340      DEFB   #00          ;
3350      DEFM   "file absent!%"
3360      CALL   ANYKEY        ;
3370      JP     PAUSE         ;
3380 ERR3   CALL   PRINT      ;Ошибка 3.
3390      DEFB   #16          ;Нет места на диске.
3400      DEFB   #10          ;
3410      DEFB   #07          ;
3420      DEFB   #10          ;
3430      DEFB   #07          ;
3440      DEFB   #11          ;
3450      DEFB   #00          ;

```

```

3460      DEFM "no space on disk!%"
3470      CALL ANYKEY      ;
3480      JP    PAUSE      ;
3490 ANYKEY
3500      CALL PRINT      ;
3510      DEFB #16        ;
3520      DEFB #12        ;
3530      DEFB #09        ;
3540      DEFB #10        ;
3550      DEFB #07        ;
3560      DEFB #11        ;
3570      DEFB #00        ;
3580      DEFB #13        ;
3590      DEFB #01        ;
3600      DEFM "press any key%"
3610      RET             ;
3620 PAUSE
3630 NEXTP XOR    A        ;Подпрограмма ожидания нажатия
3640      IN     A, (#FE)   ;любой клавиши.
3650      CPL                      ;
3660      AND    #1F        ;
3670      JR     Z,NEXTP    ;
3680      RET             ;
3690 ENTNAM CALL PRINT      ;Подпрограмма ввода версии выгрузки.
3700      DEFB #16        ;
3710      DEFB #10        ;
3720      DEFB #05        ;
3730      DEFB #10        ;
3740      DEFB #06        ;
3750      DEFB #11        ;
3760      DEFB #00        ;
3770      DEFM "enter version letter%"
3780      XOR    A        ;
3790      LD     (#5C08),A  ;
3800 LL      LD     A, (#5C08) ;
3810      AND    #DF        ;
3820      CP     #41        ;
3830      JR     C,LL       ;
3840      CP     #5B        ;
3850      JR     NC,LL      ;
3860      LD     (LETTER),A ;
3870      RET             ;

```

ЧИТАТЕЛЬ-ЧИТАТЕЛЮ

Краткое описание **LASER GENIUS ASSEMBLER/MONITOR**

© Илья Фомин, С-Петербург, 1995.

Часть 0. Вместо предисловия.

Пара слов об этом описании. Прежде всего, сообщу, что его мне любезно предоставил Владимир Ларьков).

Заинтересовавшись Laser Genius ассемблером я долго тыкал пальцем в грудь каждому встречному с целью выудить у счастливица описание данного ассемблера. Долго никто не признавался, наконец, попали мне в руки неизвестно кем созданные бумажки, текст которых я и набрал (в текстовом редакторе iS-EDIT к слову). Так что за полную достоверность не отвечаю, все проверяйте сами, но хоть список команд и их параметры появились. В известной мне версии Laser Genius-a (адаптация MOA для TR-DOS) все связанное с микродрайвом следует понимать как дисковод (вместо MDRV пишите DISK).

Ну, а если кто имеет более полное описание, сей программы, то не жмитесь, поделитесь с остальными.

Часть 1. Ассемблер.

ЗАГРУЗКА АССЕМБЛЕРА

Для загрузки инсталлятора ассемблера необходимо ввести команду: LOAD "GENASM". GENASM - загрузчик на бейсике. После запуска он выдает меню опций, по которым Вы можете выбрать рабочую структуру ассемблера. К стандартным возможностям можно добавить специальные инструментальные команды (см. 2. "Команды ассемблера") и/или ХЭШ-расширения (см. 6. "ХЭШ-расширения, язык программирования Феникс."). Так же есть возможность выгрузки ассемблера на микродрайв. Инсталлятор так же позволяет установить цвета фона, тона, значения кода символа перевода строки для работы с принтером (установление этого значения в ноль позволяет избежать проблем двойного перевода строки на некоторых моделях принтеров), а также позволяет выбрать тип подключения принтера (последовательный или параллельный).

РЕДАКТОР

Редактор делит экран на текстовые блоки, которые мы далее будем называть предложениями. Любой текст, набираемый на экране, будет введен в текущее предложение, указываемое курсором. Когда экран очищен, каждая его строка представляет собой одно предложение. Тем не менее, предложение может быть длиннее, чем одна строка. Если длина напечатанного текста превысит длину экранной строки, редактор поставит в конце строки знак продолжения и переведет курсор на следующую строку. Такие строки также объединяются в одно предложение.

Предложения объединяются в сегменты. Начало сегмента однозначно определяется номером стоящего в начале первого предложения сегмента (как в программной строке BASIC-программы). Редактор различает мнемокоманды ассемблера (в т.ч. операторы Феникса и опции ассемблирования), которые необходимо включить в текст программы, и команды, требующие немедленного выполнения без включения в ассемблерный текст (см. ниже).

ПРИЛОЖЕНИЯ

1. СПИСОК КОМАНД АССЕМБЛЕРА.

После команды могут следовать параметры, они обозначены выражениями в $\langle \dots \rangle$.

ASSEM - очищает таблицу имен и ассемблирует текущий текст.

ASSEMC - сохраняет текущую таблицу имен и ассемблирует текущий текст.

ASSEMBL - выборочно ассемблирует процедуры из библиотеки.

BASE $\langle 2, 8, 10, 16 \rangle$ - устанавливает умолчание базы в PRINT и UPRINT.

CAT - каталог с ленты/микродрайва.

CLEAR - очищает текущую таблицу имён.

CLS - очищает экранный буфер и физический экран.

CODE "**<имя>**" **<выр1>**, **<выр2>** - записывает область памяти от адреса **<выр1>** до адреса **<выр2>** в двоичный файл с именем "**<имя>**".

COPY **<стр1>**, **<стр2>** - копирует **<стр1>** в **<стр2>**.

DELETE - стирает весь текущий текст.

DELETE **<спец. стр. >** - стирает группу строк из текущего текста.

ERA "**<имя>**" - удаляет файл(ы) (микродрайв).

EXECUTE **<выр>**, **<пар>** - вызывает коды с адреса **<выр>**, посылая параметры (список параметров: **<пар>**=**<пар1>**, **<пар2>**, ..., **<парN>**) таким образом, что **IX** содержит начало списка, причём параметры записываются в обратном порядке.

1-й параметр $IX+2*(n-i)$ (мл. байт)

$1+IX+2*(n-i)$ (ст. байт)

для числового выражения; или дескриптор строки длиной три байта:

длина строки $IX+2*(n-i)$

адрес (мл) $1+IX+2*(n-i)$

(ст) $2+IX+2*(n-i)$

Замечание: т.к. **EXECUTE** вызывает коды командой **CALL** микропроцессора Z80, эти коды должны заканчиваться **RET** для нормального возврата в Genius.

EXPORT "**<имя>**" - записывает текущую таблицу имён на ленту/микродрайв.

EXIT - **RET** к программе, вызвавшей ассемблер или к ассемблеру из среды **HOUSEWORK**.

Примечание: Если при работе с Genius-ом Вы воспользовались **EXIT** или Вас выкинуло из ассемблера в **BASIC** и Вы желаете вернуться в Genius без перезагрузки, сохранив текст программы, наберите: **RANDOMIZE USR 65533**.

FIND "**<имя>**" , **<спец. стр>** - найти строку "**<имя>**".

FORM - новая страница на принтер.

HOUSEWORK - переход к среде работы с внешними носителями, поддерживает команды:

CAT	ERA
TAPE	MDRV
TAPE,IN	MDRV,IN
TAPE,OUT	MDRV,OUT

а также команду

COPY "**<имя1>**", "**<имя2>**", которая копирует файл "**<имя2>**" в новый файл "**<имя1>**".

IMPORT "**<имя>**" - присоединяет предыдущую экспортированную таблицу символов к текущей.

LENGTH - устанавливает страницу на принтере длиной 65536 строк.

LENGTH **<дес. число>** - установка размера страницы на принтере длиной **<дес.число>** строк.

LIST - распечатывает весь исходный текст.

LIST <спец. стр> - распечатывает строки текста под номерами <спец.стр>

LMISSING - распечатать на принтере имена, на которые была ссылка, но которые не были определены. (Длина поля 16).

LMISSING <дес. число> - см. выше, но длина поля <дес. число>.

LOAD "<имя>" - загрузить исходный текст с ленты/микродрайва с именем "<имя>".

LOAD "<имя>" <спец. стр> - то же, но загружаются строки <спец.стр> из файла "<имя>".

LOAD ASCII "<имя>" - загрузка "чистого" ASCII файла с именем "<имя>".

LOAD ASCII "<имя>" <опции> - загрузить файл-источник с именем "<имя>", тип которого показывает <опции>. Строки нумеруются с 0 по 10; опции:

1 - файл "чисто" ASCII (не содержит нумерации строк).

2 - строки файла заканчиваются на CR (0Dh).

4 - файл не содержит колонок после меток.

8 - пропускает первые 4 байта файла.

LOAD ASII "<имя>",<опции> <дес. число> - то же, но регулируемая нумерация строк.

LLIST - вывод всего текста на принтер.

LLIST <спец. стр> - вывод на принтер группы строк.

LTABLE - вывод на принтер текущей таблицы символов (Длина поля 16), отсортированной в ASCII порядке.

LTABLE <дес. число> - то же, но длина поля <дес.число>.

LTABLEN - то же, что и LTABLE, но в числовом порядке.

LTABLEN <дес.число> - то же, но с <дес.число>.

LUNUSED - вывод на принтер определенных, но неиспользованных символов (длина поля 16).

LUNUSED <дес. число> - то же, но длина поля <дес.число>.

MARGIN - установка левой границы в ноль.

MARGIN <дес. число> - установка левой границы.

MDRV - переключает I/O ассемблера на микродрайв (по умолчанию #1).

MDRV <дес. число> - выбор микродрайва.

MDRV. IN - переключить ввод ассемблера с микродрайва.

MDRV. OUT - переключить вывод ассемблера на микродрайв.

MISSING - вывод на экран имен, на которые есть ссылка в программе, и которые не определены (Длина поля 16).

MISSING <дес. число> - то же, но длина поля <дес.число>.

MOVE <стр1>,<стр2> - перемещение группы строк.

NEXT - продолжение поиска/замены после ESC.

PRINT <выражен> - вывод значения выражения <выражен> в текущем базисе со знаком.

PRINT <выражен>,<база> - вывод значения выражения <выражен> в базисе <база> со знаком.

REDUCE - удаление всех имен, которые не были специфицированы как CARGO во время предыдущего ассемблирования.

RENUM <нов. стр>,<шаг>,<от стр> - перенумерация параграфов. REPLACE "<имя1>",<имя2>",<спец.стр> - контекстуальная замена.

SAVE "<имя>" - запись исходного текста на ленту/микродрайв в файл "<имя>".

SAVE "<имя>",<спец. стр> - то же, но фрагмента текста.

SET SPACE <дес. число> - установка памяти редактора. (сначала около 10 кб).

STATS - информация распределения памяти.

TABLE - вывод на экран текущей таблицы символов (Длина записи 16) В порядке ASCII.

TABLE <дес.число> - то же, но длина записи <дес.число>.

TABLEN - то же, что и **TABLE**, но в числовом порядке.

TABLEN <дес.число> - то же, но длина записи <дес.число>.

TAPE - переключает I/O ассемблера на ленту.

TAPE.IN - переключает ввод ассемблера с ленты.

TAPE.OUT - переключает вывод ассемблера на ленту.

UNUSED - вывод на экран определенных, но неиспользованных символов (Длина поля 16).

UNUSED <дес.число> - то же, но длина поля <дес.число>.

UPRINT <выражен> - вывод значения выражения <выражен> в текущем базисе без знака.

UPRINT <выражен>, <база> - вывод значения выражения <выражен> в базисе <база> без знака.

VERIFY "<имя>" - сравнить текущий текст с файлом "<имя>".

VERIFY "<имя>", <спец.стр> - то же, но группу строк <спец.стр>.

WIDTH - устанавливает ширину страницы принтера 65536.

WIDTH <дес.число> - устанавливает ширину страницы принтера в <дес.число>.

Замечания:

1. Команды **SAVE**, **LOAD**, **EXPORT**, **IMPORT**, **CAT**, **COPY** (**HOUSEWORK**), **VERIFY**, а так же некоторые опции ассемблера работают с файлами, которые обычный загрузчик ZX-Spectrum-a не распознает.

2. Команда **CODE** генерирует байтовый файл, загружаемый обычным загрузчиком ZX-Spectrum. Служит для создания небольших кодовых файлов, объектный код которых помещается в памяти вместе с **Genius**-ом.

3. Для создания больших кодовых файлов используют опции ассемблирования, генерирующие объектные файлы в формате **Genius**-а, и с помощью утилиты **TRANSFER** их конвертируют в Спектрумовские.

4. Ваша библиотека может содержать ограниченное число меток на которые Вы будете ссылаться в Ваших программах. **EXPORT**, как правило, записывает все имена, в т.ч. и те, которые не нужны. В **Genius**-е предусмотрена возможность записи только желаемых имен. Для этого имена, которые надо записать, помечаются псевдо операцией **CARGO**: <label>: **CARGO** и после ассемблирования таблица символов обрабатывается командой **REDUCE**. После этого можно записать желаемую таблицу символов на внешний носитель командой **EXPORT**.

2. СПИСОК ОПЦИЙ АССЕМБЛИРОВАНИЯ.

В скобках указаны опции, устанавливаемые по умолчанию.

***screen on / *screen off** - разрешает/запрещает какой-либо вывод на экран (в т.ч. сообщения об ошибках). (on)

***printer on / *printer off** - разрешает/запрещает вывод на принтер (сообщения об ошибках и сообщения конца прохода ассемблера). (off)

***list on / *list off** - включает/выключает вывод листинга на экран. (on)

***llist on / *llist off** - то же для принтера. (off)

***form** - "новая страница" на принтер.

***title** "<имя>", <выраж> - устанавливает заголовок страницы для принтера. Если присутствует <выраж>, то устанавливает текущий номер страницы.

***maclist on / *maclist off** - разрешает/запрещает выводить листинг генерации кодов макроопределений. (off)

***include** "<имя>" - включение в исходный текст с внешнего носителя строк, которые необходимо ассемблировать.

***openout "<имя>"** - создание результата ассемблирования - объектного кода - на внешнем носителе. Адресация согласно ORG PUT - игнорируется.

***closeout** - прекращение генерации на внешний носитель.

***prompts on / *prompts off** - разрешение/запрещение запросов при работе с микродрайвом. (on)

***count on / *count off** - выводит на экран в случае, когда выключен листинг номера ассемблирующихся в данный момент строк. (off)

***report on / *report off** - включает/выключает ожидание нажатия клавиши после вывода сообщения об ошибке во время ассемблирования. (on)

***code on / *code off** - разрешает/запрещает генерацию кода во время ассемблирования. (on)

***print "<строка>"** - выводит на экран и принтер сообщение <строка>.

***pause** - останавливает ассемблирование и ждёт нажатия клавиши.

***while <выражен>**

***endw**

- цикл с пред-условием: делать пока <выражен> не равно 0.

***repeat**

***until <выражен>**

- цикл с пост-условием: делать пока <выражен> равно 0.

3. ПСЕВДООПЕРАЦИИ АССЕМБЛЕРА.

DB <список определения байтов> - DEFB, DEFM - инициализирует байты памяти во время ассемблирования.

F.e. (for example):

DB 1,2,3,0Fh,@77 => 01 02 03 0F 3F

DB "Hello!\13" => 20 0D

DW <список определения слов> - DEFW - инициализирует слова (пары байтов) в памяти во время ассемблирования.

F.e.:

DW #1234,0EFDAH => 34 12 DA EF

DS <выражение> - DEFS - освобождает место в объектном коде длиной <выражение> байтов.

<label>: EQU <выражение> - используется для определения и инициализации нового имени <label> значением <выражение>.

<label>: DL <выражение> - DEFL - то же, что и EQU, но может использоваться для переопределения слова <label>.

F.e.:

```
cnt :EQU      1
    *WHILE    cnt <= 10
    DB        cnt * cnt
cnt :DL        cnt + 1
    *ENDW
```

данный пример генерирует таблицу квадратов чисел от 1 до 10 в памяти.

...

COND <выражение>

...

ELSE

...

ENDC

- ассемблер вычисляет <выражение> и, если оно не ноль, ассемблирование продолжается с COND до ELSE а потом пропускается до ENDC. В противном случае пропускается до ELSE и ассемблируется до ENDC. После ENDC ассемблирование происходит в обычном порядке.

ORG <выражение> - устанавливает счётчик размещения значением <выражение> (относительно счётчика размещения генерируются адреса в ассемблеровской программе). Замечание: объектный код, ассемблируемый в память НЕ будет записан по адресу, указанному в ORG, а, следовательно, не может быть выполнен без псевдооперации PUT (см. ниже).

PUT <выражение> - устанавливает адрес размещения объектного кода в памяти, которое будет происходить во время ассемблирования.

<label>: MACRO <список параметров> - макроопределение. Определяет имя <label> как имя макроса, который описывается ниже слова MACRO до слова ENDM. <список параметров> - список имён через запятую, перед каждым именем обязательно ставится символ "\".

Е.е.: макрос обмена между регистровыми парами через стек:

```
exg: MACRO    \p1, \p2
      PUSH    \p1
      PUSH    \p2
      POP     \p1
      POP     \p2
      ENDM
```

Вызов осуществляется следующим образом:

```
\exg BC,DE => PUSH BC : PUSH DE : POP BC : POP DE
```

т.е. перед именем макроса также ставят "\". (NB принцип макроопределения заключается в подстановке готовых строк в исходный текст ассемблером во время трансляции, т.е. макрос ни в коем случае не является подпрограммой!).

4. ЯЗЫК ПРОГРАММИРОВАНИЯ PHOENIX.

Вычисление выражений

#DSE <компилирующее выражение> - "Определить выражение со знаком".

#DUE <компилирующее выражение> - "Определить выражение без знака".

Эти псевдооперации во время ассемблирования преобразуют <компилирующее выражение> в выполняемые коды. Вычисления во время выполнения программы происходят с сохранением временных результатов в стеке. Арифметические операции определены в специальной библиотеке Genius-a, и доступ к ним осуществляется командами CALL из программы (см. #LIB).

Присвоение

Присвоение переменным значения происходит во время выполнения программы. Genius сам во время трансляции заботится о резервировании памяти под переменные.

Е.е.:

```
#DUE state=1          => 1->(state)
#DSE x=y=0            => 0->(x) 0->(y)
#DSE y=(z=x*x)+x-20    => (x)*(x)->(z)+(x)-20->(y)
```

где x, y и z - переменные, а (x),(y) и (z) - значения ячеек памяти, соответствующих этим переменным.

Массивы

Ниже будет показано, как определять переменные для трансляции. Однако, создавая переменные, Вы можете использовать их как базу для индексированных переменных. Таким образом, это позволяет создавать одномерные массивы - вектора. Т.е., определяя блок памяти как переменную, можно обращаться к ней как к массиву.

Ф.е.: `xcoord: #DS INT, 10`

определяет место для 10 целых (20 байтов). Первое целое - переменная с именем `xcoord`, доступ к остальным осуществляется следующим образом:

`xcoord [<компилируемое выражение>]`

где `<компилируемое выражение>` может принимать значения от 0 до 9, что соответствует с `xcoord[0]` (или `xcoord`) по `xcoord[9]`.

Ф.е.: (присвоение значения)

`#DSE xcoord[2] = 100`

Определение переменных

Команда определения переменной без инициализации:

`<label>: #DS <тип>, <ассемблируемое выражение>`

где: `<label>` - имя определяемой переменной;

`<тип>` - тип переменной;

INT - целый

CHAR - символьный

PINT - поинтер целой

PCHAR - поинтер символьной

`<ассемблируемое выражение>` - значение является количеством переменных типа `<тип>`.

Команда определения переменной с инициализацией:

`<label>: #DI <тип>, <список ассемблируемых выражений>`

Здесь переменных типа `<тип>` будет столько, сколько указано элементов в `<список ассемблируемых выражений>`, причем каждой переменной будет присваиваться значение соответствующего элемента списка.

Функции

В Phoenix-е возможно определение функций так, как это происходит в языках высокого уровня. Обращение к функции `<имя>` имеет вид:

`<имя> (<список параметров>)` - если есть параметры

`<имя> ()` - если нет параметров.

Определение

`<label>: #FNC <тип>` - объявляет тип функции `<label>`, выполняемые операции этой функции должны находиться между `#BEGIN` и `#END`.

`<label>: #PRM <тип>` - объявляет параметр `<label>` типа `<тип>`. Количество объявлений `#PRM` должно совпадать с количеством элементов в списке вызывающей команды. Так же должен совпадать порядок следования. Параметров может быть до 60. `#PRM` должны находиться после `#FNC` и до `#BEGIN`.

`#BEGIN` - начало выполняемых команд функции.

`#END` - конец выполняемых команд функции и возврат.

#RETURN - возврат из функции. (Перед #END не ставится).

Ф.е.: функция $\text{sumsq}(x,y)=x*x+y*y$ - сумма квадратов. `sumsq : #FNC INT`

```
x      : #PRM      INT
y      : #PRM      INT
      #BEGIN
      #DUE      x*x + y*y
      #END
      ...
      #DUE      z= sumsq (3,4)
      ...
```

Вернет в HL значение 25.

ВНИМАНИЕ: Совместимость Phoenix-а с ассемблером достигается за счёт работы со стеком. НО результаты вычислений посылаются в регистровую пару HL (в т.ч. при вызове пользовательских функций). Об этом ниже, в "Компиляции выражений".

Поинтерные переменные

Поинтер - адрес записанной в памяти переменной. Адрес переменной находится добавлением префикса "&" к имени переменной. Доступ к данным, на которые указывает поинтер происходит с использованием оператора "*".

Ф.е.:

```
...
begin      :#DI      CHAR.  "There was no possibility of...\0"
;          определили и инициировали строку
rewrite    :#DS      CHAR,  80
;          зарезервировали место
p_str      :#DS      PCHAR,  1
p_spc      :#DS      PCHAR,  1
;          определили поинтерные переменные
;          далее p_str будет указывать на begin, а
;          а p_spc будет указывать на rewrite.
           #DUE      p_str = &begin
           #DUE      p_spc = &rewrite
;          присвоили поинтерам указатели на начала строк
;
;          А теперь пример работы
;
           #WHILE * p_spc !=0
           #DUE   * p_spc = * p_str
           #DUE   p_spc = p_spc + 1
           #DUE   p_str = p_str + 1
           #ENDW
           #DUE   * p_spc = 0
;          копия строки begin в rewrite
...

```

Разобравшись в данном примере, нетрудно понять принцип поинтерных переменных.

Компиляция выражений

Покажем на примере:

```

x      :#DS    INT,    1
x      :#DS    INT,    1
      #DUE x = y = 0 => LD HL, 0
                        LD (<Addr, x>), HL
                        LD (<Addr, y>), HL
...
      #DUE x + y      => LD HL, (<Addr, x>)
                        PUSH HL
                        LD HL, (<Addr, y>)
                        PUSH HL
                        CALL <Genius Library>
                        POP HL

```

Арифметические процедуры могут диагностировать некоторые ошибки, используя для отображения информации об ошибке ячейку памяти, резервируемую командой #STACK (см. ниже). Эта ячейка находится по адресу (IX-1). В случае ошибки в этой ячейке устанавливаются соответствующие биты:

- переполнение bit 0 (соответствует C флагу Z80)
- деление на 0 bit 2 (соответствует P/V флагу Z80)

результат операции остается в HL, так что при желании его можно принять или отвергнуть.

Условный оператор

#IF <компилируемое выражение> - Помечает начало кодов, которые необходимо выполнить в случае, если <компилируемое выражение> ИСТИНА (и не ноль). Выполнение происходит до #ELSE. В противном случае от #ELSE до #ENDIF. Далее выполнение продолжается после #ENDIF.

Ф.е.: (функция возвращает максимальное значение)

```

...
max   :#FNC    INT
x     :#PRM    INT
y     :#PRM    INT
      #BEGIN
      #IF      x > y
      #DSE     x
      #ELSE
      #DSE     y
      #ENDIF
      #END
...

```

Циклы

С пред-условием:

#WHILE <компилируемое выражение> - начало цикла. Выполнение продолжается пока <компилируемое выражение> истинно (не ноль).

#ENDW - конец цикла.

С пост-условием:

#REPEAT - начало цикла.

#UNTIL <компилируемое выражение> - конец цикла. Выполнение продолжается пока <компилируемое выражение> ложно (ноль).

Управление стеком

#STACK <ассемблируемое выражение> - псевдооперация генерирует коды, устанавливающие IX значением SP. Можно также задать новый указатель стека, он будет равен <ассемблируемое выражение>. (LD SP, word) Эта команда необходима для определения стека Phoenix-a, самое лучшее место для ее размещения - начало ассемблируемого текста (первый выполняемый оператор).

Замечание: Эта операция генерирует DEC SP для резервирования байта, используемого как флаговое пространство Phoenix-a, поэтому возврат из процедуры должен выглядеть так:

```
LD SP, IX      или      INC SP
RET            RET
```

Псевдооперация **#LIB**.

#LIB - должна быть размещена в конце ассемблерного текста. Перемещает Genius Arithmetic Library в конец выполняемого кода, т.е. создает законченный продукт. Genius Arithmetic Library - библиотека арифметических операций для Phoenix-a.

5. ОПЕРАЦИИ (ЛОГИКА И АРИФМЕТИКА).

Унарные операции (префиксы):

- минус
- ! логическое НЕ
- ^ побитное инвертирование
- * поинтерная ссылка
- & адрес переменной
- ++ инкрементирует переменную и возвращает новое значение
- декрементирует переменную и возвращает новое значение

Бинарные операции:

- + сложение
- вычитание
- * умножение
- / деление
- % остаток

Сдвиги: (x << n - x влево на n)

- << сдвиг влево
- >> сдвиг вправо
- @< сдвиг циклический влево
- @> сдвиг циклический вправо

Сравнение: (Устанавливает 1 или 0)

- > больше
- < меньше
- >= больше либо равно
- <= меньше либо равно

Логические операции:

&& логическое И

|| логическое ИЛИ

Побитные логические операции:

& побитное И

| побитное ИЛИ

^ побитное ИСКЛЮЧАЮЩЕЕ ИЛИ

Операции суффиксы:

++ инкрементирует переменную и возвращает старое значение

-- декрементирует переменную и возвращает старое значение.

6. КЛАВИАТУРА.

CS+7 - курсор вверх

CS+6 - курсор вниз

CS+5 - курсор влево

CS+8 - курсор вправо

CS+4 - табулятор назад

SS+E - табулятор вперед

SS+0 - стереть символ под курсором

CS+0 - забой

CS+3 - стирание предложения

SS+F - стирание до конца предложения

CS+1 - режим вставки/замены

SS+I - вставка предложения

SS+W - вставка одного пробела

SS+G - очистка экрана

CS+2 - смена регистра

SS+Y - "["

SS+U - "]"

SS+S - "|"

SS+D - "\"

SS+A - ESC клавиша

Часть 2: Монитор-отладчик.

1. РАЗМЕЩЕНИЕ МОНИТОРА В ПАМЯТИ (НИЖНЯЯ И ВЕРХНЯЯ ВЕРСИИ).

	начало	конец	конец	вход в	вход в
	анализатора		монитора	монитор	анал/монит
Нижняя:	25000	40589	37924	25000	25003
Верхняя:	49946	52610	65535	52611	52614

2. КОМАНДЫ МОНИТОРА.

<рег8>=<байт> - запись байта <байт> в регистр <рег8>, где <рег8> может быть: F, A, B, C, D, E, H или L.

<reg16>=<слово> - запись слова <слово> в регистровую пару <reg16>, где <reg16> может быть: AF, BC, DE, HL, IX, IY, PC или SP.

EXX - выполняет эквивалент команды Z80 "EXX" для копии значений регистров на экране, происходит обмен BC, DE, HL на альтернативные BC', DE', HL' (только на экране - без физического обмена).

EX AF - то же, что и EXX, но по команде Z80 "EX AF, AF".

MEM = <адрес> - устанавливает указатель памяти монитора на адрес <адрес> и обновляет дамп памяти на экране.

DATA <список байтов>.<список байтов> - размещает в памяти байты <список байтов> начиная с адреса MEM.

EXIT - выход из монитора в BASIC. Уничтожает всю пользовательскую информацию в т.ч. программы анализатора.

FILL <нач. адр>,<кон. адр>,<байт> - заполняет область памяти от <нач.адр> до <кон.адр> байтом <байт>.

DUMP [<нач. адр>,<кон. адр>] - выводит на полный экран дамп памяти от <нач.адр> до <кон.адр>.

LDUMP [<нач. адр>,<кон. адр>] - то же, но на принтер.

MOVE <нач. адр1>,<кон. адр1>,<нач. адр2> - переслать блок памяти от <нач.адр1> до <кон.адр2> в область памяти начиная с <нач.адр2>.

CHECK <нач. адр1>,<кон. адр1>,<нач. адр2> - сравнить блок памяти от <нач.адр1> до <кон.адр2> и область памяти начиная с <нач.адр2>.

SEARCH <нач. адр>,<кон. адр>,<список байтов> - найти байты <список байтов> в области памяти от <нач.адр> до <кон.адр>.

NEXT - продолжить поиск после обнаружения искомых байтов.

JUMP [<адрес>] - выполнение инструкций Z80 с нормальной скоростью, начиная с адреса <адрес>. Если <адрес> отсутствует, то с адреса, находящегося в PC. Замечание: Единственная возможность вернуться из такого выполнения в монитор - это точка прерывания (см. ниже). Т.к. эта команда аналог команды JMP процессора Z80.

? <слово> - распечатывает значение <слово> в различных системах исчисления.

PUSH <слово> - значение <слово> кладется в стек и SP декрементируется на 2 (нормальная эмуляция команды PUSH процессора Z80).

POP <слово> - значение SP инкрементируется на 2 как в команде POP процессора Z80.

OPTION <номер опции>,<флаг> - позволяет включить/выключить (используя <флаг> ON - включить; OFF - выключить).

Список опций:

1. OFF - шестнадцатичные числа. ON - десятичные; в дизассемблировании и некоторых операциях при выводе.

2. Если OFF, то в случае вывода на принтер выводимая информация отображается на экране.

3. ON - во время третьего прохода дизассемблирования командой FLIST генерируемый файл параллельно выводится на экран. OFF - выводятся только текущие адреса.

4. OFF - во время медленного или пошагового выполнения программ если происходит обращение JMP или CALL в область ROM, то ROM процедура будет выполняться с нормальной скоростью.

5. OFF - при выполнении команды FLIST генерируется файл, содержащий метки, что позволяет переассемблировать его. ON - файл содержит адреса.

6. OFF - очищает экран при командах JUMP, CALL и SLOW. ON - выполнение начинается с экраном монитора.

7. вывод на принтер: OFF - ZXPrinter. ON - Kempston Centronics Interface.

8. OFF - на принтер посылается перевод строки с каждым возвратом каретки. ON - только возврат каретки.

CAT - каталог микродрайва.

ERA "<имя>" - удаление файла с именем "<имя>".

DI - запрещение прерываний.

EI - разрешение прерываний.

ROM <флаг> - работа с теневой ROM Interface 1. Если <флаг> = ON то разрешен режим работы с процедурами теневой ROM т.е. обращение к ROM от адреса #0008 до #1708 заменяется обращением к теневой ROM.

MAP - информация об использовании памяти.

MDRV [<номер>] - переключает ввод/вывод на микродрайв с номером <номер>.

TAPE - переключает ввод/вывод на ленту.

LOAD "<имя>", [<адрес>] - загружает файл "<имя>" с адреса <адрес>, если адрес отсутствует, файл загружается с адреса MEM.

SAVE "<имя>", <нач. адр>, <кон. адр> - запись файла "<имя>" байтами памяти с адреса <нач. адр> по <кон.адр>.

SCRN - отобразить виртуальный экран (возврат при нажатии любой клавиши).

SCRN <флаг> - разрешить/запретить виртуальный экран. <флаг> = ON / OFF (1/0).

SCRN = <адрес> - базовый адрес для сохранения виртуального экрана в памяти. (длина 6912 байт).

Замечание: Отображение всей информации, выводимой пользовательской (отлаживаемой) программой во время SLOW, CALL, JUMP осуществляется на экран, который сохраняется при возврате в монитор (при точке прерывания (для всех трех), ESC (для SLOW) или RET (для CALL)) в определенной Вами области памяти и может вызываться командой SCRN. Пошаговое выполнение также обновляет виртуальный экран.

DISS [<нач. адр>, [<кон. адр>]] - дизассемблер, выводящий листинг в левую верхнюю область экрана. Дизассемблирование происходит с адреса <нач.адр> до <кон.адр>.

LIST [<нач. адр>, [<кон. адр>]] - то же, но листинг дизассемблера выводится на весь экран.

LLIST [<нач. адр>, [<кон. адр>]] - то же, но листинг дизассемблера выводится на принтер.

DB [<db номер>, [<нач. адр>, <кон. адр>]] - позволяет определить при дизассемблировании области байтов (псевдооперация DB для ассемблера). Можно определить 8 областей - на номер области указывает <db номер>. Область начинается с <нач.адр> по <кон.адр>. Если аргументы отсутствуют, в DB область переводится весь процесс дизассемблирования.

FLIST "<имя>", <нач. адр>, <кон. адр> - то же, что и LIST, но вывод происходит в файл с именем "<имя>", вывод зависит от опций 3, 5. Дизассемблирование происходит в три прохода и нуждается в резервировании памяти.

WORK = <нач. адр>, <кон. адр> - резервирование области памяти для FLIST от <нач. адр> до <кон.адр>. Эта память используется также для SLOW/TRACE.

WIDTH = <слово> - ширина страницы принтера.

LENGTH = <слово> - длина страницы принтера.

3. КОМАНДЫ ОТЛАДКИ.

Пошаговое выполнение

Нажатие следующих клавиш позволяет производить пошаговое выполнение программ в памяти Spectrum-a:

CS & "6" - инкрементация PC.

CS & "7" - декрементация PC.

SS & "G" - переход к следующей инструкции без выполнения текущей, значение PC устанавливается на адрес следующей инструкции.

SS & "D" - переход к следующей инструкции с выполнением текущей, значение PC устанавливается на адрес следующей инструкции.

SS & "F" - то же, но не происходит заход в подпрограммы по командам CALL и RST. Эти подпрограммы выполняются с обычной скоростью микропроцессора Z80.

Замечание: Если процедура нуждается в параметрах, передаваемых через область кодов.

F.e.: как RST8 или так:

CALL SUBROUTINE

DB PAR1, PAR2, ..., PARN

необходимо переключить опцию 4 в состояние ON для избежания ошибок при отладке.

Медленное выполнение

SLOW <режим> - запускает программу на медленное выполнение с адреса, хранящегося в PC.

Список режимов:

0 - экран не обновляется после каждой инструкции;

1 - обновляется только дамп памяти;

2 - обновляется только дисплей регистров;

3 - 1 и 2 вместе;

Режимы 4-7 повторяют режимы 0-3, но

0-3 - для SS & "D";

4-7 - для SS & "F";

Остановка медленного выполнения происходит по нажатию ESC или при совпадении PC с точкой прерывания.

TRACE [<количество инструкций>] - во время медленного выполнения значения PC при выполнении каждой инструкции заносится в память, определенную командой WORK. В память должно занестись <количество инструкций> адресов, или столько, сколько позволяет зарезервированная память.

LTRACE [<количество инструкций>] - то же, но на принтер.

Точки прерываний

Здесь и ниже <тип> это тип точки прерывания, список типов:

0 - продолжать в медленном режиме 0.

1 - продолжать в медленном режиме 1.

2 - продолжать в медленном режиме 2.

3 - продолжать в медленном режиме 3.

4 - продолжать в медленном режиме 4.

5 - продолжать в медленном режиме 5.

6 - продолжать в медленном режиме 6.

7 - продолжать в медленном режиме 7.

- 8 - декрементировать счётчик, стоп если ноль, иначе продолжение в медленном режиме 0.
- 9 - декрементировать счётчик, стоп если ноль, иначе продолжение в медленном режиме 1.
- 10 - декрементировать счётчик, стоп если ноль, иначе продолжение в медленном режиме 2.
- 11 - декрементировать счётчик, стоп если ноль, иначе продолжение в медленном режиме 3.
- 12 - декрементировать счётчик, стоп если ноль, иначе продолжение в медленном режиме 4.
- 13 - декрементировать счётчик, стоп если ноль, иначе продолжение в медленном режиме 5.
- 14 - декрементировать счётчик, стоп если ноль, иначе продолжение в медленном режиме 6.
- 15 - декрементировать счётчик, стоп если ноль, иначе продолжение в медленном режиме 7.
- 16 - продолжать в быстром режиме.

Все эти типы являются типами особых точек прерывания, в общем простая точка прерывания вызывает остановку выполнения и возврат в монитор.

Команды работы с точками прерываний

BREAK <номер>, <флаг>, <адрес> - определяет точку прерывания с номером <номер> по адресу <адрес> в состоянии <флаг>, где <флаг> показывает запрещена эта точка или разрешена (ON/OFF).

DEFBRK <номер>, <тип>, <адрес>, [<счётчик>] - определяет особую точку прерывания с номером <номер> по адресу <адрес> типа <тип>. После определения в состоянии ON. <счётчик> используется в типах 8-15.

BRK <номер>, <флаг> - устанавливает состояние точки прерывания <номер> в ON или OFF.

DELETE <номер> - уничтожает точку прерывания <номер>.

LBRK - выводит список точек прерывания.

4. АНАЛИЗАТОР.

Для удобства отладки можно применять язык программирования, напоминающий Forth. С помощью программ, написанных на этом языке можно управлять отладкой программ.

Для начала работы с анализатором необходимо определить его рабочую область командой:
PROG = <нач.адр>, <кон.адр> (объем не менее 250 байтов).

Для выполнения Forth-овских операций используется команда **EVAL**

F.e.: (занесение двух чисел в стек (не программный стек Z80, а именно стек языка Forth))

EVAL 2 3 (и ENTER) ,

тогда на экране появится:

STATE OF STACK:

#3 #2

ENTER COMMAND

Команды анализатора

PROG = <нач.адр>, <кон.адр> - смотри выше.

CLEAR - стирает программное пространство, определенное командой **PROG** и все определенные слова.

ANALYSER<флаг> - если <флаг> = OFF не происходит проверка во время медленного выполнения.

WORD <имя слова><определение>

\$ <имя слова><определение>

- определение слова <имя слова> как выполняемой последовательности <определение>.

Замечание: <имя слова>, имеющее вид STOP#, где # - число от 0 до 9

Ф.е.: STOP1

обязывает анализатор во время медленного выполнения после выполнения каждой инструкции выполнять слова STOP#. Слова STOP# должны оставлять в стеке флаг, по которому в случае, если флаг ноль медленное выполнение продолжается, иначе прекращается.

EVAL <определение> - вычисление определения.

LDEF - список определенных слов на экран.

PDEF - то же на принтер.

DEFSAVE " <имя>" - запись определенных слов в файл "<имя>". **DEFLOAD** "<имя>" - загрузка ранее определенных слов.

EDIT <имя слова> - редактирование слова.

Зарезервированные Forth-ом слова

Слово	Действие	Стек
AF, BC, DE, HL, IX, IY, SP, PC	Помещает значение регистровой пары в стек.	> n1
A, B, C, D, E, H, L, F	Значение регистра в стек	> n1
CF, PVF, NF, HF, ZF, SF	Значение флага в стек Где: CF перенос PVF четность/переполнение NF сложение/вычитание HF полуперенос ZF ноль SF знак	> f1
I	Расположение значения флага на стеке: 0 - для сброса 1 - для установки	> n1
ADDR	Значение последнего адреса доступа к памяти в стек. Если инструкция не пишет или не читает память устанавливает 0 на стек (в т.ч. для блоковых пересылок, сравнений, ввода/вывода и при стековых операциях).	> n1
RD	Признак доступа-чтения в стек	> f1
WR	Признак доступа-записи в стек	> f1
ACF	Признак доступа-просто в стек	> f1
Эти три команды устанавливают в стеке флаг = 1 если был доступ к памяти нужного режима. Замечание: ACF, RD и WR сбрасывают на начало медленное выполнение.		
+, -, *, /	Арифметика	n1 n2 > n3
AND, OR, XOR	Логика побитовая	n1 n2 > n3
NOT		n1 > n2
=	n1 равно n2 ?	n1 n2 > f1
>, <	Стандартное сравнение	n1 n2 > f1
>=, <=, <>	n1 и n2	
0=	n1 равно 0 ?	n1 > f1
0>	n1 не равно 0 ?	n1 > f1
&	Логическое И	f1 f2 > f3
?SCR1	Обновление дампа памяти если флаг на вершине стека истина	f1 >

?SCR2	то же для регистров	f1 >
!	Запись слова n1 по адресу n2	n1 n2 >
C!	то же, но запись байта	n1 n2 >
MEM	Значение мониторовского MEM в стек.	> n1
MEM!	Установка MEM из стека	n1 >
?PAUSE	Для истинного флага анализатор ждет нажатия клавиши	f1 >
IF	Если флаг на вершине стека ложь, то интерпретация анализатором выполняемого слова заканчивается	f1 >
C@	Выдать байт, на который указывает адрес на вершине стека(т.е. (n1) -> n2)	n1 > n2
@	то же, но для слова	n1 > n2
BIT	Текст бита n2 в байте/слове	n1 n1 n2 > f1
DUP	Дубликация вершины	n1 > n1 n1
SWAP	Обмен	n1 n2 > n2 n1
OVER	Второй через первый	n1 n2 > n1 n2 n1
ROT	Ротация вершины	n1 n2 n3 > n2 n3 n1
ON		> 1
OFF		> 0
DROP	Скидывание вершины	n1 >
NOP	Как NOP для Z80	>
KEY	Значение последней нажатой клавиши	> n1
CALL	Вызов процедуры по адресу n1	n1 >
	Похожа на CALL для Z80, но как параметры в процедуру передается:	
	HL: Адрес процедуры анализатора выталкивающей вершину Forth- стека в регистровую пару BC.	
	DE: то же, но процедура кладет BC на вершину стека.	
	IX: Адрес таблицы регистров:	
	IX + 0 SP	
	IX + 2 IX	
	IX + 4 IY	
	IX + 6 PC	
	IX + 8 BC	
	IX + 10 DE	
	IX + 12 HL	
	IX + 14 байт флагов F	
	IX + 15 A	
	IX + 16 BC'	
	IX + 18 DE'	
	IX + 20 HL'	
	IX + 22 байт флагов F'	
	IX + 23 A'	

Полезные определения (заодно примеры)

\$RANCE: ROT SWAP OVER >= ROT ROT <= &

RANCE - проверка входит ли n1 в диапазон n1 n2 n3 > f1 от n2 до n3 ?

\$MEMWR: ADDR >= SWAP OVER ADDR <= & WR &

MEMWR - защита области памяти от n1 до n2 n1 n2 > от записи. пример использования:

\$STOP9: #8000 #8FFF MEMWR #0 #1FF MEMWR OR

защищает от записи области #8000 - #8FFF и #0 - #1FF.

\$LXOR: 0 > SWAP 0 > XOR

LXOR - логическое исключающее ИЛИ f1 f2 > f3

Ещё одно полезное определение

Печать значения с вершины стека в шестнадцатичном виде. Часть на ассемблере (ручное кодирование или ассемблирование на Genius ассемблере).

```

      ORG    #ABE0
      CALL  DROPBC
      LD     A,22
      RST    #10
      XOR    A
      RST    #10
      RST    #10
      LD     A,B
      CALL  SIGN
      LD     A,C
SIGN:  PUSH  AF
      RLCA
      RLCA
      RLCA
      RLCA
      CALL  TETRA
      POP   AF
TETRA: AND   #0F
      ADD   A,#90
      DAA
      ADC   A,#40
      DAA
      RST   10H
      RET

```

DROPBC: JP (HL)

; Для правильной работы убедитесь,

; что перед запуском IY = #5C3A

Часть на Forth-e

\$PRINT: #ABE0 CALL

PRINT Печать вершины стека n1

n1 >

ЗАПОМНИТЕ: Команды анализатора могут выполняться ТОЛЬКО при медленном выполнении для использования слов STOP#.

Edit by -=LV=- St-Petersburg

Камертон и музыкальный сопроцессор

© Рюмик С.М., г. Чернигов, 1995.

Применение музыкальных сопроцессоров АУ-3-8910/12 (в дальнейшем по тексту АУ) по праву способствовало уникальному долголетию компьютеров ZX-Spectrum.

Музыкальные сопроцессоры появились в конце 1985 года в модели с расширенной памятью ZX-Spectrum-128 и затем присутствовали во всех последующих фирменных и отечественных разработках с памятью 128 и более Кб.

При всем разнообразии схем включения музыкальных сопроцессоров, один вопрос до сих пор решается неоднозначно, какой же должна быть частота, подаваемая на тактовый вход АУ-микросхемы? В переводе фирменного описания указана цифра 1,7734 МГц с точностью 0,01% [1]. В реальных схемах частоту выбирают в диапазоне 1-2 МГц, рассчитывая услышать "именно те звуки, которые были задуманы при написании программы" [2].

Объяснить этот феномен, а также ответить на вопросы: какая же должна быть "истинная" частота и так ли был прав сэр Клайв Синклер, устанавливая определенные константы в ПЗУ-128, поможет предлагаемая публикация.

1. Общие сведения из теории музыкальной акустики.

Частоты акустических колебаний в пределах 20 - 20000 Гц называются звуковыми.

Высота звука - это качественная оценка частоты колебаний звука. С ростом частоты высота звука увеличивается.

Человеческое ухо чувствительно к изменению высоты тона более, чем к какому-либо другому параметру звука. Человек способен различить в диапазоне до 4000 Гц изменение частоты в 1 Гц. А по биениям частот двух тонов он может обнаружить разность частот до десятых долей Гц! [3].

Каждый звук в мелодии имеет свою частоту. Соответствие ноты и частоты звука подчиняется строгим математическим законам. В 1700 году немецкий ученый и музыкант Андреас Веркмейстер предложил математическую модель музыкальной шкалы, основанную на теории иррациональных чисел. Речь идет о логарифмически равномерной или, по-другому, равномерно темперированной музыкальной шкале, которой мы пользуемся и поныне.

Звукоряд насчитывает семь полных и две неполных октавы. Это (снизу - вверх): субконтроктава (три верхних звука), контроктава, большая октава, малая октава, первая, вторая, третья, четвертая октавы и пятая октава (три нижних звука). В октаве 12 нот, итого в звукоряде 90 нот, что соответствует клавиатуре пианино.

Нотный состав музыкальной шкалы приведен в таблице 1.

Таблица 1.

Нота	до	до-диез	ре	ре-диез	ми	фа	фа-диез	соль	соль-диез	ля	ля-диез	си
Обозначение	C	C#	D	D#	E	F	F#	G	G#	A	A#	B

- знак повышения высоты тона на полтона.

Интервал между любыми двумя рядом расположенными нотами табл.1 называется полутоном. Частоту любой ноты F(i) можно вычислить, зная частоту рядом стоящей ноты F(i-1), по формуле:

$$F(i) = 2^{\frac{1}{12}} * F(i-1) \quad (1)$$

В качестве стандарта звукового и частотного соответствия принята нота "ля" первой октавы, имеющая частоту ровно 440 Гц. Недаром камертон (устройство, являющееся эталоном высоты звука) воспроизводит именно 440 Гц.

Все остальные 11 нот октавы являются производными от "ля" и не имеют абсолютно точного значения частоты, так как число "корень двенадцатой степени из двух" в формуле (1) является иррациональным, то есть приближенным ($\approx 1,05946\dots$).

Цент - это единица частотного интервала, равная 1/100 полутона. Октава равна 12 полутонам или 1200 центам.

2. Структурная схема музыкального сопроцессора.

Музыкальный сопроцессор в справочнике [4] назван программируемым генератором звуков. По принципу работы он наиболее близок к классу музыкальных синтезаторов.

Основой любого электромузыкального инструмента (ЭМИ) является генераторный блок, который, как известно [5], выполняют либо с 12 отдельными тональными генераторами, либо с одним, работающим совместно с делителями частоты. Именно второй способ применен в архитектуре АУ.

На рисунке 1 приведена структурная схема музыкального сопроцессора [4]. Для упрощения анализа в структуре оставлены только элементы, влияющие на высоту звука, форма сигнала и спецэффекты не рассматриваются.

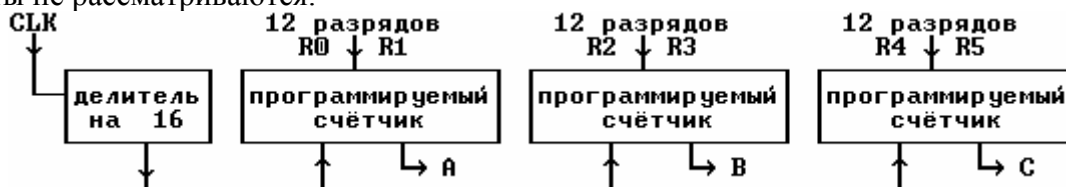


Рис.1.

Управление генераторами тона А, В, С, осуществляется через 6 внутренних управляющих регистров R0-R5. Каждый канал снабжен собственным программируемым счетчиком. Выходная частота получается делением синхронизирующей тактовой частоты CLK сначала на $K_{сч}=16$ с последующим делением полученного результата на 12-разрядную величину, соответствующую частоте тона.

Значение 12-разрядной величины получается путем комбинации содержимого 4 младших разрядов грубого тона (R1, R3, R5 - число 0-15) и 8 разрядов регистра точного тона (R0, R2, R4 - число 0-255).

Все три канала А, В, С, работают независимо друг от друга.

3. Программирование музыкального сопроцессора.

С точки зрения программиста музыкальный сопроцессор в Спектруме представляет собой устройство, имеющее два порта: выходной порт с адресом #BFFD (49149) и входной/выходной порт с адресом #FFFD (65533).

Из 16 внутренних регистров АУ в установке высоты тона участвуют шесть: R0-R5.

Для работы с регистрами первоначально в порт #FFFD заносится его порядковый номер, для R0 это #00, R1 - #01, R2 - #02 и т.д. Затем в порт #BFFD заносится конкретное число, определяющее в случае регистров R0-R5 коэффициент деления входной частоты.

Программирование высоты тона музыкального сопроцессора осуществляется следующими способами:

- через оператор вывода OUT на БЕЙСИКЕ-48 или БЕЙСИКЕ-128;
- через команду АССЕМБЛЕРА OUT в машинных кодах;

- через оператор PLAY на БЕЙСИКЕ-128.

Команда PLAY гораздо более гибкая, чем обыкновенный BEEP, так как позволяет программировать мелодию непосредственно в нотах выбранной октавы. Формат команды PLAY при воспроизведении, например, камертонной ноты "ля" первой октавы:

PLAY "O4A"

где цифра "4" после заглавной буквы "O" обозначает порядковый номер октавы, а заглавная буква "A" является символом ноты "ля".

Оператор PLAY охватывает 9 октав, число после буквы "O" может изменяться только в пределах от 0 (субконтроктава) до 8 (пятая октава). Действительно, задайте: PLAY "O9A", и получите ошибку: "Out of range".

Буква, обозначающая ноту, может быть как заглавная ("A"), так и строчная ("a"), причем строчная буква обозначает понижение одноименной ноты ровно на октаву.

Коэффициенты деления оператора PLAY зашиты в ПЗУ-0 (ПЗУ Спектр-128) и занимают область 216 байтов по адресам #1096-#116D.

4. Определение коэффициентов деления и частоты CLK.

Из трёх способов программирования звука только последний, через оператор PLAY, даёт абсолютный результат, то есть, имеется привязка к конкретным нотам. Это означает, что, запрограммировав звучание ноты "ля" первой октавы через команду PLAY и сравнив полученную частоту камертона, следует ожидать одинаковые результаты.

Так как высота ноты для Спектр-128 однозначно зависит от тактовой частоты, подаваемой на вход CLK сопроцессора, то идеальное звучание одной ноты возможно только для одной, вполне определенной частоты CLK.

Поиск частоты можно провести двумя путями.

Первый способ, практический. Подключить к входу CLK внешний задающий генератор в диапазоне 1-2 МГц и, изменяя его частоту, добиться совпадения звучания с эталоном. Способ сложен в реализации, так как измерение надо произвести для всех 90 нот звукоряда.

Второй способ, теоретический. Рассчитать требуемое значение частоты CLK, используя зашитые в ПЗУ компьютера Спектр-128 коэффициенты деления. Этот способ предпочтительнее, так как позволяет при высокой точности результата провести математическую обработку.

Для нахождения коэффициентов деления можно использовать тот факт, что после выполнения команды PLAY в регистрах грубой и точной установки высоты тона остаются коэффициенты, которые можно прочесть, обратившись к порту #FFFD. Значение общего десятичного коэффициента деления K_i определится по формуле:

$$K_i = T_i + 256 * G_i \quad (2)$$

где T_i и G_i - соответственно десятичные коэффициенты деления точного и грубого регистров для i -й ноты.

Частоту i -й ноты $F(i)$ можно определить по формуле (1), зная эталонную частоту. Представим формулу (1) в удобном для вычисления виде:

$$F(i) = \frac{F_{\text{ля}}}{16} * 2^{\frac{i-1}{12}} \quad (3)$$

где: $F_{\text{ля}}/16$ - частота самой низкой ноты звукоряда F1, которая ровно в 16 раз меньше частоты эталона $F_{\text{ля}}=440$ Гц;

i - от 1 до 90 - порядковый номер ноты.

Частотный диапазон исследуемого звукоряда F1-F90 равен 27,5-4698,6 Гц. Частота CLK i -й ноты определится по формуле:

$$CLK_i = F(i) * K_i * K_{\text{сч}} \quad (4)$$

где $K_{сч}=16$ - коэффициент счёта общего для трех каналов делителя, находящегося внутри АУ (рис.1).

Очевидно, что для каждой из 90 нот существует своя "истинная" CLK, то есть частота, при которой конкретная нота будет звучать абсолютно точно. Итого 90 частот CLK, которые можно вычислить, используя программу листинга 1.

Листинг 1.

```

10 REM AY-3-8910/12
20 LET I=0: LET R=65533: FOR N=1 TO 8
30 FOR M=1 TO 23 STEP 2
40 IF M=7 THEN LET N=N+1
50 LET A$="O"+"012345678"(N)+"N10"+" A#A B C#C D#D E F#F G#G"(M TO M+1)
60 PLAY A$
70 OUT R,0: LET T=IN R: OUT R,1: LET G=IN R
80 IF T=255 OR G=255 THEN GO TO 70
90 LET I=I+1: LET K=T+256*G
100 LET F=440/16*2^((I-1)/12)
110 LET CLK=F*K*16
120 IF I>90 THEN STOP
130 PRINT A$( TO 2);A$(6 TO );"F";I;" K=";K;" CLK=";CLK;" Hz"
140 NEXT M: LET N=N-1: NEXT N

```

Результат работы удобно представить в виде гистограммы (рис.2). Гистограмма - это график, по горизонтальной оси которого отложена исследуемая величина (в данном случае это CLK), а по вертикальной оси - частота ее повторения при выборке из ограниченного количества испытаний (в данном случае из 90 "истинных" CLK).

На гистограмме можно выделить рабочую зону, где находится 80 нот или 89% всех значений, две свободные зоны и две области "фальшивых" нот, имеющих предельно большой разброс CLK. Наибольшее число "попаданий" приходится на диапазон 1773000 - 1774000 Гц (31 значение или 34%), следовательно здесь и должна быть расчетная CLK.

5. Определение логически обоснованной частоты CLK.

Листинг 1 помог вычислить 90 значений CLK. Но в реальности частота CLK одна для всех нот, поэтому каждая нота будет звучать чуть-чуть "не так", с погрешностью.

Коэффициенты деления K_i , зашитые в ПЗУ 128К, были по какой-то методике заранее рассчитаны программистами сэра Клайва Синклера. Не зная этой методики, можно пойти от обратного - при известных коэффициентах деления вычислить пределы логически обоснованной частоты CLK.

Например, коэффициент деления для ноты F1 "ля" субконтроктавы равен 4031, именно 4031, а не 4030 или 4032. Поэтому можно вычислить три частоты CLK:

$$CLK_0 = 27.5 \text{ Гц} * 4031 * 16 = 1773640 \text{ Гц}$$

$$CLK_{\min} = 27.5 \text{ Гц} * 4030 * 16 = 1773200 \text{ Гц}$$

$$CLK_{\max} = 27.5 \text{ Гц} * 4032 * 16 = 1774080 \text{ Гц}$$

Расположим полученные частоты на оси, как показано на рис.3. Логически обоснованным интервалом будем называть интервал между частотами MIN и MAX, определяемых из условия равенства логарифмов отношений частот:

$$MIN_i = F_i * K_{сч} * \text{SQR}(K_i * (K_i - 1)) = 1773420 \text{ Гц (для F1)}$$

$$MAX_i = F_i * K_{сч} * \text{SQR}(K_i * (K_i + 1)) = 1773860 \text{ Гц (для F1)}$$

где SQR - математическая функция "квадратный корень".

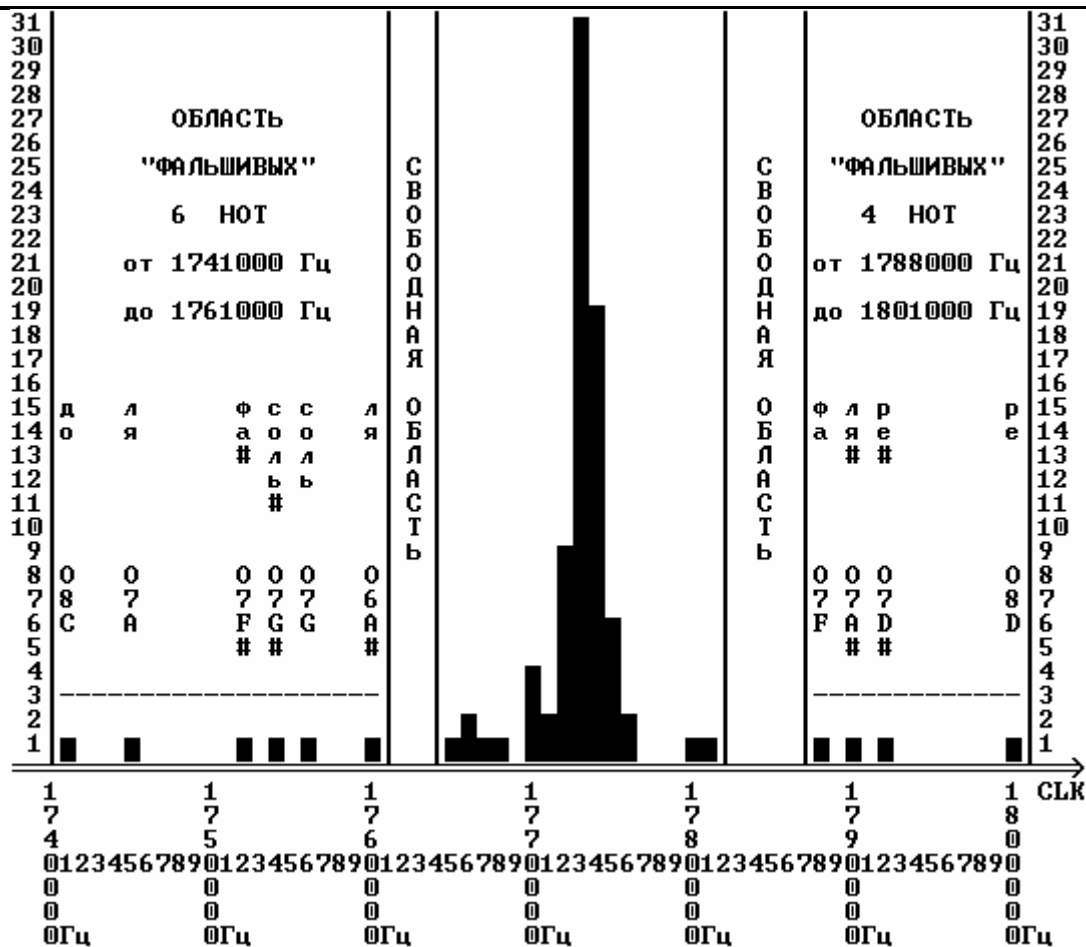


Рис. 2.

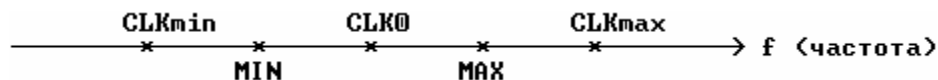


Рис. 3.

Приведенный пример рассчитан на самую низкую ноту, для которой логический интервал CLK самый минимальный. Поэтому можно утверждать, что, если коэффициент K1 выбран продуманно, а не произвольно, то "истинная" CLK находится в пределах MIN-MAX, иначе был бы выбран другой коэффициент. Аналогично для всех 90 нот можно вычислить интервалы MIN_i - MAX_i , совместить их друг с другом на оси частот и посмотреть, какой же интервал, в конце концов, останется общим для всех.

Вычисления можно поручить компьютеру, дополнив при помощи команды MERGE листинг 1 строками из листинга 2.

Листинг 2.

```

10 LET e=1E-7: LET min=e: LET max=1/e
120 IF I<91 THEN GO TO 150
130 PRINT "MIN=";min;" Hz  MAX=";max;" Hz": STOP
150 LET mi=F*16*SQR (K*(K-1))
160 LET ma=F*16*SQR (K*(K+1))
170 IF mi-max>e OR min-ma>e THEN PRINT "ERROR F";I,"K=";K: GO TO 140

```

```

180 IF mi-min>e THEN LET min=mi
190 IF max-ma>e THEN LET max=ma
200 GO TO 140

```

Результат работы заставляет задуматься, общий интервал для 90 нот:

MIN=1773441,5 Гц

MAX=1773520 Гц

ошибочная нота "ре-диез" малой октавы (F31, октава 03). Если изменить ошибочный коэффициент K31=712 на 713, то интервал MIN-MAX останется прежним, а ошибка исчезнет.

Известны исследования однотипной ситуации для ЭМИ [5], показывающие, что в ряду коэффициентов деления существуют "магические" числа, которые позволяют свести к минимуму погрешности нот для вполне определенных входных частот.

Оказывается, что и для имеющегося ряда коэффициентов деления существуют другие комбинации чисел, при которых интервал MIN-MAX находится в главном пике гистограммы и отсутствуют ошибочные коэффициенты (таблица 2).

Таблица 2.

КОЭФФИЦИЕНТЫ	K1	K6	K17	K31	MIN MAX, Гц	dF, Гц
без коррекции	4031	3020	1600	712	1773441-1773520	ERR.F31
K31	4031	3020	1600	713	1773441-1773520	79
K6	4031	3019	1600	712	1773420-1773423	3
K6, K31	4031	3019	1600	713	1773423-1773441	18
K1, K6	4030	3019	1600	712	1773414-1773420	6
K1, K6, K17	4030	3019	1599	712	1773395-1773414	19

Из таблицы 2 видно, что наиболее широкий интервал dF=MAX-MIN имеет вариант с коррекцией K31 для частоты F31. Думается, что при расчете коэффициентов ПЗУ-128 программистами сэра Клайва Синклера проводилась подобная оптимизация, однако коэффициент K31 был упущен из виду.

Необходимость в как можно более широком интервале dF очевидна.

На вход CLK поступает кварцованная частота. Точность настройки современных кварцевых резонаторов 1-20 МГц колеблется от $(\pm 5 - \pm 15) \cdot 10^{-6}$ для вакуумных резонаторов и до $(\pm 10 - \pm 50) \cdot 10^{-6}$ для обычных герметизированных. Здесь и далее "1E-6" - число "десять в минус шестой степени".

В пересчете на частоту 1,77 МГц допустимые отклонения составят от ± 8 до ± 88 Гц.

Следовательно, расчетной следует признать частоту CLK = (1773480 ± 40) Гц при точности настройки кварцевого резонатора не хуже $\pm 20 \cdot 10^{-6}$. Коэффициент K31 должен быть увеличен до 713.

6. Распределение абсолютных погрешностей нот.

Введем понятие абсолютной погрешности звучания ноты. Это та величина, на которую отличается звучание "идеальной" ноты (1) от реальной.

Малые музыкальные интервалы удобно измерять в центах, абсолютная погрешность i-той ноты определяется по формуле:

$$|S_i| = 1200 * \log_2 \left(\frac{F_i}{F_i(0)} \right) \quad (5)$$

где S_i - абсолютная погрешность, в центах;

F_i - реальная частота ноты;

$F_i(0)$ - "идеальная" частота ноты.

Задача состоит в том, чтобы для дискретных частот CLK из рабочей зоны гистограммы рис. 2 вычислить массив из 90 значений погрешностей $|S_i|$ и проанализировать результат методом "срезов". "Срез" - это в данном случае определенный уровень погрешности в центах.

Стандартного ряда для "срезов" не существует, поэтому можно воспользоваться косвенными сведениями: "1 цент" - это погрешность прибора для настройки музыкальных инструментов; "3 цента" - минимальная погрешность, которую способен различить человек на слух [3]; "5 центов" - максимальная погрешность установки частоты тона ЭМИ [6]; "10, 20 центов" - произвольно взятые целочисленные значения грубых погрешностей.

Программа на БЕЙСИКЕ, производящая вычисления массивов погрешностей по "срезам", получается, если добавить через MERGE строки из листинга 3 в листинг 1.

Листинг 3.

```

10 DIM C(90): LET e=1E-7
120 IF I>90 THEN GO TO 140
130 LET C(I)=CLK
150 FOR Z=1 TO 12 STEP 2
160 IF Z=7 THEN LET Z=10
170 IF Z=12 THEN LET Z=20
180 FOR C=1770000 TO 1778000 STEP 500
190 LET CENT=0: FOR I=1 TO 90
200 LET S=ABS (1200/LN 2*LN (C/C(I)))
210 IF Z-S>e THEN LET CENT=CENT+1
220 NEXT I
230 PRINT "CENT<" ; Z ; "  L=" ; INT (CENT*100/90+0.5) ; "%  CLK=" ; C ; "  Hz"
240 NEXT C: NEXT Z

```

Результаты работы программы выводятся в форме: CENT<1 L=4% CLK=1770000 Hz. Это означает, что для частоты CLK равной 1770000 Гц из 90 нот 4% имеют абсолютную погрешность звучания менее 1 цента.

Для наглядности полученные результаты приведены в таблице 3.

Таблица 3.

"СРЕЗЫ"	Частота CLK, Гц																
	1770000	1770500	1771000	1771500	1772000	1772500	1773000	1773500	1774000	1774500	1775000	1775500	1776000	1776500	1777000	1777500	1778000
1 цент	4	7	7	11	14	29	44	49	56	40	28	16	9	6	2	1	0%
3 цента	22	39	54	66	72	76	79	76	77	74	74	66	64	46	38	17	12%
5 центов	78	81	84	84	83	83	82	82	81	81	81	77	78	78	77	68	67%
10 центов	88	87	89	89	89	89	89	89	89	89	89	89	88	88	86	86	86%
20 центов	96	96	96	96	97	97	96	96	96	96	94	94	94	94	93	93	93%

Как видно, расчетная частота CLK=1773480 Гц с точки зрения минимума абсолютных погрешностей не является оптимальной! И, вообще, говорить об оптимальной частоте можно только для конкретных "срезов".

Например, чтобы иметь максимальный процент нот, удовлетворяющих требованиям ЭМИ (5 центов), частота CLK должна быть в пределах 1771000 – 1771500 Гц и т.д.

Усредняя результаты таблицы 2, можно расширить диапазон допустимой частоты CLK до 1771500 – 1775000 Гц, при этом снимаются жесткие ограничения на точность настройки кварцевого резонатора.

Разумеется, в фирменном ZX Spectrum-128 возможности варьировать частотой CLK не было, поскольку вся схема жестко привязана к одному кварцевому генератору.

7. Расчёт относительных погрешностей музыкальных интервалов.

Кроме абсолютных погрешностей, указывающих, на сколько отличаются ноты от "истинных" по частоте, не менее важны относительные погрешности музыкальных интервалов.

Музыкальная мелодия описывается последовательными интервалами между составляющими ее тонами. Для музыкального интервала характерно определенное отношение частот звуков. Именно отношение частот между звуками мелодии делают её узнаваемой.

Например, перенести мелодию выше - это значит воспроизвести её иными звуками (более высокими по частоте), но с точным сохранением отношений частот тонов в каждом интервале. Характер мелодии при этом на слух не искажается.

Можно выделить 12 основных музыкальных интервалов в пределах октавы [7]: от малой секунды (1 полутон) до октавы (12 полутонов).

Рассчитать, насколько фальшиво будут звучать все 12 интервалов в пределах полного звукоряда из 90 нот поможет формула (6). Формула расчёта не будет иметь зависимости от частоты CLK, так как сравниваются отношения частот, зависящие только от коэффициентов деления.

$$|dS(i, x)| = 1200 * \log_2 \left(\frac{K_i}{K_{i+x}} \right) - 100 * x \quad (6)$$

где x от 1 до 12 - количество полутонов в музыкальном интервале;
 i от 1 до (90- x) - порядковый номер ноты.

Вычисления относительных погрешностей по формуле (6) можно выполнить, присоединив через MERGE листинг 4 к листингу 3. Результаты вычислений приведены в таблице 4.

Листинг 4.

```
130 LET C(I)=K
180 FOR X=1 TO 12
200 LET S=ABS (1200/LN 2*LN (C(I)/C(I+X))-100*X)
215 IF I+X=90 THEN GO TO 230
230 PRINT "CENT<";Z;" L=";INT (CENT*100/I+0.5);"% MUZ=";X
240 NEXT X: NEXT Z
```

Таблица 4.

"СРЕЗЫ"	МУЗЫКАЛЬНЫЕ ИНТЕРВАЛЫ (количество полутонов)												L _{сред.}
	1	2	3	4	5	6	7	8	9	10	11	12	
1 цент	49	45	47	47	39	39	39	48	40	39	47	63%	45.2%
3 цента	66	58	70	69	67	65	71	72	73	59	65	76%	67.6%
5 центов	76	76	79	74	72	74	76	76	81	76	73	78%	75.9%
10 центов	88	88	85	85	87	82	87	87	89	89	86	83%	86.3%
20 центов	93	93	93	94	94	96	94	93	96	94	96	90%	93.8%

Распределение по "срезам" музыкальных интервалов достаточно равномерно. В среднем чуть более точно будут звучать октавы ($x=12$), чуть менее точно - малые септимы ($x=10$), чистые кварты ($x=5$).

Примерно третью часть всех интервалов в звукоряде человек с музыкальным слухом отнесёт к фальшивым ("срез" 3 цента). Кроме того, имеется около 7% очень грубых ошибок более 20 центов. Очевидно, что признать музыкальный сопроцессор эталонным источником звука для полного звукоряда нельзя.

Как показано [5], для известного максимального коэффициента деления ($K_1=4031$) существуют оптимальные коэффициенты $K_i(\text{opt})$, которые минимизируют относительные погрешности:

$$K_i(\text{opt}) = \text{INT} \left(\frac{K1}{2^{\frac{i-1}{12}}} \right) \quad (7)$$

Где INT - функция округления до ближайшего целого числа.

Можно добавить через MERGE строки из листинга 5 в листинг 1 и сравнить оптимальные коэффициенты с имеющимися в ZX-Spectrum-128.

Листинг 5.

```
120 IF I=1 THEN LET K1=K
130 IF I<91 THEN GO TO 160
150 STOP
160 LET Kopt=INT (0.5+K1/2^((I-1)/12))
170 IF Kopt<>K THEN PRINT "ERROR F";I;" K=";K;" K(opt)=";Kopt
180 GO TO 140
```

Результат указывает на пять неоптимальных нот. Должно быть K2=3805, K4=3390, K13=2016, K15=1796, K31=713.

Если провести расчет погрешностей музыкальных интервалов по формуле (6) для оптимальных коэффициентов, то выяснится, что это отразится только на улучшении параметров по "срезу" 1 цент - было Lcp=45.2%, стало Lcp=46.5%. Следовательно, коэффициенты деления, зашитые в ПЗУ-128, близки к оптимальным.

8. Выводы.

1. Музыкальные сопроцессоры AY-3-8910/12 не предназначены для эталонного воспроизведения звука в широком диапазоне частот. При допустимой частоте CLK=1-2 МГц [4] эквивалентная частота генераторного блока составит всего 62-125 кГц вследствие деления частоты на 16 внутри музыкального сопроцессора (рис.1).

Отсюда низкие значения коэффициентов деления на высоких частотах (K90=24) и большая максимальная погрешность округления [5]: $d=0.5/K90 \cdot 100\%=2,1\%$.

2. Расчётное значение частоты CLK для ZX-Spectrum-128 равно 1773480 Гц с точностью $+20 \cdot 10^{-6}$, при этом коэффициент K31 следовало бы увеличить на единицу. K31=713, ячейка #10D2 в ПЗУ-0: имеется #C8, должно быть #C9.

Однако на практике улучшение звучания ноты F31 для расчётной частоты CLK составит всего 0,05 цента, что можно зафиксировать лишь точными измерительными приборами. Поэтому перешивать ПЗУ128 сэра Клайва Синклера торопиться не надо во избежание головной боли от возможных проблем несовместимости ПЗУ.

При проектировании новых систем с музыкальными сопроцессорами следует проводить предварительное математическое моделирование с целью определения оптимальных коэффициентов деления.

3. Наиболее точно для музыкального сопроцессора звучат самые низкие ноты звукоряда, наиболее фальшиво - самые верхние. Абсолютные погрешности звучания нот для CLK=1773480 Гц приведены в таблице 5.

Таблица 5.

ОКТАВА	00	01	02	03	04	05	06	07	08
ЦЕНТЫ	<0.2	<0.4	<0.7	<1.6	<3.3	<5.8	<12.9	<27.1	<31.6

Использовать в музыкальных произведениях ноты четвертой (07) и пятой (08) октав нецелесообразно, так как даже неискушенный слушатель заметит фальшь. На гистограмме рис. 2 перечислены 10 "фальшивых" нот, которые явно не рекомендуется применять.

4. При введении отдельного кварцевого генератора для частоты CLK можно получить результирующие абсолютные погрешности нот лучше, чем на фирменном ZX-Spectrum-128. Значение

оптимальной частоты выбирается по данным табл.3 в диапазоне 1771500 - 1775000 Гц. Схема подключения отдельного генератора может быть выполнена по аналогии с приведенной в [1], при этом следует изменить коэффициент деления счётчика $K_{\text{дел}}$ согласно таблице 6.

Таблица 6.

№	КВАРЦЕВЫЙ РЕЗОНАТОР	ТОЧНОСТЬ НАСТРОЙКИ	$K_{\text{дел}}$	РАСЧЁТНАЯ CLK
1	РК169МВ-8ВТ-12399,5 кГц	+30*1E-6	7	1771357 Гц
2	РК172-ТВ-1-8862,9 кГц	+15*1E-6	5	1772580 Гц
3	РК100-4АЛ-8866,99 кГц	+5*1E-6	5	1773398 Гц
4	РК100-4АЛ-8867,24 кГц	+5*1E-6	5	1773448 Гц
5	РК100-4БН-12425 кГц	+5*1E-6	7	1775000 Гц
6	РК100-4АЛ-17734,47 кГц	+5*1E-6	10	1773447 Гц

5. Если значение CLK будет находиться вне рабочей зоны рис. 2, то вместо абсолютных погрешностей, которые и так будут велики, следует обратить внимание на относительные погрешности, которые определяют узнаваемость мелодии.

Для получения минимальных относительных погрешностей музыкальных интервалов следовало бы изменить значения коэффициентов деления K2, K4, K13, K15, K31. Соответственно перешить ячейки ПЗУ-0: #1098 с #DC на #DD; #109C с #3D на #3E; #10AE с #DF на #E0; #10B2 с #03 на #04; #10D2 с #C8 на #C9.

Однако улучшения на слух не будет заметно, так как изменения коснутся только "среза" в 1 цент. Кроме того, ноты F2 - F15 редко используются на практике.

При желании все-таки изменить коэффициенты в ПЗУ-128 и иметь более точное звучание музыкальных интервалов, лучше сделать это через замещающее "тенивое" ПЗУ.

6. Если тактовая частота CLK равна 1 МГц, то АУ-мелодия будет звучать ниже по высоте почти на октаву (10 полутонов).

Если CLK равна 2 МГц, то наоборот - звуки будут выше примерно на 2 полутона (208 центов).

Темп мелодии и относительные музыкальные погрешности не зависят от частоты CLK, чем и объясняется феномен высокой узнаваемости мелодии при широком варьировании тактовой частоты.

7. Из широко распространенных схем компьютеров 128К и приставок-128 более удачными следует признать решения с CLK=1,75 МГц ("АТМ-TURBO-2" фирмы MicroART, самодельный Львовский вариант 128К и т.д.).

При этом высота каждой ноты будет ниже по сравнению с фирменным ZX-Spectrum-128 на 23 цента или на 1,3%.

Абсолютные погрешности звучания для 80 нот рабочей зоны гистограммы составят от 15 до 30 центов или от 0,9% до 1,7%. Относительные погрешности музыкальных интервалов определяются данными таблицы 5 и не будут отличаться от имеющихся в фирменном компьютере.

Литература.

1. Кунгуров А. Несколько устройств для "Спекки". - ZX-РЕВЮ 1994, №6, стр. 17-18.
2. Шепелев Г. Особенности подключения АУ-3-8910 к Спектруму. - "Радиолобитель", 1994, №3, стр.13.
3. Акустика. Справочник / А.П. Ефимов и др. Под ред. М.А. Сапожкова. - М.: Радио и связь, 1989.
4. Системные программы для ZX-Spectrum-128К. Справочник. - М.: "VA PRINT", 1993.
5. Моисеев А. Выбор коэффициентов деления частоты. - "Радио", 1990, №3, стр. 63-64.
6. Володин А. Основные технические требования к ЭМС. - "Радио", 1980, №2, стр. 42.
7. Советский энциклопедический словарь / Гл. ред. А.М. Прохоров. - М.: Сов. энциклопедия, 1987.

Поиск "вечных жизней"

© Сергей Бородавкин, г. Гродно (Беларусь), 1995.

Хочу предложить программу, которая облегчит жизнь многим хакерам и сохранит их драгоценное время. Программа позволяет автоматизировать поиск адресов, в которых хранится количество жизней, патронов, бомб, количество оставшегося времени и т.д. Одним словом, можно искать адреса всего того, что меняется в процессе игры. А после этого получать долгожданное бессмертие. Ведь как неприятно, когда не находишь среди опубликованных POKES названия нужной игры. А теперь любой желающий может попробовать свои силы в этом увлекательном занятии. Что для этого надо? Две вещи: система TRDOS любой версии и эта небольшая программа. Идея такова. В различные моменты времени сбрасываем на диск содержимое ОЗУ MAGIC-кнопкой. Причём первый раз сбрасываем, когда у вас имеется *N чего-нибудь*. Где под *чем-нибудь* подразумеваются или жизни, или снаряды, или мины, или время, или ..., а *N* - это их количество. Второй раз сбрасываем игру на диск, когда уже количество стало равно *N-1* (*N+1*, если увеличивается). И третий - когда *N-2* (*N+2*, если увеличивается). Абсолютно не важно, если MAGIC-файлы после сброса не работают из-за испорченного стека. Теперь остальное можно доверить программе, которая найдет эти изменившиеся на единицу байты и покажет их адрес. Главное, получить три MAGIC-файла, удовлетворяющих выше сказанному условию. Число их (три) получено экспериментальным путём. При двух файлах выдавалось много лишних адресов, не имеющих для нас никакого значения из-за того, что приличное количество байт в двух файлах отличалось на единицу. Короче говоря, двух файлов оказалось недостаточно для однозначного определения нужного адреса.

Программа.

		ORG	25000	;адрес размещения программы.
61A8 CD6B0D	BEGIN	CALL	#0D6B	;очищаем экран.
61AB CD7762		CALL	PRN	;печатаем название программы.
61AE 11E364		LD	DE, TXMAG	;печатаем просьбу
61B1 012100		LD	BC, #21	;вставить диск с
61B4 CD3C20		CALL	#203C	; "магик" файлами.
61B7 CDF663		CALL	KEY	;ждем нажатия на любую клавишу.
61BA 010509		LD	BC, #905	;считываем первые 9 секторов
61BD 210080		LD	HL, #8000	;в память с адреса #8000
61C0 110000		LD	DE, 0	;нулевой дорожки.
61C3 CD133D		CALL	#3D13	;входная точка подпрограммы.
61C6 21409C		LD	HL, #9C40	;очищаем от
61C9 11419C		LD	DE, #9C41	; "мусора"
61CC 01F401		LD	BC, #01F4	;буфер для
61CF 3600		LD	(HL), 0	;хранения имен
61D1 EDB0		LDIR		; "магик" файлов.
61D3 CD3E62		CALL	NAME	;помещаем в этот буфер
				;имена "магик" файлов.
61D6 AF		XOR	A	;первоначально обнуляем ячейку
61D7 320463		LD	(NMAG), A	;числа "магик" файлов.
61DA 3E16		LD	A, #16	;устанавливаем
61DC D7		RST	#10	;позицию
61DD AF		XOR	A	;печати
61DE D7		RST	#10	;в левый
61DF AF		XOR	A	;верхний
61E0 D7		RST	#10	;угол.
61E1 210040		LD	HL, #4000	;очищаем
61E4 110140		LD	DE, #4001	;часть
61E7 01FF0F		LD	BC, #0FFF	;экрана
61EA 3600		LD	(HL), 0	;от

61EC	EDB0		LDIR		;надписи.
61EE	CD8B62		CALL	PRNAM	;вывод имен MAGIC-файлов.
61F1	3A0463		LD	A, (NMAG)	;проверяем, есть ли на диске
61F4	FE01		CP	1	;хотя бы один "магик" файл.
61F6	3008		JR	NC, YES	;если ДА, то продолжаем дальше.
61F8	CD0164		CALL	NO	;если нет, то пишем надпись
				;об отсутствии	"магик" файла.
61FB	CDF663		CALL	KEY	;ждем нажатия на любую клавишу и
61FE	18A8		JR	BEGIN	;переходим на начало программы.
6200	CDB062	YES	CALL	SELEC	;выбираем 1-ый "магик" файл.
6203	CD0563		CALL	TRSEC	;его расположение на диске.
6206	ED531363		LD	(TS1), DE	;запоминаем это значение.
620A	CDB663		CALL	BEEP	;сигнал о том, что файл выбран.
620D	CDB062		CALL	SELEC	;выбираем 2-ой "магик" файл.
6210	CD0563		CALL	TRSEC	;его расположение на диске.
6213	ED531563		LD	(TS2), DE	;запоминаем это значение.
6217	CDB663		CALL	BEEP	;сигнал о том, что файл выбран.
621A	CDB062		CALL	SELEC	;выбираем 3-ий "магик" файл.
621D	CD0563		CALL	TRSEC	;его расположение на диске.
6220	ED531763		LD	(TS3), DE	;запоминаем это значение.
6224	CDB663		CALL	BEEP	;сигнал о том, что файл выбран.
6227	CD6B0D		CALL	#0D6B	;очищаем экран.
622A	CDC463		CALL	MENU	;выводим меню поиска и
					;обрабатываем нажатие клавиш.
622D	CD6B0D		CALL	#0D6B	;очищаем экран.
6230	3E02		LD	A, 2	;после очистки открываем
6232	CD0116		CALL	#1601	;2-ой канал.
6235	CD1963		CALL	FIND	;процедура поиска.
6238	CDF663		CALL	KEY	;ждем нажатия на любую клавишу и
623B	C3A861		JP	BEGIN	;переходим на начало программы.
623E	3AE488	NAME	LD	A, (#88E4)	;в А заносим общее количество
					;файлов, имеющихся на диске.
6241	47		LD	B, A	;копируем это значение в В для
					;организации цикла.
6242	11409C		LD	DE, #9C40	;адрес буфера для хранения имен
					; "магик" файлов.
6245	21F87F		LD	HL, #7FF8	;первоначальное значение. Оно
					;должно быть на 8 меньше адреса,
					;куда считана нулевая дорожка.
6248	C5	NAME1	PUSH	BC	;сохраняем BC.
6249	011000		LD	BC, #10	;в BC - длина описателя файла.
624C	09		ADD	HL, BC	;прибавляем ее к HL. После этого
					;в HL находится адрес байта
					;спецификации файла.
624D	C1		POP	BC	;восстанавливаем BC.
624E	7E		LD	A, (HL)	;проверяем является ли файл
624F	FE43		CP	#43	;блоком кодов.
6251	2803		JR	Z, NAME2	;если ДА, то проверяем дальше.
6253	10F3		DJNZ	NAME1	;повторяем проверку столько раз,
					;сколько файлов на диске.
6255	C9		RET		;Все файлы проверены.
6256	E5	NAME2	PUSH	HL	;сохраняем HL.
6257	C5		PUSH	BC	;сохраняем BC.
6258	010500		LD	BC, 5	;в BC помещаем смещение между
					;байтом спецификации файла и
					;байтом размера файла в сект.
625B	09		ADD	HL, BC	;прибавляем его к HL. Теперь в
					;HL находится адрес байта разме-
					;ра файла в количестве секторов.
625C	C1		POP	BC	;восстанавливаем BC.

625D 7E		LD	A, (HL)	; проверяем длину файла,
625E FEC0		CP	#C0	; сравнивая ее с #C0.
6260 E1		POP	HL	; восстанавливаем HL.
6261 2803		JR	Z, NAME3	; если длина равна #C0, то это
				; MAGIC-файл; идем дальше.
6263 10E3		DJNZ	NAME1	; повторяем проверку столько раз,
				; сколько файлов на диске.
6265 C9		RET		; Проверены все файлы.
6266 C5	NAME3	PUSH	BC	; сохраняем BC.
6267 E5		PUSH	HL	; сохраняем HL.
6268 010800		LD	BC, 8	; уменьшаем HL на 8. После этого
626B ED42		SBC	HL, BC	; он указывает на начальный адрес
				; имени MAGIC-файла.
626D 011000		LD	BC, #10	; перебрасываем блок описателя
				; файла в адрес буфера для
6270 EDB0		LDIR		; хранения имен MAGIC-файлов.
6272 E1		POP	HL	; восстанавливаем HL.
6273 C1		POP	BC	; восстанавливаем BC.
6274 10D2		DJNZ	NAME1	; повторяем проверку столько раз,
				; сколько файлов на диске.
6276 C9		RET		; Проверены все файлы.
6277 3E01	PRN	LD	A, 1	; в A помещаем номер канала.
6279 CD0116		CALL	#1601	; открываем 1-ый канал.
627C 110C64		LD	DE, TXTPR	; адрес названия программы.
627F 018C00		LD	BC, #8C	; длина названия программы.
6282 CD3C20		CALL	#203C	; печатаем название программы.
6285 3E02		LD	A, 2	; в A помещаем номер канала.
6287 CD0116		CALL	#1601	; открываем 2-ой канал для даль-
				; нейшей печати в основной экран.
628A C9		RET		; возврат.
628B 11409C	PRNAM	LD	DE, #9C40	; в DE помещаем адрес, где лежат
				; имена "магик" файлов.
628E 1A	PRNA1	LD	A, (DE)	; если там находится 0,
628F B7		OR	A	; то есть имена закончились,
6290 C8		RET	Z	; то выходим.
6291 3E0D		LD	A, #0D	; смещаем позицию печати
6293 D7		RST	#10	; на строку ниже
6294 3E17		LD	A, #17	; с табуляцией
6296 D7		RST	#10	; в центр
6297 3E0C		LD	A, #0C	; строки.
6299 D7		RST	#10	; второй рестарт необходим для
629A D7		RST	#10	; правильной работы табуляции.
629B 010800		LD	BC, 8	; число символов в имени файла.
629E CD3C20		CALL	#203C	; печать имени.
62A1 3A0463		LD	A, (NMAG)	; считаем количество
62A4 3C		INC	A	; "магик" файлов
62A5 320463		LD	(NMAG), A	; на диске.
62A8 210800		LD	HL, 8	; помещаем
62AB 19		ADD	HL, DE	; в DE адрес
62AC EB		EX	DE, HL	; следующего имени и
62AD 18DF		JR	PRNA1	; идем на начало процедуры
				; с новым адресом.
62AF C9		RET		; возврат.
62B0 0E01	SELEC	LD	C, 1	; в регистре C будем хранить
				; порядковый номер "магик" файла,
				; выделенного на экране.
62B2 212C58		LD	HL, #582C	; первоначальный адрес атрибутов.
62B5 E5	SELO	PUSH	HL	; сохраняем HL.
62B6 C5		PUSH	BC	; сохраняем BC.
62B7 210058		LD	HL, #5800	; устанавливаем атрибуты

62BA 110158		LD	DE, #5801	; для невыделенных имен
62BD 017F02		LD	BC, #027F	; "магик" файлов,
62C0 3604		LD	(HL), 4	; не затрагивая надписей
62C2 EDB0		LDIR		; названия программы.
62C4 C1		POP	BC	; восстанавливаем BC.
62C5 E1		POP	HL	; восстанавливаем HL.
62C6 0608		LD	B, 8	; делаем цикл для
62C8 360E	SEL1	LD	(HL), #0E	; выделения другим цветом
62CA 23		INC	HL	; одного имени
62CB 10FB		DJNZ	SEL1	; "магик" файла.
62CD 110800		LD	DE, 8	; возвращаем
62D0 ED52		SBC	HL, DE	; в HL прежнее значение.
62D2 AF		XOR	A	; обнуляем системную переменную
62D3 32085C		LD	(#5C08), A	; кода последней нажатой клавиши.
62D6 3A085C	SEL2	LD	A, (#5C08)	; проверяем ее
62D9 FE61		CP	#61	; на нажатие клавиши "a".
62DB 2809		JR	Z, DOWN	; если "a" нажата, то идем
				; на соответствующую процедуру.
62DD FE71		CP	#71	; проверяем нажатие клавиши "q".
62DF 2814		JR	Z, UP	; если "q" нажата, то идем
				; на соответствующую процедуру.
62E1 FE0D		CP	#0D	; проверяем нажатие ENTER.
62E3 C8		RET	Z	; если ENTER нажата, то возврат.
62E4 18F0		JR	SEL2	; если ничего не нажато, то ждем.
62E6 3A0463	DOWN	LD	A, (NMAG)	; следим, чтобы выделение не
62E9 B9		CP	C	; пошло ниже последнего файла.
62EA 2002		JR	NZ, DOWN1	; если нет выхода за заданные
				; пределы, то переход на DOWN1.
62EC 18C7		JR	SEL0	; если есть выход за пределы, то
				; возвращение к началу процедуры.
62EE 0C	DOWN1	INC	C	; увеличиваем C на единицу.
62EF 112000		LD	DE, #20	; устанавливаем адрес атрибутов
62F2 19		ADD	HL, DE	; на одну строку ниже и
62F3 18C0		JR	SEL0	; идем на начало процедуры
				; с новым адресом атрибутов.
62F5 3E01	UP	LD	A, 1	; следим, чтобы не было выделения
62F7 B9		CP	C	; выше первого файла.
62F8 2002		JR	NZ, UP1	; если нет выхода за заданные
				; пределы, то переход на UP1.
62FA 18B9		JR	SEL0	; если есть выход за пределы, то
				; возвращение к началу процедуры.
62FC 0D	UP1	DEC	C	; уменьшаем C на единицу.
62FD 111F00		LD	DE, #1F	; устанавливаем адрес атрибутов
6300 ED52		SBC	HL, DE	; на одну строку выше и
6302 18B1		JR	SEL0	; идем на начало процедуры
				; с новым адресом атрибутов.
6304 00	NMAG	DEFB	0	; ячейка памяти для хранения общ.
				; количество MAGIC-файлов на диске.
6305 213E9C	TRSEC	LD	HL, #9C3E	; первоначально в HL помещаем число
				; на 2 меньшее адреса буфера
				; для хранения имен MAGIC-файлов.
6308 111000		LD	DE, #10	; длина описателя файла.
630B 41		LD	B, C	; в B заносим порядковый номер
				; выбранного "магик" файла.
630C 19	TRS1	ADD	HL, DE	; прибавляем длину описателя.
630D 10FD		DJNZ	TRS1	; количество сложений равно
				; порядковому номеру выбранного
				; MAGIC-файла. После этой процедуры
				; HL указывает на адрес, в
				; котором хранится дорожка и сектор

630F 5E		LD	E, (HL)	;выбранного MAGIC-файла.
6310 23		INC	HL	;помещаем в регистр E номер
6311 56		LD	D, (HL)	;сектора, а в регистр
				;D - номер дорожки выбранного
				; "магик" файла.
6312 C9		RET		;возврат.
6313 0000	TS1	DEFW	0	;ячейка для хранения местоположения
				;1-го MAGIC-файла.
6315 0000	TS2	DEFW	0	;ячейка для хранения местополо-
				;жения 2-го MAGIC-файла.
6317 0000	TS3	DEFW	0	;ячейка для хранения местополо-
				;жения 3-го MAGIC-файла.
6319 3E0C	FIND	LD	A, #0C	;сохраняем число дорожек, кото-
631B 320B64		LD	(NTR), A	;рые занимает MAGIC-файл.
631E ED5B1363	FIND0	LD	DE, (TS1)	;считываем
6322 010510		LD	BC, #1005	;одну дорожку
6325 210070		LD	HL, #7000	;1-го файла
6328 CD133D		CALL	#3D13	;в адрес #7000.
632B ED5B1563		LD	DE, (TS2)	;считываем
632F 010510		LD	BC, #1005	;одну дорожку
6332 210080		LD	HL, #8000	;2-го файла
6335 CD133D		CALL	#3D13	;в адрес #8000.
6338 ED5B1763		LD	DE, (TS3)	;считываем
633C 010510		LD	BC, #1005	;одну дорожку
633F 210090		LD	HL, #9000	;3-го файла
6342 CD133D		CALL	#3D13	;в адрес #9000.
6345 210070		LD	HL, #7000	;помещаем в HL адрес 1-го файла.
6348 110080		LD	DE, #8000	;помещаем в DE адрес 2-го файла.
634B DD210090		LD	IX, #9000	;помещаем в IX адрес 3-го файла.
634F 1A	FIND1	LD	A, (DE)	;в A значение байта 2-го файла.
6350 46		LD	B, (HL)	;в B значение байта 1-го файла.
6351 90		SUB	B	;сравниваем их.
6352 FEFF		CP	#FF	;здесь может быть также команда
				;CP 1 -это зависит от нашего
				;выбора в процедуре MENU.
6354 200A		JR	NZ, NEXT	;если отличаются не на единицу,
				;то переход к следующим значениям.
6356 1A		LD	A, (DE)	;заносим
6357 47		LD	B, A	;в B значение байта 2-го файла.
6358 DD7E00		LD	A, (IX+0)	;в A значение байта 3-го файла.
635B 90		SUB	B	;сравниваем их.
635C FEFF		CP	#FF	;здесь может быть также команда
				;CP 1 -это зависит от нашего
				;выбора в процедуре MENU.
635E 2829		JR	Z, PRNUM	;если и эти значения отличаются
				;на единицу, то переходим к про-
				;цедуре печати адреса.
6360 23	NEXT	INC	HL	;увеличиваем на единицу адрес
				;1-го файла.
6361 13		INC	DE	;увеличиваем на единицу адрес
				;2-го файла.
6362 DD23		INC	IX	;увеличиваем на единицу адрес
				;3-го файла.
6364 7C		LD	A, H	;проверяем не закончили ли мы
6365 FE80		CP	#80	;просмотр дорожки.
6367 20E6		JR	NZ, FIND1	;если не закончили, то возвра-
				;щаемся на начало процедуры с
				;новыми значениями.
6369 2A1363		LD	HL, (TS1)	;увеличиваем на единицу
636C 24		INC	H	;содержимое ячейки места

636D 221363		LD	(TS1), HL	;1-го "магик" файла.
6370 2A1563		LD	HL, (TS2)	;увеличиваем на единицу
6373 24		INC	H	;содержимое ячейки места
6374 221563		LD	(TS2), HL	;2-го "магик" файла.
6377 2A1763		LD	HL, (TS3)	;увеличиваем на единицу
637A 24		INC	H	;содержимое ячейки места
637B 221763		LD	(TS3), HL	;3-го "магик" файла.
637E 3A0B64		LD	A, (NTR)	;уменьш. на 1 содержимое ячейки,
6381 3D		DEC	A	;в которой храним число остав-
6382 320B64		LD	(NTR), A	;шихся для считывания дорожек.
6385 C8		RET	Z	;возврат если там находится 0,
				;т.е. уже прошли все 12 дорожек.
6386 1896		JR	FIND0	;переход на считывание в память
				;следующих дорожек.
6388 C9		RET		;возврат.
6389 E5	PRNUM	PUSH	HL	;сохраняем от "порчи" HL,
638A D5		PUSH	DE	;DE и
638B DDE5		PUSH	IX	;IX.
638D E5		PUSH	HL	;копируем через стек
638E C1		POP	BC	;содержимое HL в BC.
638F 59		LD	E, C	;сохраняем в E содержимое C.
				;в нем содержится младший байт
				;истинного адреса, изменившейся
				;на единицу ячейки памяти.
6390 78		LD	A, B	;в A старший байт адреса 2-го
				;файла. Он имеет значение в
				;пределах #70-#80.
6391 C690		ADD	A, #90	;для удобства прибавим к нему
				;#90, после этого там будет
				;число в пределах 0-#10.
6393 F5		PUSH	AF	;сохраняем регистр A.
6394 3A0B64		LD	A, (NTR)	;в B через A помещаем число ос-
6397 47		LD	B, A	;тавшихся дорожек для считыван.
6398 F1		POP	AF	;восстанавливаем регистр A.
6399 50		LD	D, B	;в D число оставшихся дорожек
				;для считывания.
639A 0610		LD	B, #10	;выполняем процедуру, равносиль-
639C 92	PRNU1	SUB	D	;ную выражению: A=A-#10*D. После
639D 10FD		DJNZ	PRNU1	;этого в A находится старший
				;байт истинного адреса, изменив-
				;шейся на единицу ячейки памяти.
639F 47		LD	B, A	;копируем в B регистр A.
63A0 4B		LD	C, E	;переписываем в C из E сохранен-
				;ное значение. Сейчас в BC нахо-
				;дится истинный адрес, изменив-
				;шейся на единицу ячейки памяти.
63A1 CD2F2D		CALL	#2D2F	;помещаем его на стек калькуля-
63A4 CDE32D		CALL	#2DE3	;тора и печатаем.
63A7 3E20		LD	A, #20	;после числа
63A9 D7		RST	#10	;помещаем три
63AA 3E20		LD	A, #20	;пробела для
63AC D7		RST	#10	;отделения его
63AD 3E20		LD	A, #20	;от следующего
63AF D7		RST	#10	;выводимого числа.
63B0 DDE1		POP	IX	;восстанавливаем IX.
63B2 D1		POP	DE	;восстанавливаем DE.
63B3 E1		POP	HL	;восстанавливаем HL.
63B4 18AA		JR	NEXT	;на проверку следующих значений.
63B6 E5	BEEP	PUSH	HL	;сохраняем HL.
63B7 D5		PUSH	DE	;сохраняем DE.

63B8	216400		LD	HL, #64	;высота звука.
63BB	119600		LD	DE, #96	;продолжительность его звучания.
63BE	CDB503		CALL	#03B5	;сама процедура.
63C1	D1		POP	DE	;восстанавливаем DE.
63C2	E1		POP	HL	;восстанавливаем HL.
63C3	C9		RET		;возврат.
63C4	3E02	MENU	LD	A, 2	;открываем
63C6	CD0116		CALL	#1601	;2-ой канал.
63C9	119864		LD	DE, TXMEN	;печатаем
63CC	012E00		LD	BC, #2E	;сообщение о том,
63CF	CD3C20		CALL	#203C	;что нам надо искать.
63D2	DD214F63		LD	IX, FIND1	;привязка к адресу процедуры по-
					иска FIND, для того чтобы там
					;вставить необходимые команды.
63D6	3EF7	MENU0	LD	A, #F7	;опрашиваем
63D8	DBFE		IN	A, (#FE)	;клавиши "1"-"5".
63DA	CB47		BIT	0, A	;нажата клавиша "1"?
63DC	2806		JR	Z, MENU1	;если да, то переход на
					;соответствующую процедуру.
63DE	CB4F		BIT	1, A	;нажата клавиша "2"?
63E0	280B		JR	Z, MENU2	;если да, то переход на
					;соответствующую процедуру.
63E2	18F2		JR	MENU0	;если не нажаты, то ждем.
63E4	DD360401	MENU1	LD	(IX+#04), 1	;подставляем в процедуру поиска
63E8	DD360E01		LD	(IX+#0E), 1	;в нужные ячейки единицу. После
					;этого мы имеем в нужных местах
63EC	C9		RET		;команду CP 1.
63ED	DD3604FF	MENU2	LD	(IX+#04), #FF	;подставляем в процедуру по-
63F1	DD360EFF		LD	(IX+#0E), #FF	;иска в нужные ячейки #FF.
					;После этого мы имеем в нужных
63F5	C9		RET		;местах команду CP #FF.
63F6	AF	KEY	XOR	A	;обнуляем системную переменную
63F7	32085C		LD	(#5C08), A	;кода последней нажатой клавиши.
63FA	3A085C	KEY1	LD	A, (#5C08)	;проверяем
63FD	B7		OR	A	;ее.
63FE	C0		RET	NZ	;если клавиша нажата, то возврат.
63FF	18F9		JR	KEY1	;если нет, то ждем.
6401	11C664	NO	LD	DE, TXTNO	;адрес сообщения.
6404	011D00		LD	BC, #1D	;его длина.
6407	CD3C20		CALL	#203C	;печать сообщения.
640A	C9		RET		;возврат.
640B	00	NTR	DEFB	0	;число дорожек, которые
					;необходимо считать.
640C	11061001	TXTPR	DEFB	#11, 6, #10, 1	;устанавливаем PAPER и INK
6410	46494E44		DEFM	"FIND THE INFINITE ADDRESS V 1.00"	
6414	20544845				
6418	20494E46				
641C	494E4954				
6420	45204144				
6424	44524553				
6428	53205620				
642C	312E3030				
6430	11011006		DEFB	#11, 1, #10, 6	;устанавливаем PAPER и INK
6434	20205752		DEFM	" WRITTEN BY SERGEY BORODAVKIN "	
6438	49545445				
643C	4E204259				
6440	20534552				
6444	47455920				
6448	424F524F				
644C	4441564B				


```

6450 494E2020
6454 20202020          DEFM      "      GRODNO-94 TEL. 33-36-53      "
6458 47524F44
645C 4E4F2D39
6460 34205445
6464 4C2E2033
6468 332D3336
646C 2D353320
6470 20202020
6474 11061001          DEFB      #11,6,#10,1 ;устанавливаем PAPER и INK
6478 20512C41          DEFM      " Q,A - SELECT      ENTER - АСЦЕПТ "
647C 202D2053
6480 454C4543
6484 54202020
6488 20454E54
648C 4552202D
6490 20414343
6494 45505420
6498 160A06      TXMEN  DEFB      #16,#0A,6      ;устанавливаем позицию печати.
649B 312E2046          DEFM      "1. FIND INC. ADDRESS"
649F 494E4420
64A3 494E432E
64A7 20414444
64AB 52455353
64AF 160C06          DEFB      #16,#0C,6      ;устанавливаем позицию печати.
64B2 322E2046          DEFM      "2. FIND DEC. ADDRESS"
64B6 494E4420
64BA 4445432E
64BE 20414444
64C2 52455353
64C6 160B04      TXTNO  DEFB      #16,#0B,4      ;устанавливаем позицию печати.
64C9 1002          DEFB      #10,2      ;устанавливаем цвет INK.
64CB 4D414749          DEFM      "MAGIC FILES NO PRESENT !"
64CF 43204649
64D3 4C455320
64D7 4E4F2050
64DB 52455345
64DF 4E542021
64E3 160B02      TXMAG  DEFB      #16,#0B,2      ;устанавливаем позицию печати.
64E6 1004          DEFB      #10,4      ;устанавливаем цвет INK.
64E8 494E5345          DEFM      "INSERT DISC WITH MAGIC FILES"
64EC 52542044
64F0 49534320
64F4 57495448
64F8 204D4147
64FC 49432046
6500 494C4553

```

После ассемблирования записываем полученный кодовый файл:

SAVE "ftia"CODE 25000,860

Загрузчик данного блока кодов может быть таким:

```

5 REM PROGRAM FTIA
10 PAPER 0 : BORDER 0 : INK 5 : CLEAR 24999
20 RANDOMIZE USR 15619 : REM : LOAD "ftia" CODE
30 RANDOMIZE USR 25000

```

Как работать с программой

Создайте по выше указанной методике на диске три MAGIC-файла. Запустите программу. Далее по запросу вставьте диск с этими MAGIC-файлами и нажмите любую клавишу. Вы увидите имена своих MAGIC-файлов в той последовательности, как они у вас записаны на диске: 1-ый файл сверху, 2-ой ниже и т.д. Переименовывать MAGIC-файлы не обязательно. Клавишами "Q" и "A" выберите 1-ый сброшенный файл, выбор фиксируйте клавишей "ENTER". Аналогично выберите 2-ой и 3-ий файл. После этого появиться меню поиска. Если вы нажмете "1", то будет осуществлен поиск адресов, увеличившихся на единицу. Если "2", то уменьшившихся на единицу.

Пример

В качестве примера рассмотрим поиск бессмертия в известной игре "1942". После запуска этой программы, в дисковод помещаем рабочий диск с достаточным количеством свободного пространства для записи MAGIC-файлов. Стартуем игру, выбирая в меню программы 4-ый пункт. Количество жизней 1-го игрока отображается в виде изображений самолётов в правом верхнем углу экрана. Первый раз сбрасываем игру MAGIC-кнопкой, когда у нас есть четыре самолёта. Для наглядности переименуем полученный MAGIC-файл в NAME1. Потом даём себя один раз убить. У нас остается три самолёта и после этого сбрасываем игру "волшебной" кнопкой во второй раз. Даём имя файлу NAME2. И последний, третий раз сбрасываем программу в момент, когда у нас на экране остаётся два самолёта. Меняем имя файла на NAME3. Подготовительная часть закончена. Теперь запустите предлагаемую программу FTIA и вставьте рабочий диск с полученными MAGIC-файлами. После нажатия на любую клавишу вы увидите эти файлы:

@ - имена предыдущих MAGIC-файлов, если таковые имеются.

NAME1 - 1-ый сброшенный файл.

NAME2 - 2-ой.

NAME3 - 3-ий.

Клавишами "Q" и "A" выберите 1-ый сброшенный файл и нажмите "ENTER". Аналогично поступаем со 2-ым, а затем и с 3-им файлом. После этого вам будет предложено меню поиска:

1. FIND INC. ADDRESS

2. FIND DEC. ADDRESS

Нажмите клавишу "2" и через некоторое время вы увидите долгожданный адрес, в котором хранится количество жизней: 52208.

Дальше уже дело техники. Находим в отладчике этот адрес:

```
52303      LD      A, 5
           LD      (52208), A
```

Заменяя число 5 на другое, вы получите необходимое вам число жизней.

Для того чтобы получить полное бессмертие, необходимо немного разбираться в машинном коде. Анализируя программу, устанавливаем, что необходимо исключить команду, которая уменьшает содержимое найденной ячейки памяти:

```
52471      DEC     (IX+8)
```

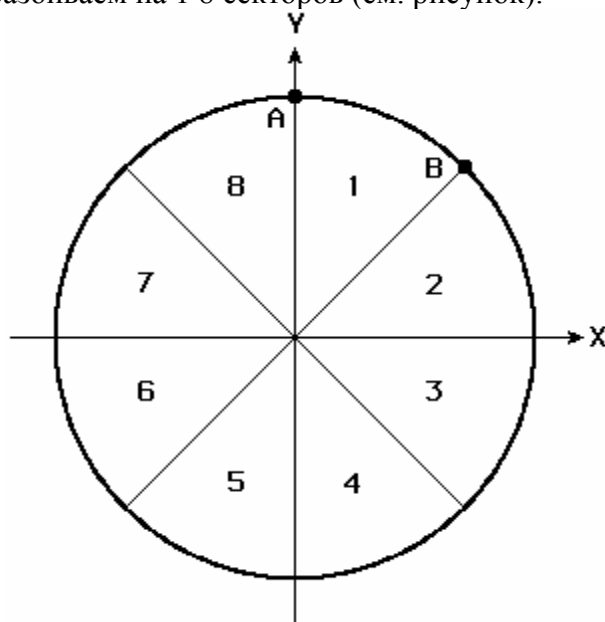
где в данный момент значение (IX+8) равно 52208.

Построение окружности и круга

© Михаил Усачев, г. Шахты, 1995.

Хочу предложить идейку, которую "подсунул" мне мой друг: алгоритм построения круга, при этом получается настоящий круг, а не многоугольник.

Алгоритм: весь круг разбиваем на 18 секторов (см. рисунок).



При движении от начала координат по оси X вправо, с приближением к абсциссе точки B, ордината все чаще уменьшается на единицу, но не больше, чем на единицу. Необходимо лишь узнать, на каком шаге уменьшить ординату на единицу, чтобы построить сектор. Результаты вычислений для первого сектора можно использовать для построения остальных секторов.

Я реализовал этот алгоритм в конкретной программе, результат работы которой впечатляет. Последовательность кругов с радиусами от 40 до 1 выстраивается примерно за полсекунды.

Но так как любую программу можно улучшить (ускорить), то мое предложение такое: попробуйте самостоятельно реализовать этот алгоритм. Свой же вариант я не привожу, чтобы не "загонять" программистов в его рамки.

Процедура построения окружности и круга

© Герасим Троеглазов, с. Гайтер (Хабаровский край), 1995.

Существует несколько методов рисования окружности, но для программ, использующих векторную графику, тригонометрический метод отпадает - очень медленный. Есть ещё уравнение окружности, его мы и рассмотрим.

Наберём программу:

```
9000 LET x=128: LET y=88: LET r=40
9010 FOR a=0 TO r
9020 LET x0=SQR (r*r-a*a)
9030 PLOT x-x0,y+a:PLOT x+x0,y+a
9040 PLOT x-x0,y-a:PLOT x+x0,y-a
9050 NEXT a
```

На скорость пока не будем обращать внимания (БЕЙСИК - ведь), а посмотрим на верх и низ окружности: вид портят расстояния между точками. Обойдем эту проблему так: разобьём окружность

на восемь секторов, каждый из которых отличается от других только положением на окружности. Рассчитаем один, и по этим же данным построим остальные:

```
9000 LET x=128: LET y=88: LET r=40
9010 FOR a=0 TO r*COS (PI/4)
9020 LET x0=SQR (r*r-a*a)
9030 PLOT x-x0,y+a:PLOT x+x0,y+a
9035 PLOT x-a,y+x0:PLOT x+a,y+x0
9040 PLOT x-x0,y-a:PLOT x+x0,y-a
9045 PLOT x-a,y-x0:PLOT x+a,y-x0
9050 NEXT a
```

С окружностью разобрались, возьмёмся за круг. Это дело куда проще предыдущего:

```
9000 LET x=128: LET y=88: LET r=40
9010 FOR a=0 TO r
9020 LET x0=SQR (r*r-a*a)
9030 PLOT x-x0,y+a: DRAW x0*2,0
9040 PLOT x-x0,y-a: DRAW x0*2,0
9050 NEXT a
```

Теперь примемся за АССЕМБЛЕР. Здесь надо решить две математические проблемы: квадратный корень и возведение в квадрат. Если использовать встроенный калькулятор, то скорость построения будет мало отличаться от Бейсика.

Одним из возможных решений является таблица квадратов чисел от нуля до максимального радиуса окружности. Если максимальный радиус принять за 127 (окружность большего размера на экране не поместится), на каждое число по два байта, итого: $128*2=256$ байт. Не так уж и много. Следующая программа заполнит эту таблицу:

```
9000 LET adr=49700
9010 FOR a=0 TO 127
9020 LET n=a*a
9030 LET n1=INT (n/256): LET n2=n-n1*256
9040 POKE adr,n2: POKE adr+1,n1
9050 LET adr=adr+2
9060 NEXT a
9070 SAVE "data" CODE 49700,256
```

Перейдем к программе на АССЕМБЛЕРЕ. Она расположена после таблицы, с адреса 50000. Точками входа являются: 50000 - рисование окружности и 50370 - рисование круга. Программа рисования окружности аналогична БЕЙСИК-программе, только части линии, выходящей за пределы экрана не отображаются. Круг же имеет некоторые отличия: рисование линий идет через любой заданный шаг по горизонтали, про шаг=0, черчение идет линия к линии. Кроме того, линия чертится не по точкам, а по байтам, что во много раз увеличивает скорость программы. А байты для заполнения берутся из шаблонов UDG-графики, поэтому круг получается сразу же заполненный графическим узором.

Переменные, при помощи которых происходит перенос данных в программу:

23297 (#5B01) - координата X окружности или круга (0...255).

23298 (#5B02) - координата Y окружности или круга (0...175).

23299 (#5B03) - радиус окружности или круга (0...127).

23301 (#5B05) - определяет символ UDG-графики, при этом 0 соответствует "А", 1 - "В" и т.д.

23302 (#5B06) - шаг линий по горизонтали при рисовании круга (0...радиус круга).

23303 (#5B07) - флаг наложения (0 - ИЛИ; 1 - ИСКЛЮЧАЮЩЕЕ ИЛИ).

Листинг программы.

```
1          ORG      50000
2 DATA    EQU      49700
```

```

3 KOORDX      EQU    #5B01
4 KOORDY      EQU    #5B02
5 RADIUS      EQU    #5B03
6 SYSTEM      EQU    #5B04
7 SYMBOL      EQU    #5B05
8 STEP        EQU    #5B06
9 FLAG        EQU    #5B07
10            ENT     $
11            LD      A, (RADIUS)
12            LD      (LC480), A
13            CALL    LC4B3
14            NOP
15            NOP
16            NOP
17            NOP
18            LD      B, A
19 LC35E       PUSH    BC
20            LD      A, (LC480)
21            LD      L, A
22            LD      H, #00
23            CALL    LC46A
24            EX      DE, HL
25            LD      L, B
26            LD      H, #00
27            CALL    LC46A
28            EX      DE, HL
29            AND     A
30            SBC     HL, DE
31            CALL    LC43A
32            DEC     L
33            DEC     L
34            LD      A, L
35            LD      (LC47F), A
36            LD      B, A
37            LD      A, (KOORDX)
38            SCF
39            SUB     B
40            JR      C, LC390
41            LD      E, A
42            POP     BC
43            PUSH    BC
44            LD      A, (KOORDY)
45            ADD     A, B
46            LD      C, E
47            LD      B, A
48            CALL    LC481
49 LC390       SCF
50            LD      A, (LC47F)
51            LD      B, A
52            LD      A, (KOORDX)

```

53		ADD	A, B
54		JR	C, LC3A7
55		LD	E, A
56		POP	BC
57		PUSH	BC
58		LD	A, (KOORDY)
59		ADD	A, B
60		LD	C, E
61		LD	B, A
62		CALL	LC481
63	LC3A7	SCF	
64		LD	A, (LC47F)
65		LD	B, A
66		LD	A, (KOORDX)
67		SUB	B
68		JR	C, LC3BE
69		LD	E, A
70		POP	BC
71		PUSH	BC
72		LD	A, (KOORDY)
73		SUB	B
74		LD	B, A
75		LD	C, E
76		CALL	LC481
77	LC3BE	SCF	
78		LD	A, (LC47F)
79		LD	B, A
80		LD	A, (KOORDX)
81		ADD	A, B
82		JR	C, LC3D5
83		LD	E, A
84		POP	BC
85		PUSH	BC
86		LD	A, (KOORDY)
87		SUB	B
88		LD	B, A
89		LD	C, E
90		CALL	LC481
91	LC3D5	SCF	
92		POP	BC
93		PUSH	BC
94		LD	A, (KOORDX)
95		SUB	B
96		JR	C, LC3EC
97		LD	E, A
98		LD	A, (LC47F)
99		LD	B, A
100		LD	A, (KOORDY)
101		ADD	A, B
102		LD	B, A

103	LD	C, E
104	CALL	LC481
105	LC3EC	SCF
106	POP	BC
107	PUSH	BC
108	LD	A, (KOORDX)
109	SUB	B
110	JR	C, LC403
111	LD	E, A
112	LD	A, (LC47F)
113	LD	B, A
114	LD	A, (KOORDY)
115	SUB	B
116	LD	B, A
117	LD	C, E
118	CALL	LC481
119	LC403	SCF
120	POP	BC
121	PUSH	BC
122	LD	A, (KOORDX)
123	ADD	A, B
124	LD	E, A
125	JR	C, LC41A
126	LD	A, (LC47F)
127	LD	B, A
128	LD	A, (KOORDY)
129	ADD	A, B
130	LD	B, A
131	LD	C, E
132	CALL	LC481
133	LC41A	SCF
134	POP	BC
135	PUSH	BC
136	LD	A, (KOORDX)
137	ADD	A, B
138	LD	E, A
139	JR	C, LC431
140	LD	A, (LC47F)
141	LD	B, A
142	LD	A, (KOORDY)
143	SUB	B
144	LD	B, A
145	LD	C, E
146	CALL	LC481
147	LC431	POP
148		DEC
149		LD
150		CP
151		JP
152		RET

153	LC43A	PUSH	BC
154		PUSH	DE
155		PUSH	IX
156		PUSH	AF
157		LD	IX, DATA
158		LD	B, #80
159		LD	C, #00
160	LC447	LD	E, (IX+0)
161		LD	D, (IX+1)
162		INC	IX
163		INC	IX
164		LD	A, B
165		CP	#00
166		JR	Z, LC45F
167		DEC	B
168		PUSH	HL
169		SCF	
170		SBC	HL, DE
171		POP	HL
172		INC	C
173		JR	NC, LC447
174	LC45F	LD	A, C
175		NOP	
176		LD	L, A
177		LD	H, #00
178		POP	AF
179		POP	IX
180		POP	DE
181		POP	BC
182		RET	
183	LC46A	PUSH	IX
184		PUSH	DE
185		ADD	HL, HL
186		EX	DE, HL
187		LD	IX, DATA
188		ADD	IX, DE
189		LD	L, (IX+0)
190		LD	H, (IX+1)
191		POP	DE
192		POP	IX
193		RET	
194	LC47F	DEFB	#47
195	LC480	DEFB	#48
196	LC481	LD	D, C
197		LD	E, B
198		LD	A, B
199		CP	#B0
200		RET	NC
201		LD	A, #AF
202		SUB	E


```

203      LD      E, A
204      AND     A
205      RRA
206      SCF
207      RRA
208      AND     A
209      RRA
210      XOR     E
211      AND     #F8
212      XOR     E
213      LD      H, A
214      LD      A, D
215      RLCA
216      RLCA
217      RLCA
218      XOR     E
219      AND     #C7
220      XOR     E
221      RLCA
222      RLCA
223      LD      L, A
224      LD      A, D
225      AND     #07
226      LD      B, A
227      INC     B
228      LD      A, #FE
229 LC4A8  RRCA
230      DJNZ    LC4A8
231      LD      B, #FF
232      XOR     B
233      LD      B, A
234      LD      A, (HL)
235      OR      B
236      LD      (HL), A
237      RET
238 LC4B3  CALL  #2D28
239      RST     #28
240      DEFB    #A2, #A3, #04
241      DEFB    #1F, #04, #38
242      CALL    #2DD5
243      LD      B, A
244      RET
245 ; * Точка входа для круга *
246      CALL    LC5DF
247      LD      A, (RADIUS)
248      LD      B, A
249 LC4C9  PUSH    BC
250      LD      L, B
251      LD      H, #00
252      CALL    LC46A

```

253	EX	DE, HL
254	LD	A, (RADIUS)
255	LD	L, A
256	LD	H, #00
257	CALL	LC46A
258	AND	A
259	SBC	HL, DE
260	CALL	LC43A
261	LD	A, L
262	LD	(LC52E), A
263	LD	A, (LC52E)
264	LD	B, A
265	LD	A, (KOORDX)
266	SUB	B
267	CALL	C, LC51C
268	LD	E, A
269	LD	(LC52C), A
270	POP	BC
271	PUSH	BC
272	LD	A, (KOORDY)
273	SUB	B
274	LD	(LC52D), A
275	LD	C, E
276	LD	B, A
277	CALL	LC531
278	POP	BC
279	PUSH	BC
280	LD	A, (KOORDY)
281	ADD	A, B
282	LD	B, A
283	LD	A, (LC52C)
284	LD	C, A
285	INC	B
286	CALL	LC531
287	POP	BC
288	LD	A, (STEP)
289	LD	C, A
290	LD	A, B
291	INC	C
292	SUB	C
293	LD	B, A
294	JR	NC, LC4C9
295	RET	
296	LC51C	NEG
297	LD	D, A
298	LD	A, (LC52E)
299	ADD	A, A
300	SUB	D
301	SRL	A
302	LD	(LC52E), A

303	LD	A, #00
304	RET	
305	LC52C	DEFB #64
306	LC52D	DEFB #55
307	LC52E	DEFB #1C
308	LC52F	DEFB #04, #00
309	LC531	LD A, B
310	CP	#AF
311	RET	NC
312	LD	D, C
313	LD	E, B
314	LD	A, #AF
315	SUB	E
316	LD	E, A
317	AND	A
318	RRA	
319	SCF	
320	RRA	
321	AND	A
322	RRA	
323	XOR	E
324	AND	#F8
325	XOR	E
326	LD	H, A
327	LD	A, D
328	RLCA	
329	RLCA	
330	RLCA	
331	XOR	E
332	AND	#C7
333	XOR	E
334	RLCA	
335	RLCA	
336	LD	L, A
337	PUSH	HL
338	LD	BC, #57FF
339	AND	A
340	SBC	HL, BC
341	JR	C, LC55C
342	POP	HL
343	RET	
344	LC55C	POP HL
345	LD	A, H
346	AND	#07
347	LD	IX, (#5C7B)
348	LD	B, #00
349	LD	C, A
350	ADD	IX, BC
351	LD	A, (SYMBOL)
352	ADD	A, A

```

353      ADD    A,A
354      ADD    A,A
355      LD     C,A
356      ADD    IX,BC
357      LD     A,(IX+0)
358      LD     (SYSTEM),A
359      LD     A,D
360      AND     #07
361      LD     B,A
362      INC     B
363      LD     (LC52F),A
364      LD     A,(LC52E)
365      ADD    A,A
366      LD     D,A
367      LD     A,(SYSTEM)
368      LD     E,#FE
369 LC58A    RRC     E
370      AND     E
371      DJNZ   LC58A
372 LC58F    XOR     (HL)
373      LD     (HL),A
374      LD     A,(LC52F)
375      LD     B,A
376      LD     A,#08
377      SUB     B
378      LD     B,A
379      LD     A,D
380      SUB     B
381      LD     D,A
382      JR     C,LC5CE
383      INC     HL
384      LD     A,L
385      AND     #1F
386      SUB     #01
387      RET     C
388 LC5A5    LD     A,D
389      SUB     #08
390      JR     C,LC5B9
391      LD     D,A
392      LD     A,(SYSTEM)
393 LC5AE    XOR     (HL)
394      LD     (HL),A
395      INC     HL
396      LD     A,L
397      AND     #1F
398      SUB     #01
399      RET     C
400      JR     LC5A5
401 LC5B9    LD     A,D
402      AND     #07

```

```

403      LD      B,A
404      LD      A,#08
405      SUB     B
406      LD      B,A
407      LD      E,#FE
408      LD      A,(SYSTEM)
409 LC5C6    AND     E
410      RLC     E
411      DJNZ    LC5C6
412 LC5CB    XOR     (HL)
413      LD      (HL),A
414      RET
415 LC5CE    LD      A,D
416      ADD     A,#08
417      LD      D,A
418      AND     #07
419      LD      B,A
420      LD      E,#FE
421      LD      A,(HL)
422 LC5D8    AND     E
423      RLC     E
424      DJNZ    LC5D8
425      LD      (HL),A
426      RET
427 LC5DF    LD      A,(FLAG)
428      CP      #00
429      JR      Z,LC5EB
430      CP      #01
431      JR      Z,LC5F7
432      RET
433 LC5EB    LD      A,#B6
434 LC5ED    LD      (LC58F),A
435      LD      (LC5AE),A
436      LD      (LC5CB),A
437      RET
438 LC5F7    LD      A,#AE
439      JR      LC5ED

```

Некоторые комментарии к программе.

Строка 11 - радиус заносится в А и переход на расчет формулы $A=R*\cos(\pi/4)$.

Строка 18 - в В счётчик точек сектора окружности.

Строка 20 - начало расчета формулы уравнения окружности $B=\sqrt{R^2-A^2}$.

Строка 40 - PLOT X0-B,Y0+A. Строка 54 - PLOT X0+B,Y0+A.

Строка 68 - PLOT X0-B,Y0-A. Строка 82 - PLOT X0+B,Y0-A.

Строка 96 - PLOT X0-A,Y0+B. Строка 110- PLOT X0-A,Y0-B.

Строка 125 - PLOT X0+A,Y0+B. Строка 139- PLOT X0+A,Y0-B.

Строка 147 - переход к следующей точке сектора, если $B>0$, иначе - выход.

Строка 153 - расчёт квадратного корня из Н. Вычисление идёт по таблице квадратов чисел, расположенной по адресу 49700 и заполненную сформированной выше последовательностью (файл "data" CODE). Результат в регистре Н.

Строка 183 - возведение числа в Н в квадрат, также по таблице. Результат в Н.

Строка 196 - печать точки на экране. Программа взята из книги "Элементарная графика".

Строка 239 - $A=R*\cos(\pi/4)$.

Строка 246 - точка входа для рисования круга; следует переход в подпрограмму установки режима наложения.

Строка 247 - расчёт уравнения окружности.

Строка 263 - вычисление левого края отрезка.

Строка 272 - отрезок внизу $Y=Y_0+A$.

Строка 280 - отрезок вверху $Y=Y_0+A$.

Строка 288 - $A=A+\text{шаг}$.

Строка 294 - переход к следующей линии.

Строка 296 - расчёт длины отрезка при выходе за левый край отрезка.

Строка 305 - рабочие переменные.

Строка 309 - черчение отрезка.

Строка 312 - расчёт адреса в экране.

Строка 338 - расчёт адреса шаблона.

Строка 358 - зануление крайних левых битов в байте, взятом из шаблона в левом конце отрезка.

Строка 374 - средняя часть отрезка; байты устанавливаются полностью.

Строка 392 - правый байт отрезка; зануление правых битов.

Строка 415 - Если длина отрезка меньше 8 и он находится в одном байте экрана, то происходит зануление справа зануленного слева байта.

Строка 427 - Установка режима наложения (ИЛИ-ИСКЛЮЧАЮЩЕЕ ИЛИ).

Длина блока кодов - 683 байта:

C350	3A	03	5B	32	80	C4	CD	B3	A1
C358	C4	00	00	00	00	47	C5	3A	25
C360	80	C4	6F	26	00	CD	6A	C4	F7
C368	EB	68	26	00	CD	6A	C4	EB	8A
C370	A7	ED	52	CD	3A	C4	2D	2D	3E
C378	7D	32	7F	C4	47	3A	01	5B	0A
C380	37	90	38	0C	5F	C1	C5	3A	6D
C388	02	5B	80	4B	47	CD	81	C4	CC
C390	37	3A	7F	C4	47	3A	01	5B	E4
C398	80	38	0C	5F	C1	C5	3A	02	40
C3A0	5B	80	4B	47	CD	81	C4	37	19
C3A8	3A	7F	C4	47	3A	01	5B	90	55
C3B0	38	0C	5F	C1	C5	3A	02	5B	33
C3B8	90	47	4B	CD	81	C4	37	3A	20
C3C0	7F	C4	47	3A	01	5B	80	38	5B
C3C8	0C	5F	C1	C5	3A	02	5B	90	A3
C3D0	47	4B	CD	81	C4	37	C1	C5	F4
C3D8	3A	01	5B	90	38	0E	5F	3A	A0
C3E0	7F	C4	47	3A	02	5B	80	47	8B
C3E8	4B	CD	81	C4	37	C1	C5	3A	FF
C3F0	01	5B	90	38	0E	5F	3A	7F	FD
C3F8	C4	47	3A	02	5B	90	47	4B	7F
C400	CD	81	C4	37	C1	C5	3A	01	CE
C408	5B	80	5F	38	0D	3A	7F	C4	C8
C410	47	3A	02	5B	80	47	4B	CD	91
C418	81	C4	37	C1	C5	3A	01	5B	74
C420	80	5F	38	0D	3A	7F	C4	47	CC

C428	3A	02	5B	90	47	4B	CD	81	F3
C430	C4	C1	05	78	FE	FF	C2	5E	13
C438	C3	C9	C5	D5	DD	E5	F5	DD	B6
C440	21	24	C2	06	80	0E	00	DD	7C
C448	5E	00	DD	56	01	DD	23	DD	7B
C450	23	78	FE	00	28	09	05	E5	C8
C458	37	ED	52	E1	0C	30	E8	79	10
C460	00	6F	26	00	F1	DD	E1	D1	39
C468	C1	C9	DD	E5	D5	29	EB	DD	3E
C470	21	24	C2	DD	19	DD	6E	00	7C
C478	DD	66	01	D1	DD	E1	C9	47	1F
C480	48	51	58	78	FE	B0	D0	3E	69
C488	AF	93	5F	A7	1F	37	1F	A7	B0
C490	1F	AB	E6	F8	AB	67	7A	07	8F
C498	07	07	AB	E6	C7	AB	07	07	7B
C4A0	6F	7A	E6	07	47	04	3E	FE	C1
C4A8	0F	10	FD	06	FF	A8	47	7E	FA
C4B0	B0	77	C9	CD	28	2D	EF	A2	17
C4B8	A3	04	1F	04	38	CD	D5	2D	4D
C4C0	47	C9	CD	DF	C5	3A	03	5B	9D
C4C8	47	C5	68	26	00	CD	6A	C4	21
C4D0	EB	3A	03	5B	6F	26	00	CD	79
C4D8	6A	C4	A7	ED	52	CD	3A	C4	7B
C4E0	7D	32	2E	C5	3A	2E	C5	47	BA
C4E8	3A	01	5B	90	DC	1C	C5	5F	EE
C4F0	32	2C	C5	C1	C5	3A	02	5B	F4
C4F8	90	32	2D	C5	4B	47	CD	31	00
C500	C5	C1	C5	3A	02	5B	80	47	6E
C508	3A	2C	C5	4F	04	CD	31	C5	0E
C510	C1	3A	06	5B	4F	78	0C	91	95
C518	47	30	AE	C9	ED	44	57	3A	8D
C520	2E	C5	87	92	CB	3F	32	2E	5B
C528	C5	3E	00	C9	64	55	1C	04	92
C530	00	78	FE	AF	D0	51	58	3E	D1
C538	AF	93	5F	A7	1F	37	1F	A7	61
C540	1F	AB	E6	F8	AB	67	7A	07	40
C548	07	07	AB	E6	C7	AB	07	07	2C
C550	6F	E5	01	FF	57	A7	ED	42	96
C558	38	02	E1	C9	E1	7C	E6	07	4B
C560	DD	2A	7B	5C	06	00	4F	DD	35
C568	09	3A	05	5B	87	87	87	4F	B4
C570	DD	09	DD	7E	00	32	04	5B	07
C578	7A	E6	07	47	04	32	2F	C5	15
C580	3A	2E	C5	87	57	3A	04	5B	E9
C588	1E	FE	CB	0B	A3	10	FB	AE	9B
C590	77	3A	2F	C5	47	3E	08	90	17
C598	47	7A	90	57	38	30	23	7D	0D
C5A0	E6	1F	D6	01	D8	7A	D6	08	71
C5A8	38	0F	57	3A	04	5B	AE	77	C9
C5B0	23	7D	E6	1F	D6	01	D8	18	E1

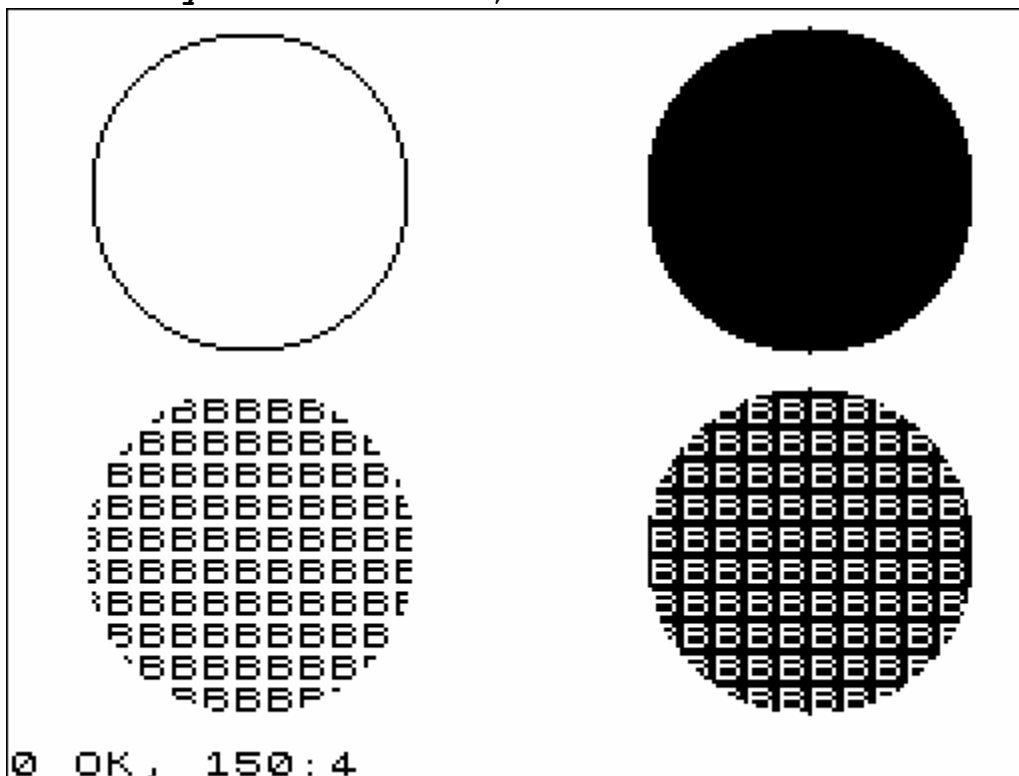
C5B8	EC	7A	E6	07	47	3E	08	90	ED
C5C0	47	1E	FE	3A	04	5B	A3	CB	EF
C5C8	03	10	FB	AE	77	C9	7A	C6	C9
C5D0	08	57	E6	07	47	1E	FE	7E	C2
C5D8	A3	CB	03	10	FB	77	C9	3A	93
C5E0	07	5B	FE	00	28	05	FE	01	31
C5E8	28	0D	C9	3E	B6	32	8F	C5	25
C5F0	32	AE	C5	32	CB	C5	C9	3E	23
C5F8	AE	18	F2	00	00	00	00	00	75

Демонстрационная БЕЙСИК-программа (автостарт со 2 строки):

```

1 GO TO 100
2 BORDER 7: PAPER 7: INK 0: CLEAR 49699:LOAD "data" CODE 49700
3 LOAD "circle" CODE 50000
4 FOR u=USR "a" TO USR "a"+7: POKE u,255: NEXT u: RUN
40 POKE 23297,x: POKE 23298,y: POKE 23299,r
45 RANDOMIZE USR 50000
49 RETURN
50 POKE 23297,x: POKE 23298,y: POKE 23299,r
55 RANDOMIZE USR 50370
59 RETURN
100 POKE 23302,0: POKE 23303,1: LET r=40
110 LET x=60: LET y=130: GO SUB 40
120 LET x=200: LET y=130: POKE 23301,0: GO SUB 50
130 LET x=60: LET y=40: POKE 23301,1: GO SUB 50
140 LET x=200: LET y=40: POKE 23301,0: GO SUB 50
150 LET x=200: LET y=40: POKE 23301,1: GO SUB 50

```



Результат работы демонстрационной БЕЙСИК-программы Вы можете увидеть на рисунке.

На этом заканчиваю. Может, кто и найдет другой способ черчения окружности и круга, более быстрый, ведь в "ЭЛИТЕ" же нашли!

Нестандартная загрузка

© Павел Старков, г. Красноярск, 1995.

В ZX-РЕВЮ-94, № 6 была напечатана статья о нестандартной загрузке. В дополнение к ней я решил выслать Вам две процедуры. Первая - загружает рисунок последовательно, линия за линией, а затем загружаются атрибуты, но не как обычно, сверху вниз, а вертикальными столбцами слева направо. Процедуры построены с использованием подпрограммы ПЗУ с точкой входа #05A9.

Замечу, что предварительно закодированный экран имеет длину такую же, как и обычный стандартный экран - 6912 байт.

```

1      ORG      50000
2      ENT      $
3      LD       IX,16384
4      LD       DE,32
5      XOR      A
6      SUB      1
7      EX       AF,AF'
8      DI
9      PUSH     IX
10     CALL     #0562
11 L1    POP     IX
12     DEFB     #DD
13     INC      H
14     DEFB     #DD
15     LD       A,H
16     AND      7
17     JR       NZ,P1
18     DEFB     #DD
19     LD       A,L
20     ADD      A,32
21     DEFB     #DD
22     LD       L,A
23     JR       C,P1
24     DEFB     #DD
25     LD       A,H
26     SUB      8
27     DEFB     #DD
28     LD       H,A
29 P1    LD       DE,32
30     PUSH     IX
31     CALL     #05A9
32     DEC      E
33     INC      E
34     JR       NZ,ERR
35     DEFB     #DD
36     LD       A,H
37     CP       88
38     JR       C,L1
39     POP      AF
40     LD       B,32
41 M2    PUSH     BC

```

```

42      PUSH    IX
43      LD      B, 24
44 M3    PUSH    BC
45      INC     DE
46      CALL    #05A9
47      LD      BC, 31
48      ADD     IX, BC
49      POP     BC
50      DJNZ    M3
51      POP     IX
52      INC     IX
53      POP     BC
54      DJNZ    M2
55      SCF
56      JP      #053F
57 ERR   POP     IX
58      OR      A
59      JP      #053F

```

При завершении процедуры загрузки флаг C будет установлен, если загрузка прошла успешно, или сброшен, если была зафиксирована ошибка.

Длина процедуры = 97 байтов:

C350	DD	21	00	40	11	20	00	AF	31
C358	D6	01	08	F3	DD	E5	CD	62	DE
C360	05	DD	E1	DD	24	DD	7C	E6	26
C368	07	20	0E	DD	7D	C6	20	DD	7D
C370	6F	38	06	DD	7C	D6	08	DD	F4
C378	67	11	20	00	DD	E5	CD	A9	0B
C380	05	1D	1C	20	26	DD	7C	FE	1E
C388	58	38	D6	F1	06	20	C5	DD	6A
C390	E5	06	18	C5	13	CD	A9	05	A9
C398	01	1F	00	DD	09	C1	10	F3	25
C3A0	DD	E1	DD	23	C1	10	E7	37	10
C3A8	C3	3F	05	DD	E1	B7	C3	3F	E9
C3B0	05	00	00	00	00	00	00	00	78

Для кодирования экрана под нее предназначена программа-шифратор на Бейсике:

```

1 CLEAR 39999: FOR x=1 TO 60: READ a: POKE 49999+x,a: NEXT x
10 LOAD "" SCREEN$
15 RANDOMIZE USR 5E4
20 SAVE "scr" CODE 40000,6912
30 GO TO 10
100 DATA 17,64,156,33,0,64,6,192,197,229,1,32,0,237,176
110 DATA 225,36,124,230,7,32,10,125,198,32,111,56,4,124,214,8
120 DATA 103,193,16,229,33,0,88,6,32,197,229,6,24,197,237,160
130 DATA 1,31,0,9,193,16,246,225,35,193,16,237,201

```

Дизассемблер кодового блока программы-шифратора:

```

1      ORG      50000
2      ENT      $
3      LD      DE, #9C40
4      LD      HL, #4000
5      LD      B, #C0

```

```

6 LC358    PUSH    BC
7          PUSH    HL
8          LD      BC, #0020
9          LDIR
10         POP     HL
11         INC     H
12         LD      A, H
13         AND     #07
14         JR      NZ, LC370
15         LD      A, L
16         ADD     A, #20
17         LD      L, A
18         JR      C, LC370
19         LD      A, H
20         SUB     #08
21         LD      H, A
22 LC370    POP     BC
23         DJNZ    LC358
24         LD      HL, #5800
25         LD      B, #20
26 LC378    PUSH    BC
27         PUSH    HL
28         LD      B, #18
29 LC37C    PUSH    BC
30         LDI
31         LD      BC, #001F
32         ADD     HL, BC
33         POP     BC
34         DJNZ    LC37C
35         POP     HL
36         INC     HL
37         POP     BC
38         DJNZ    LC378
39         RET

```

Вторая процедура загружает рисунок по прямоугольникам 6x8 знакомест (всего их - 16). Загружаются они в хаотичном (заранее определяемом) порядке. После загрузки каждого прямоугольника, сразу же загружаются его атрибуты.

```

1          ORG     50000
2          ENT     $
3          CALL    LD_FL
4          RET     Z
5          RST     8
6          DEFB    26
7 LD_FL    LD      HL, LOAD1
8          LD      (CAL+1), HL
9          XOR     A
10         SUB     1
11         EX      AF, AF'
12         DI
13         LD      HL, DATA

```

```

14      LD      B,16
15 L1    PUSH    BC
16      LD      A,(HL)
17      DEFB    #DD
18      LD      L,A
19      INC     HL
20      LD      A,(HL)
21      DEFB    #DD
22      LD      H,A
23      INC     HL
24      CALL    LOAD
25      POP     BC
26      DJNZ    L1
27      XOR     A
28      EXX
29 EXIT   LD      HL,10072
30      EXX
31      JP      #053F
32 LOAD   LD      B,48
33      PUSH    IX
34 L2     PUSH    IX
35      EXX
36      LD      DE,8
37 CAL    CALL    LOAD1
38      INC     E
39      DEC     E
40      JR      NZ,ERR1
41      EXX
42      POP     IX
43      DEFB    #DD
44      INC     H
45      DEFB    #DD
46      LD      A,H
47      AND     7
48      JR      NZ,L3
49      DEFB    #DD
50      LD      A,L
51      ADD     A,32
52      DEFB    #DD
53      LD      L,A
54      JR      C,L3
55      DEFB    #DD
56      LD      A,H
57      SUB     8
58      DEFB    #DD
59      LD      H,A
60 L3     DJNZ    L2
61      POP     IX
62      DEFB    #DD
63      LD      A,H

```

```

64      RRCA
65      RRCA
66      RRCA
67      AND    #03
68      OR     #58
69      DEFB   #DD
70      LD     H,A
71      LD     B,6
72 L4    EXX
73      LD     DE,8
74      CALL   #05A9
75      INC    E
76      DEC    E
77      JR     NZ,ERR2
78      EXX
79      LD     DE,24
80      ADD    IX,DE
81      DJNZ   L4
82      RET
83 LOAD1 CALL   #0562
84      LD     IX,#05A9
85      LD     (CAL+1),IX
86      RET
87 ERR1  POP    BC
88      POP    BC
89 ERR2  POP    BC
90      POP    BC
91      JR     EXIT
92 DATA DEFW   #4000,#4890
93      DEFW   #4880,#5058
94      DEFW   #4010,#40C8
95      DEFW   #5048,#40D8
96      DEFW   #5050,#40C0
97      DEFW   #5040,#4888
98      DEFW   #4008,#4018
99      DEFW   #40D0,#4898

```

При выходе из программы флаг Z установлен, если не было ошибки и сброшен, если была.

Длина процедуры = 175 байтов:

C350	CD	56	C3	C8	CF	1A	21	CD	98
C358	C3	22	89	C3	AF	D6	01	08	DA
C360	F3	21	DF	C3	06	10	C5	7E	32
C368	DD	6F	23	7E	DD	67	23	CD	4C
C370	7E	C3	C1	10	F1	AF	D9	21	DF
C378	58	27	D9	C3	3F	05	06	30	D0
C380	DD	E5	DD	E5	D9	11	08	00	B9
C388	CD	CD	C3	1C	1D	20	4A	D9	24
C390	DD	E1	DD	24	DD	7C	E6	07	58
C398	20	0E	DD	7D	C6	20	DD	6F	15
C3A0	38	06	DD	7C	D6	08	DD	67	1C
C3A8	10	D8	DD	E1	DD	7C	0F	0F	88

C3B0	0F	E6	03	F6	58	DD	67	06	03
C3B8	06	D9	11	08	00	CD	A9	05	EE
C3C0	1C	1D	20	17	D9	11	18	00	F5
C3C8	DD	19	10	ED	C9	CD	62	05	7B
C3D0	DD	21	A9	05	DD	22	89	C3	8A
C3D8	C9	C1	C1	C1	C1	18	98	00	18
C3E0	40	90	48	80	48	58	50	10	3B
C3E8	40	C8	40	48	50	D8	40	50	F3
C3F0	50	C0	40	40	50	88	48	08	6B
C3F8	40	18	40	D0	40	98	48	00	43

Бейсик-шифратор для кодирования экрана:

```

1 CLEAR 39999: FOR x=0 TO 132: READ a: POKE 5E4+x,a: NEXT x
10 LOAD ""SCREEN$
15 RANDOMIZE USR 5E4
20 SAVE "scr"CODE 40000,6912
30 GO TO 10
100 DATA 17,64,156,6,16,33,181,195,197,126,221,111,35,126
110 DATA 221,103,35,229,205,106,195,225,193,16,239,201
120 DATA 6,48,221,229,197,221,229,221,229,225,1,8,0,237,176
130 DATA 221,225,221,36,221,124,230,7,32,14,221,125,198,32
140 DATA 221,111,56,6,221,124,214,8,221,103,193,16,218,221,225
150 DATA 221,124,15,15,15,230,3,246,88,221,103,6,6,197,221
160 DATA 229,225,1,8,0,237,176,1,32,0,221,9,193,16,239,201
170 DATA 0,64,144,72,128,72,88,80,16,64,200,64,72,80,216,64
180 DATA 80,80,192,64,64,80,136,72,8,64,24,64,208,64,152,72

```

Дизассемблер кодового блока шифратора:

```

1      ORG      50000
2      ENT      $
3      LD       DE,#9C40
4      LD       B,#10
5      LD       HL,LC3B5
6 LC358  PUSH    BC
7      LD       A,(HL)
8      DEFB     #DD
9      LD       L,A
10     INC      HL
11     LD       A,(HL)
12     DEFB     #DD
13     LD       H,A
14     INC      HL
15     PUSH     HL
16     CALL     LC36A
17     POP      HL
18     POP      BC
19     DJNZ     LC358
20     RET
21 LC36A  LD      B,#30
22     PUSH     IX
23 LC36E  PUSH     BC
24     PUSH     IX

```

25		PUSH	IX
26		POP	HL
27		LD	BC, #0008
28		LDIR	
29		POP	IX
30		DEFB	#DD
31		INC	H
32		DEFB	#DD
33		LD	A, H
34		AND	#07
35		JR	NZ, LC391
36		DEFB	#DD
37		LD	A, L
38		ADD	A, #20
39		DEFB	#DD
40		LD	L, A
41		JR	C, LC391
42		DEFB	#DD
43		LD	A, H
44		SUB	#08
45		DEFB	#DD
46		LD	H, A
47	LC391	POP	BC
48		DJNZ	LC36E
49		POP	IX
50		DEFB	#DD
51		LD	A, H
52		RRCA	
53		RRCA	
54		RRCA	
55		AND	#03
56		OR	#58
57		DEFB	#DD
58		LD	H, A
59		LD	B, #06
60	LC3A3	PUSH	BC
61		PUSH	IX
62		POP	HL
63		LD	BC, #0008
64		LDIR	
65		LD	BC, #0020
66		ADD	IX, BC
67		POP	BC
68		DJNZ	LC3A3
69		RET	
70	LC3B5	DEFW	#4000, #4890
71		DEFW	#4880, #5058
72		DEFW	#4010, #40C8
73		DEFW	#5048, #40D8
74		DEFW	#5050, #40C0

75	DEFW	#5040, #4888
76	DEFW	#4008, #4018
77	DEFW	#40D0, #4898

ADVENTURE GAMES

Алексей и Евгений Федяевы из Москвы прислали нам полный словарь игры **SORCERER OF CLAYMORGUE CASTLE**. Они заметили, что игра распознает вводимые команды по первым пяти буквам. Вот её полный словарь:

ANY	DAM	HIT	PULL	
AROUND	DAMN	HOLD	PUSH	TAKE
ASHES	DANCE	HOLE	PUT	TAP
AT	DEAN	HYPER		TASTE
AUTO	DEEP		QUEEN	THROW
AWAKE	DESCREEN	ICON	QUIT	TIGHT
AWAY	DIG	IGNITE		TOSS
	DIVE	INVENTORY	RAISED	TOUCH
BALLROOM	DIZZY	INVOKE	RANGE	TOWARD
BATTELEMENTS	DOOR		RAT	TOWEL
BLISS	DOWN	JUMP	RATS	TREE
BLOW	DRAIN		RELEASE	TREES
BOTTOM	DRAWBRIDGE	KICK	REMOVE	TUG
BOX	DRINK	KILL	REPLAY	
BREAK	DROP	KITCHEN	ROOF	UNLOCKED
BREATH	DROPLETS	KNOCK	ROOM	UNRAVEL
BRICK	DRY		ROPE	UNTIE
BUILD	DUNGEON	MAKE	RUN	UP
BURN	DUST	ME		USE
		METAL	SAVE	USING
CABINET	EAST	METHUSELAH	SAY	
CAN	EDGE	MIRROR	SCORE	WAKE
CARRY	EDGES	MONSTER	SCREAM	WAKEN
CAST	ENTER	MOVE	SEED	WALK
CASTLE	EXAMINE	MYSELF	SECU	WALL
CEILING			SELF	WALLS
CELL	FASTEN	NICHES	SESAME	WASH
CENTREPIECE	FEEL	NORTH	SHAFT	WATER
CHANDELIER	FIRE		SING	WAVE
CHOP	FIREFLY	OFF	SINK	WEST
CLEAN	FLOOR	OPEN	SLAY	WICKED
CLIMB	FOREST	OPENED	SLICE	WITH
CLOSE	FOUNTAIN	OVEN	SMASH	WOOD
CRATER		OUTLETS	SOUTH	WORKSHOP
CRACK	GAME		SPELL	WRING
CROSS	GET	PAW	SQUARED	WRINKLES
COOK	GLASS	PERMEABILITY	SQUEEZE	
COUNT	GO	PICK	SQUEZ	YELL
COURTYARD	GROTT	PICKUP	STAIRCASE	YOHO
CURSE		POOL	STOVE	YOU
CUT	HANDS	POTION	STREAM	
	HELP	PRESS	SWIM	

Также Алексей и Евгений прислали нам словарь программы HEAVY ON THE MAGIC с переводом слов и пояснениями. Итак:

- А -

AI	- имя призрака
AN	- частица (перед существ.)
AS	- который, в качестве
AM	- частица
AND	- и
ARE	- множественное число
ASK	- спроси
ARM	- рука, оружие
AGE	- возраст, стареть
APEX	- имя друга (против "-", THANKS")
ALAS	- увы
AXIL	- имя героя игры
ALUM	- алюминий
ABLE	- способный, умелый
AGILE	- подвижный, живой (МЕСТО)
ACHAD	- магическое заклинание
ABOUT	- о, около, недалеко
ATTACK	- нападение
ATTACKS	- атаки
ARROWS	- стрелы (МЕСТО)
ASMODEE	- демон, разрушает дверь
ASTOROT	- демон-телепортатор
AAAAARGH	- крик агонии
ADEPTUS MAJOR	- старший эксперт
ADEPTUS MINOR	- младший эксперт
ADEPTUS EXEPT	- высший эксперт

- В -

BE	- частица
BAG	- сумка, мешок
BOOK	- книга
BEST	- лучший
BONE	- кость
BALL	- шар, мяч (вместо PELLET)
BIRD	- птица (EGG+FIRE=NEST,PHOENIX)
BOUND	- граница, прыжок
BRASS	- латунь
BLAST	- взрыв
BLIND	- ослепление, штора
BRAIN	- мозг
BRONZE	- бронза
BELEZBAR	- демон, исследует предметы

- С -

CRY	- крик, плач
COLD	- холодный, холодно
CAKE	- лепешка, торт, кекс
CALL	- звать, кричать

CAN'T	- "не могу"
CLASP	- ножка саламандры (огонь)
CHARM	- чары, очаровывать, обаяние
CLAWS	- когти, скрести
CARRY	- везти, переносить
CHASM	- пропасть (МЕСТО)
CRYPT	- склеп (МЕСТО)
CHEST	- сундук
CLICK!	- "щелк!"
CHROMA	- хром
COPPER	- медь
COBALT	- кобальт
CORPSE	- останки, труп
COMPLEX	- комплекс (МЕСТО)
CUNNING	- хитрость, коварство, умение
CABINET	- тумбочка
CRACKLE	- треск
CYCLOPS	- циклоп (враг)
CAULDRON	- магический котел
COLLODON	- (МЕСТО)
CERTAINLY	- конечно, несомненно
- D -	
DIE	- умирать
DROP	- бросить
DOOR	- дверь
DEAD	- мертвый
DOWN	- вниз
DEAR	- дорогой
DONE	- законченный, готовый
DEMON	- демон
DOUBT	- сомнение, сомневаться (МЕСТО)
DRAGON	- дракон
DISPLAY	- показ, показывать
DESTROYS	- повреждение, поврежденный
- E -	
EYE	- глаз
EGG	- яйцо (вместо него - SHELL)
EAST	- восток
EXIT	- выход
EXITS	- выходы
EMPTY	- пусто
ENTER	- входить
ENTRY	- вход, ворота
EFFECT	- эффект
ENOUGHT	- достаточный
ELEVEN	- одиннадцать
EXAMINE	- исследовать
ELEMENTAL	- элементал (дух стихии), стихийный
ERLSTONE	- алмаз (пройти в LICHGATE)

EXPERIENCE	- опыт
- F -	
FOR	- для
FROM	- от, из
FULL	- полный, целый, обильный
FOOT	- стопа, волчья лапа (амулет)
FIRE	- огонь
FLOX	- (МЕСТО)
FOOD	- еда
FOUR	- четыре
-FORGET IT	- отмена
FREE	- пустой, свободный
FALL	- падение (от воды, "WATER, FALL")
FIRST	- первый
FLASK	- флажок (от пропасти, амулет)
FREEZE	- заморозить
FLICKER	- мерцание
FURNACE	- жаровня (МЕСТО)
FOUNTAIN	- фонтан
- G -	
GOLD	- золото
GOOD	- хорошо
GRADE	- статус
GHOST	- привидение (враг)
GREAT	- великий, большой
GLAD	- рад
GUARDS	- стражники (колонны)
GOBLIN	- гоблин (враг)
GARLIC	- чеснок (против вампира)
GORBURG	- кровавый город (МЕСТО)
GRIMOIRE	- книга заклятий
GRAUMERPHY	- ?
- H -	
HAD	- иметь, обладать
HIT	- бить
HALT	- стой
HEAD	- голова
HELP	- помощь
HAVE	- иметь, обладать
HOLDS	- держит
HORNS	- рога, телец
HYDRA	- гидра (враг)
HONEY	- мед
HEAVY	- тяжело
HEAVEN	- рай, небеса
HEMLOCK	- болиголов
HORRIBLY	- страшный, ужасный
- I -	
I	- частица "я"

IS, IT, IT'S	- частицы
IN	- предлог "в"
IV	- ?
ILL	- болезнь, плохо, зло
INSCRIBED	- начертано
IRON	- железо
INVOKE	- вызывать, пробуждать (демона)
ICTHYS	(МЕСТО)
INVENTORY	- список предметов
IPSISSIMUS	- статус?
- J -	
JAR	- банка
- K -	
KEY	- ключ
KILL	- убивать
KNOWS	- знать
KITCHEN	- кухня (место)
- L -	
LEG	- нога
LIFT	- поднять
LEFT	- влево, оставлять
LIKE	- нравиться
LOAF	- булка, хлеб
LEAF	- лист
LUCK	- удача
LAST	- последний, прошлый
LAZA	- пароль к двери PRACTICUS
LOCKED	- закрыто
LONG	- длинный
LOOKS	- смотри, взгляды
LEVEL	- этаж, уровень
LIMAX	- враг (поднимающийся камень)
LITTLE	- маленький
LITHIC	- литий
LUNACY	- сумашествие
LEATHER	- кожа
LICHGATE	- горный проход, ворота из лишайника (МЕСТО)
- M -	
MANTIS	- богомол (против BELEZBAR)
ME	- мне
MY	- мой
MADE	- сделано
MEAT	- мясо
MAGUS	- маг
MAGOT	- демон, знает где лежат предметы
MIGHT	- могущество, сила
MAGIC	- магия
MISERY	- горе, несчастье (МЕСТО)
MERELY	- только, просто

MAGNAM	- магний
METHOS	- (МЕСТО)
MEDUSA	- медуза (враг)
MIRROR	- зеркало (против MEDUSA)
MONSTER	- чудовище
MORFANG	- (МЕСТО)
MADNESS	- сумасшествие
MAGISTER TEMPLI	- магистр замка
- N -	
NO	- нет
NOT	- не
NOW	- сейчас, теперь
NOTE	- записка, заметка
NANI	- (МЕСТО)
NAME	- имя
NEST	- гнездо
NIGHT	- ночь
NIDUS	- место
NORTH	- север
NEARBY	- рядом, близлежащий
NICKEL	- никель
NUMBER	- число, номер
NUGGET	- серебряный самородок (против WEREWOLF)
NOUGAT	- нуга, щербет (положить вместо NUGGET)
NOTHING	- ничего
NEOPHYTE	- неофит, новичок
NORTH-EAST	- северо-восток
NORTH-WEST	- северо-запад
- O -	
OF	- частица
ON	- предлог "на"
OAK	- дуб
OUT	- выход, наружный, прочь
ONE	- один
OGRE	- великан, людоед
OUTER	- внешний, космический
ORDER	- порядок, приказ
OBJECT	- предмет, объект
OPTIONS	- опции (выход в меню)
OVERLOAD	- перегрузка
- P -	
POT	- котел (AI=SKULL+ULNA+THIGH)
PIT	- шахта, яма
PAW	- лапа
PILE	- куча, свая, шерсть
POUCH	- мешок
PRIDE	- гордость
PORTA	(МЕСТА)
PLACE	- место, плато

PITTED	- ?
POISON	- яд, отравка
PILLAR	- столб, колонна
PURITY	- чистота, невинность (МЕСТО)
PELLET	- хрустальный шар (BALL - вместо него)
PEBBLE	- галька, горный хрусталь, линза
PICK UP	- взять, поднять
PHOENIX	- феникс
PSYCHIC	- психологический
POLISHED	- полированный, изысканный
PILEFOOT	- шерстяная лапа (МЕСТО)
PRECENCE	- представляю, к вашим услугам
PARADISE	- рай (МЕСТО)
PRACTICUS	- практикующий
PARCHMENT	- пергамент
PROTECTED	- защищать, оборонять
PHILOSOPHUS	- философ

- Q -

QUADRA (МЕСТО)

- R -

RIB	- ребро
ROOM	- комната (МЕСТО)
RUBY	- рубин (против ASMODEE)
ROOK	- ладья, грач (МЕСТО)
ROOF	- крыша
RIGHT	- вправо
RABAK	- голова дракона (МЕСТО) (-, WATER)
RAINS	- дожди
REACH	- достигать, богатый
REALLY	- реальный, действительный
REALITY	- реальность, действительность
RAISED	- подъем, повышение
REMAINS	- остатки, останки

- S -

SAY	- говорить, сказать
SEA	- море, океан
SIGN	- знак
SUCH	- такой
SHOW	- показ, показывать
SEEK	- искать
SOME	- некоторый, несколько
STAR	- звезда, звездный
SLUG	- ленивец (враг)
SEEN	- был увиден
SLAT	- планка (против CYCLOPS)
SOUTH	- юг
STAIR	- ступенька, лестница
SKULL	- череп (к AI)
SKILL	- мастерство

SPEAK	- говорить
SPELL	- заклинание
SPELLS	- заклинания
SHADE	- тень, затенять
SWORD	- меч (против ASTAROT)
SNAKE	- змея (против HYDRA)
SHAME	- стыд, смущение
SHELL	- скорлупа (вместо EGG)
SOTHIC	(МЕСТО)
SCROLL	- свиток
SHAPED	- призрак
STINGS	- жала, ожег
SCALES	- чешуя (МЕСТО)
SILVER	- серебро
SEVERAL	- некоторый, особый
STAMINA	- выносливость
SMEARED	- пятно
SILENCE	- молчание (пароль к двери ZELATOR)
SLYMOLE	- хитрый, скрытый; мол, крот
SECUNDA	(МЕСТО: SECUNDA PORTA)
SKELETON	- скелет, каркас
SITUATION	- ситуация
SORONOROS	(пароль к двери PHILOSOPHUS)
SOUTH-EAST	- юго-запад
SOUTH-WEST	- юго-восток
SUNFLOWER	- подсолнух (против MAGOT)
SALAMANDER	- саламандра (против FIRE)
STALAGMITE	- сталагмит
STALACTITE	- сталактит

- Т -

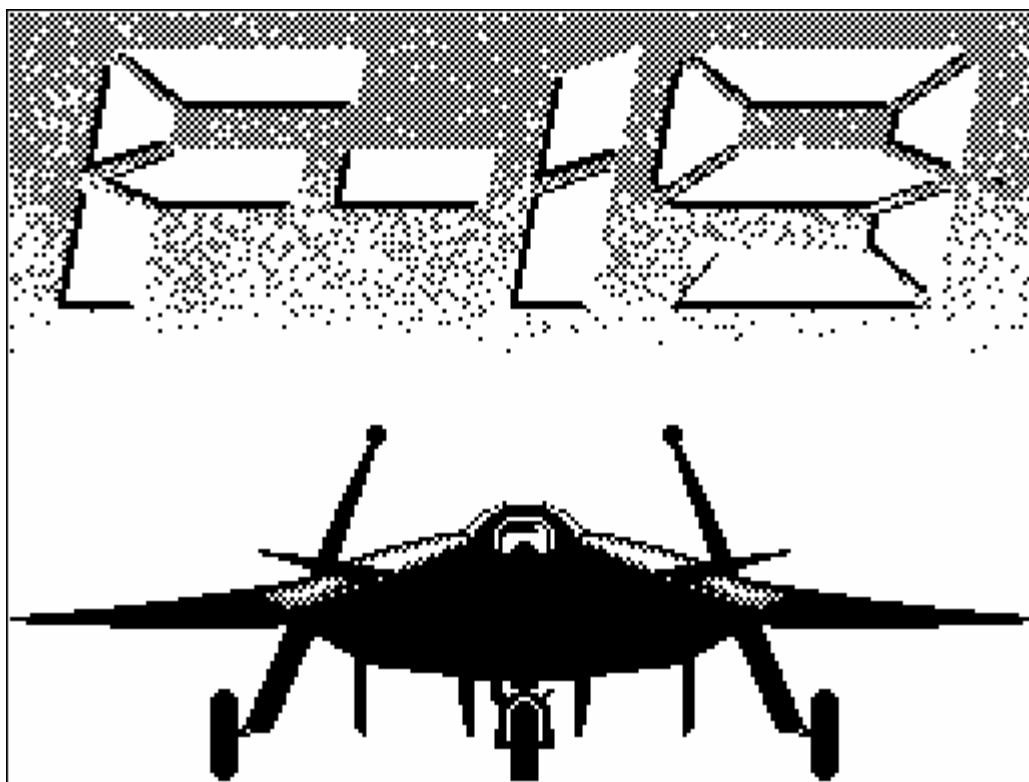
TO	(указывает на движение)
TA	- ?
THE	- частица
TWO	- два
TIN	- олово, жесть
TEA	- чай
TAKE	- брать
THIS	- это
TOLL	- пошлина
TARO	- клеймо (МЕСТО: TOMBS OF TARO)
TALK	- разговор
TIME	- время
TABLE	- стол
TROLL	- троль (враг)
THING	- вещь
TOMBS	- могилы (МЕСТО: TOMBS OF TARO)
TASTY	- вкусный, вкусно
TIGHT	- тугой
THIGH	- бедро (компонент AI)

THAT'S	- то
TERTIA	(МЕСТО: TERTIA PORTA)
TWENTY	- двадцать
THANKS	- спасибо
TRICKLE	- "крак" (потрескивание дров в огне)
TALISMAN	- талисман
TROLLWYND	- город троллей (МЕСТО)
TRADITIONAL	- традиционный
TRANSFUSION	- переливание, перекачивание
	- U -
ULNA	- локтевая кость (компонент AI)
UNDINE	- дух воды, водяная ведьма
UNGUENT	- мазь
	- V -
VERY	- очень
VAMPIRE	- вампир (враг)
VENETIAN	- венецианский ?
	- W -
WAY	- дорога, путь
WHO	- кто
WEST	- запад
WITH	- с, вместе с
WOLF	- волк
WORD	- слово
WAYS	- пути
WELL	- хорошо
WANT	- желать, хотеть
WHAT	- что
WARM	- теплый, согреваться
WASTE	- неразумно тратить, транжирить
WOEDS	- слова
WATER	- вода
WYVERN	- виверн, крылатый дракон (враг)
WRAITH	- скелет (враг)
WHOOSH	- "буль" (шум воды)
WOODWORM	- древесный червь
WOLFDORP	- плато волков (МЕСТО)
WORMRING	- кольцо червей (МЕСТО)
WONTOOTOO	- ?
WRAITHVALE	- долина скелетов (МЕСТО)
	- Y -
YOU	- ты, вы
YOUR	- ваш, твой
YUM	- "ням"
	- Z -
ZINC	- цинк
ZELATOR	- соискатель

ПРИМЕЧАНИЕ ИНФОРКОМА: Следует внимательней относиться к некоторым словарям в программах. Ведь часто, ради экономии памяти, в программе хранятся не сами слова, а их

сокращения, скажем, первые четыре буквы. (Такой способ применен в игре "MESSAGE FROM ANDROMEDA"). Слово при этом может быть неправильно истолковано, и не поможет игроку, а лишь запутает его. Помните об этом, составляя для себя такие "микрословарики".

СОВЕТЫ ЭКСПЕРТА



F-19 STEALTH FIGHTER

© 1989 MICROPROSE SOFTWARE

Эксперт: Дмитрий Усманов, Московская область, 1995.

Эту замечательную программу выпустила фирма MICROPROSE для компьютеров IBM, где она сразу же стала одним из самых популярных воздушных имитаторов. Видя такой успех своего детища, MICROPROSE перенесла эту игру на другие модели компьютеров, в том числе и на SPECTRUM.

Надо сразу сказать, что авторы спектрумовской версии не учли специфику графики этого компьютера, поэтому события, разворачивающиеся на экране подчас проигрывают в реалистичности другим программам этого жанра. Но, к счастью, этот недостаток с лихвой возмещает прекрасно продуманная динамика и точный баланс игры. В неё будет интересно сыграть и начинающему пилоту и бывалому асу - ведь "F-19" имеет широкий спектр настройки сложности игры.

Итак, программа загрузилась, и на экране возникли технические характеристики Вашего самолёта. В игре имитируется реально существующая модель LOCKHEED "SCUNK WORKS", созданная в США. Это истребитель-бомбардировщик "невидимка". Кстати, если Вы думаете, что управляя "невидимым" самолётом Вы будете скрыты от вражеских радаров, то Вы заблуждаетесь. Просто Ваш самолёт даёт эхосигнал гораздо меньшей интенсивности, чем обычные летательные аппараты, и поэтому его труднее "засечь" на дальних дистанциях.

Но вернёмся к техническим данным:

- размах крыльев - 31 фут 8 дюймов (9,65 м)
- полная длина - 59 футов (17,98 м)
- полная высота - 13 футов 2 дюйма (4,01 м)
- расчётный вес при взлёте - 17 т
- двигатели: установлены два двигателя фирмы GENERAL ELECTRICS модели F404-GE-100A

- полная загрузка - 34000 фунтов (15,3 т)
- радиус беспосадочного перелёта - 520 миль (836,6 км)
- максимальная высота полёта - 64000 футов (19507 м)
- максимальная скорость над уровнем моря - 530 узлов (981 км/ч)
- максимальная скорость на высоте 36000 футов - 640 узлов (1885 км/ч)
- на самолёте жестко закреплена 6-ствольная пушка калибром 20 мм в корпус встроены 4 оружейных отсека, максимальная общая загрузка которых не должна превышать 6000 фунтов (2,7 т).

Через несколько секунд экран с техническими данными исчезнет и Вам покажут фирменный вступительный мультфильм, а затем Вы попадете в меню идентификации самолётов. На экране возникнет схематичный рисунок одного из воздушных судов и Вам предстоит правильно указать его название. Такая проверка введена авторами, как способ защиты от копирования, ведь правильный ответ можно было узнать только из прилагаемой к игре инструкции. Над моей версией программы поработали отечественные хакеры, так что при любом ответе играм переходит в экран выбора пилота.

Здесь сверху мы видим личную карту того пилота, который будет выполнять следующую миссию. В эту карту помимо имени и звания входят следующие данные:

MISSION FLOWN - количество боевых вылетов;

LAST MISSION POINTS - количество очков, набранных в последнем вылете;

BEST MISSION POINTS - рекордное количество очков за карьеру этого пилота;

TOTAL POINTS - сумма всех очков, заработанных лётчиком за время полётов.

Ниже находится меню, в котором Вы можете выбрать следующие опции:

START NEW PILOT - начать карьеру пилота заново. Вас попросят ввести свое имя (максимум 15 символов) и зачислят в эскадрилью вооруженных сил США в звании младшего лейтенанта (2nd Lt.)

Сохранить на внешнем носителе свою характеристику Вы сможете, выбрав опцию SAVE THIS PILOT, а загрузить - LOAD NEW PILOT.

Начать выполнение задания можно, указав на строку BEGIN A MISSION. (Если эта строчка окрашена в зелёный цвет, значит пилот погиб и вылет невозможен). Если же все в порядке, Вам предложат выбрать район боевых действий:

1. LIBYA TRAINING
2. LIBYA
3. PERSIAN GULF
4. NORTH CAPE
5. CENTRAL EUROPE

Выше компьютер выводит дополнительную справочную информацию по данному региону:

1. LIBYA TRAINING (Тренировка в Ливии). Это имитация возможных боевых действий в районе Ливии даст Вам возможность поэкспериментировать в методах атаки и управлении самолётом. Для облегчения тренировок вражеские попадания не будут приносить вреда Вашему самолёту. Если Вы выбрали этот пункт, то потом Вам предложат выбрать ещё и вид тренировок:

1.1. BOMBING PRACTICE - Вам предстоит научиться наносить бомбовые удары. Целью послужат незащищенные объекты, расположенные в заливе Сидра.

1.2 AIR-AIR PRACTICE - Имитация воздушного боя над территориями воздушных баз Бенгази и Бени. Вражеские ракетные установки систем ПВО в этом районе отсутствуют.

1.3. DRESS REHEARSAL - Полная имитация боевого вылета против Триполи, защищаемого как ракетными установками ПВО, так и истребителями.

2. LIBYA (Ливия) - Взлетая с авианосцев и наземных баз в центральном районе Средиземного моря, Вы должны выполнить задание против Ливии, одной из самых военизированных стран третьего мира.

3. PERSIAN GULF (Персидский залив) - Вы взлетаете с территории дружественных арабских стран на юге Персидского залива. Вашей целью является Иран, основной источник международного терроризма.

4. NORTH CAPE (Северный мыс) - Вам предстоит совершить скрытые вылеты, стартуя с аэродромов на севере Норвегии. Ваша цель расположена в России, на Кольском полуострове - огромный наземный, воздушный и морской военный комплекс.

5. CENTRAL EUROPE (Центральная Европа) - Базы НАТО в Западной Европе поддерживают Вас в выполнении миссий против самой мощной в мире военной группировки - войск стран Варшавского Договора в Восточной Европе.

После того, как выбран район боевых действий, Вы выбираете стадию военного конфликта в данное время:

1. COLD WAR
2. LIMITED WAR
3. CONVENTIONAL WAR

1. COLD WAR (Холодная война). Бескомпромиссная политика дипломатов и международный терроризм повышают напряженность во всем мире. Однако вооруженные силы все ещё пребывают в мирном режиме.

2. LIMITED WAR (Локальная война). Дипломатическая ситуация заставляет общество провести небольшую военную операцию в этом районе. Вражеские вооруженные силы находятся в состоянии боевой готовности.

3. CONVENTIONAL WAR (Обычная война). В этом районе началась полномасштабная война обычными средствами (т.е. без применения ядерного оружия). Вооруженные силы противника готовы к сражениям или уже участвуют в них.

Затем Вы попадете в меню выбора вида миссии (если Вы выбрали практику, то это меню отсутствует):

1. AIR-AIR MISSIONS (Миссии "воздух-воздух"). Вашей целью будет являться важный, хорошо защищенный самолёт, который не может быть перехвачен и уничтожен обычными средствами.

2. STRIKE MISSIONS (миссии с наземными целями). Чаще всего Вам предстоит совершать воздушные удары и изредка - проводить аэрофотосъемку различных наземных или плавучих объектов.

После этого Вы должны выбрать одну из степеней подготовки противника:

1. GREEN OPPONENTS
2. REGULAR OPPONENTS
3. VETERAN OPPONENTS

1. GREEN OPPONENTS (Новички). Тренировка и профессионализм личного состава вражеской армии очень слаба. Их оборудование устарело и плохо обслуживается.

2. REGULAR OPPONENTS (Регулярные войска). Личный состав вражеских вооруженных сил имеет современное оборудование и хорошую тренированность. Однако они получили лишь небольшой боевой опыт. У них зачастую запаздывает реакция на сложившуюся обстановку и случаются ошибки в принятии решений.

4. VETERAN OPPONENTS (Ветераны). Противник владеет оборудованием самой последней модели с дополнительными усовершенствованиями. Весь личный состав врага уже участвовал в боях. Следует ожидать от них быстрой реакции.

И, наконец - последнее настроенное меню. В нем надо выбрать степень реалистичности посадки.

1. NO CRASHES
2. EASY LANDING
3. REALISTIC LANDING

1. NO CRASHES (Без катастроф). Самолёт не может разбиться о поверхность. Вместо этого он начинает подпрыгивать на земле и, если двигатели включены, сам взлетает. Однако, если взлёт не произойдет в течении приблизительно 8 секунд, самолёт все-таки разваливается. Любое приземление на взлетную полосу автоматически считается мягкой посадкой. Истребитель выдерживает большие повреждения.

2. EASY LANDING (Простая посадка). Для мягкой посадки Ваша скорость должна быть ниже 300 узлов, угол тангажа должен находиться в пределах от -9 до 45 градусов, крен не должен превышать 45 градусов, скорость снижения - небольшая.

3. REALISTIC LANDING (Реалистичная посадка). Для мягкой посадки требуется, чтобы скорость при заходе на ВПП была менее 200 узлов, тангаж в пределах от 0 до 20 градусов и крен не более 16 градусов. Скорость снижения - очень маленькая.

Как можно заметить, внизу под меню располагается графа RISK. В ней постоянно меняется оценка Вашего риска, она может принимать следующие значения:

LOW	- низкая степень риска;
MODERATE	- умеренная;
HIGH	- высокая;
EXTREME	- предельная;
ULTIMATE	- наивысшая.

Естественно, чем Выше степень риска, тем больше очков Вы заработаете, вернувшись на базу.

После выбора всех параметров, Вам покажут полётное задание на следующий вылет. Здесь в графе MISSION BRIEFING выводится текстовое описание миссии. Советую обратить внимание в этом тексте на то, какую именно цель Вам необходимо уничтожить или сфотографировать, есть ли вторичная цель. Ниже, в графе FLIGHT PLAN, выводятся данные о характере полёта:

TAKEOFF - место взлёта;

LANDING - место посадки; Если перед названием базы стоят буквы CV, значит это - авианосец.

TIME - здесь сообщается о времени суток (DAY/NIGHT). Дневная миссия отличается от ночной только тем, что в первом случае изображение на лобовом стекле чёрно-белое, а во втором - цветное. Кстати, чтобы избежать наложения цветов в игре, авторы использовали такой приём:

когда самолёт летит над сушей, небо окрашено в синий цвет, а поверхность - в зелёный. Если же Ваш путь проходит над водой, то небо окрашено в голубой цвет, а поверхность - в синий. Смена палитры происходит резко и поначалу такие "скачки" сбивали меня с толку, но потом к этому привыкаешь, так очень удобно определять момент, когда самолёт пересекает береговую линию.

FUEL: MIN.REC. Здесь показан минимальный рекомендуемый запас топлива для данной миссии. Если Вы планируете после выполнения задания вступить в воздушный бой, то топлива может и не хватить (без дополнительных бензобаков самолёт несет в себе 10000 фунтов горючего).

Нажав "ENTER", Вы перейдете в основное меню:

1. GO ON LEAVE
2. SELECT NEW MISSION
3. INTELLIGENCE BRIEF
4. MISSION BRIEF
5. ARM YOUR PLANE

1. GO ON LEAVE. Возврат в экран выбора пилота.

2. SELECT NEW MISSION - выбор новой миссии, если предложенная Вас чем-то не устраивает. Вам вновь предстоит пройти через настроечные меню, начиная с выбора типа задания (AIR-AIR или STRIKE).

3. INTELLIGENCE BRIEF. Здесь Вы ознакомитесь со сводкой разведданных по этому региону. Вы узнаете тип вражеских радаров и категории ракет "земля-воздух", которыми владеет противник. Кроме того, Вам покажут тип самолётов на вооружении у врага и степень подготовки пилотов. Для

начинающих я приведу перевод доклада разведки по Ливии в состоянии холодной войны, с GREEN OPPONENTS:

Вражеские силы ПВО включают в себя пульсирующие радары, управляющие ракетными комплексами категорий SA-5, SA-2, SA-N-5 с ракетами "земля-воздух". На многочисленных аэродромах противника размещены эскадрильи истребителей-перехватчиков "МиГ-23". Личный состав баз ПВО и пилоты имеют низкий уровень профессионализма. Появление в этом районе самолётов вражеских десантных отрядов быстрого реагирования не ожидается.

4. MISSION BRIEF. Выбрав эту опцию Вы ещё раз увидите текст миссии и план полёта данной миссии.

5. ARM YOUR PLANE. Меню вооружения самолёта. Войдя в него, уже невозможно вернуться обратно - Вы отправляетесь на задание. Но перед этим Вам предстоит заполнить четыре оружейных отсека в самолёте по своему усмотрению. Вам доступно следующее оружие:

SIDEWINDER - управляемая ракета класса "воздух-воздух". Боеголовка захватывает цель по инфракрасному излучению, исходящему от двигателей и дальше наводится сама, не требуя подсветки лучом Вашего радара. Применяется на коротких дистанциях, расстояние до цели должно превышать 300 футов. В отсеке помещаются 4 ракеты.

AMRAAM - управляемая ракета, требует постоянной подсветки лучом Вашего радара. Применяется на средних дистанциях, расстояние до цели должно превышать 300 футов. В отсеке помещаются 3 ракеты.

HARM - управляемая ракета. Следует по подсвечивающему лучу вражеского радара. Радиус действия от 1000 фт до 6000 фт. К сожалению, я так и не смог использовать эту ракету по назначению, в игре она реализована как обычная ракета "воздух-земля", с радиусом действия 13000 фт. Может быть, кто-нибудь сможет с этим разобраться? В отсек загружается только одна такая ракета.

PENGUIN - управляемая ракета. После захвата цели боеголовкой не требует подсветки радаром. Применяется против кораблей. Радиус действия от 1000 фт до 30000 фт. В отсеке помещаются две подобные ракеты.

HARPOON - противокорабельная управляемая ракета. После запуска глиссирует по поверхности воды. Радиус действия от 300 фт до 30000 фт. В оружейном отсеке одна ракета.

MAVERICK - управляемая ракета, применяется против кораблей, бункеров, ракетных установок. Рекомендуемое расстояние до цели при запуске от 5000 фт до 30000 фт. В одном отсеке размещаются две ракеты.

PAVEWAY - бомба с лазерным наведением. Применяется для уничтожения бункеров и зданий. Запускается, когда расстояние до цели находится в пределах от 500 футов до 2000 футов. В отсеке две штуки.

ROCKEYE - парашютная бомба, применяемая для разрушения зданий, ракетных установок и кораблей. Сбрасывается с высоты от 500 фт до 2000 фт. В отсеке 2 бомбы.

DURANDAL - парашютная бомба, предназначена для уничтожения взлетно-посадочных полос противника. Сбрасывается с высоты от 500 до 2000 фт. В отсеке размещаются две бомбы.

SLICK - обычная авиационная бомба, предназначена для разрушения зданий, кораблей, бункеров и ракетных установок. Сбрасывать следует с высоты от 2000 фт до 8000 фт. В отсеке помещаются три бомбы этого типа.

SNAKEYE - парашютная бомба, применяемая против зданий и кораблей. Сбрасывается с высоты от 500 фт до 2000 фт. В одном отсеке помещаются 2 бомбы.

ROCKEYE II - бомба с лазерным наведением на цель, направленного действия. Сбрасывается на здания и бункеры с высоты от 500 фт до 2000 фт. В отсеке - две бомбы.

FIREYE - обычная бомба, применяется против зданий, кораблей, бункеров и ракетных установок. Сбрасывать с высоты от 2000 фт до 8000 фт. В отсеке - две бомбы.

FAE - парашютная фугасная бомба. Предназначена для разрушения бункеров и укрытий для подводных лодок. Сбрасывается с высоты от 500 фт до 2000 фт. В отсеке устанавливаются две бомбы.

IN CLUSTER - парашютная бомба, предназначенная для уничтожения зданий, бункеров и ракетных установок. Сбрасывается с высоты от 500 до 2000 фт. В отсеке - 2 бомбы.

B-1 MINELETS - парашютная бомба для уничтожения бункеров и ВПП. Сбрасывается с высоты от 500 до 2000 фт. В отсеке 2 бомбы.

IR CAMERA - инфракрасная камера для аэрофотосъёмки. Использует пленку шириной 135 мм с 500 кадрами. Съёмка производится с высоты в 20-24000 фт.

1500lbs FUEL - дополнительный топливный бак, несущий в себе 675 кг топлива (т.е. 946 литров). Увеличивает запас топлива и дальность беспосадочного перелета на 15%.

В верхней части экрана находятся четыре оружейных отсека, уже заполненные тем оружием, которое рекомендуется для данной миссии. Если Вы захотите изменить вооружение, сначала подведите синий маркер к нужному отсеку, выбрав опцию CHANGE BAY. Затем наведите курсор на нужное Вам оружие и нажмите "ENTER". Обратите внимание, под отсеками размещен указатель общего количества топлива в самолёте (без дополнительных баков самолёт поднимает 10000 фунтов (4536 кг) горючего. Обязательно сравните эту цифру с минимальным рекомендованным запасом, который Вам сообщили в FLIGHT PLAN. Практика показывает, что в реальном полёте расход горючего всегда превышает расчётный.

Если вся подготовка к полёту закончена, выбирайте опцию ARMING COMPLETE. Вы окажетесь в кабине истребителя, на ВПП аэродрома или авианосца. Экран разделён на две части - в верхней вид сквозь стекло кабины, а в нижней - приборная доска. Повернуть голову и посмотреть направо или налево можно, нажав клавишу V. На лобовое стекло выведены шкалы нескольких приборов (см. рис.1):

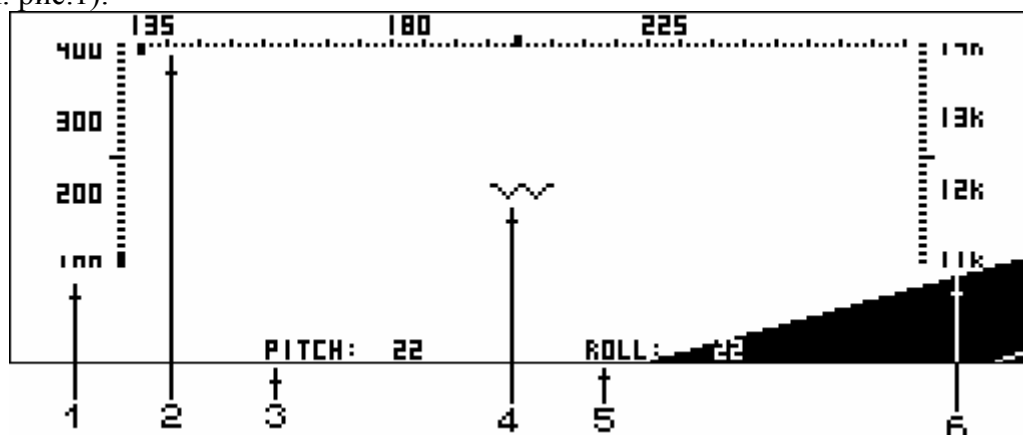


Рис. 1.

1. Спидометр, показывает скорость Вашего самолёта в узлах. Шкала от 0 до 100 узлов показана сплошной линией, выше - с разметкой. Для взлета необходимо, чтобы скорость превышала сплошную шкалу.

2. Компас: сверху расположена подвижная шкала, отградуированная от 0 до 359 градусов. Курс самолёт указывает неподвижная метка в центре. На подвижную шкалу нанесены восемь основных направлений света. Если метка указывает на:

- 0 градусов - вы летите на север;
- 45 - на северо-восток;
- 90 - на восток;
- 135 - на юго-восток;
- 180 - на юг;
- 225 - на юго-запад;
- 270 - на запад;

315 - на северо-запад.

Кроме того, под этой шкалой размещена подвижная метка, показывающая направление на текущую цель. В самом начале борткомпьютер Вашего истребителя настроен на первичную цель (PRIMARY TARGET). Изменить настройку на другие цели, внесенные в борткомпьютер, можно нажав клавишу "X". Кроме объектов на территории противника в память борткомпьютера обязательно заносятся координаты аэродрома, куда Вы должны совершить посадку. Совместив подвижную и неподвижную метки, Вы устанавливаете верный курс.

3. PITCH - угол тангажа. Если это число положительное, значит нос самолёта направлен вверх, если отрицательное - вниз.

4. Неподвижный маркер указывает точно на центр лобового стекла и служит только для облегчения управления самолётом.

5. ROLL - показатель угла крена самолёта. Если он отрицательный, Вы совершаете переворот против часовой стрелки, если положительный - то по часовой стрелке.

6. Альтиметр показывает высоту над землей в футах, буква "k" после числа означает тысячу футов. Ниже одной тысячи футов шкала поделена на отрезки по 100 фт.

Кроме того, если в поле действия вашего подсвечивающего радара попадает одна из целей, на экране появляется другой прицел - подвижный. Цель захватывается на расстоянии менее 70000 фт. Подвижный прицел имеет различные формы:

а) Две квадратные скобки (сверху и снизу) - цель схвачена, но установленное оружие не предназначено для уничтожения таких объектов.

б) Прямоугольный прицел означает, что цель схвачена, но находится вне зоны действия установленного оружия.

в) Круг - цель схвачена и находится в зоне досягаемости оружия, однако возможна ошибка при наведении.

г) Две концентрических окружности - боеголовка оружия прочно захватила цель, можно открывать огонь.

д) Окружность, соединенная с центральным маркером линией показывает в какую область упадет сброшенная бомба. Этот прицел автоматически устанавливается при бомбометании.

е) Окружность с точкой в центре - показывает область, куда будет произведен выстрел из встроенной пушки. Прицел появляется при установке этого оружия.

Внизу расположена приборная доска (см.рис.2). Как ни странно, на ней отсутствует авиагоризонт, очевидно этот самолёт не предназначен для выполнения фигур высшего пилотажа. (Вместе с тем, их вполне можно выполнить).

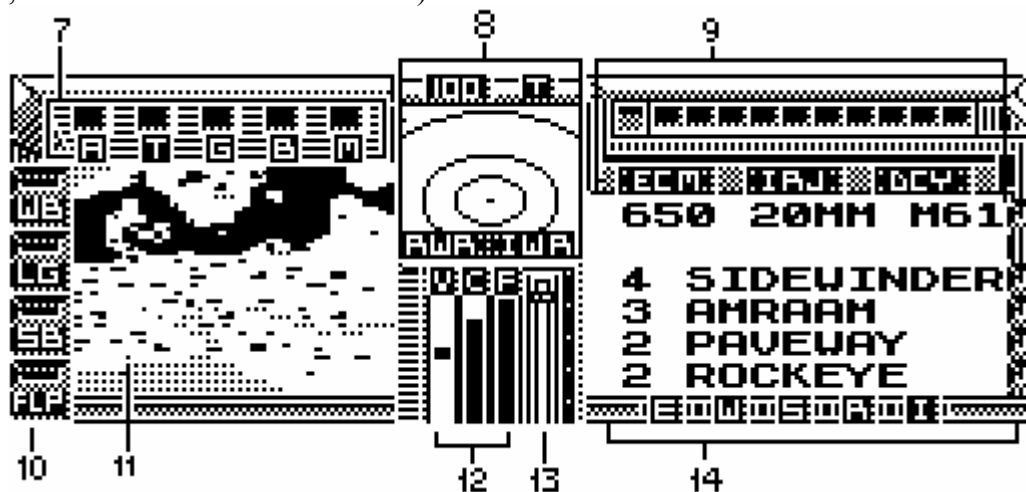


Рис. 2.

На приборной доске размещены следующие группы датчиков:

7. Панель состояния боевых систем. На ней размещены буквенные индикаторы:

A - в обычном состоянии этот индикатор окрашен в красный цвет. Если же на Вашем самолёте закрыть оружейные люки и убрать шасси, то можно будет включить систему невидимости клавишей "H" (HIDE). Тогда буква "A" начнет мигать до тех пор, пока Вас не обнаружит радар противника.

T (TARGET). Показывает вид цели, которую будет искать борткомпьютер. Если этот индикатор окрашен в синий цвет, то компьютер ищет воздушные цели, если в зелёный - то наземные. Вид цели автоматически настраивается на выбранное оружие. Переключить его можно, нажав клавишу "M" (MODE).

G (GUN). Если этот индикатор выделен желтым цветом, значит текущим оружием установлена пушка.

B (BOMB). Этот индикатор загорается, если Вы подготовили к использованию бомбу.

M (MISSILE) Означает использование ракеты.

8. По центру приборной панели расположен радар с сопутствующими индикаторами. На радаре квадратами обозначаются нейтральные цели, мигающими квадратами - военные. Треугольники символизируют самолёты противника, а крестики - запущенные в Вас ракеты.

Над радаром показан его радиус действия. Изменить радиус работы радара, и соответственно, масштаб изображения можно клавишей "Z". Радар работает в трёх режимах с максимальной дальностью захвата в 100, 25 и 12 тысяч футов.

Правее этого индикатора расположена литера "T" - она окрашивается в желтый цвет, если в зоне действия радара появились наземные цели. Если же Вы установили борткомпьютер на поиск воздушных целей, то эта буква будет окрашена в желтый цвет постоянно. Под радаром размещены два датчика:

RWR - Если этот индикатор белый, Вас никто не обнаружил, если же он стал желтым - значит Ваш самолёт "засекли" вражеские радары, но Вы находитесь вне зоны действия их ракет, если он стал синим - радар противника потерял Вас. Красный цвет индикатора сигнализирует о том, что Вы вошли в зону действия ракет противника, Ваш самолёт четко отслеживают. Если же красный индикатор начал мигать, значит в Вас запустили ракету с радарной подсветкой.

IWR - Этот индикатор может быть окрашен лишь в два цвета: белый - все в порядке; мигающий красный - в Вас запущена ракета с инфракрасной системой наведения.

9. Панель защитных систем истребителя. Стало уже привычным в имитаторах отстреливать фольгу (CHAFF) если в Вас запущена зенитная ракета и термолушки (FLARES) против ракет с ИК наведением. Но на F-19 STEALTH FIGHTER'e такие системы не используются, их заменили более современные устройства отвода ракет, которые создают помехи:

ECM - Эта система нейтрализует ракеты с подсветкой радарным лучом. Она включается клавишей "R". При включении индикатор ECM окрашивается в желтый цвет и автоматически выключается через некоторое время или отведя все запущенные в Вас ракеты. Перед выключением индикатор загорается красным цветом, чтобы пилот был готов вновь его подключить.

IRJ - Наводит помехи в инфракрасном диапазоне и отводит ракеты с тепловым наведением. Он включается клавишей "I". Также, как и предыдущий индикатор, этот окрашивается сначала в желтый, а потом в красный цвет.

DCY - Индикатор отстрела ложной цели (DECOY), которая с высокой эффективностью отводит любые типы ракет. К сожалению, на борту умещается всего 15 ложных целей. Отстрелить одну из них можно клавишей "D", на время действия индикатор загорается желтым цветом.

Над этой панелью расположена цветная шкала. Сегменты в ней могут окрашиваться в синий, фиолетовый или красный цвета. Я так и не разобрался, какую информационную нагрузку несёт этот прибор, но выяснил, что при небольшой тяге двигателей шкала синяя, при выборе воздушной цели - фиолетовая, а наземной - желтая.

10. На эту панель выведены индикаторы приборов, влияющих на полёт:

WB (WEAPON BAY) - открыть или закрыть оружейные люки. При закрытых люках (синий цвет индикатора) улучшаются аэродинамические характеристики Вашего истребителя и резко

возрастает скорость. Открытые оружейные люки (голубой цвет) позволяют вести огонь. Люки автоматически открываются при выборе оружия клавишами (1-4). Клавиша "Y" закрывает люки и устанавливает наземный тип цели на бортом компьютере. Кроме того, люками управляет клавиша "U" (открыть/закрыть).

LG (LANDING GEARS). Индикатор шасси окрашен в синий цвет, если они убраны и в голубой, если выпущены. Управление шасси осуществляется клавишей "G". Выпускать шасси можно только на скорости менее 300 узлов при этом снижается скорость и маневренность самолёта.

SB (SURFACE BREAKS). Наземные тормоза резко снижают скорость при посадке, также при включенных тормозах падает скорость полёта. Включить или выключить их можно клавишей "B". Если цвет индикатора голубой - тормоза задействованы.

FLP (FLAPS). Индикатор выпуска закрылок. Если он синий, закрылки убраны (скорость при этом возрастает), а если голубой - выпущены (при этом улучшается устойчивость). Управление закрылками осуществляется клавишей "F".

11. Карта - имеет два режима работы, которые переключаются клавишей "L". В крупномасштабном режиме на карту региона выводится положение Вашего самолёта, а также всех занесенных в бортом компьютер объектов. В мелкомасштабном режиме можно визуально определить курс самолёта и расстояние до цели. В отличие от IBM-овской версии, в этой на карту не выводятся самолёты и ракеты противника.

12. Здесь размещены технические индикаторы:

V - скорость вертикального перемещения самолёта. Если этот показатель красный - Вы снижаетесь, если голубой - взлетаете.

C - реальная тяга, развиваемая двигателями. Запустить двигатели можно клавишей E (ENGINE)

F - показатель оставшегося горючего. Если Вы подключаете дополнительный бак, когда основные ещё наполнены, то это топливо пропадёт впустую.

13. Здесь подвижный маркер показывает отклонение ручки тяги двигателей. Нажав "K" Вы увеличиваете тягу, а, нажав "J" - уменьшаете. Показатель реальная тяги двигателей может не совпадать со степенью отклонения ручки.

14. Здесь расположен многофункциональный текстовый экран. Выводит справочную информацию, работая в пяти основных режимах. О том, какой из режимов установлен сейчас, можно узнать по пяти индикаторам внизу дисплея.

E (EXTEND MODE) - расширенный режим работы, в котором на экран выводится информация бортового компьютера. Главным образом она касается идентификации целей и реакции на Вашу атаку.

W (WEAPON MODE) - этот режим Вы можете включить клавишей "W", тогда на экран будет выведена информация о вооружении самолёта. Верхняя строчка говорит об оставшемся боезапасе пушки, ниже расположены последовательно данные о содержимом четырёх оружейных отсеков Вашего истребителя. Переключать оружие в отсеках Вы можете клавишами "1"- "4" соответственно. Клавиша "5" подключает пушку. Справа от названия оружия загорается индикатор, указывающий на то, что это оружие установлено. Если в одном из отсеков расположен дополнительный топливный бак, то, выбрав его, даже не нажимая "ENTER", Вы автоматически начнёте перекачивать оттуда горючее. Будьте осторожны, можно растратить топливо впустую!

S (STATUS). Нажав клавишу "S", Вы выведете на информационный экран справочные данные о состоянии систем самолёта:

DECOYS - количество оставшихся ложных целей. Если вместо числа стоит OUT - значит, эта система повреждена.

JAMMERS - состояние противоракетных систем (ECM и IRJ). Если всё в норме - О.К., иначе - OUT, тогда нажатие на клавиши "I" или "R" вызовет лишь кратковременную вспышку на индикаторах. Вам остается лишь надеяться на маневрирование.

FIRE CNTL - управление системами ведения огня. При лёгких повреждениях (LGT) возникают сбои в работе оружия, а при серьёзных (HVY) оно вообще не выбирается.

FLGH CNTL - состояние систем управления полётом. При небольшом ущербе самолёт начинает "рыскать", теряется скорость, при больших перестают подчиняться двигатели, а если загорелось FAI - двигатели выведены из строя.

FUEL TANK - состояние топливных баков. LGT - небольшая пробоина, HVY - топливо вытекает с большой скоростью, надо подумать о возвращении на базу.

BAY DOORS - состояние люков в оружейных отсеках. Если в этой графе возникла надпись JAM, значит, люки заклинило и ни закрыть, ни открыть их больше не удастся. Обычно заклинивает открытые люки.

Вернёмся к режимам работы дисплея:

R (RADIO MESSAGE MODE) - в этом режиме на экране возникают радио-сообщения с земли, например о том, что Вы находитесь над заданным районом и т.п.

I - Экран в этом режиме выводит данные о работе системы INS. Включается эта система клавишей "N" (надпись INS ENGAGED), тогда самолёт зафиксирует элероны и элеваторы, сохраняя таким образом тангаж и крен. Управление будет осуществляться с помощью хвостовых рулей. По неясной для меня причине, включение системы INS сопровождается принудительным переключением карты на крупный масштаб. Если система отключена, возникает надпись INS DISENGAGED.

Итак, мы разобрались с панелью управления, можно взлетать:

ТЕХНИКА ВЗЛЕТА

Вы начинаете выполнение миссии, находясь на взлётно-посадочной полосе, по радио Вам сообщают, что всё готово к взлёту. Запускайте двигатели клавишей "E", затем клавишей "K" доведите тягу до максимума. Когда реальная тяга (показатель C) перестанет увеличиваться, снимите наземные тормоза клавишей "B". Самолёт начнёт двигаться, набирая скорость. Когда скорость достигнет 100 узлов, поднимайте истребитель клавишей "A" - Вы взлетели.

НАНЕСЕНИЕ РАКЕТНЫХ УДАРОВ

Приближаясь к цели, нажмите клавишу "T", включив систему идентификации целей. Любая захваченная борткомпьютером цель будет описана в информационном экране. На лобовом стекле в левом нижнем углу возникнет расстояние (в тысячах футов) до цели. Если компьютер захватил постороннюю мишень, нажимайте клавишу SPACE до тех пор, пока на дисплее под описанием цели не появится надпись "PRIME (или SECONDARY) TARGET". Запускать ракеты можно, когда прицел превратится в окружность, но на высших уровнях сложности лучше подождать чёткого захвата (две концентрические окружности), иначе - промах (TARGET MISSED). Иногда цель не уничтожается с одного залпа (проверьте, быть может, Вы выбрали не тот вид оружия?) и тогда на экране возникнет надпись INEFFECTIVE HIT ON TARGET. На самых высших уровнях сложности игры возможна ситуация, когда цели исчезают с экрана Вашего радара - возможно, это работают системы подавления противника. (Такой эффект я наблюдал только в ночных вылетах в Центральной Европе).

БОМБОВЫЕ УДАРЫ

При бомбометании необходимо совместить круглый прицел с мишенью и лишь тогда произвести сброс. Попасть бомбой по цели довольно сложно, требуется определенный опыт. Кстати, бомбы с лазерным наведением борткомпьютер приравнивает к ракетам - ими пользоваться гораздо проще.

АЭРОФОТОСЪЕМКА

Включив камеру, Вы увидите в левом верхнем углу лобового стекла надпись FLM - это количество оставшихся кадров, а в правом верхнем углу надпись TRG - количество фотографий цели. О том, что Вы приблизились к искомому объекту Вас предупредят по радио сообщением NOW YOU OVER PHOTO AREA. Когда Вы покинете район съёмки, на борт поступит другое сообщение - LEAVING PHOTO AREA. По каждому нажатию ENTER камера снимает 10 кадров.

СТРЕЛЬБА ИЗ ПУШКИ

Применять пушку в этой игре совсем непросто, она эффективна только на малых расстояниях. Старайтесь снизить скорость до минимума, иначе подвижный прицел, указывающий направление выстрела, будет слишком далеко "улетает" при каждом движении. Не забудьте учесть ещё и упреждение - в среднем пули летят до цели в течении двух секунд. Пушка стреляет очередями по 10 патронов.

ВОЗДУШНЫЙ БОЙ

На низких уровнях сложности Ваши системы отвода ракет ECM и IRJ будут надежно защищать Вас от нападений, главное не забудьте их вовремя включать. Но на сложных уровнях не надейтесь на это. Если Вы "тупо" летите по прямой, не маневрируя, то Вас нашьпигуют самыми разнообразными ракетами в первом же бою. Постоянно меняйте скорость и не позволяйте самолётам противника висеть у Вас на хвосте. Если же ситуация не позволяет маневрировать (например, Вы готовитесь поразить наземную цель), то при приближении ракеты (смотрите на радар) отстреливайте ложную цель (DECOY). Кстати, в отличие от современных истребителей-перехватчиков Ваш F-19 не такой маневренный. Если Вы попытаетесь поймать в прицел вражеский самолёт, разворачиваясь по кругу, то на низкой скорости Вам это не даст сделать пилот противника, а на высокой его самолёт просто промелькнет на экране. Вам не хватит времени, чтобы вновь выровнять истребитель и удержать цель в зоне действия радара. Лучше используйте тот факт, что противник всегда старается зайти Вам в хвост. Если самолёт близко, увеличьте скорость, закрыв, например, оружейные люки. Потом сделайте мертвую петлю, установите ракету - и противник окажется перед Вами. Надо лишь несколько секунд удержать его перед собой, чтобы захватить цель. Пилоты врага тоже хорошо уходят от ракет (я говорю о ветеранах), поэтому постарайтесь перед запуском как можно ближе подвести прицел к центру экрана, иначе ракета будет приближаться к цели по спирали, оставляя противнику время на маневр.

Если самолёт противника прочно "повис" у Вас на хвосте, и повторяет все Ваши маневры, имеет смысл резко снизить скорость - Вы увидите, как Ваш преследователь пролетел вперед, представляя из себя отличную мишень. Надо сказать, что запас ракет "воздух-воздух" у вражеских перехватчиков намного превосходит Ваш, так что лучше сразу избавиться от опасного соседства. Когда на обратном пути Вы пересечете границу или подлетите к своему аэродрому, большинство вражеских самолётов разворачиваются и улетают, предпочитая не связываться с зенитной артиллерией союзников.

КАТАПУЛЬТИРОВАНИЕ

Осуществляется нажатием клавиши SYMBOL SHIFT, постарайтесь случайно её не нажать в пылу боя. На низших уровнях сложности можно спокойно катапультироваться на любой высоте без опасений разиться. В остальных случаях следует помнить, что на малых высотах парашют не успевает снизить скорость падения, и Вы разбиваетесь. Кроме того, я заметил, что вероятность гибели при прыжке с парашютом над чужой территорией значительно возрастает, особенно если до этого Вы нанесли противнику ощутимый урон. Есть предположение, что на территории врага Вас подбирают и расстреливают на месте. Если же Вас захватят в плен и предадут суду, то родные Штаты спасут Вас, обменяв на какого-нибудь шпиона. Вот как это событие описывает газета "ПРАВДА", передовица которой возникает на экране:

"Информационное агентство ТАСС сообщает: Сегодня в главном здании суда состоялся открытый судебный процесс над захваченным империалистическим летчиком-шпионом. Страна потребовала наказать его по всей строгости закона за его террористический акт! Поднятые обломки его самолёта лишний раз доказали, что капиталистических свиней можно остановить только при помощи силы. Между тем, вашингтонские бандиты предложили обменять незаконно арестованного советского дипломата на этого преступника. Советский посол осудил такое соглашение, но с неохотой согласился на условия".

Если же Вы катапультируетесь над союзной территорией, то Вас подберут спасательные вертолёты по маяку, и Вы тоже благополучно вернетесь на базу.

ТЕХНИКА ПРИЗЕМЛЕНИЯ

Сложность посадки сильно меняется, в зависимости от того какую степень реалистичности Вы выберете. В любом случае, при подлёте к ВПП следует сбросить скорость до 300 узлов (быстро снизить скорость можно, сделав "бочку" или качая крыльями при поднятом носе истребителя). Когда скорость станет меньше 300 узлов, выпускайте шасси клавишей "G" и закрылки клавишей "F" для улучшения маневренности. Садитесь, точно выполняя условия посадки для своего вида реалистичности. Соприкоснувшись с взлетным полем, заглушите двигатели клавишей "E" и включите тормоза клавишей "B". Впрочем, если Вы садитесь на авианосец, это делать необязательно - специальные посадочные тросы быстро затормозят Ваш самолёт.

ВОЗВРАЩЕНИЕ

Когда Вы так или иначе вернётесь на родную базу, Вам покажут разбор полёта, где расскажут, удачно ли Вы посадили самолёт, выполнили ли цель миссии (впрочем, это можно понять по выражению Вашего лица на экране). При разборе полёта учитываются также дополнительно уничтоженные цели противника (ADDITIONAL TARGETS), но они принесут Вам очки только если в этом районе идёт полномасштабная война. Оценивается и количество сбитых истребителей противника. Если Вам в одном вылете удастся подбить более четырёх самолётов, то Вас объявят асом и резко увеличат очки. В конце подводится итог: количество набранных очков выводится в графе "PERFOMANCE". Дополнительно к этому ещё и рассчитывается степень скрытности вылета (она снижается при захвате радаром, визуальном контакте и т.п.) Этот показатель выводится в графе STEALTH PERFOMANCE. Если Вы сделали что-нибудь особо героическое, то командование военно-воздушных сил США (USAF) может наградить Вас или повысить в звании.

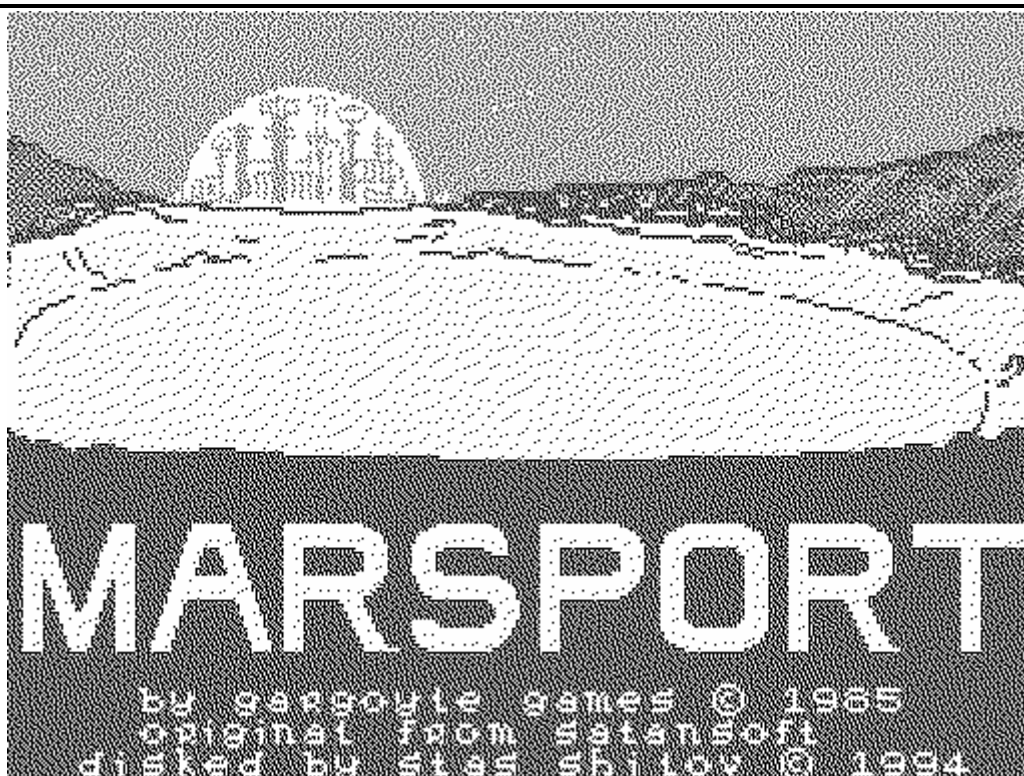
Затем Вам покажут картинку из повседневной жизни летчиков - офицерский бар, где Вы предпочитаете напиваться в одиночку при неудачах и в компании при успехах. Если же Вы погибли, на базе в знак траура приспустят флаги, а Вам покажут вид на местное кладбище. В конце Вы возвращаетесь в меню пилота, где можете начать всё заново.

Напоследок я привожу список всех управляющих клавиш в этой игре (задействована вся клавиатура, кроме клавиши "C")

Q/A, 9/8	- управление элеваторами (взлёт/снижение)
O/P, 6/7	- управление элеронами (крен влево/вправо)
1-4	- выбор оружия из соответствующего отсека
5	- выбор пушки
ENTER, 0	- огонь
W	- переход в режим вооружения
E	- запуск/останов двигателей
R	- включение системы отвода ракет с наведением по лучу
I	- защита от ракет с ИК-наведением
T	- идентификация захваченной цели
Y	- закрыть оружейные люки + поиск наземных целей
U	- закрыть/открыть оружейные люки
S	- состояние систем самолёта
D	- отстрел ложной цели
F	- выпустить/убрать закрылки
G	- выпустить/убрать шасси
H	- включить систему помех
J	- уменьшить тягу двигателей
K	- увеличить тягу двигателей
L	- изменить масштаб карты

Z	- изменить радиус действия радара
X	- поменять настройку компаса
V	- смотреть: вперед/направо/налево
B	- включить/выключить наземные тормоза
N	- вкл./выкл. систему фиксации элеваторов и элеронов
M	- поиск целей: наземных/воздушных
SPACE	- переключить борткомпьютер на другую цель (в том случае, если их много) SYMBOL
	SHIFT - катапультирование
CAPS SHIFT	- пауза.

УДАЧНЫХ ПОЛЁТОВ!



MARSPORT © GARGOILE GAMES

Эксперт: Станислав Шилов, 1995.

МИССИЯ ИГРЫ.

Нужно выкрасть из города MARSPORT секретные планы под названием "Земной шар", которые создал некто с именем Х. Мюллер. Чтобы это сделать, Вам придется исходить весь город вдоль и поперек, использовать 68 предметов, причём некоторые дважды, и вдобавок к этому Вам будут мешать марсиане, сторожевые дроиды и т.д. Начинается игра после входа героя в город из космического пространства (SPACEFIELD).

OPTIONS.

После загрузки программы на экране появляется меню.

OPTIONS

- 1 - enter marsport
- 2 - save game
- 3 - restore game
- 4 - demo mode

Enter to marsport - вход в игру.

Save game - с помощью этой опции Вы можете сохранить свое состояние на магнитном носителе. Вы должны ввести букву от А до Z, которая будет соответствовать выгрузке. Таким образом, на магнитном носителе можно создать 26 разных выгрузок.

Restore game - восстановление состояния игры.

Demo mode - выбрав эту опцию, Вы увидите, что герой начинает ходить вокруг сектора. Но не ждите, что компьютер будет что-то показывать, не дождетесь! Этот режим полезен, когда нужно попасть в противоположную часть сектора. При повторном нажатии клавиши '4' можете дальше ходить по городу самостоятельно.

УПРАВЛЕНИЕ.

КЛАВИШИ УПРАВЛЕНИЯ:

1, 0, SPACE, CapsShift - стрельба из энергетического оружия;
 2, 3, 7, 8, 9 - переключение текущего предмета;
 4 - DEMO MODE (см. в предыдущем пункте);
 5 - PAUSE;
 6 - OPTIONS;
 ENTER - вход в лифты и комнаты;
 Z, C, B, M - движение героя влево;
 X, V, N, SymbolShift - движение героя вправо;
 A, D, G, J, L - поворот компаса против часовой стрелки;
 S, F, H, K - поворот компаса по часовой стрелке;
 Q, E, T, U, O - взять предмет;
 W, R, Y, I, P - положить предмет;

ЛОКАЦИИ. ОПРЕДЕЛЕНИЕ МЕСТОПОЛОЖЕНИЯ.

Марсианский порт состоит из 10 этажей (LEVELS), имеющих соответствующие названия:

Alba (mars dome) - марсианский купол;
 Byer (admin) - администрация;
 Coma (reception) - приём;
 Daly (stores) - склады;
 Elis (residences) - жилой комплекс;
 Farr (recreation) - развлечения;
 Gill (gardens) - сады;
 Hale (computers) - компьютерное управление;
 Iaxa (stores) - склады;
 Joly (main plant) - главное основание;

Каждый этаж состоит из 9 секторов имеющих соответствующие буквы: A,B,C,D,E,H,G,F,I.

Общая структура всех этажей показана на рисунке 1.

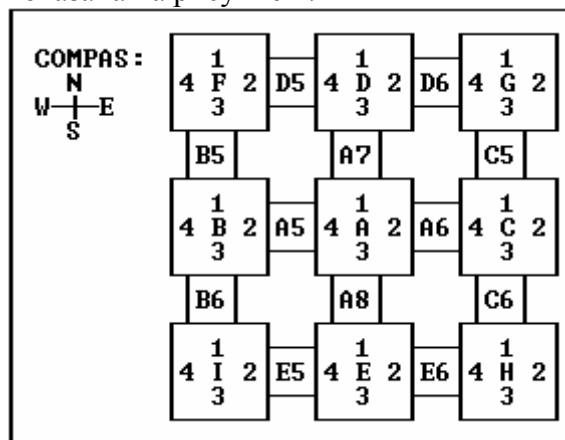


Рис. 1.

Если на стене висит знак "LEVEL C-D1", то это значит, что Вы находитесь на этаже СОМА в секторе D с северной стороны. Помимо секторов, есть переходы между ними. Их обозначения показаны на рисунке 1. Этажи соединяют лифты. Лифты бывают двух типов: вверх (UP TUBE) и вниз (DOWN TUBE). Не во всех лифтах Вы сможете кататься вверх и вниз. Есть односторонние лифты.

Если в таком лифте Вы подниметесь вверх или вниз, то попасть обратно с помощью этого же лифта уже не сможете, т.к. Вы выйдете из двери (NO ENTRY), а в нее невозможно войти.

КОМНАТЫ И ЯЩИКИ.

В городе есть множество комнат. Одни Вам попадутся только один раз, другие - несколько. Есть комнаты, в которые можно попасть только с помощью ключа. Ниже идёт список комнат с информацией о них.

INFORMAT	- в этой комнате Вы можете получить информацию о дальнейших действиях.
ONE WAY	- это односторонний переход между секторами, если Вы воспользовались им, то уже не сможете пройти обратно тем же путём.
RESTRICTED	- двери служебного пользования, есть на каждом этаже. Открываются только после входа в М-Центр.
GAMES ROOM	- игровая комната. Этих комнат только три, все на этаже FARR. В этих комнатах есть две суб-игры, в которые Вы можете поиграть: разведчик и пираты (SCOUT AND PIRATES) и подобие игры в крестики-нолики (TRIPLES). Информацию об этих играх Вы можете получить в соседних комнатах (INFORMAT). Через третью комнату лежит путь к М-Центру. Ключ к этой комнате: игра (GAME).
BAKERY	- кулинария. Ключ: торт (CAKE).
BANKER	- банкир. Ключ: тесто или деньги (DOUGHT).
BOILER	- котельная. Ключ: чайник (KETTLE).
SAFETY	- комната безопасности.
SANCTIUM	- центр. Ключ: KEY TO SANCTIUM.
CONTROL	- управление. Ключ: KEY1 TO M-CENTRAL.
ORATORY	- комната ораторов.
CHEMISTS	- химическая комната. Ключ: башмаки (BOOTS).
MAP_ROOM	- комната с картами.
DIRECTOR	- комната директора.
M_CENTRAL	- М-Центр. Ключи: KEY1, KEY2, KEY3.
HOT_HOUSE	- теплица. Ключ: пакет со льдом (ICE-PACK).
ICE_CREAM	- мороженое.
SPACEFIELD	- космическое пространство. С этой комнаты начинается игра. В этой комнате она и заканчивается, если в неё войти с секретными планами (PLANS).
LABORATORY	- лаборатория. В этой комнате лежат планы (PLANS).
SOIL_DEPOT	- комната земли. Ключ: карта земли(EARTH MAP)
MUSIC_ROOM	- музыкальная комната. Ключ: корнет(CORNET).
PLANT_ROOM	- комната растений. Ключ: герань (GERANIUM).
BAR	- бар.

В городе, кроме дверей и лифтов, есть множество ящиков или открывающихся шкафчиков. Ниже перечислены все виды.

LOCKER	- в этом ящике можно оставлять различные предметы.
REFUSE	- мусорка. С помощью этого ящика можно избавиться от ненужных предметов.
FACTOR	- этот ящик используется для получения из нескольких предметов одного нужного предмета.
SUPPLY	- снабжение. Здесь хранятся предметы.
VIDTEX	- здесь можно получить подсказки.
CHARGE	- зарядка оружия.
KEY...	- в этот ящик нужно положить ключ.

Есть ещё два ящика HERALD и WARDEN, но они не открываются, если к ним подойти. О них читайте ниже.

ПРЕДМЕТЫ.

Таблица 1. Названия и назначение предметов в игре MARSPORT.

№	Название предмета	Где взять предмет	Использование предмета
01	POWER GUN энергетическое ружье	LEVEL D-D2	Для уничтожения воинов (SEPT WARRIORS), сторожей (HERALD) и охранников (WARDEN)
02	HERALD (сторож-дроид)	?	?
03	GUNPERMIT разрешение на оружие	LEVEL E-G2	Положить в ящик KEY, который находится рядом с POWER GUN, после чего можно взять оружие
04	CHARCOAL древесный уголь	LEVEL D-G1	Требуется для создания газовой маски. Часть T-TOKEN
05	GAUSE марля	LEVEL J-I3	Требуется для создания газовой маски (GAS MASK).
06	EMPTY TANK пустой баллон	LEVEL A-A1	Часть воздушного баллона (AIR TANK).
07	OXYGEN воздух	Комната "CHEMISTS" Вход: D-C3	----//----
08	AIR TANK воздушный баллон	FACTOR: 06+07	Нужен для прохода комнаты "DANGER"-CA2,CC4, где нет воздуха
09	WATER (вода)	LEVEL D-D1	Ингредиент для создания торта
10	FLOUR (мука)	LEVEL D-C2	Ингредиент для создания торта
11	BAKINGTIN печной порошок	LEVEL D-A3	Ингредиент для создания торта
12	CAKE (торт)	FACTOR: 09+10+11	Ключ к комнате "BAKERY" Часть R-TOKEN
13	GERANIUM (герань)	Комната "HOT_HOUSE" Вход: G-C1	Ключ к комнате "PLANT_ROOM"
14	EARTH MAP земная карта	Комната "MAP_ROOM" Вход: I-G1	Ключ к комнате "SOIL_DEPOT"
15	SUN CHART солнечная карта	-//-	Часть N-TOKEN
16	MARS MAP марсианская карта	-//-	?
17	EARMUFFS наушники	Комната "SAFETY" Вход: J-C1	Защита от звуковой бомбы (SONIC BOMB) Комната "DANGER" Вход: G-B1, G-F3. Часть Y-TOKEN
18	PROJECTOR	Комната "ORATORY" Вход: B-I1	Часть T-TOKEN
19	HYPER GUN гипер-ружье	FACTOR: 01+32	Для уничтожения военачальников (WARLORDS), в остальном как в POWER-GUN.
20	BOOTS (башмаки)	Там же где 17	Ключ к комнате "CHIMISTS"

21	DOUGHT (деньги)	Комната "BAKERY" Вход: D-F3, D-B1	Ключ к комнате "BANKER"
22	GASMASK газовая маска	FACTOR: 04+05	Защита от газа в DANGER-IC4, IA2
23	VALLIUM проволока или наркотик (см. словарь)	Там же где 07	Защита от нервно-паралитической бомбы (NERVE BOMB) в DANGER-BB1,BF3
24	BOMB (бомба)	Комната "DANGER" Вход: E-C3, E-H1	Быстро положить в мусорку-REFUSE, после чего идти дальше.
25	KETTLE (чайник)	LEVEL F-F4	Ключ к комнате "BOILER"
26	EYESHIELD (очки)	FACTOR: 15+34+35	Защита от световой бомбы (NOVA BOMB) в ASTRONOMY-AB3, AI1. Часть Y-TOKEN
27	KEY 1	Комната "DIRECTOR" Вход: A-D1	Ключ к центральному входу (M_CENTRAL APPROACH). Положить в ящик "KEY", который находится комнате "CONTROL"-HA3
28	KEY 2	Комната "BANKER" Вход: B-E1, B-A3	Положить в "KEY" слева от центрального входа.
29	KEY 3	LEVEL F-G2	Положить в "KEY" справа от центрального входа.
30	HELMET (шлем)	Комната "BOILER" Вход: J-I4	Часть ARTIFACT
31	TOPEE тропическая шапка	LEVEL F-B4	----//----
32	ARTIFACT- энергетическое оружие звездных пилотов	FACTOR: 30+31	Положить в "FACTOR" вместе с POWER GUNом, получите HYPER GUN
33	GAME - SNAKES & LADDERS	FACTOR: 47+48	Ключ к комнате "GAMES_ROOM"-FG3
34	GLASS (стекло)	LEVEL I-H4	Часть EYESHIELD.
35	FRAME (рама)	LEVEL I-E1	----//----
36	LEADSUIT свинцовый костюм	FACTOR: 43+44	Защита от радиации в комнате DANGER-BA2, BC4. Часть R-TOKEN
37	SYRINGE (шприц)	Там же, где 07	?
38	TEA (настойка)	Комната "BAR" Вход: F-E3	Часть противоядия (ANTIDOTE)
39	INSECT насекомое	LEVEL G-D4	----//----
40	POINTE	Там же, где 18	----//----
41	ANTIDOTE противоядие	FACTOR: 38+39+40	Защита от вируса в комнате DANGER-CE2, CH4
42	ICE-PACK пакет со льдом	Комната "ICE CREAM" Вход: I-A1	Ключ к комнате "HOT_HOUSE"-GC1
43	DAIS	Там же где 18	Часть LEADSUIT
44	LUTE (лютя)	Комната "MUSIC ROOM" Вход: F-A1	----//----
45	CORNET	Там же, где 42	Ключ к комнате "MUSIC_ROOM"-FA1

46	(DUST) BOMB пылевая бомба	Комната "DANGER" Вход: G-A4, G-B2	Положить в KEY в той же комнате и можно идти дальше.
47	STEPS ступеньки	LEVEL C-B4	Часть GAME-SNAKES&LADDERS
48	MACHINES- CALCULATORS	LEVEL B-E3	----//----
49	KEY - TO SANCTIUM	FACTOR: 64+65+66+67+68	Ключ к неприкосновенному месту в городе (SANCTIUM).
50	N-CODEX	LEVEL A-E2	?
51	H-CODEX	LEVEL B-G1	?
52	I-CODEX	LEVEL C-H1	?
53	W-CODEX	LEVEL D-E4	?
54	U-CODEX	LEVEL E-F1	?
55	T-CODEX	LEVEL F-E2	?
56	6-CODEX	LEVEL G-G2	?
57	G-CODEX	LEVEL H-I4	?
58	D-CODEX	LEVEL I-D1	?
59	E-CODEX	LEVEL J-I2	?
60	PLANS - EARTHSPHERE план земного шара	Комната "LABORATORY" Вход: H-I1, H-I3	С этим предметом нужно выйти из города. После этого город взрывается, и игра заканчивается. Выход там же, где и вход.
61	MUTE (немой)	Там же, где 44	Часть E-TOKEN
62	MANIFESTO (манифест)	Там же, где 18	Часть E-TOKEN
63	STRAINER	Там же, где 38	Часть N-TOKEN
64	E-TOKEN	FACTOR: 61+62	Часть ключа к SANCTIUM
65	N-TOKEN	FACTOR: 63+15	----//----
66	T-TOKEN	FACTOR: 04+18	----//----
67	R-TOKEN	FACTOR: 12+36	----//----
68	Y-TOKEN	FACTOR: 17+26	----//----

ЭТАПЫ ИГРЫ.

Игра состоит из трех этапов, равносильных по своей сложности.

1. Доступ к М-Центру (m-central approach: %)
2. Вход в центр (entry to sanctum : %)
3. Выход из города (exit from marsport city)

Если Вы входите в новые сектора и правильно используете предметы, то Вам начисляются проценты за прохождение этапа. Этап считается пройденным, если набрано не менее 90%. Мне удалось набрать в первом этапе 96%, во втором - 90%. Однако можно предположить, что эти этапы пройдены не полностью. О третьем этапе читайте в пункте ПРОБЛЕМЫ.

МАРСИАНЕ И ДРОИДЫ.

В городе можно различить два вида марсиан: воины (SEPT WARRIORS) и военачальники (WARLORDS). Все марсиане опасны для героя. Воинов можно расстреливать из простого энергетического оружия (POWER GUN), военачальников - только из гипероружия (HYPER GUN). В городе есть и два вида дроидов: сторож (HERALD) и охранник (WARDEN). Сторожа бывают

опасны, когда у героя в руках ARTIFACT (см. предмет 32 в таблице 1) или предметы его составляющие HELMET и TOPEE (см. предметы 30-31 в таблице 1). Охранники всегда опасны. Программа всегда предупреждает об опасных секторах.

ОПАСНЫЕ КОМНАТЫ.

В городе 9 комнат с табличкой "DANGER", через которые нужно пройти.

Ниже перечислены все виды этих комнат.

DANGER1 - бомба (BOMB). В этой комнате нужно выкинуть бомбу в мусорный ящик (REFUSE), после чего можно идти дальше.

DANGER2 - газовая бомба (GAS BOMB). Через эту комнату без газовой маски не пройти.

ASTRONOMY (DANGER3) - световая бомба. Защита: защитные очки (EYESHIELD).

DANGER4 - утечка воздуха (AIR RUNS OUT). Защита: воздушный баллон (AIR TANK).

DANGER5 - пылевая бомба (DUST BOMB). Нужно взять бомбу и положить ее в ящик (KEY).

DANGER6 - звуковая бомба (SONIC BOMB). Защита: наушники (EARMUFFS).

DANGER7 - радиация (RADIATION). Защита: свинцовый костюм (LEADSUIT).

DANGER8 - активный вирус (VIRUS). Защита: противоядие (ANTIDOTE).

DANGER9 - нервно-паралитическая бомба (NERVE BOMB). Защита: проволока или наркотик (VALLIUM).

ПРОБЛЕМЫ.

Как говорилось выше, основная задача игры - выкрасть из города MARSPORT секретные планы (PLANS), которые угрожают Земле войной с Марсом. Когда Вы их возьмете, включится часовая бомба, которая уничтожит город. Вам остается 90 секунд, чтобы покинуть его, через комнату SPACEFIELD, с которой начиналась игра. Мне этого сделать не удалось, т.к. приходится обходить весь город, чтобы попасть в точку старта, а бомба взрывается примерно на середине пути. Таким образом, выход из города является третьим, самым сложным этапом игры. Возможно, есть другие невидимые пути, ведущие к выходу, т.к. я побывал не во всех секторах.

Остались ни разу не использованными тринадцать предметов, что видно из таблицы 1 (см. столбец "Где использовать", напротив этих предметов стоит - ?). Предмет 2 (табл. 1) я вообще не нашел. Но, найдя в выгрузке, где находятся предметы, я изменил её. Предмет 2 оказался сторожевым дроидом (HERALD), про которого в VIDTEX написано, что он знает все в чужом городе (heralds know all that is alien). Но что же он знает, я не разобрался.

Осталась неиспользованной карта марса (MARS MAP). Очевидно, её надо использовать при выходе из города. Неизвестно, что делать со шприцем (SYRINGE).

Также непонятно, что делать с Древней рукописью (CODEX) (см. в таблице 1 предметы 50-59). Но у меня есть предположения по поводу разгадки CODEX. Если слово CODEX заменить на INDEX (INDEX-указатель), то понятно, что CODEX на что-то указывает, очевидно - на часть сектора. И, если взять во внимание, предмет 6 - CODEX, то указатель указывает на какой-то невидимый переход между секторами. Можно из CODEX выкинуть все буквы, неиспользуемые для обозначения локаций. Т.е., первоначальный CODEX состоит из букв: N, H, I, W, U, T, 6, G, D, E. Убираем буквы: N, W, U, T, при этом остаются: H, I, 6, G, D, E. Сравнивая значения с данными карты, у меня получились две (возможно) "невидимые" локации - LEVEL D-E6 и LEVEL G-E6. Таким образом, осталось всего 4 буквы из - CODEX D, E, G, 6. Но, побродив вокруг этих предполагаемых локаций, я не смог в них войти. Может быть, их вообще не существует. Кстати, в отношении выбора букв из CODEX имеется подсказка: "7 из 10 сделают Вас большой головой!" (7 from 10 makes you a bighead!). По этому сообщению можно сразу догадаться, что речь идет о выкидывании 7 букв из 10 существующих, после

чего 3 буквы указывают на невидимую локацию. Интересно расположение частей CODEX. Каждая часть находится на своем этаже.

В заключение предлагаю карты локаций. Хочу сразу предупредить, что в картах, не отмечены уровни (levels), но с помощью Рис. 1. можно элементарно вычислить положение героя.

Удачи Вам!

Карты локаций игры MARSPORT.

Условные обозначения:

Комнаты DANGER:

- D1 bomb (бомба)
- D2 gas bomb (газовая бомба)
- D3 nova bomb (световая бомба)
- D4 air runs out!(нет воздуха)
- D5 dust bomb (пылевая бомба)
- D6 sonic bomb (звуковая бомба)
- D7 radiation (радиация)
- D8 virus (активный вирус)
- D9 nerve bomb (нервно-пар. бомба)

- Ящики: L locker R refuse
F factor C charge
S supply W warden
V vidtex H herald
K key

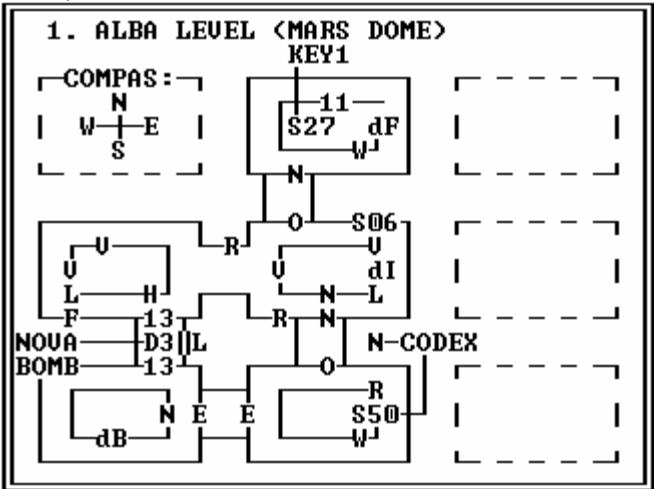
- военачальники (WARLORD)

Комнаты и двери:

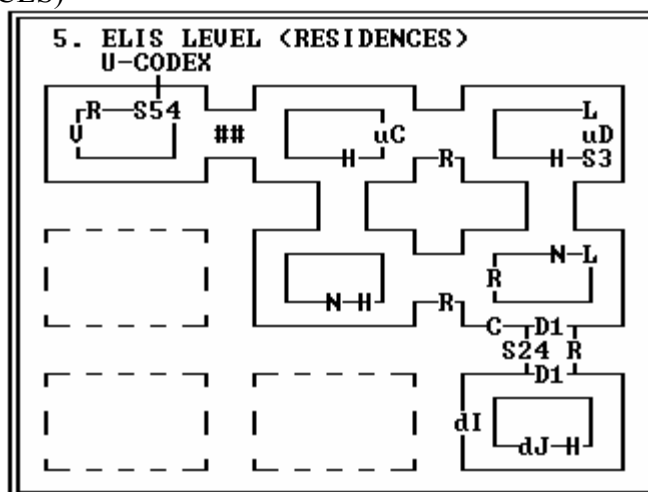
- u up tube to level A-J
- d down tube to level A-J
- I infomat
- O one way
- N no entry
- E restricted
- G games room

- | | |
|-------------|---------------|
| 01 banker | 12 m central |
| 02 bakery | 13 astronomy |
| 03 boiler | 14 hot house |
| 04 safety | 15 ice cream |
| 05 bar | 16 spacefield |
| 06 sanctum | 17 laboratory |
| 07 control | 18 soil depot |
| 08 oratory | 19 music room |
| 09 chemists | 20 plant room |
| 10 map room | |
| 11 director | |

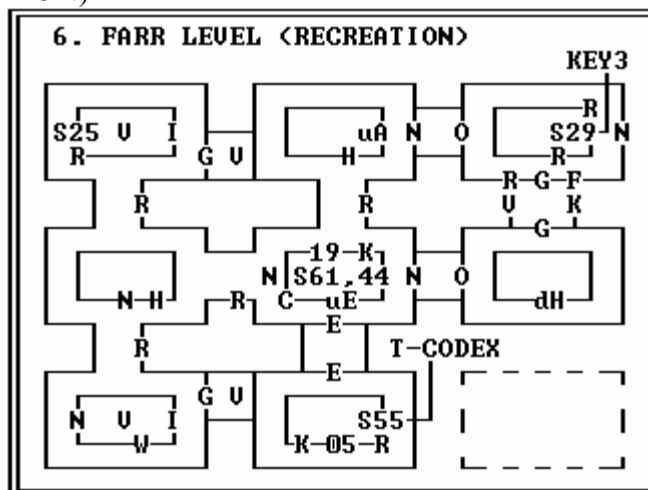
1. ALBA LEVEL (MARS DOME)



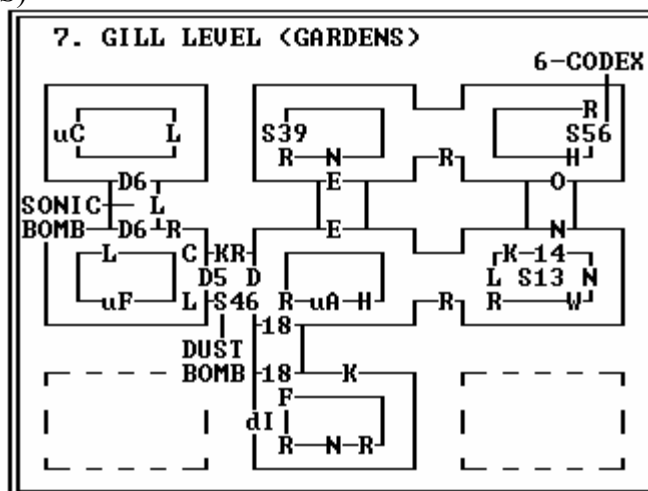
5. ELIS LEVEL (RESIDENCES)



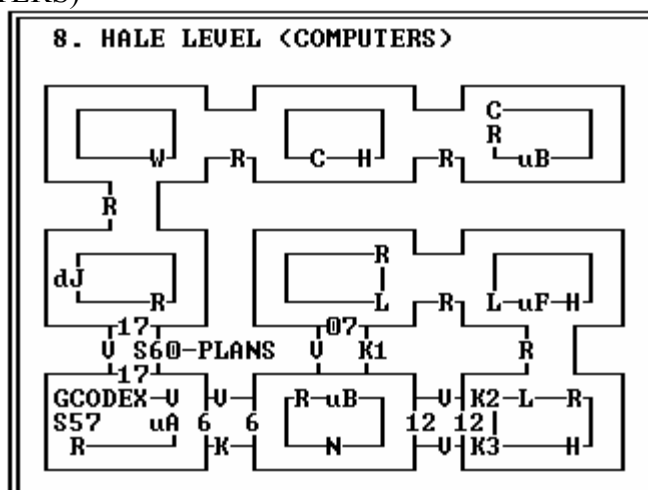
6. FARR LEVEL (RECREATION)



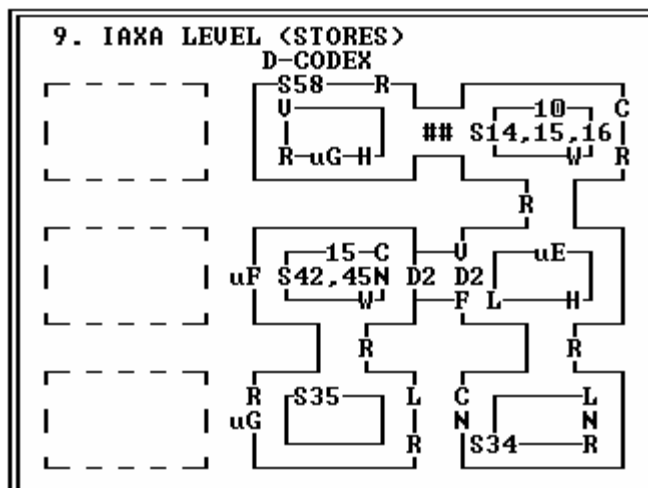
7. GILL LEVEL (GARDENS)



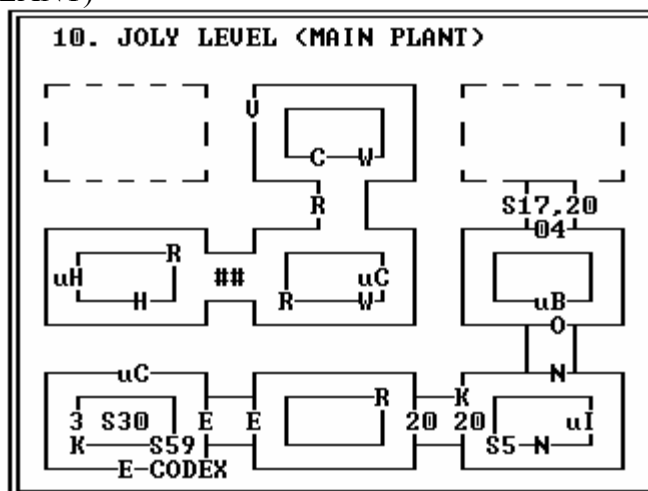
8. HALE LEVEL (COMPUTERS)



9. IAXA LEVEL (STORES)



10. JOLY LEVEL (MAIN PLANT)



----- © SHS 1995 -----

POKES

Представляем читателям очередную часть "полного собрания" POKES, которое составлено по вашим письмам. Но прежде хотим сказать несколько вступительных слов. Мы получаем много писем с POKES. Настолько много, что для того, чтобы разобраться со всем их многообразием и избежать повторов, ведём специальную базу данных. Все POKES, когда-либо публиковавшиеся на страницах ZX-РЕВЮ или ZX-FORUM, заносятся в эту базу данных. Поэтому, анализируя ваши письма, мы выбираем только те POKES, которые ещё нигде не были опубликованы, или для которых в письмах содержатся иные значения. Но вот что мы замечаем в последнее время. Очень много писем поступает с повторами наших ранних публикаций. Такое впечатление, что взяты наши старые подшивки (ZX-РЕВЮ-91, ZX-РЕВЮ-92), и те же самые POKES направлены опять же нам (часто с теми же ошибками). Скорее всего, эти POKES переписаны из других источников, которые, в свою очередь, переписаны у нас. Мы не испытываем никакой обиды по этому поводу. Но мы получаем порой списки POKES такого объёма, которые стоило скорее отправлять не письмом, а посылкой. И вот, мы получаем такое многостраничное послание, на переписывание которого автор потратил не один день, руководствуясь самыми благими намерениями. После проверки по нашей базе данных пары страниц, мы убеждаемся, что это – повторы. Поэтому, с сожалением, приходится прекращать бесполезную работу. Возможно, там и содержится несколько новых значений, но потери времени, требуемые на их "добычу" намного превышают ценность самих POKES. К тому же, скорее всего, POKES к этим играм будут продублированы в других письмах. Время, потраченное впустую автором такого письма – вот это, действительно, обидно. Ведь за это время можно было вполне найти POKES хотя бы для одной игры, ранее не "обессмерченной". И это было бы нужно всем! Поэтому убедительная просьба: не загружайте себя, почту и нас бесполезной работой по переписыванию POKES из разных источников, а займитесь самостоятельной работой. Мы получаем письма, в которых содержатся всего несколько значений, найденных самостоятельно, и, как правило, ранее неизвестных. Вот за это огромное СПАСИБО их авторам.

Солидную порцию POKES, среди которых практически не было повторов, прислал нам Владислав Патрахин из Оленегорска, причем, все значения с комментариями о конкретных параметрах. Он пишет, что все POKES он разыскал на компьютере "Скорпион" во время работы программы. Хотим отметить также вклад Андрея Леконцева из Ижевска – тоже приличное количество собственных новых POKES с комментариями. Большим накопившимся опытом по части POKES и, особенно, советов поделился наш постоянный читатель из Воркуты Михаил Судаков. Систематически присылал последние результаты своих поисков Павел Перин из Первоуральска. Много POKES прислали Виталий Казаков и его друг Д. Денисенко из Армавира (правда, пришлось выбрять из них только те, которые ранее не были опубликованы). Мы благодарим и всех других читателей, принявших участие в составлении разделов POKES и ПАРОЛИ И СОВЕТЫ. Вот только некоторые из них: Михаил Карпов из г. Череповец, Антон Созинов из г. Краснотурьинск Свердловской области, Леонид Горнов из Нижнего Новгорода, Сергей Новиков и Е. Зотов из Москвы, Евгений и Алексей Федяевы из Москвы, Арсений Омелько из Прокопьевска, Борис Круглыхин из Новосибирска, Павел Старков, Павел Лалетин и Алексей Леготин из Красноярска, Григорий Максименко и Павел Костенко из Рубцовска, Алексей (фамилия не указана) из Улан – Удэ, А. Савосин из г. Мытищи Московской области, А. Рылкин из Первоуральска, Алексей Пичугин из Саратова, Андрей Темников из г. Сургут, Н. Пинкин из Каменск – Шахтинска Ростовской области, Александр Коробенков из г. Старобельск Луганской области, Дмитрий Иванищев из Волгодонска, Андрей Рубин из г. Энгельс, Григорий Иванов из Кемеровской области, Павел Бобровников и Ministr Oleg из Челябинска, Владимир Бычков и Андрей Капустин из Тулы, В. Иванов из Смоленска, С. Смирнов из Уссурийска, Сергей Рюмик из Чернигова, С. Назаренко из Краснодара, Flat 50 из Воркуты, Сергей Странин из г. Балаково Саратовской области и ещё многие другие (чтобы отметить всех, понадобилось бы несколько страниц!)....

И. Ручкин из г. Знаменск Астраханской области пишет об игре **SIM CITY**. В таблице, приведённой ниже в двух смежных ячейках в двухбайтном виде хранится число, вычитаемое из Вашего текущего банковского счёта при строительстве определенных объектов.

Объект	Ячейки
Аэропорт	27145/27146
Морской порт	27093/27094
Угольная электростанция	27060/27061
Атомная электростанция	27044/27045
Стадион	26978/26979
Пожарный пост	26945/26946
Полицейский участок	26919/26920
Индустриальный район	26893/26894
Коммерческий район	26867/26868
Жилой район	26841/26842
Парк	26811/26812
Линия электропередач под водой	26794/26795
Линия электропередач на суше	26791/26792
Железная дорога под водой	26783/26784
Железная дорога на суше	26780/26781
Автомобильная дорога под водой	26772/26773
Автомобильная дорога на суше	26769/26770
Бульдозер – "9"	26698/26699
Бульдозер – "1"	26561/26562
???	26537/26538
???	26498/26499
???	26179/26180

Обычно во все эти ячейки я засылаю нули. Но это ещё не всё. В памяти хранится небольшая, начиная с адреса 25398 по адрес 25425, таблица. В ней хранятся числа, указывающие минимальную сумму на Вашем счету, которую нужно иметь для постройки выбранного объекта. Если же такой суммой Вы не располагаете, то некоторые из пиктограмм будут недоступны (при этом их изображение на пиктограммах будет как бы "размазанное") – отсюда длина таблицы: по 2 байта на каждую из 14 пиктограмм.

Оставшиеся проблемы – нашему корреспонденту не удалось найти POKES для "Бульдозера – 56", предназначенного для особо – крупных объектов, таких, как стадион, аэропорт. Кроме того, при пересечении магистралей (например, железнодорожной с автомагистралью) будет вычитаться некоторая сумма. Наконец, не удалось найти POKES, отменяющие бюджетные расходы на полицию, пожарных и дороги.

Наши постоянные корреспонденты Павел и Герасим Троеглазовы из Хабаровского края пишут о том, что некоторые POKES делающие вечной энергию, только портят игру. Например, **HEAVY ON THE MAGIC**. POKES помогают только до того момента, пока Вас какой – нибудь демон не забросит в FURNACE. Выйти из него и начать игру заново Вы не сможете никакими способами, кроме как полностью перезагрузить компьютер. Герасим предлагает загрузчик, с помощью которого герой будет бессмертен до тех пор, пока Вы не нажмете одновременно SS+SPACE. В программе два счётчика энергии: экранный и программный (внутренний). Внося POKES для прекращения убывания энергии, мы останавливаем лишь экранный счётчик, а внутренний – убывает в соответствии с игрой. Нажав SS+SPACE, мы уравниваем показания экранного счетчика с показаниями внутреннего.

10 BORDER 0: PAPER 0: INK 0: CLEAR 29999

20 LOAD ""CODE: LOAD "" CODE: LOAD "" CODE

```

30 POKE 33205,195: POKE 33206,176: POKE 33207,255: POKE 33251,0: POKE
  33252,0: POKE 33253,0
40 FOR A=0 TO 26: READ S: POKE USR "L"+A,S:NEXT A
50 DATA 245,62,127,219,254,203,71,40,4,241,195,187,129,203,79,40,4,241,
  195,187,129,241,214,1,195,187,129
60 RANDOMIZE USR VAL "18434"

```

После загрузки программы **обязательно** установите STAMINE минимальной (опция "6" в меню). Эти изменения сделаны для версии, взломанной "Super fish", поэтому количество "LOAD" и адрес старта могут быть другими.

Предлагаем некоторые загрузчики с POKES, присланные нашими читателями.

ACADEMY (фирменный вариант)

```

1 REM (40 пробелов, можно больше)
10 FOR i=23760 TO 23800: READ a: POKE i,a: NEXT i
20 DATA 221,33,0,64,17,30,27,55,62,208,205,2,8,221,33,0,97,17,232,156,55,
  62,31,205,2,8,221,33,232,253,17,24,2,55,62,37,205,2,8,251,201

```

Выполните: RUN. Далее наберите новые строки:

```

10 BORDER 0: PAPER 0: INK 7: CLEAR 24831: RANDOMIZE USR 23760
20 POKE 31378,190: POKE 31386,190: POKE 31249,190: POKE 31305,190
30 FOR i=23760 TO 23772: READ a: POKE i,a: NEXT i: RANDOMIZE USR 23760
32 DATA 33,227,94,17,255,255,1,0,0,195,0,97,201

```

DEATH WISH 3

Как только загрузится и запустится Бейсик – загрузчик, нажмите BREAK и введите: POKE 24114,201. Теперь запустите загрузчик. После завершения загрузки и остановки, удалите все строки и наберите:

```

1 FOR i=33120 TO 33134: READ a: POKE i,a: NEXT i: DATA 62,255,55,205,86,
  5,62,183,50,185,153,50,37,169,201
2 POKE 33087,96: POKE 33088,129: RANDOMIZE USR 32768

```

FLYING SHARK

POKE 43006,0 POKE 43007,0 POKE 43008,0 – бессмертие;

POKE 48980,0 POKE 48981,0 POKE 48982,0 – можем бросать бомбы без ограничения;

POKE 48605,0 POKE 48606,0 POKE 48607,0 или POKE 47035,0 – уменьшает количество врагов, но, наверное, слишком радикально; кто не хочет переборщить, пусть попробуют POKE 42464,N.

Обе приведенные программы уменьшают количество жизней через индексы, без использования аккумулятора или абсолютного адреса. Те же эффекты можно получить, найдя все DEC (HL) или DEC (DE).

MAG MAX

Вставить 20 строку в загрузчик:

```

20 CLEAR 24999: POKE 23800,195: RANDOMIZE USR 23760: POKE 58472,0:
  RANDOMIZE USR 23803

```

MULTI PLAYER SOCCER MANAGER

POKE 23301,99 – безопасность

POKE 23303,10 – деньги

POKE ADDR,255 POKE ADDR+1,255

При этом $ADDR = 24878 + D * 576 + M * 36$

где M – номер дивизиона команды;

D – место в дивизионе.

ПРИМЕЧАНИЕ: номер дивизиона и место считаются только до начала матчей, сразу после загрузки программы.

NETHER EARTH

Выполнить MERGE "", затем ввести:

```
20 POKE 23800,201: RANDOMIZE USR 23760: POKE 23800,237: POKE 44399,18:
RANDOMIZE USR 23800
```

REBEL

```
1 BORDER : PAPER 0: INK 0: CLEAR 65535: LOAD "" SCREEN$: LOAD ""CODE
28400: FOR i=65364 TO 65380: READ a: POKE i,a: NEXT i
2 DATA 62,6,50,38,245,50,196,245,50,71,242,49,0,0,195,0,192
3 POKE 49958,0:POKE 51139,0:POKE 49239,0: RANDOMIZE USR 65364
```

SIGMA 7 (SIGMA SEVEN)

```
20 CLEAR 24999: POKE 23800,195: RANDOMIZE USR 23760: POKE 34159,0: POKE
34164,0: POKE 60068,0: POKE 60073,0: POKE 60396,0: POKE 60401,0:
RANDOMIZE USR 23803
```

SIR FRED

```
10 BORDER 0: PAPER 0: INK 7: CLS: LET k=16384: FOR i=k TO k+38: READ a:
POKE i,a: NEXT i
12 RANDOMIZE USR k
15 DATA 49,255,255,221,33,0,91,17,88,164,62,0,55,205,86,5,175,50,139,233,
50,140,233,50,175,121,50,176,121,62,201,50,14,183,50,27,183,195,68,181
```

ПРИМЕЧАНИЕ: подчеркнутый фрагмент обеспечивает возможность проходить через что угодно, имея меч. Нужно взять его в руки, а потом нажать влево или вправо – и Вы увидите, что получится. Но этим приёмом злоупотреблять нельзя, так как если на другом экране, на который Вы собираетесь перейти, на вашем пути будет стена, то есть вероятность в ней застрять!

SIR LANCELOT

Поправки надо ввести в блок без заголовка длиной 9344 байт, который загружается по адресу 33424:

POKE 33590,N (N до 10) POKE 33892,0 POKE 33893,0 – бессмертие;

POKE 33697,0 POKE 33698,0 – "экстра"–бессмертие, мы не гибнем от рук врагов и от исхода времени, но если ошибемся, придётся перезапускать компьютер;

POKE 33790,0 POKE 33791,0 POKE 33792,0 – умираем в тишине и спокойствии;

POKE 33690,195 – можем быстро, не утруждая себя, пересмотреть все комнаты; если темп пересмотра слишком велик, придержите ENTER.

А теперь переходим к основной таблице POKES.

Название игры	Значения POKES
3D TANKDUEL	51311,0
ACTION FORCE	51905,183 – жизни 51446,201 – энергия
ACTION FORCE 2	51459,182 – время
AFTER THE WAR 1	52364,167 – энергия
AIR WOLF 1	45583,201
ANDROID 2	53880,0 – время
ANTIRIAD	57086,0
ARABIAN NIGHT	57832,0
ARKANOID 2	33753,0 33754,0 33755,0 – возобновляющиеся кубики не появляются 33728,0 33729,0 33730,0 – эффект 37214,0 37215,0 37216,0 – супер эффект
ARTURA	31943,0 31944,0 31945,0 – энергия
ASTRO MARINO CORP.1 (AMC 1)	40866,182 – жизнь 45096,0 – время 30505,58 – гранаты
ASTRO MARINO CORP.2 (AMC 2)	40866,182 – жизнь 45096,0 – время 32159,58 – гранаты
ATF	31428,N – жизни 31603,0 – бесконечные жизни / 35717,0 35718,0 – ограничение веса 36451,62 36452,100 36453,0 – ограничение скорости 32990,0 – жизни / 32814,N / 32990,0
ATOM ANT	40542,0 40543,0 40544,0 – жизни 40535,0 40536,0 40537,0 – время
BALL CRAZY	28485,0 28489,201
BALLBREAKER	35695,N 35840,0 – жизни / 35700,250 – ракеты / 39844,0 – ракеты
BALLBREAKER 2	38874,0 35729,250 – жизнь 35734,250 – ракеты / 31709,0
BARBARIAN 3	37480,0 37481,195 / 37133,0
BASE	51932,183 – жизни / 51932,60
BATMAN 3	24917,201 24918,0 – неуязвимость
BATTY	47633,N 48437,183 / 48437,0
BEACH BUGGY	45850,255 48603,255 – время 45878,0 46309,0 46549,0 (46548,0???) – машина летит, не задевая препятствий
BESTIAL WARRIOR	40607,N (0...9)
BIGNOSES USA	42823,194
BILLY THE KID	29382,0 29383,0 35817,0 35818,0 – выстрелы
BIONIC COMMANDOS	34244,0 34458,0 – переход на следующий уровень 34200,0 34201,0 – "замораживание" врагов
BLACK RIDER	32832,183 – энергия 38584,183 – патроны 42402,183 – нет врагов
BLADE ALLEY	53172,255

Название игры	Значения POKES
BLAZING THUNDER	35054,183 36484,0 36514,0 37334,0 / 33350,0 / 34701,0
BLIND PANIC	33640,0 33611,0 – нет врагов
BOULDER DASH 2	34732,0 34733,0 34734,0 – время 40185,182 – жизни
BOULDER DASH 4	39664,N
BUBBLER	57129,0
BUCKROGERS	51033,255 / 61137,60
BUGGY RANGER	45778,255
BUMPY	47836,0 47771,0 51717,0 52296,0 52807,0 51576,0 52155,0 / 42200,0 – приз 42906,0 – жизни
BUTCHER HILL	27612,183 27624,183
CABAL	31673,57 31679,57 – гранаты
CAPTAIN FIZZ	65521,24 – проход в двери
CAR TOON	42448,0
CASANOVA	53154,0 / 48420,0 – жители неподвижны
CAULDRON	40062,24 40063,1
CENTIPEDE	28629,0 28630,0
CHAIN REACTION	43339,0 – время 42499,0 43643,0 40642,0 – энергия
CHASE H.Q.	40029,0 40769,0 – турбо / 45221,0 – турбо
CHRION (P.YC.)	31311,0 – патроны 31381,0 – бомбы 33039,0 – энергия 33762,0 – топливо
CHRONOS	56096,201 – неуязвимость 56894,201 – жизни 28107,n 26345,n – работа с Megalaser: n=255 – нет n=1 – выключен n=0 – включен
CHUBBY CRISTLE	29851,182
CIRCUS	29015,N
CLIFF HANGER	27882,255
COLISEUM	56888,N между USR / 57030,N 57062,N
COMMANDO	61955,201 (61956,201???) – враги не стреляют
COMMANDO 4	50693,0 – время 51483,201 – энергия
COMMANDO TRACER	49258,0 49265,0 – энергия
COSMIC KANGA	36212,0
CRAZY CARS	29422,0 – время
CRAZY KONG	24601,0
CROM	51056,0 56220,0
CROSSWIZE	32955,0
CURO JIMENES	40224,N 40336,N
CYBER ZONE	63478,0 – жизни 64213,0 – самолеты
D.N.A. WARRIOR	38631,0 38632,0
DANDY	40825,15 40826,39
DARIUS	62415,0 – энергия 57121,N – жизни
DARIUST	39291,0
DEEP STRIKE	47531,0
DEJA VU	50141,0 50142,0 50143,0 50153,41 / 50290,0 – жизни 47230,0 47231,0 – патроны

Название игры	Значения POKES
DESERT BURNER	40578,0
DIAMOND'S	40654,187
DIZZY 2	25184,0 – постоянно включено суперменю, активизируемое клавишей "C"
DIZZY 2 (TREASURE ISLAND)	29312,58 29321,58 – неуязвимость
DIZZY 3.5	32697,N 32880,0 32881,195 / 29620,5(201) – больше жизней
DIZZY 5	40575,0 40576,0 40577,0 – не мотает головой 52130,24 – энергия
DIZZY 6	37550,0 37526,20 – старт с 20-ю вишнями / 30891,182
DIZZY 7 (CRYSTAL KINGDOM)	34615,201 – от воды 34739,201 – от огня 37092,201 / 34457,0
DIZZY DOWN THE RAPIDS	31072,255 31116,255 31082,255 – патроны / 28394,255
DIZZY X (JOURNEY TO RUSSIA)	35743,182 / 35552,255
DOUBLE DRAGON	38760,N – кредиты
DRAGON BREED (R – TYPE 2)	36363,0 36364,0 36365,0 – неуязвимость 37662,0 – сквозной выстрел / 26054,N – кредиты / 37397,24 – неуязвимость
DRAGON SPIRIT	49611,0 – энергия
DRAGON'S LAIR 2	35766,0
DRUID	30039,176
E – MOTION	35493,0
EARTH SHAKER	35344,183 34552,201 – жизни 59498,22 43139,18 – помощь для прохождения / 35339,24 – переход на другой экран / 31758,0
EDD DUCK	33683,0 – жизнь 32956,0 – неуязвимость
EGG HEAD	57921,34 57922,226 RANDOMIZE USR 57890
EGG HEAD 2	58156,201 58899,201 58184,53
EHEN MONTY	37002,0
EIDOLON	57138,0 59145,0 60013,201 – энергия
ELEPHANT ANTIKS	32866,255 32875,255
ELIMINATOR	39149,250
ELITE	56999,255 – заправка на 25,5 световых лет / 55410,250 – при заправке от звезды – топливо на 250 световых лет / 42950,N – грузоподъёмность / 46848,201 – уничтожает врага с 1 попадания / 46841,0 – уничтожает врага с 1 попадания / 54827,0 54828,0 54829,0 – межгалактический привод 42919,0 42920,0 42921,0 – спасательная капсула 42870,0 42871,0 42872,0 – ракеты 37485,0 – энергетическая бомба 46222,N – грузоподъёмность 46647,N – грузоподъёмность дополнительного контейнера
ELITE (M1 или M128)	42188,0
ELITE	28820,0 – энергетическая бомба 39353,0 – энергия 54758,0 – горючее
(JACK'O'LANTERN)	39795,0 – ракеты 39823,0 – спасательная капсула

Название игры	Значения POKES
ENERGY	43873,201
ETHNIPOD	25200,N – жизни 1 игрока / 25281,N / 48603,N – жизни 2 игрока / 25301,N – игрок 1 48258,N – игрок 2
EXOLON	40221,0 – бессмер. 33646,0 – патроны 37456,0 – гранаты
EXORSISM	34501,0 34502,0
EXPLO – 31	26070,N
F1 TORNADO	60873,182
FALCON PATROOL 2	45363,250
FARGO WELL	53658,N – жизни 62015,3 – бессмертие
FAST FOOD	47844,0
FERNAN 2	47301,255
FIGHTING WARRIOR	61233,183 60991,182 / 60994,0 / 62868,0
FINDERS KEEPERS	30394,255
FIRE & FORGET	25273,0 25414,0
FIRE FLY	45453,183
FREDDY HARDEST	64012,0 64013,0 64014,0 / 64011,0 64015,0 64016,0 64017,0 / 58625,0 58629,0 58631,0 55923,1
FREDDY HARDEST 2	58230,0 58234,0 58235,0 49078,1
FRIGHT MARE	44051,0 48465,0
FRONT LINE	62499,0
FROST BYTE	33989,0 28236,0 36560,0 30991,0
FRUIT MACHINE	49386,0
SIMULATOR	
FX 2020	59494,201
GAME OVER	30648,255 30658,255
GHOST BUSTERS	1–й уровень: 48240,183 – жизни 49815,0 – энергия 49494,0 – лазер 49624,0 – молния 49712,0 – защита 2–й уровень: 51186,0 – жизни 55877,0 – энергия 3–й уровень: 49285,0 – жизни 53878,0 – энергия
GHOULS'N'GHOSTS	35536,255 – кредиты
GOLDEN FUTURE	35762,0 35763,201
GONZZALESS	39291,0
GOODY	44846,255 / 45590,0
GULPMAN	26904,N (49...165) – лазер 27367,0 – лазер 26927,N (49...255) – жизни 27311,185 – жизни 25405,57 25407,56 25409,54 25411,55 25413,48 – управление SINCLAIR–джойстиком
GUN HEAD	47150,0
H.E.R.O.	53518,201 53934,201 54918,0
HAMTE DAMTE	42755,0 42756,0 42757,0 42748,0 42749,0 42750,0
HAW TO ABE HERO	62001,0 – время
HAWK STORM	32163,0 34869,0 35306,0 36604,0 / 33451,N – снаряды 33397,N – энергия

Название игры	Значения POKES
HE – MAN	45882,N 46781,183
HEAD OVER HEELS	42195,0 – жизни 42050,24 – патр. 42985,0 – быстрый бег
HIGH PHOTON	38737,0
HILEAS FOGG BALLON	50284,255 64544,255
HOLLYWOOD POKER	38565,175
HONG KONG	33102,183 / 33102,0 43763,0 / 33128,0
HOPPIN'MAD	41968,0
HUMPHREY	25782,201
HUNDRA	41373,0 41375,0 41378,0 43380,201
HYPER	25198,0 25199,0
HYPER ACTIVE	48122,0 48140,201 – энергия / 29993,255 35849,255
I'BALL 2	43384,N между USR
IMPOSSAMOLE	49125,0 49126,0 – неуязвимость
INDI JONES (LAST CRUSADE)	44356,183 44516,183 41570,183 – энергия / 41570,167 – энергия
INDI JONES (TEMPLE OF DOOM)	33948,0
INFERNAL COMBUSTION	41896,0
INSIDE OUTING	43589,0 43590,0 43591,0
INTERALLIA	48708,201 – защита от пушек 52960,0 – защита от лазеров
INTO THE EAGLES NEST	36640,0 41136,0 40512,0 36641,0 41137,0 40513,0
IRON MAN	35165,190 53402,0
IS – CHESS 48	25297,16 26537,26 26559,24 26563,25 26566,254 26567,23 – управление SINCLAIR–джойстиком
ISLAND OF DR. DESTRUCTO	41763,N – жизни 41778,N – дни
JACK THE NIPPER	43203,201
JACK THE NIPPER 2	43251,0 43314,0 43315,0 34631,0 / 34426,0 – неуязвимость
JACKSON CITY	34419,0 34420,0 – жизни 34427,0 34428,0 34429,0 – старт с места гибели
JET STORM	53440,0 – топливо 53943,0 53957,0 – оружие / 50273,0 – fuel 50790,0 – ammo 51016,0 – bombs
JETPAC	25018,0
JOE BLADE	31865,0 – субигра 37797,0 – энергия 38497 – бомба
JOE BLADE 2	36269,0 – время
JOE BLADE 3	37346,0 / 43058,0 – энергия 45327,0 – время 45362,0 – время субигры
JOE BLADE 4	40972,195 – энергия 41590,0 – время 41648,0 – время субигры / 39559,0 39560,0 39561,0 39562,0 41587,201 41647,0 41648,0 36113,99

Название игры	Значения POKES
JUNGLE WARRIOR	54687,N / 44976,201 – жизни 39701,183 – патроны 45177,40 45178,2 45179,0 45180,183 45181,201 – энергия / 45177,183 45178,201 – энергия / 44905,0 44637,201 – энергия
KENDO WARRIOR	37435,195
KLAX	39143,0 – кредиты 38569,55 – падения
KUNG – FU KNIGHTS	45687,N / 45692,0 между USR
KWIK SNAX	37297,N перед первым USR / 34125,0 50449,0 – таймер
LASER	35969,182 65074,182
LAST MISSION	43177,0 – отключенная гравитация 44566,0 – бессмертие / 44764,0 44745,0 – лазер не перегревается 45682,0 – жизнь / 45690,0 – жизнь
LAST VAMPIRE	27983,194 30018,0 – энергия
LE CENTRACE	24544,0
LICENCE TO KILL	45102,201 45103,194 45108,0 / 54608,183 57106,183 58674,184 59436,183 59701,183
LIVINGSTONE 2	45831,183 / 46332,0 46333,0 46334,0 / 51393,N (0...127)
LOGO+	24189,0 – время
LUNA CRABS	40671,N (24...52) – жизни 40666,0 – жизни 33036,71 33040,95 33044,239 33048,103 – управление SINCLAIR-джойстиком
MAD MIX 1	39947,N 40065,0 40041,0 40304,195
MAIN BLOW	45607,N – жизни для обоих игроков 50924,0 – вертолёт 48294,0 – джип
MASTER BLASTER	49447,0 – защита 49988,0 – жизни / 50910,201 50916,201 51051,201 51172,201 51177,201 (51177,7???) 53161,201
MAZEMANIA	29549,255 – замораживает врагов / 42582,0 – энергия / 42582,0 – энергия
MEGA – BUCKS	65000,201 – время 41203,0
MERLIN	45127,N
MERMAID MADNESS	31277,0 31288,0 31307,201
MIAMI COBRA GT	40113,0 40956,0 41433,0 44788,0 45306,0 – турбо / 62001,0 – время
MICHEL 2	27908,0 – жизни 1 уровень 34487,0 34491,195 – жизни уровень 31215,0 – жизни 3 уровень 33516,0 – жизни 4 уровень 35530,0 – жизни 5 уровень 24922,N 25211,N 25882,N – жизни / 27560,0 31256,0 33527,0 34499,0 35537,0
MIDNIGHT RESISTANCE	29432,N – жизни 37503,0 37524,0 – оружие / 36194,255
MIG – 29	41987,N 42965,183

Название игры	Значения POKES
MISS PACMAN	52887,0 – жизни 1 57106,0 – жизни 2
MIVISIS	34779,0
MOLECULE MAN	64622,0 64623,0
MONTE – CARLO	30707,0 30708,0 30736,0 – деньги
CASINO	/ 34207,255 – деньги / 32983,57 – деньги
MONTY FREE	34117,0 – время
MONTY N3	38858,62 38859,5 38860,50 38861,111 38862,134
MONTY ON THE RUN	34715,182
MOT 2	23484,N – энергия / 40745,0 40748,0 / 41529,0 / 62039,0 62040,0 – энергия
MOT 3	32723,N – энергия / 44005,0 / 47741,15 47742,195 – энергия
MOTOS	33300,N / 42241,0
MR.GAS	26837,7 26838,0
MULTI PLAYER	23301,99 – безопасность 23303,10 – деньги
SOCCER MANAGER	
MUTANT MONTY	51742,0 51746,195
MUTANT ZONE 2	37814,201
MYTH	25716,245 25720,241 26854,245 26858,241 / 25303,N 25902,N 26441,N 27040,N / 1 уровень – 27797,0 2 уровень – 27738,0 3 уровень – 27770,0 4 уровень – 27855,0 5 уровень – 27678,0 (блок длиной 33792 для каждого уровня загружать в COPY–COPY с адреса 23040) / 65207,48 65208,41 – не зависнет при загрузке уровня с ошибкой, а попросит перезагрузить заново
MYTHOS	29701,0
N.O.M.A.D.	40167,0 34569,201 – неуязвимость
NEBULUS	32921,0 32928,0 43520,0 43332,0
NETHER WORLD	32096,0 32361,0 – энергия
NIGHT STEEL	30736,0
NIGHTMARE ON	48371,0
ROBINSON STREET	/ 53267,0 – жизни 49937,N – количество врагов 56046,0 – враги безопасны
NINJA COMMANDO	29076,255 51769,0
NINJA SCOOTER	43920,183 43921,0 – время
NINJA WARRIOR	29666,0 30383,195
NONTERRAQUEOUS	26093,0
NUDGE	43060,255 43052,255 43044,255 – кредиты
NUREK	49372,0 49385,0
O.F.T.R.	30771,255 27081,N

Название игры	Значения POKES
OBLITERATOR	33595,0 33596,195 / 48529,183 / 26803,N – энергия 27558,N – энергия после пополнения
OLLI & LISSA	35871,0
OLLI & LISSA 2	37510,0 37511,0
ONE MAN AND HIS DROID	57350,0 – время
OPERATION THUNDERBOLT	42918,183 – энергия 1 42853,183 – энергия 2
OPERATION WOLF	40681,0 – патроны 40710,0 – гранаты
OVERLANDER	31313,0 – жизнь 33854,0 – топливо
PACLAND	32319,0 / 35242,201 35280,0
PEOPLE FROM SIRIUS	31276,0 – энергия 32760,201 – патроны 43170,0 43172,0 43174,0 43175,0 – падающая в пропасть, окажетесь на другом берегу
PETER PACK RAT	27112,100 / 27212,N / 27312,255
PHANTIS	54216,0 – бессмертие
PHANTOM CLUB	49138,0 – неуязвимость 49081,0 – энергия
PHILEAS FOGG'S BALLOON	58003,0 50284,N
PHODEY HONGCONG	29710,0
PI*R**2	38481,0 38752,0
PIROMANIA	43682,0
PLUMETY	25222,0 25146,0 33219,201 30081,0 25209,0 25210,0 25211,0
POLE POSITION	35025,0 – время
POWER BOAT	31344,0 31350,0
POWER PYRAMID	51175,0
PREDATOR	39575,0 39594,0 – патроны 36166,201 – гранаты
PRINCE CLUMSY	45715,201 43760,24 43715,0
PROFESSIONAL SKI	36148,57 36153,57 36158,57 36163,57 36168,57 36173,57 36178,57 – время
SIMULATOR	32109,62 32110,255 32111,0 – переход на другой уровень для 1 игрока 32138,62 32139,255 32140,0 – переход на другой уровень для 2 игрока
PSSST	30684,0 / 42391,0
PULSOIDS	53887,N
PUZZNIC	46062,201 – время
PYJAMARAMA	48669,0 / 48679,0 / 48670,0 48686,58 – жизни 59189,201 – нет врагов 44092,201 – максимальная скорость 44095 и 44096 – 2 – байтное число – скорость (1 – максимально, 65535 или 0 – минимально) 33904,201 – энергия и не мешает конвейер
QUAZATRON	58283,62 58284,63 58285,50 58286,147 58287,175 – энергия, KLP2 не погибает даже после поражения в схватке GRAPPLE

Название игры	Значения POKES
R-TYPE	37374,201 / 37525 – кредиты 37129,201 – никого нет 49992,201 – враг не стреляет 37571,70 37573,201 / 38479,0 – бессмертие 38678,73 – появление на месте гибели
RAIDERS	25962,183 / 25962,0
RAMPAGE	58900,0 – враги иногда не стреляют 56924,0 – нет задержки после разрушения всех зданий
RASTAN SAGA	37877,0 – энергия
RED HEAT	33644,183 – жизни 36297,0 – патроны 31650,195 31651,249 31652,121 – враги не нападают
RED LOTUS	42735,255
RED STORM	37337,201
RENEGADE 1	41036,0 / 41048,195
REPTON	63657,N / 63922,0
REPTON MANIA 2	63646,182
REX 1	37045,N – взрывы 37040,255(иногда первую игру можно получить без вечных жизней)
REX 2	56046,0 56047,0 56048,0 – не надо вводить код
RICK DANGEROUS	34607,0 35376,0 – жизни 34541,0 38303,0 – молнии 34574,0 39745,0 – бомбы / 55424,N – аммуниция 55235,N – жизнь / 55524,0 60955,0 61046,0 / 55480,0 – жизнь / 55279,N
RICK DANGEROUS 2	38303,0 – патроны 39745,0 – гранаты 37844,201 – прозрачность 34607,0 35376,0 – жизни
RIPTOFF	34585,N 57666,N
RIVER RAID	24251,0
ROBIN HOOD	50616,201 между USR
ROBOCOP	45736,0 – время / 46113,24 46229,182 / 46107,0 46108,0 46229,0 46230,195 46149,0 46150,201
ROBOCOP 2	44140,195 – энергия
ROCK'N'ROLLER	42548,N / 29928,0 – можно ездить по машинам
ROCKFORD	57048,183
ROCKMAN	56344,0 56345,0 56346,0
ROLLAROUND	32604,0 – время 32679,0 32692,0 – жизни
ROLLER	40297,0 40979,0 41047,0
ROLLING THUNDER	39792,183 / 48472,0 – нет врагов 35280,0 35281,0 35282,0 – время
RUFF & REDDY	32849,255 / 35789,0 33099,N
RUNNING MAN	29049,201 – энергия 28307,201 – время
S.D.I.	33832,0 33905,0

Название игры	Значения POKES
SABRE WOLF	45599,N 43575,0
SABRINA	26157,N
SAIGON COMBAT UNIT 1	43044,N 43127,201 – энергия 40051,0 – жизни
SAIGON COMBAT UNIT 2	43229,N 43312,201 43725,201
SAMANTA FOX	61039,7 – деньги
SAMURAI WARRIOR	37866,183 – энергия 45092,183
SANXION	36585,182 / 25181,255 25186,0 25187,0 25188,0
SAVAGE 1	34398,255 / 37299,255 39446,0
SAVAGE 3	34459,255 44251,0
SCORE	46416,60 – энергия / 45027,N – жизни 49543,0 – энергия 31150,0 – неуязвимость мяча 45120,0 – бессмертие
SCUMBALL	52824,0 52866,0 52890,0 52928,0 52954,0 54402,0 54423,0
SEA HAWK	52054,0 – неуязвимость / 53775,0 53776,0 53777,0 / 50626,201 / 54834,N
SENDA SALVAGE 2	56350,0 56351,0 56352,0 52584,0 56420,0 56449,0
SENTINEL	37407,0 37408,0 – непобедимость 32450–59: 0...0 – энергия
SH.BEAST	33220,0
SHINOBI	34715,0
SHOOT OUT	60310,N 61476,0
SHORT CIRCUIT 2	35926,0 35932,0 – энергия 36488,0 – жизни
SHURIKEN PARTH ONE	40000,152 40001,40 40002,201 – энергия
SIDEWINDER 2	36081,182 / 32821,0 32822,0 32823,0 32824,0
SILK WORM	38562,195 – вертолет 38506,195 – джип
SIM CITY	27145,1 27146,0 – Airport=1\$ 27093,1 27094,0 – Seaport=1\$ 27044,1 27045,0 – Nuclear=1\$ / 36759,255 36760,255 – деньги / 33581,255 35583,255 19329,0 19330,0
SIR FRED	25243,0 25375,0 25952,0 46671,0 / 59787,0 59788,0 46862,201 46875,201
SIR WOOD 1	40742,0
SIR WOOD 3	16575,N 36267,0
SKATE USA	43096,254 52954,254
SNAKE PIT	28615,24 – неуязвимость 28524,57 28528,56 28532,54 28536,55 29206,48 29222,48 – управление SINCLAIR–джойстиком 29224,201 – меняет смысл игры: вместо борьбы за выживание – соперничество за съедение яблок
SNARE 1	27251,0 30561,0 – время 29888,0 – нет врагов
SOLARIS	32794,N – жизни 35780,0 – бессмертие
SOLDIER OF FORTUNE	42181,N

Название игры	Значения POKES
SOLDIER OF LIGHT	42176,60 43302,2 / 50035,N 50552,183 51119,183 – жизни 62272,201 62273,0 – энергия / 62272,192 – энергия
SON OF BLAGER	27421,N – жизни 27413,N (1...12) – уровень
SONIC BOOM	31439,183 31813,182/30361,255 31590,255 32051,255
SOUL OF A ROBOT	25812,N – жизни 25876,0 – бессмертие
SPACE RAIDERS	46543,201 / 25962,0 – жизни 26594,239 26607,98 26619,90 26641,66 – управление SINCLAIR–джойстиком
SPHEROID	51595,N / 52930,0 52931,0 – энергия
SPIKE HAROLD	34813,0 / 34816,24 – жизни 34808,24 – неуязвимость
SPITFIRE 2	34386,0 / 34383,0 55783,0
SPOKED 89	51962,0 51963,0 50458,0 50459,0 50460,0
STARFOX	28268,0 между USR
STARGLIDER	54421,0 – ракеты
STEG	32315,183 32336,183 36449,0 37444,183 37445,183
STORM	37667,132
STORMLORD	31360,0 – жизнь 32260,0 – прыжок / 34657,N – жизни 34640,0 – жизни 58102,201 – время
STREET GANG	39254,182 / 37109,0 37125,0 37361,0 37389,0
STRIP GANG	25296,9 (25296,0???) 25321,0 – выигрываете всегда Вы
SUPER HANG-ON	49913,0 – время для 1-й версии игры 49698,0 – время для 2-й версии игры
SUPER KID IN SPACE	61003,N
SUPER STUNTMAN	25517,0
SUPERTANK	41548,0
SWITCH BLADE	38480,183 / 38478,0
SWTORM LORD 1	34657,N 56877,183 – жизни 56863,201 – неуязвимость
TAI – CHI	25491,2(1) – неуязвимость
TAPPER	41003,0 41004,0
TARGET RENEGADE	62931,201 – время
TELADON	35969,182 65074,182
TEMPEST	33533,0 – игрок 1 33605,0 – игрок 2
TERRA COGNITA	42774,255 45427,255
TERRAMEX	36844,N
TERROMETER	46233,0 46234,0 46235,0
TETRIS 2	60277,255
THE DEEP	50377,0 – бомбы 27038,0 27041,0 53737,0 – жизни / 52817,183 – жизни 50374,N – жизни 40157,201 – защита от ракет
THE'A'TEAM	24874,N
THREE WEEKS IN PARADISE	50066,0 (блок длиной 38582 в COPY-COPY с адреса 26490)
THUNDERBIRDS	38816,60

Название игры	Значения POKES
TILT	36921,0 / 55033,0
TIME MASHINE	41000,24 43278,0 44871,194 46700,60 – время/46213,0
TIME TRAX	39690,0 39691,0
TITANIC 1	59195,0 59196,0 59197,0 – жизни 58692,24 58734,24 58874,24 58916,24 58529,24 59043,0 – кислород
TITANIC 2	59536,0 59537,0 59538,0 – жизни 59389,24 58791,24 59035,24 59077,24 59217,24 59259,24 – кислород
TOI 1	50297,0
TOI 2	49808,0
TOI 3	49693,0
TOI 4	49485,0
TOMAHAWK	24601,0
TOOBIN	52910,0
TOP CAT	64786,182 – время / 64796,182 – время 62996,0 62997,24 – энергия
TOWER OF EWIL	59386,0
TRANSFORMERS	58199,0 58200,0 58201,0 – защита 55717,0 – оружие 54293,0 – энергия
TUBE MIX	55546,201 – время
TUJAD	37097,201 – нет маленьких монстров
TURBO BOAT	58795,N – жизни 51009,201 – неуязвимость
SIMULATOR	
TURBO DRIVER	27529,N 27798,0
TURBO THE TORTOISE	50312,N 51937,0 51938,0 51939,0 51940,0 / 46428,N – касания 48088,N – жизни
TURTLES	45218,0 – время 34449,255 49390,255 43463,255 – энергия
TUSKER	38627,0 – жизни 41800,0 41801,0 – неуязвимость
TUTANKHAMUN	27279,N (1...224) – жизни 27783,0 – жизни 31581,239 31593,87 31597,79 31601,95
TWIN TURBO V8	39150,0 – время / 42006,N
TWIN WORLD	60906,167 60942,167
U.C.M.	46108,N
U.N.SQUADRON	59087,N
ULISES	35057,0
VATMAN	49861,0 – нет врагов (конец игры) 49916,250 – жизни
VENOM STRIKES BACK	45809,201
VENTURAMA	53629,0 53630,0
VICTORY ROAD	34791,0 39386,N – игрок 1 34795,0 39403,N – игрок 2 / 39386,0 – между USR / 39386,N / 39400,0 между USR
VIDEO POKER	45021,N – деньги
VIDEO POOL	33802,0 33803,0 33804,0 / 38753,0 / 44780,0

Название игры	Значения POKES
VINDICATOR	35972,0 – нет монстров 36074,0 – монстры не стреляют 38388,0 – неуязвимость 33448,0 – жизнь 1 части 34139,0 – жизнь танка 34203,0 – бессмертие самолёта 34116,N – бомбы, ракеты
VIRUS	37835,0 – мощные бомбы 44945,0 – бессмертие
VIXEN	51789,0
VIXEN 2	51794,0
WAR	36333,N
WATERMIL	26144,N – жизни 24751,0 – жизни 25206,201 – время
WDW 2	48719,255 48724,255
WEC LE MANS	26018,0 25874,0 25845,201 / 33050,0 33051,0
WELL'S FORGO	53658,255 – жизни 53737,N – увеличение очков в N раз
WEST BANK	52518,0 – жизни 52507,201 – проход уровня
XECUTOR	48272,183 – жизни 45418,0 45419,0 – никого нет
YOGI BEAR 3	32785,N – жизни / 37232,201 – время 33569,0 33570,0 33571,0 33572,0 – жизни
ZIG ZAG	43564,0
ZOMBI	45183,36 45184,0 45190,194 – время 53024,201 – защита от укуса зомби
ZYTHUM	51836,201 – защита от ям 54727,201 – неуязвимость
НЛО ВРАГ	38693,0 – защита от криссалидов 39199,0 – оружие 44425–28: 255...255 –
НЕИЗВЕСТЕН	очень много денег
ОТРЯД БЫСТРОГО	41432,183 – топливо
РЕАГИРОВАНИЯ	/ 40917,0 – бессмертие 41550,0 – топливо 40974,0 – патроны
ПРИКЛЮЧЕНИЯ	59904,N – монеты
БУРАТИНО (РУС.)	

ПАРОЛИ И СОВЕТЫ

ALIENS (+7 levels)

2 – 7324G
3 – 2727H

4 – 1506E
5 – 3761H

6 – 7100D
7 – 7123G

BATTLE COMMAND

Число перед паролем означает количество пройденных миссий:

0 9999999999
1 AZZ39899PU
2 BNX4979955
3 HARU96995H

4 SAMN96999W
5 QONHT599PY
6 JAAOTP99TU
7 JTOJTOT9HC

8 QMQATNTTT8
9 CHJITNTT4Z
10 PPSMTLTTUY

BITE THE DUST

START
NOODLE
BUFFET
GOLFBALL
COURGETTE

BUNKBED
BLANCMANGE
GLOBULE
TURNIP
GOBLIN

WALLABY HAEMOGLOBIN
GRUNT GOURMET
ELBOW GOBLET
YELLOWBANANA GIBLETS
SNAPDRAGON

В этой программе обнаружен "жучок": как известно, программа не позволяет использовать редактор, чтобы изменить уровни с перечисленными выше кодами. Введя код ASYMPTOTE, Вы выйдёте в меню редактирования.

CHASE H.Q.

В режиме выбора клавиш набрать слово: SNOCKED. В левом верхнем углу должна появиться надпись: "TEST:ACTIVE". Это означает, что Вы вошли в режим выбора блоков игры. При нажатии на клавишу "1" можно загружать 2 – й блок, при нажатии на "2" – третий, "3" – четвертый. Кроме того, в этом режиме обеспечивается бессмертие.

CONFUSED

2 – METABOLISM
3 – SYNTHESIZE
4 – WELLDEFJEF
5 – SKODALOAD

6 – PROCESSORS
7 – RIMRAMROMS
8 – SINCLAIRE
9 – DIRECTORYS

10 – YUTILITIES
11 – ATTRIBUTES
12 – CENTURIANS
13 – MICRONAUTS

DRAGON BREED (R – TYPE 2)

Для неуязвимости нажать EDIT (CS+1) (пауза), затем вместе Q W E R (по другим данным – C.S. W E R). Бордюр сменит цвет с чёрного на синий. Выход в игру – CAPS LOCK (CS+2). Эти манипуляции с клавиатурой приведут к бесконечной жизни (но для каждого догружаемого уровня процедуру придётся повторять).

EARTH SHAKER (русифицированная версия)

Вступление в игру
Ванна с пузырями
Голова в облаках
Ледяная страна
Алмазы в глубине
Два сердца
Главный ликвидатор
Главный реактор
Урановые рудники
Антигравитация
Нефтяная скважина

Дьявольская грязь
Подземелье
Медовые соты
Головоломка
Скрытый водосток
Трубы из космоса
Западня
Хитрая уловка
Граница
Скрытый родник
Сигнал будильника

Туманный водопад
Огненные стены
Китайский канал
Цифровой замок
Спасение
Твой дом
Укрытие
Ради добра
Без дураков

LEMMINGS (Disked by SHISOFT 3.11.92)

1		21	NFHBAKLFDJ	41	BAKJODIJEM
2	IKHNDHBCCX	22	NHBIKLLGDY	42	MKHM LIBKEN
3	OHNDHBADCU	23	HBEOLLNHDJ	43	OJODIBALES
4	JNDHBINECO	24	BINLNFHIDX	44	HODIBMNMEM
5	NDHBAKLFCW	25	BAKJMFHJDK	45	MLIBEKL NEM
6	DHBIKLNGCP	26	IKHMMNHBKDJ	46	LIBMKLMOEV
7	HBAONLDHCM	27	OHOFHBALDQ	47	IBANNMLPEP
8	BMNLLLHICO	28	JMNHBINMDQ	48	BMNNODIQEW
9	BEKHMLHJCR	29	MNHBEJLNDL	49	BEKJNFIBFK
10	IKHODHBKQ	30	FHBMKNOODK	50	IJL NIBCFU
11	OHODHBALCN	31	HBENLOFPDL	51	NJL NIBADFR
12	HMLHBMOMCQ	32	BIOLMNHQDX	52	JL NIBINEFK
13	ODHBAKNNCR	33	BEKHLLIBEL	53	LNIBEKLFFW
14	LHBIKLMOCO	34	IKHLDIBCEY	54	FIBIKNNGFX
15	HBANLMLPCK	35	NHLLIBEDEQ	55	IBENNNFHFX
16	BMNNODHQCT	36	JLLIBINEEX	56	BIOLLNIIFR
17	BAJL NHB DY	37	LLIBAJNFEQ	57	BAKHMNIJFT
18	IJNFHBCDL	38	LIBMKLLGEM	58	IJMNIBKFN
19	OJNFHBADDJ	39	IBAOLLHEV	59	OJOFIBALFV
20	JL NHBINEDX	40	BMOLLLIIES	60	HMNIBMOMFW

Анализ построения паролей наводит на мысль о том, что, возможно, в них содержатся (как, например, и в игре REX 2) какие-то игровые параметры, с которыми игрок переходит на новый уровень. При беглом взгляде бросаются в глаза второй и третий от конца символы. Вас ни на какие раздумья не наводит периодичность их следования? Возможно, любители этой игры обнаружат и другие закономерности.

LOGOS

5 – CODE	30 – MOON	55 – GIFT	80 – WAVE
10 – GAME	35 – ROCK	60 – KICK	85 – YEAR
15 – DISK	40 – SLOT	65 – SAFE	90 – ZERO
20 – HOME	45 – TEST	70 – EXIL	95 – LOVE
25 – CALL	50 – EVIL	75 – TREE	

MAZE MANIA

5 – HARLECH	9 – JUPITER	13 – STAYPUFT
-------------	-------------	---------------

REPTON-1

ASP	EARTHWORM	BASILISK	LEVIATHAN
SCREEN ONE	SEASNAKE	CEPHALOPPOO	OPHIDIAN
CROCODILE	ANEMONE	ANNELID	KING COBRA

RIPTOFF

1 – UHUEALOO!	11 – GLOBBITS	21 – TRI GLASBO
2 – WEE PAUL	12 – OBERKORNIT	22 – FAR 2 TUFF
3 – MONOCHROME	13 – SKREEM	23 – PUDDLEGLUM
4 – MOUSE	14 – FRUSTRATIN	24 – KANGIS BLO
5 – 2061	15 – RAINDROPS2	25 – SPAFOX9
6 – SIMPLE	16 – GALLOSND	26 – HARLEQUIN
7 – MCAULEY	17 – KIT – KAT	27 – BASGUILLAI
8 – YOU B – – – – D	18 – GEEZAHAND	28 – SAWQ BHURD
9 – BAD	19 – FLOSONUMF	29 – BLASNO KLA
10 – LICORISH	20 – GOLGIS PRY	30 – CIRK ROHE

SLABAGE

1 – 1111	11 – 1872	21 – 4957	31 – 8989	41 – 3778	51 – 3355
2 – 3627	12 – 1873	22 – 1956	32 – 1644	42 – 6567	52 – 0999
3 – 5789	13 – 3721	23 – 1957	33 – 5399	43 – 9822	53 – 1357
4 – 3423	14 – 4921	24 – 4822	34 – 2198	44 – 9388	54 – 2468
5 – 1239	15 – 7432	25 – 1933	35 – 2199	45 – 0345	55 – 3398
6 – 2763	16 – 2091	26 – 1764	36 – 2010	46 – 2567	
7 – 4621	17 – 1900	27 – 1935	37 – 7611	47 – 2356	
8 – 8912	18 – 2000	28 – 8462	38 – 4621	48 – 0912	
9 – 9110	19 – 3942	29 – 5465	39 – 7832	49 – 9012	
10 – 0012	20 – 4672	30 – 2245	40 – 7577	50 – 7433	

ПРИМЕЧАНИЕ. Пароли 7 и 38 уровней почему-то совпадают. Поэтому не получается попасть на 38 уровень, вводя пароль – попадаем на 7 уровень.

SPECVENTURE

STAR	ALU#	Chr\$	CROM	LISA	GAME
MAST	SPOK	????	SINC	EARc	difi
QUAE	Bugs	ULAc	Wawe	Leia	PV=K
Z80A	LOST	TRIM	VAGA	Kybr	

SPHERICAL

RADAGAST	SKYFIRE	GLIEP	GUMBACHACHMAL
YARMAK	MIRGAL	MOURNBLADE	ILLUMINATUS
ORCSLAYER	GHANIMA	JADAWIN	

STEG

2 RDNUHCCMGU	5 ODQMFUVLIC	8 NEDGFLDVRL
3 EDOUTIOCKO	6 MEBHETPIAG	9 OEFVHAGHLV
4 HDPFUVLCCM	7 LECGODTRHK	10 PEGTTHIGLD

Сказанное выше замечание о структуре паролей LEMMINGS, видимо, справедливо и для STEG – определенная закономерность прослеживается, если приглядеться ко 2 и 3 символам от начала.

THUNDER BIRDS

2 – RECOVERY	3 – ALOISUS	4 – ANDERSON
--------------	-------------	--------------

TUBE MIX

5 – 4672	15 – 6912	25 – 5634	35 – 7729	45 – 0450	55 – 1299
10 – 8921	20 – 3429	30 – 3342	40 – 6410	50 – 7352	60 – 1979

VENOM STRIKES BACK

Некоторые подробности по поводу использования паролей к этой игре (см. таблицу ниже). Пароли вводятся в меню по клавише "P". Затем игра начинается как обычно, но идти нужно до экрана с телепортами. Когда введены первые три пароля, телепорты соответствуют номерам паролей. Четвертый код работает только, если введены первые три. И телепорты уже можно только угадать, куда какой ведёт.

VIAJE LA TERRA

Пароль 2-го уровня будет: EVAMARIASEFUE. Для третьего уровня пароли будут различаться:

1 игра: SMITHS	1 и 3 игры: REM
2 игра: POGUES	2 и 3 игры: SISTER
1 и 2 игр: GODFATH	1, 2 и 3 игры: LOU REED
3 игра: SUGAR S	

X – OUT

Войдя в режим выбора и оснастки корабля, не спешите его (корабль) выбирать, а наведите курсор на пиктограмму "СТРЕЛКА ВНИЗ" и нажмите "ОГОНЬ". На поле появится довольно – таки "навороченный" корабль. Не вздумайте на него ещё что – нибудь "навесить", а просто начинайте игру. Когда у Вашего корабля кончится энергия, то он исчезнет, и появится новый – слабый и без дополнительного оборудования, однако игра не закончится. После 10–15 смертей Вам может повезти и Вы получите более "крутой" корабль. Эти действия нужно проводить после каждого уровня, и ничего не покупать на заработанные деньги.

Пароли и советы для других игр сведены в таблицу.

Название игры	Пароли и советы
BALLBREAKER	Когда тебя убивает ALIEN, нажать "S" и начать прохождение комнаты сначала
BASE	Пароль: NEWRM
BATMAN 3	Нажать MICK для перехода на следующий уровень
BOMBSCAR	Пароли: ZEPHA QUART DELTA XYLEM NITRO CRYPT YTRON ASTRA
BREAK THRU	Если со старта игры нажать ПРЫЖОК и не отпускать – игра окончится минут через десять
COMET GAME (THE)	Нажать BREAK, затем вместе 9 и 0 – перейдете в субигру, назначение которой неизвестно
DEVIANTS	TROLLDOR ZACARONT DEVIANTS DENZIENT CEVBURUS XANTHIUS DIZZIDUS GUZZLOID VESTRONA JABBADOR CYBENIA
DIZZY 7 (CRYSTAL KINGDOM)	Для бессмертия в главном меню одновременно нажать: S T A R
FIRE GEAR	Пароли: 2 level – 502 3 level – 738 4 level – 164 5 level – 940
IMPACT (русская версия)	Пароли для русской версии 11 – МАША 21 – САША 31 – НИНА 41 – ДИМА 51 – ЮЛИЯ 61 – ПАША 71 – ЛЕНА
INDI JONES (LAST CRUSADE)	Написать DOT в титульном экране, после этого: CS+1 – рестарт уровня CS+2 – переход к следующей точке CS+3 – следующий уровень
MAIN BLOW	Для бессмертия во время игры нажать: C H I P
MEGA-BUCKS	Телефонные номера комнат: 38312 16349 87327 59112 68495
MINE	Пароль: QQQ11
MORTADELO Y FILEMON	Для бесконечных жизней нажать одновременно C U P
MOUNTAIN BIKE RACER	Во время игры нажать одновременно Z E P L I N – бесконечное время; ещё раз – возвращает ход времени
PUZZNIC	Набрать HELPMЕ (без пробела) в меню RETRY – получите бесконечное RETRY и время; нажать BTR – следующий уровень
ROAD RUNNER	В меню выбора управления нажмите: BJR – бесконечные жизни LOVE – переход на конечную фазу; JACQUI – ????
ROBOCOP 2	Задать клавиши: MSYULE; в игре нажать GTI
ROCK'N'ROLLER	В режиме DEMO – игры нажмите H E L P и не отпускайте до тех пор, пока не появятся авторы игры – бордюр мигнёт красным и Вы получите бесконечные жизни

Название игры	Пароли и советы
SACRED SWORD	Пароли: 2 – АТАНУАЛПА 3 – МАЧУПИЧЧУ
SAM CRUISE	Номера телефонов 8864 7162 9408 3183 4090 1776 5278 6839 7337 2522 6142 3979 4715 7832 1369
SAVAGE 1	Для бесконечных жизней: когда появится надпись "PROBE SOFTWARE..." нажмите одновременно F E R G
SCRAMBLE SPIRITS	В меню выбора управления нажмите M и 1 игрок станет неуязвимым
SEA HAWK	Пароль 2 уровень: 090166 (090156???)
SKATE USA	Пароль следующего уровня: THIS BOY CAN WAIT
TEENAGE MUTANT	Во время игры нажать CS/G/0/1 – вкл./выкл. режима бесконечной энергии
HERO TURTLES	Для бесконечных жизней во время гибели (OOOCH!) нажать одновременно SS/P/D
THREE WEEKS IN	Пароли:
PARADISE	2 – RECOVERY
THUNDERBIRDS	3 – ALOYSIUS
	4 – ANDERSON
TITANIC 2	Код для полной версии: SUSIE
VENOM STRIKES BACK	Пароли: MAYHEM, TRANSMOGRIFY, VALKYR, PETALS OF DOOM
VIGILANT	В таблице рекордов ввести GREEN CRYSTAL
VINDICATOR	Пароли:
	2 часть – VALSALVA MANOEUVRE
	3 часть – EUSTACHIAN TUBES
ZYNAPS 128	Для бессмертия нажать "4"(пауза), затем "C"

КОМПЬЮТЕРНАЯ НОВЕЛЛА

УЗНИК ШАХТЫ

Компьютерная новелла по игре THUNDERBIRDS

© Андрей Андреев, г. Шадринск, 1995.

Когда Алан открыл глаза, в комнате было еще темно, но настойчивый гудок телефона не умолкал. Бросив взгляд на будильник, он прикинул, кому же приспичило звонить ему в 5 часов утра, и снял трубку.

- Тревога! Готовность номер 1. - проговорила быстро трубка голосом дежурного.

То, что услышал Алан, заставило его, наконец, проснуться. Быстро одевшись, приняв душ, он, прыгнув в свой джип, через 15 минут был уже около штаб-квартиры прославленного спасательного отряда "Громовые птицы". Алан направился прямо в кабинет шефа. В приёмной уже с нетерпением переминался с ноги на ногу Брайнс. Быстро поздоровавшись со своим напарником, Алан вместе с ним зашел к своему непосредственному начальнику Биллу Гильберту.

- Ну что ж, ребята, для вас есть новое задание, - шеф уселся на свое место и продолжил - В одной из шахт в Северной Каролине случилась беда: завалило человека, и только вам под силу вызволить его.

- Какие проблемы, - Алан весело улыбнулся - Всегда, пожалуйста!

- Когда можно приступить? - спросил Брайнс, поправив очки.

- Уже все готово, можете приступить, - ответил Гильберт.

Выйдя из кабинета шефа, Алан огорченно промолвил:

- Ну почему все неприятности случаются именно в понедельник? - Брайнс его не слушал, он стремительным шагом направлялся к взлётной площадке вертолёта. Едва поспевая за приятелем, плёлся и Алан.

Во время полёта они молчали. Наконец, спустя два часа вертолёт приземлился около злополучной шахты.

Встретивший их человек назвался начальником службы безопасности и проводил до штаба спасательного отряда. Получив подробные инструкции и карту шахты, друзья решили разделить: Алан попробует пробраться к шахтеру с поверхности, а Брайнс - из подрытого туннеля. Узнав о том, что в шахту прекращена подача электроэнергии, спасатели запаслись фонарями, а Алан, непонятно для чего, взял с собой масленку.

Огромных усилий стоило Брайнсу пробраться по узкому, скользкому тоннелю до входа в штольни.

"Все бы ничего, если бы куда-то не подевалась эта карта", - так размышлял Брайнс, пробираясь по этажам все выше и выше, уворачиваясь от падающих сталактитов. Остановившись передохнуть, спасатель разносил в пух и прах владельца этой шахты за то, что он пожалел денег на ремонт уже давно устаревшего оборудования. Под ногами подозрительно зачавкало. Брайнс посмотрел вниз - поблескивая в свете фонаря, на него медленно, но неуклонно наступала вода. Вот тут-то его прорвало. Тихий, интеллигентный молодой человек, обладающий исключительными манерами поведения, матерился, как самый настоящий сапожник.

Проклиная всё на свете, Брайнс бросился к лестнице. Неожиданно он почувствовал, что летит. Наконец, достигнув, пола штольни, Брайнс осветил предмет, ставший причиной его падения. Это оказался самый обыкновенный гаечный ключ. Повертев предмет в руках, у Брайнса созрел план: "Надо починить помпу, не до такой уж степени скупердяй этот владелец шахты - она где-то должна быть". С этой мыслью спасатель побежал искать лифт. В одном из закоулков он наткнулся на огромный, слегка заржавевший молоток. "В хозяйстве пригодится", - подумал Брайнс и взял его с

собой. В этот момент фонарь предательски замигал, С большим сожалением спасатель бросил его на скользкий от всё прибывающей воды пол.

Подгоняемый мыслью о затоплении он увидел лифт, но, как на зло, трос застрял между камней. Поэтому кабина лифта не двигалась с места. Не теряя ни секунды, Брайнс принялся освобождать трос. Тут пришлось слегка поработать молотком. "Ну, наконец-то", - пробормотал он, вскочив в начинающий медленно двигаться вверх лифт. В нём спасатель позволил себе немного отдохнуть.

Тихо скрипя заржавевшим тросом, лифт поднялся до следующего уровня. К своему удивлению, Брайнс обнаружил, всего в нескольких шагах от себя отличный плунжер и небольшой ящик с динамитом. Взяв это, он выкинул ненужный теперь молоток - тихий всплеск напомнил ему о надвигающейся опасности. Брайнс вскочил обратно в лифт, и начался утомительный подъём наверх.

Наконец показался долгожданный уровень №8. Здесь, как помнил спасатель, согласно карте, должна была находиться водооткачивающая помпа. Вот и она, отливающая блеском высветленных поверхностей, Не теряя времени, Брайнс кинулся к ней.

- Так я и думал: контакты реле совсем расшатались. С этими словами он принялся за работу, ловко орудуя гаечным ключом. Спустя несколько минут, помпа довольно заурчала и начала откачивать воду из нижних штолен. Тревожно смотря в темноту, Брайнс подумал о фонаре, который оставил внизу. "Ну что ж, придётся возвращаться", - подумал он и побежал назад к лифту. Благополучно спустившись вниз, спасатель забрал фонарь, который, как ни странно, ещё работал.

Теперь к Брайнсу вернулось его привычное спокойствие. Без особого труда он преодолевал один за другим уровни злополучной шахты. Остановиться его заставил обвал: огромный камень преградил дорогу. Протерев очки, Брайнс решил взорвать препятствие, благо все необходимое у него было с собой. Заложив динамит около камня, он аккуратно растянул, захваченную с собой проволоку. Отойдя подальше, спасатель замкнул контакты плунжера - раздался оглушительный взрыв. Воздушной волной Брайнса откинуло назад метров на десять. Слегка придя в себя, он зашагал дальше.

- Что за чудо: от камня и следов не осталось, - изумился Брайнс. Пройдя ещё немного, он обнаружил застрявшего шахтера.

- Крепко же тебя завалило, приятель, - сказал Брайнс, пытаясь освободить его.

- Бесполезно,- грустно промолвил пострадавший - Я уже пробовал, а тут ещё эта проклятая вагонетка!

- Ничего, дружище, потерпи, сейчас должен сюда пробраться мой напарник, и мы что-нибудь придумаем.

* * *

Алан же явно не торопился с выполнением поставленной задачи. Выкурив ещё не одну сигарету, он наконец зашел в штольни. В нос ему ударил неприятный затхлый запах шахты. В этот момент почва под ногами слегка задрожала. "Что за черт, я же не пил с утра ничего, кроме кофе!" - подумал Алан и решил узнать, что поэтому поводу думает начальник безопасности шахты. Однако, подойдя к выходу, он обнаружил, что дверь от сильного толчка заклинило. "Теперь совсем здорово", - Алан в сердцах громко чертыхнулся, но делать было нечего, и он отправился на подмогу Брайнсу.

С трудом пробираясь по темному лабиринту коридоров шахты, спасатель наткнулся на первое препятствие: прямо перед ним зияла огромная глубокая яма. "Надо что-нибудь придумать", - решил Алан, опуская на землю уже довольно надоевшую ему масленку. Оглядевшись внимательно вокруг, он обнаружил лестницу-стремянку. В глазах у него загорелись огоньки. "По-моему она для чего-нибудь да сгодится", - промелькнула мысль в сознании Алана. Схватив лестницу, он побежал к яме. Через неё, едва успев остановиться, спасатель перекинул стремянку и по такому импровизированному мостику перебрался на другой край ямы, не забыв прихватить с собой масленку.

Алан продолжал спускаться всё ниже и ниже. Проходя мимо совсем заржавевшей вагонетки, он мысленно сказал в адрес владельца шахты что-то недоброе, а вслух пробормотал: "Надо же, они до сих пор пользуются такой рухлядью!". Пройдя примерно метров сто по тускло освещенному тоннелю, спасатель неожиданно чуть не упал в ещё одну яму. "Как же через неё перебраться?" - вслух рассуждал Алан, удрученно идя обратно. Около вагонетки он устало остановился. "А что если..." - мелькнуло у него в сознании, и уже, не давая себе полного отчёта, он попытался сдвинуть с места упрямую вагонетку. Вдруг Алан остановился и со всего размаху шлепнул себя по лбу. От такого резкого движения из кармана у него выпала маслёнка. Тонкая струнка масла полилась на заклинившую ось. Вот тут-то Алана осенило. Он схватил маслёнку и быстро смазал все заржавевшие детали. Слегка толкнув вагонетку, Алан ощутил, что она начинает стремительно набирать скорость. Теперь, крепко схватившись за её борт, он стремительно мчался вперед...

Только спустя 10 минут Алан поверил, что жив. Тщательно осмотрев себя, он удивленно заметил: всё на месте.

- Надо было мне в каскадеры идти, а не в спасатели, - горько усмехнулся Алан и решил посмотреть то, что осталось от вагонетки. Остановившись недалеко от ямы, он осветил фонарём небольшое углубление: прислоненный к стене стоял отбойный молоток. "Ну вот, это уже нечто современное", - хмыкнул Алан и прихватил его с собой. Свернув налево, спасатель заметил лифт. Не раздумывая ни секунды, Алан шагнул к нему и уверенно нажал на кнопку "вниз". Слегка подрагивая, кабина лифта начала опускаться. Спустившись вниз, Алан зашагал по тоннелю и был приятно удивлён: свет здесь горел нормально. Выкинув ненужный теперь фонарь, он спустился в склад, более похожий на обыкновенную свалку. В куче всякого старья и хлама Алан обнаружил относительно новый моток шланга.

Взвалив его на плечо, спасатель поплёлся к компрессору, стоявшему неподалёку. Соединив всё это вместе, Алан начал ломать стену отбойным молотком. Через пять минут работа была закончена, но в самый последний момент один острый осколок вонзился в голень. Алан с трудом удержал в себе крик боли, чуть не потеряв сознание. Вдруг на его лице появилась некое подобие улыбки: к нему на помощь спешил Брайнс.

Теперь, когда все трое были в сборе, Брайнс подал сигнал готовности наверх. Спустя некоторое время к ним опустился трос. Спасатели быстро прикрепили к нему вагонетку. Трос сильно натянулся, но она не двигалась с места, и только после нескольких томительных минут ожидания удалось чуть-чуть сдвинуть её.

- Ну, давай, родимая, - шептал Брайнс. От волнения у него все лицо покрылось испариной.

- Пошла! Идёт! - радостно кричал Алан. Он помог выбраться шахтеру, у которого от радости текли слезы на чёрном от грязи и копоти лице.

- Теперь уже всё, - устало сказал Брайнс. - Алан, отправляйся тоже сейчас наверх, троих трос не выдержит. Я доберусь по низу.

- Удачи! - крикнул Алан. - Но лучше подожди второго рейса. В ответ на это второй спасатель показал свои часы с неустанно бегущей секундной стрелкой.

- Держу пари. Я успею за 10 минут выбраться из этого склепа, - прокричал Брайнс вслед поднимающимся вверх людям.

- О'кэй! - донеслось ему, и многократное эхо повторило это. Брайнс стремительно преодолевал один за другим уровни, пробегая по длинным, мрачным тоннелям шахты.

Смертельно усталый, но счастливый спасатель брёл по направлению к штабу. Легкий вечерний ветерок теребил его волосы, приятная прохлада плавно опускалась на плечи.

Около штаба спасателей, на подножке санитарной машины с забинтованной ногой сидел Алан. Увидев друга, он улыбнулся и сказал:

- Ты как всегда вовремя.

Обессиленный Брайнс присел рядом.

- Я думаю мы с тобой заслужили небольшой отпуск, - продолжал Алан. Брайнс, похлопав его по плечу сказал:

- Посмотрим, что скажет по этому поводу мистер Гильберт.

Вот так закончилась одна из спасательных операций, которую успешно выполнили Алан и Брайнс, спасатели из легендарной группы "Громовые птицы".

SOS или "Русалочка" в беде

Пятого июня погода выдалась великолепная. Яркое и нежное солнце рассылало свои лучи всем загорающим на пляже Майами. Легкое дуновение ветерка можно было принять за воздушный поцелуй. Именно о таком деньке мечтал Алан Томпсон уже вторую неделю. Но вместо того, чтобы флиртовать на пляже с девочками, он на бешеной скорости мчался на своем джипе - "Рэнглер" вдоль побережья.

"Ну почему все несчастья случаются в мой выходной?" - думал он, с легкостью вписываясь в крутые повороты. Ответ Алан находил просто - он входил в команду "Громовые птицы", а спасатели из этой группы первыми прибывали туда, где случилась беда.

Спустя полчаса "Рэнглер" Алана уже стоял на стоянке, а хозяин автомобиля нервно выкуривал уже вторую сигарету в кабинете начальника одной из крупнейших военных баз на побережье, слушая рассказ полковника Саймона. Изредка он бросал взгляд на своего нового напарника Гордона Блэкстоуна.

А ситуация складывалась критическая. В каких-то 50 милях от знаменитых пляжей Майами затонула экспериментальная атомная подводная лодка с ядерными боеголовками на борту. По крайней мере, экипаж эвакуировали...

- Самое неприятное для нас является то, что началась утечка из реактора и через затопленные отсеки, возможно распространение токсинов, а это означает гибель всего живого на расстоянии многих миль, - продолжал свой доклад полковник.

- Разрешите, я прерву вас, - сказал глава секретной службы Стерлинг, - по последним сведениям, китайцы очень заинтересовались теми торпедами, что на борту. Их корабли непрерывно курсируют недалеко от места аварии, и придётся, похоже, демонтировать их в кратчайшие сроки.

- Короче господа, - прервал их Алан, - нам придётся не только обезвредить реактор, выкачать воду из отсеков, но и снять эти торпеды?!

- Да, демонтаж не входил в ранее обговоренные планы, но, чёрт возьми, эти китайцы совсем обнаглели... Вы не волнуйтесь, вам нужно будет только имитировать пуск торпед из носовых и кормовых аппаратов - и "узкоглазых", как ветром сдует.

На этом совещание закончилось. Алан и Гордон направились выбирать снаряжение. Без тени колебания оба натянули снаряжение для подводного плавания - мало ли, что. Гордон взял с собой упаковку противорадиационных таблеток. Алан же вместо этого прихватил с собой свежий номер журнала "PLAYBOY".

А тем временем в закрытом доке техники заканчивали подготовку подводного катера нового поколения - "Слэйер", у которого было одно очень примечательное свойство - он мог стыковаться с субмаринами, приблизительно так же, как и космические корабли.

Получив последние указания и напутствия, друзья с трудом разместились внутри "Слэйера", он ведь был рассчитан на одного лишь человека. На нём спасатели отправились к терпящей бедствие субмарине.

Подводное путешествие прошло благополучно. Из панорамного иллюминатора открывался чудесный вид. Впечатление портила одна очень надоедливая акула, которой очень уж хотелось попробовать на вкус спасателей из "Громовых птиц".

Со стыковкой проблем не было: автоматика работала, как часы. Парни из команды "Громовые птицы" оказались внутри огромной стальной громадины с таким ласковым названием - "Русалочка".

Действие радиации спасатели почувствовали сразу. Гордон без промедления заглотил всю пачку противорадиационных пилюлей. А Алан же, тяжело вздохнув, пошел обследовать подводную лодку. Тем временем Гордон карабкался по лестнице на капитанский мостик. Поднявшись, он увидел капитана - последнего члена экипажа на корабле.

- Вам может потребоваться это, - сказал обладатель капитанской фуражки и протянул магнитную карточку-пароль.

Резко развернувшись, капитан быстрым шагом направился ко второму шлюзу. Гордон опомнился только тогда, когда закрылась и начала наполняться водой шлюзовая камера.

- Но у вас ведь нет гидрокостюма! - завопил спасатель.

- Капитан умирает вместе со своим судном, - донесся глухой голос из шлюзовой камеры, - все кончено!

- Нет! Ещё совсем ничего не кончено! Подождите!!!

Но ответа уже не последовало. Гордон присел, обхватив голову руками.

В это время Алан бегал от каюты к каюте, ища драгоценные противорадиационные пилюли. Спустившись на один уровень ниже, он обнаружил то, что так усердно искал. Безо всякого промедления Алан так же, как и Гордон, быстро проглотил все таблетки и почувствовал, что воздействие радиации на него временно прекратилось. Теперь без лишней суеты он продолжил осмотр подводной лодки.

Успокоившись, Гордон поплёлся искать своего напарника, чтобы рассказать о случившемся. В отсеке №24 он нос к носу столкнулся с Аланом.

Томпсон довольно спокойно отнесся к встревоженному рассказу друга: за несколько лет службы в "Громовых птицах", он насмотрелся всякого, но сердце болезненно сжалось - было все-таки жаль капитана.

- Слушай, Гордон, поищи пока ту штуку, к которой подойдет эта магнитная карточка: я имею в виду головной компьютер.

- Алан, ты думаешь, нам удастся через эту машину остановить реактор?

- Я надеюсь на это. К тому же ты ведь специалист по компьютерным системам.

- Да, тогда насколько я знаю ни в одну систему нельзя войти с одним паролем, нужен ещё, как минимум, один.

- Вот и отлично, тогда я поищу вторую магнитную карточку. Порешив на этом, друзья разошлись. Занятый теперь больше поисками компьютера, а не судьбой бедного капитана, Гордон обследовал несколько отсеков. Заглянул он, также в каюткомпанию, а, пройдя ещё пару отсеков, зашел в тупик.

В то время, когда Гордон искал центральный компьютер, Алан медленно опускался в люк. Следующий уровень был затоплен водой. Спасатель медленно плыл вдоль поручней.

- Надо бы откачать воду из отсека, - вслух сказал Томпсон, нажимая на кнопку запуска водооткачивающей помпы. Но ни какого эффекта не было, за исключением того, что в открытый рот попала вода... "Пойду-ка я погляжу, что там, в следующем отсеке", - подумал спасатель, открывая дверь в следующий отсек. Вспоминая инструктаж, Алан продолжал идти, а точнее плыть в выбранном направлении: на этом уровне находился носовой торпедный аппарат. Оказавшись в этом отсеке, спасатель подошел к приборной панели. Небольшой пассаж по кнопкам управления и в заключение нажатие на рычаг-сбрасыватель - "холостая" торпеда направилась к цели. "Высокие технологии - это здорово! Отсек заполнен водой, а все функционирует!" - так думал Алан, тцетно ища кнопку водооткачивающей помпы. Наконец он нашел её и нажал... И опять никакого эффекта!

"Делать мне здесь нечего, будем пробираться обратно", - с такой мыслью Алан начал двигаться обратно, по пути нажимая на все выключатели помп. Лишь когда спасатель нажал на последнюю кнопку, началась долгожданная откачка воды. Минут через пять Алан уже снял маску и спокойно разгуливал по отсекам, ещё недавно заполненным водой. В одном из них, Томпсон обнаружил под решеткой странный кусок пластика, похожий на магнитную карточку. После того, как

Алан взял её в руки и протер от грязи и ила, у него не осталось сомнений в том, что сейчас держит то, что необходимо Гордону. Не медля ни секунды, спасатель рванулся, больно ударяясь об острые углы, на поиски своего напарника.

Гордон в это время уже пытался разобраться с системой кодировки пароля. За таким занятием застал его Алан.

- Эй, Блэкстоун! Я нашел второй пароль, давай попробуем вместе. Без лишних разговоров друзья подошли к пультам и начали набирать код доступа. Но тут их постигла неудача: компьютер не принимал пароль Алана.

- Эх, чёрт! - с этими словами Томпсон кинул ненавистную карточку на пол и начал её с остервенением пинать.

- Алан, успокойся! Если она не подошла, то где-то все-таки лежит та, которая нам нужна!

- Ладно, - Алан взял себя в руки, - пойду, поищу какой-нибудь ещё кусок пластика.

С этими словами Томпсон направился к люку, ведущему на нижний уровень. В проходе он силой ударил кулаком о переборку. От неожиданности Гордон вздрогнул и устремил свой взгляд на спускающегося вниз напарника.

Обида постепенно утихла, и Алан обследовал нижний уровень субмарины. Ничего примечательного он не нашел, за исключением кормового торпедного аппарата. Прodelав те же манипуляции, что и носовом торпедном отсеке, спасатель облегченно вздохнул: теперь надоедливые китайцы не будут крутиться вокруг "Русалочки", как шакалы вокруг умирающего льва.

Теперь ему и Гордону оставалось отключить реактор и откачать воду из оставшихся отсеков. Решив заняться сначала последним, Алан начал пробираться в нос подводной лодки. Прodelав долгий путь по умирающему гиганту, спасатель очутился перед люком, ведущим на последний, самый нижний уровень. Натянув маску и включив подачу кислорода, Алан спустился в люк. С легким плеском вода приняла спасателя в свои объятия.

Сперва Алан пробовал на ощупь отыскать нужную ему карточку, но всё было бесполезно. Поверьте: не очень-то удобно в гидрокостюме и сумраке искать маленький пластиковый предмет.

К сожалению, эти поиски ни к чему не привели, кроме того, что спасатель открыл двери всех отсеков и больно стукнулся о проплывающий мимо огнетушитель. Чисто механически Алан взял его одной рукой, а другой - потер ушибленное место. И самое главное: он обнаружил пробоину. Томпсон медленно подплыл к её зияющему краю и посмотрел вниз. А там, в каких-то трёх метрах, виднелось дно, покрытое слоем липкого ила, которого, вообще-то, было полно и в затопленных отсеках субмарины.

Теперь Алан уже не раздумывал, а начал действовать: осторожно взявшись за острые края пробоины, он выскользнул в океан. Стараясь не поднимать тучи ила, спасатель медленно брел по дну и тщательно обыскивал каждое подозрительное место.

Но тут Алан посмотрел влево, и неприятный холодок пробежал по его спине. "Проклятье! - чуть не закричал Томпсон, - как же я сразу её не заметил!"

К нему на огромной скорости приближалась какая-то "рыбка". Но вскоре можно было различить у нее вертикальный плавник на спине и несколько рядов острых зубов.

"Акула!!! - завопил Алан, поправляя съехавшую набор кислородную маску. - Да ведь это та же самая акула, которая доставала наш "Слэйер". Клянусь!"

Тем временем гигантская рыба приближалась. До субмарины было очень далеко, просто не успеть. Алан до боли сжал кулаки. В правой руке он ощутил железную рукоятку огнетушителя. Не раздумывая ни секунды, Томпсон запустил этот предмет прямо в раскрытую пасть акулы. Огнетушитель исчез внутри необъятной рыбы, словно растворился. Акула продолжала двигаться, но тут случилось неожиданное: на глазах её стало "распирать". Несколько секунд и огромная рыба с вздутым животом валялась около ног Томпсона. Теперь она была не опасна. Алан поспешно отвернулся от представившегося его глазам зрелища: из разорванного брюха акулы вываливались

внутренности, окруженные серо-белой пеной. На ходу, борясь с рвотой, спасатель кинулся подальше от этого места.

Всего в пятидесяти метрах от мертвой акулы он обнаружил то, что давно искал: из ила виднелась ещё одна магнитная карточка. Схватив её, Алан рванулся обратно к подводной лодке, при этом обошел исполинскую рыбину издалека.

Снова спасатель очутился внутри субмарины. Теперь уже медленно, с видом победителя, Алан плыл из отсека в отсек, включая водооткачивающие помпы. Через некоторое время из отсеков была откачена почти вся вода.

Теперь оставалось только остановить реактор, и все. Думая об этом, Алан бежал в тот отсек, в котором безуспешно бился с компьютером Блэкстоун.

Гордон за последний час совсем измучился: злобредный компьютер неустанно требовал второй пароль, и никакие ухищрения ему не помогали. Обессиленный Блэкстоун откинулся на стуле.

Алан появился неожиданно, и, прислонившись к переборке, растягивая слова сказал:

- Ну что, партнер, ничего не выходит. Да?

Ответом послужил гневный взгляд Гордона. Томпсон, не торопясь, подошел и устроился напротив своего напарника.

- А у меня что-то для тебя есть - промолвил Алан, сладко улыбаясь и доставая заветную карточку. - И даю слово, если она не подойдет, то я повторю геройский поступок капитана.

Увидев её, Гордон радостно взвизгнул, выхватил из рук Томпсона пароль и вставил во второй терминал.

- А теперь, - быстро проговорил Блэкстоун, обращаясь к своему партнеру, - ты будешь повторять мои действия на втором компьютере, и не забудь вставить первый пароль.

- Не беспокойся.

Работа закипела. Две пары рук стремительно летали по клавишам. Спустя пять минут была запущена новая программа, а реактор был заблокирован.

Теперь уже не опасаясь радиации, спасатели заглянули в следующий отсек, где был расположен реактор, отделенный толстенными свинцовыми перегородками.

Перед их взором предстала - святая святых: ядерный реактор. Действие радиации здесь все ещё ощущалось, тем более что таблетки уже не предохраняли спасателей от воздействия излучения.

По молчаливому согласию Алан и Гордон устремились к своему подводному катеру. По дороге Томпсон рассказал своему другу о битве с акулой.

- Нет ничего проще, - комментировал Блэкстоун, - огнетушитель ударился обо что-то твердое внутри твоей "рыбки" и начал забивать её брюхо пеной, а когда не хватило места - разорвало.

На миг Алан вспомнил ту сцену с акулой, и его передернуло.

Усевшись в катер, друзья вывели его из шлюза и сделали прощальный круг около "Русалочки", которая теперь, возможно, снова вернется в строй.

- Знаешь, Алан, - сказал Гордон, указывая на подлодку - а она огромная!

Томпсон только хмыкнул в ответ.

В закрытом доке напряженно ждали возвращения маленького "Слэйера". Собралось много высокопоставленных чиновников. Прилетел даже шеф команды "Громовые птицы" - Билл Гильберт.

И вот долгожданный катер появился на поверхности. Из него выскочили двое молодых людей и принялись разминать затекшие мышцы. Глядя на них, мистер Гильберт подумал о том, что ребята заслужили отпуск, а особенно Алан.

"Жизнь прекрасна, - думал Томпсон, направляясь к своему джипу. - Вот Гордон сейчас вернется к жене и детям, а я присоединюсь к какой-нибудь цыпочке на пляже и целую неделю проваляюсь с ней, занимаясь пустяками".

* * *

ПРИЛОЖЕНИЕ

Предлагаем вниманию читателей документацию к авторскому программному пакету, распространяемому на дистрибутивном диске АС 17. С его помощью Вы сможете, например, создавать полноценные инструкции к Вашим программам, придавая им логическую завершенность. Однако мы думаем, что область применения пакета шире - он позволяет легко и просто выпускать компьютерный журнал, принимая на себя все технические проблемы и оставляя Вам исключительно творческую роль по его содержанию и оформлению (Для практической работы Вам понадобится лишь пара редакторов: текстовый плюс графический).

PRESENTATION MASTER



© Сергей Ханцис, г. Ижевск, 1995.

От автора.

Пакет программ, который я имею честь Вам представить, называется PRESENTATION MASTER. Его цель - демонстрация текстовой и графической информации. Другими словами, PRESENTATION MASTER (PM) позволяет создать что-то вроде "информационного ролика" - электронную инструкцию, рекламный каталог и т.п. Особое внимание уделено показу текстов. Собственно, вначале я и хотел сделать только демонстратор текстов. К этому меня побудила острая необходимость в электронных инструкциях для своих программ - распространять программы с описанием на бумаге дорого, а описание в виде текстовых файлов неудобно для пользователя, т.к. не у каждого может быть соответствующий редактор. Распространять же с текстами чужой редактор без разрешения автора не будет ни один уважающий себя программист.

Итак, цель была ясна - написать демонстратор текстов, позволяющий организовать более-менее приличную электронную инструкцию. Для этого необходимо, чтобы демонстратор:

- ✓ имел удобное управление (желательно перенастраиваемое) - перемещение по тексту вверх и вниз на строку и страницу, в начало и в конец текста;
- ✓ был универсальным, т.е. "понимал" бы тексты различных текстовых редакторов и различные кодировки текстовых файлов;
- ✓ имел развитые средства вывода: печать в окно любых размеров, несколько шрифтов, настраиваемые цвета, средства выделения отдельных слов;
- ✓ если это не слишком сложно, позволял вставлять в текст иллюстрации;
- ✓ отличался высоким качеством исполнения.

При обдумывании этой задачи встал вопрос, как задавать параметры для демонстратора - окно вывода, имя демонстрируемого файла, шрифт, цвета, клавиши управления и т.п. - ведь эти параметры должен устанавливать автор текста, а не пользователь, для которого этот текст предназначен. Выход представлялся таким: автор (или организатор демонстрации) создает файл специального формата, в котором указывает все необходимые параметры, а демонстратор при запуске прочитает этот файл и

выполнит все, что там написано. С развитием этой идеи стало понятно, что этот специальный файл должен представлять собой обычный текст, в котором записаны команды, которые демонстратор должен выполнять. Причем эти команды могут не только устанавливать параметры для текстового демонстратора, но и производить другие полезные действия - например, рисовать окна, организовывать меню, демонстрировать картинки и т.п. Таким образом, демонстратор превращается в некий интерпретатор специального языка, в котором показ текста - лишь одна из многих функций. Осталось лишь спроектировать набор команд.

Так родился язык PM-LAN - язык системы PRESENTATION MASTER. Большая часть данного описания системы посвящена командам PM-LAN. По опыту работы с первой версией системы в набор команд будут внесены соответствующие изменения. Поэтому у меня большая просьба - присылайте свои отзывы. Авторам ценных предложений при их желании вышлю следующую версию PM. Замечания и предложения можно присылать по адресу:

426063, г. Ижевск, ул. Восточная, д. 36, кв. 68. Тел. (3412)76-19-13 Ханцис Сергей

Язык PM-LAN

Язык PM-LAN предназначен для составления демонстрационных роликов в системе PRESENTATION MASTER. Это узкоспециализированный язык, он не подходит для написания обычных программ. В нем Вы не найдете каких-либо особо гибких конструкций или типов данных. Кроме того, этот язык интерпретируемый, поэтому он не поражает своим быстродействием. Да это и не требуется для задач данного класса.

По структуре PM-LAN - командный язык. Это значит, что каждое действие вызывается соответствующей командой, в которой, если необходимо, указываются параметры. Такое построение больше всего похоже на структуру MS-DOS'овских batch-файлов. Здесь нет ни функций, ни процедур - только команды. Программирование на таких языках несколько напоминает программирование на ассемблере - ведь ассемблер, по сути, тоже командный язык, только не проблемно-, а машинно-ориентированный.

Программа на PM-LAN создается в обычном текстовом редакторе и записывается в обычный текстовый файл (требования к текстовым файлам см. в главе "Технические подробности"). Каждая команда занимает отдельную строку текста. Все параметры отделяются друг от друга и от команды пробелом (или пробелами не делает различий между прописными и строчными буквами, поэтому команды можно писать, выделяя начала образующих слов. Так, KeyPress для интерпретатора ничем не отличается от KEYPRESS или keypress. Естественно, это не относится к тексту внутри кавычек.

В своих программах Вы можете использовать переменные, которые могут содержать либо целое числовое значение в интервале 0..65535, либо текстовую строку длиной до 62 символов. Всего имеется 26 переменных с именами от A до Z (имена переменных, как и команды, можно писать и маленькими, и большими латинскими буквами). Кроме непосредственного указания имени, к переменным можно адресоваться как к одному массиву из 26 элементов через псевдофункцию VAR. Выражение var 0 заменяет переменную A, var 1 - B, и так до var 25 - Z. В качестве параметра var может быть и переменная: var S. Можно составить и вложенную конструкцию, например: var var R - что означает: переменная, номер которой (от 0 до 25) записан в переменной, номер которой указан в переменной R. Вначале все переменные не содержат никакой информации и считаются неопределенными. При определении переменной интерпретатор сам разберется, какого типа данные определяются - строка или число. Тип содержащихся в переменной данных можно менять неоднократно.

В программу можно вставлять комментарии в круглых скобках. Если интерпретатор встречает открывающую скобку '(', он пропускает все вплоть до закрывающей скобки ')'. Комментарии могут располагаться где угодно: в конце строки, между параметрами, перед командой и т.д. Поскольку внутри комментария игнорируются и переходы на новую строку, можно одну команду "растянуть" на несколько строк:

```
Command (первый параметр:
) 68 (второй параметр:
) "текст" (конец команды)
```

Что-то же самое, что и Command 68 "текст".

Теперь перейдем к описанию команд PM-LAN. Если у Вас возникнут какие-либо неясности с синтаксисом, изучите входящую в состав пакета демонстрационную программу (она записана в файле autoexes). В ней содержится много примеров использования почти всех команд системы.

Для начала обговорим некоторые соглашения. Пусть в дальнейшем

<var> означает переменную любого типа или неопределенную;

<num> означает числовой параметр (числовая переменная или целая числовая константа в диапазоне от 0 до 65535), например: 0 654 009543;

<str> означает строковой параметр (строковая переменная или строковая константа в кавычках длиной до 62 символов, например: "текст" "ZX Spectrum" "" (пустая строка);

<vnum> означает числовой параметр, передаваемый только через переменную;

<vstr> означает строковой параметр, передаваемый только через переменную;

если какой-либо параметр является необязательным, он заключается в квадратные скобки, например: [<num>];

если в команде требуется указать назначение конкретного параметра, то тип этого параметра указывается после назначения через двоеточие, например, так: Command <имя_файла:str>;

у некоторых команд есть параметры-спецификаторы (чаще необязательные), которые тем или иным образом изменяют действие этих команд. Спецификаторы записываются по тем же правилам, что и команды. В этом случае при описании синтаксиса таких команд спецификаторы указываются в том виде, в каком они

есть, например: Command <str> [Key];

если в команде возможен выбор одного из вариантов, то эти варианты разделяются символом '|', например: Command ON|OFF - что означает два варианта использования команды: Command ON и Command OFF.

Чтобы было проще разобраться в использовании команд, после их описания приводятся примеры. Итак, приступим.

```
SET <var> <num>|<str>
```

Присвоить переменной var значение второго параметра.

```
Set A 35
```

```
Set var K "string"
```

```
Set I R
```

Set var T var S (в дальнейшем возможность адресации к переменным по var будет подразумеваться)

```
ADD <vnum> <num>
```

```
ADD <vstr> <str>
```

Сложить значения переменной и второго параметра и присвоить результат этой же переменной. В первом варианте применения команды выполняется арифметическое сложение по модулю 65536, переполнение игнорируется. Во втором - выполняется конкатенация (слияние) строк.

```
Set A 20
```

```
Add A 15 (теперь A=35)
```

```
Set L "Presentation "
```

```
Add L "Master" (теперь L="Presentation Master")
```

SUB <vnum> <num>

Вычесть из значения переменной значение второго параметра и присвоить результат этой же переменной. Заем игнорируется, т.е. если из 0 вычесть 1, получится 65535.

```
Set A 100
Sub A 68 (теперь A=32)
```

MUL <vnum> <num>

Умножить значения параметров и присвоить результат переменной. Если результат умножения превышает 65535, переменной будет присвоено максимально допустимое числовое значение - 65535.

```
Set A 123
Mul A 321 (теперь A=39483)
```

DIV <vnum> <num>

Разделить значение переменной на значение второго параметра и присвоить целую часть частного этой переменной. При попытке деления на 0 выполнение программы будет прервано и выведено сообщение об ошибке (подробнее об ошибках см. главу "Сообщения об ошибках").

```
Set A 128
Div A 10 (теперь A=12)
```

RANDOM <vnum>

Присвоить переменной <vnum> случайное числовое значение в интервале от 0 до старого значения <vnum>.

```
Set R 1234
Random R
```

CODE <var> <str>

Присвоить переменной <var> значение кода первого символа строки <str>.

```
Code A "1.0TR" (теперь A=49)
Set S "D"
Code S S (теперь S=68)
```

VAL <var> <str>

Перевести строку <str> в числовую форму и результат присвоить переменной <var>. Перевод оканчивается на первом нецифровом символе. Если строка не представляет собой символьную запись числа, выполнение программы прерывается и выводится сообщение об ошибке.

```
Set B "12345Ni"
Val B B (теперь B=12345)
```

ASK <var>

Присвоить переменной <var> символ, соответствующий нажатой на клавиатуре клавише. Выполнение программы приостанавливается до тех пор, пока пользователь не нажмет какую-либо клавишу. Примечание:

по техническим причинам ввод символа "" (двойная кавычка) заблокирован во всех командах ввода.

```
Ask K (например, если нажмут клавишу T, то K="t",
-----"----- CS/Z, то K="Z")
```

READ <var>

Прочитать строку с клавиатуры и записать её в переменную <var>. В процессе ввода можно использовать следующие управляющие клавиши:

- CS/0 - забой последнего введенного символа;
- CS/1 - переключение РУС/LAT;
- CS/2 - фиксация верхнего регистра;
- Enter - конец ввода.

При нажатии Enter позиция печати не переносится на новую строку. Примечание: русские символы вводятся в альтернативной кодировке.

```
Read S
```

READNUM <var>

Прочитать строку с клавиатуры, перевести её в число и поместить это числовое значение в переменную <var>. При вводе заблокированы все клавиши, кроме цифр, Enter и CS/0 (действие последних двух клавиш см. в команде READ). Команда устроена таким образом, что пользователь не может, ввести некорректное число или не ввести ничего.

```
ReadNum L
```

KEYPRESS <var>

Присвоить переменной <var> значение 1, если на клавиатуре нажата какая-либо клавиша, и 0 в противном случае. Выполнение программы не задерживается; клавиши Caps Shift и Symbol Shift опрашиваются наравне с другими.

```
KeyPress F
```

UPCASE <vstr>

Привести все буквенные символы в строке <vstr> к верхнему регистру. Команда выполняется корректно как для латинских, так и для русских букв в альтернативной кодировке.

```
Set N "Привет пользователям ZX Spectrum!"
UpCase N
(теперь N="ПРИВЕТ ПОЛЬЗОВАТЕЛЯМ ZX SPECTRUM!")
```

LINELEN <num>

Ограничить буфер ввода для команды READ <num> символами. Значение <num> может изменяться в пределах от 1 до 62. В команде READ пользователь не сможет ввести символов больше, чем установлено командой LINELEN. На команду READNUM данная установка не распространяется, эта команда всегда позволяет вводить не более 5 символов.

```
LineLen 8
```

TEXTOUTSPACE <x1:num> <y1:num> <x2:num> <y2:num>

Установить текущее поле вывода для текста. Координаты поля задаются в знакоместах 8x8 точек от левого верхнего угла экрана: x1,y1 - координаты левого верхнего угла поля, x2,y2 - правого нижнего. Диапазон значений для x1 - 0..31, y1 - 0..23, x2 - x1..31, y2 - y1..23. При определении поля позиция печати перемещается в его левый верхний угол; старое содержимое поля не очищается.

```
Set X 4
TextOutSpace X 1 30 21
```

MODE <num>

Установить режим печати текста в соответствии со значением <num>. <num> может иметь значения 5, 6 или 8, что определяет ширину выводимых символов - соответственно 5, 6 или 8 точек (высота символов всегда равна 8 точкам). Следует учитывать, что в режимах 5 и 6 текст может позиционироваться по горизонтали с дискретностью до 1 точки, а в режиме 8 - только на 8 точек (по атрибутным знакоместам). Зато в последнем режиме символы печатаются намного быстрее.

Т.к. поле вывода текста задается в знакоместах режима 8, возникает вопрос, сколько символов режимов 5 и 6 уместится в поле заданной ширины. Чтобы получить ответ на этот вопрос, разделите нацело ширину поля на 5/8 для режима 5 и на 3/4 для режима 6. Полученное значение и будет равно количеству уместяющихся символов.

```
Mode 5
```

TEXTATTR OFF|<атрибут:num>

Установить атрибут печати текста. Если задан параметр OFF, текст будет печататься без атрибутов, т.е. символы будут принимать цвет, который установлен в данном месте экрана. Атрибут же рассчитывается обычным образом: цвет фона умножается на 8 и к нему прибавляется цвет тона. Кроме того, прибавляется 64 для повышения яркости и 128 для мерцания.

```
TextAttr OFF
TextAttr 105 (105=64+5*8+1 - яркость, голубой фон, синий тон)
```

PRINT [<num1>|<str1>] [<num2>|<str2>] ...

Печатать на экран в текущее поле вывода значения параметров. Количество параметров не ограничено. После печати всех параметров позиция печати не перемещается на новую строку.

```
ReadNum T
Print "Вы ввели число " T
```

NSP

Переместить позицию печати на новую строку. При достижении нижней границы текущего поля вывода содержимое поля скроллируется вверх.

```
NSP
```

LOCATE <x:num> <y:num> [<сдвиг:num>]

Переместить позицию печати в знакоместо с координатами x,y. Координаты задаются в знакоместах текущего режима и отсчитываются от левого верхнего угла текущего поля вывода (левое верхнее знакоместо имеет координаты 0,0). <сдвиг> применим только для режимов 5 и 6 и устанавливает, на сколько точек должен быть смещен выводимый текст вправо (точное

горизонтальное позиционирование). Значение <сдвиг> может изменяться от 0 до значения текущего режима минус 1 (т.е. до 4 или 5 для режимов 5 и 6 соответственно). <сдвиг> также нельзя использовать, если <х> указывает на последнее знакоместо строки в текущем поле (например, если $x=41$ при ширине поля 32 (во весь экран) в режиме 6). Примечание: наличие третьего параметра тогда, когда он недопустим, не ведет к ошибке; в этом случае он просто игнорируется.

```
Locate 21 15
Locate 7 2 2
```

CLS [<атрибут:num>] [GIRD]

Очистить экран и закрасить его указанным атрибутом (если атрибут не указан, закрасить текущим атрибутом). Если атрибут указывается, он становится текущим. При использовании спецификатора GIRD экран заполняется мелкой (в одну точку) шахматной сеткой. Команда устанавливает поле вывода размером во весь экран (выполняется TextOutSpace 0 0 31 23) и перемещает позицию печати в его левый верхний угол. Примечание: необходимо различать текущий атрибут, которым производится очистка экрана или поля и ПО которому производится печать текста, и атрибут печати, устанавливаемый командой TEXTATTR, КОТОРЫМ печатается текст.

```
CLS
CLS 15 (15=1*8+7 - синий фон, белый тон)
CLS 56 Gird
CLS Gird
```

CLSPC [<атрибут:num>]

Очистить текущее поле вывода текста и закрасить его указанным атрибутом (или текущим, если не атрибут не указан). Указанный атрибут становится текущим. Позиция печати перемещается в левый верхний угол поля.

```
ClSpC
ClSpC 32 (32=4*8+0 - зеленый фон, черный тон)
```

SCROLL UP|DOWN

Скроллировать текущее поле вывода текста вверх при параметре UP и вниз при параметре DOWN. Освобождающаяся строка очищается и закрашивается текущим атрибутом.

```
Scroll Up
Scroll Down
```

BORDER <цвет:num>

Установить цвет бордюра экрана равным значению <цвет>. Допустимые значения - от 0 до 7.

```
Border 1
```

WINDOW <х:num> <у:num> <ширина:num> <высота:num> <атрибут:num> [TOP] [SHADE] [STORE]

Нарисовать окно в соответствии с параметрами. Координаты и размеры окна задаются в атрибутных знакоместах. <х>, <у> - координаты левого верхнего угла окна, отсчитываемые от левого верхнего угла экрана; допустимые значения: для <х> - от 0 до 29, для <у> - от 0 до 21. Параметры <ширина> и <высота> задают соответственно ширину и высоту окна, допустимые значения - от 3 до

границ экрана. Параметр <атрибут> устанавливает цвета окна и становится текущим атрибутом. Спецификатор TOP указывает, что верхняя строка окна должна быть проинвертирована (например, для размещения заголовка) (инвертирование производится атрибутное). SHADE окаймляет окно справа и снизу "тенью". Наконец, спецификатор STORE устанавливает режим, при котором часть экрана, затиравая окном, сохраняется в памяти для последующего восстановления. Команда переустанавливает поле вывода текста таким образом, чтобы печать производилась внутри окна с отступом в одно знакоместо со всех сторон (если присутствует спецификатор TOP, отступ сверху не делается для того, чтобы дать возможность печатать текст в верхней инверсной строке окна). Как всегда при установке поля вывода, позиция печати перемещается в верхний левый угол поля.

Все параметры окна (в том числе и позиция печати) помещаются в специальный оконный стек. Благодаря этому можно организовать иерархическую оконную систему: открывая окна, Вы последовательно заносите их данные на стек, а затем, удаляя окна в обратном порядке специальной командой (см. ниже), переходите каждый раз к предыдущему окну. Глубина стека такова, что можно открыть одновременно не более 14 окон, причем неважно, использовался спецификатор STORE для запоминания информации с экрана или нет. Кроме того, каждое окно с сохранением экранной информации организуется в памяти как файл, а одновременно в памяти может находиться не более 25 резидентных файлов (подробнее об организации файлов см. раздел о кэшировании диска в команде TEXTVIEW), не говоря уже о том, что для сохраняемой информации может просто не хватить памяти. Во всех случаях нехватки памяти и переполнения стека программа останавливается и выдается сообщение об ошибке.

```
Window 2 5 20 7 15
Window 19 11 4 3 32 Shade Store
Window 0 0 32 24 41 Top
```

DELWIN

Вернуться к предыдущему окну. Параметры предыдущего окна снимаются с оконного стека и становятся текущими. Если часть экрана под окном сохранялась, она будет восстановлена. (Подробнее см. описание команды WINDOW).

```
DelWin
```

MENU <vnum> <x1:num> <y1:num> <длина1:num> [<x2> <y2> <длина2>] ...

Организовать меню в соответствии с параметрами. Команде в качестве параметров передаются координаты <x>, <y> и <длина> каждого поля выбора. Пользователь с помощью клавиш ВЛЕВО, ВПРАВО, ВВЕРХ, ВНИЗ (см. команду KEY) может установить условный маркер на любое из перечисленных полей (при этом оно инвертируется), причем неважно, в каком порядке перечислены поля в команде - система сама разберется, какое поле правее, левее, выше или ниже при переходе маркера. Координаты и длина каждого поля задаются в атрибутных знакоместах относительно левого верхнего угла экрана; количество полей может быть от 1 до 21. Первоначально маркер устанавливается на поле с номером, равным значению переменной <vnum> (первое поле в списке имеет номер 1). Если пользователь нажмет ВЫБОР, действие команды окончится, а переменная <vnum> получит значение номера текущего (т.е. выбранного) поля. При нажатии клавиши ВЫХОД эта переменная получит значение 0. Примечание: команда MENU не печатает никаких сообщений на экране, поэтому Вы сами должны напечатать заголовки каждого поля как раз в том месте экрана, где располагается соответствующее поле.

```
Set A 1
TextOutSpace 1 1 9 9
```

```
Print "ДА"
NSP
Print "НЕТ"
Menu A 0 1 4 0 2 5
```

DELBAR

Восстановить инвертированное командой MENU поле выбора. При выходе из команды MENU выбранное поле остается инверсным. Команда DELBAR восстанавливает это поле. Примечание: команда MENU не проверяет, было ли до ее вызова инвертировано какое-либо поле от предыдущего меню; поэтому DELBAR надо вызывать до организации следующего меню. Если Вы этого не сделаете, поле так и останется инвертированным.

```
DelBar
```

TEXTVIEW <имя_файла:str>

Показать текст, записанный в файле с указанным именем. Это самая сложная и самая главная команда системы, и она требует отдельного описания. См. главу "Демонстрация текста".

```
TextView "pmhelp1"
```

KEY UP|DOWN|LEFT|RIGHT|FIRE|HOME|END|QUIT [<символ>] ["-"] ["-"]

Определить клавиши выполнения указанной функции для команд TEXTVIEW и MENU. Определяемые функции таковы:

UP - ВВЕРХ	FIRE - ВЫБОР
DOWN - ВНИЗ	HOME - В НАЧАЛО ТЕКСТА
LEFT - ВЛЕВО	END - В КОНЕЦ ТЕКСТА
RIGHT - ВПРАВО	QUIT - ВЫХОД

Каждой функции можно определить не более трех клавиш. Клавиши указываются соответствующим им символом, например, клавиша T - символом t, CS/T - T. Клавиши, не имеющие своего печатного символа, можно определить, указав символ '\ ' и после него - код клавиши, например CS/1 - \7. Если ни одной клавиши не будет указано, действие данной функции будет заблокировано.

```
Key QUIT (отключение функции выхода)
Key Left o 5 \8 (клавиши O, 5, CS/5)
Key Down axk (клавиши A, X, K)
Ask K
Code K K
Key FIRE \13\32\K (клавиши ENTER, SPACE и клавиша, нажатая
пользователем)
```

GRAPHOUTSPACE <x1:num> <y1:num> <x2:num> <y2:num>

Установить поле вывода графики в соответствии с указанными координатами. Координаты задаются в атрибутных знаках и отсчитываются от левого верхнего угла экрана: <x1>, <y1> задают левый верхний угол поля, <x2>, <y2> - правый нижний. Допустимые значения такие же, как в команде TEXTOUTSPACE.

```
GraphOutSpace 1 1 30 22 (поле с отступом 1 от границ экрана)
```

SCRVIEW <имя_файла:str> [<атрибут:num>] | [COMPRESSED]

Вывод графического изображения из файла <имя_файла> на экран. Вывод осуществляется в текущее поле, установленное командой GRAPHOUTSPACE. При указании в команде атрибута изображение выводится не с собственными цветами, а с указанным атрибутом. Спецификатор COMPRESSED позволяет просматривать скомпрессированные изображения. Декомпрессия производится встроенным в файл изображения декомпрессором: файл загружается в свободную память и запускается с начального адреса. Декомпрессор может изменять любые регистры и статус (но не тип) прерываний. Вызов декомпрессора осуществляется с запрещенными прерываниями и с загруженным в регистр ВС стартовым адресом. Важно, чтобы процедура декомпрессии была перемещаемой и не использовала, кроме себя и экрана, никакой памяти. Примечание: при выводе скомпрессированной картинки поле вывода графики игнорируется и вывод производится на весь экран.

```
ScrView "demols"
ScrView "screen" 56 (56=7*8+0 - чёрным по белому)
ScrView "dizzy1" Compressed
```

SPRVIEW <имя_файла:str> <x:num> <y:num> <ширина:num> <высота:num> <номер:num>

[<атрибут:num>] Показать спрайт, записанный в указанном файле. Координаты вывода <x>, <y> задаются в атрибутивных знаках относительно левого верхнего угла экрана (поле графического вывода (см. команду GRAPHOUTSPACE) игнорируется). <ширина> спрайта задается также в знаках, а вот <высота> - в точках. Система выводит спрайты, закодированные в самом распространенном формате: сначала последовательно слева направо и сверху вниз располагаются байты раstra, затем - атрибуты. Если высота спрайта не кратна 8, атрибуты должны присутствовать и для "лишних" точек. Например, при высоте спрайта 19 (2*8+3) атрибуты указываются для трех линий. Параметры, указанные в команде, должны совпадать с параметрами отображаемого спрайта. Если в команде указан <атрибут>, то считается, что атрибуты в спрайте не закодированы, и спрайт выводится с указанным атрибутом. Параметр <номер> сообщает номер спрайта в файле (в одном файле может быть до 256 однотипных спрайтов; они должны следовать один за другим без разделителей). Первый спрайт (располагающийся в начале файла) имеет номер 0.

```
SprView "demolp" 11 9 4 32 0 5
SprView "Exolon 1" 20 17 3 40 1
```

PUTPIC <имя_файла:num> <x:num> <y:num>

Показать иллюстрацию, записанную в указанном файле (файл имеет расширение <P>). Показывается иллюстрация, созданная входящей в пакет PRESENTATION MASTER утилитой PM-ILLUSTRATOR. Главная цель этой команды - вставка иллюстраций в текст, о чем см. описание команды TEXTVIEW, но можно и просто вывести иллюстрацию на экран. При выводе поле графики игнорируется. Параметры <x>, <y> указывают атрибутивные координаты левого верхнего угла иллюстрации; все остальные параметры записаны прямо в файле. По ширине иллюстрация должна умещаться в экране, а по высоте она может выходить за его нижний край.

```
PutPic "F1B" 4 12
```

GOTO <метка>

Выполнить безусловный переход на указанную метку. Имя метки может состоять из любых русских и латинских букв, цифр, специальных символов (кроме разделителей - пробела, конца строки и т.п.). Определяется метка указанием двоеточия ':' и сразу после него - имени метки, например:

:метка

В имени метки, в отличие от команд, различаются прописные и строчные буквы. Метка, как и команда, занимает отдельную строку текста. Если в тексте программы встретятся две или несколько одинаковых меток, GOTO совершит переход на первую из них. В отличие от многих алгоритмических языков, в которых использова-

ние безусловного перехода считается недостатком, в языке РМ-LAN такой переход является необходимым средством, т.к. нет другого способа организовать цикл, ветвление и т.п. Примечание: GOTO - довольно "мед-

ленная" команда, причем время ее исполнения прямо пропорционально расстоянию от начала текста до метки, т.к. в поисках этой метки приходится сканировать весь текст программы.

```
GoTo Label@1Ю
```

```
. . .
```

```
:Label@1Ю
```

ON <num> GOTO <метка0> [<метка1>] [<метка2>] ...

Перейти на метку с номером, равным значению <num>. Переход на первую метку (расположенную после GOTO) осуществляется при <num> равном 0, на следующую - при 1 и т.д. Количество меток не ограничено.

```
Set G 2
```

```
On G GoTo exit m1 m2 m3 m4 (перейдет на m2)
```

IF [NOT] <str1> = <str2>

IF [NOT] <num1> < | = | > <num2>

Установить флаг разрешения интерпретации в соответствии со значением логического выражения. Значение операций сравнения обычное: = - равно, < - меньше, > - больше. Строки можно проверять только на равенство, числа - на любое из перечисленных условий. Если значение выражения истинно (при использовании спецификатора NOT - ложно), программа выполняется дальше как обычно, иначе все последующие команды игнорируются, пока не встретится ELSE или ENDIF (см. ниже).

```
Read L
```

```
UpCase L
```

```
If L="ВПЕРЕД"
```

```
. . .
```

```
Read K
```

```
If K < 12
```

```
. . .
```

```
If not K>32768 (если меньше либо равно ...)
```

ELSE

Инвертировать флаг разрешения интерпретации. Если раньше интерпретация была запрещена, она будет разрешена, и наоборот. Таким образом, ELSE может встречаться и без соответствующего IF. Особое внимание обратите на тот факт, что флаг интерпретации всего один, т.е. нельзя составлять вложенных конструкций из IF; точнее, такие конструкции могут не дать ожидаемого результата. Так, например, несколько следующих один за другим IF просто заменяют логическое И нескольких

условий. Так же и ELSE относится не к последнему IF, как в обычных языках, а ко всем условиям сразу.

```
Set A 20
Random A
If A=17
Print "Я так и думал!"
Else
Print "Что-то тут не так..."
```

ENDIF

Разрешить интерпретацию. Подробнее см. IF и ELSE.

```
Ask C
If C="y"
Set A 34
Random A
If A<7
Print "Вот это удача!" ( C="y", A<7 )
Else
Print "Ну и ладно!" ( C="y", not A<7 либо not C="y" )
EndIf
```

ALT

Включить альтернативную кодировку текста. Действует как на текст, выводимый командой TEXTVIEW (для чего в основном и применяется), так и на всю остальную выводимую на экран текстовую информацию.

```
Alt
```

ASCII

Включить кодировку текста, применяемую в редакторе TLW2+ (см. также команду ALT). Если последние две команды применяются для установки режима для TEXTVIEW и кодировка файла, содержащего программу, отличается от кодировки просматриваемого этой командой текста, обязательно после TEXTVIEW поставьте команду включения "родной" кодировки программы.

```
ASCII
```

SOUND ON|OFF

Включить (при параметре ON) или выключить (при параметре OFF) генерацию звуковых эффектов командами TEXTVIEW, ASK, READ, READNUM, MENU, а также всеми специальными командами генерации звука (см. ниже).

```
Sound ON
Sound OFF
```

BREAK ON|OFF

Включить или выключить (соответственно ON и OFF) реакцию на нажатие клавиши CS/BREAK при работе с диском. Если реакция включена, при нажатии BREAK генерируется сообщение об ошибке, иначе нажатие BREAK не приведёт ни к какому результату.

Break On
Break Off

ERROR ON|OFF

Включить или выключить (ON или OFF) генерацию сообщений об ошибках интерпретатора. Если данный режим включен, при обнаружении программной ошибки система остановит программу, выдаст соответствующее сообщение и покажет команду, в которой обнаружена ошибка. Если же режим выключен, при обнаружении ошибки компьютер просто сбросится (продолжать программу при ошибке невозможно, т.к. это почти обязательно привело бы к цепи вытекающих из нее других ошибок). Сказанное относится и к сообщению о логическом конце программы (см. также главу "Сообщения об ошибках").

Error on
Error off

CLICK

Генерировать короткий щелчок.

Click

DRUM

Генерировать шумовой эффект длительностью около 0.1 сек.

Drum

BEEP

Генерировать звуковой тон длительностью около 0.5 сек.

Beep

PAUSE [<время:num> [KEY]]

Выдержать паузу продолжительностью <время>/50 секунды. Если <время> равно нулю или не указано, пауза будет бесконечной; при максимальном значении данного параметра (65535) длительность паузы будет приблизительно 21 минута 51 секунда. Если указан спецификатор KEY, пауза окончится либо по истечении указанного времени, либо при нажатии любой клавиши; при этом до начала отсчета времени ожидается отпущение всех клавиш.

Pause (полное зависание)
Pause 0 (то же)
Pause 50 (задержка 1 сек.)
Pause 3000 key (задержка 1 мин. с опросом клавиатуры)
Pause 1 key (фактически – ожидание отпущения клавиш)

END

Конец программы. Если интерпретатор встречает эту команду, программа завершается и выдается соответствующее сообщение (если это разрешено командой ERROR). Наличие данной команды в конце программы не обязательно. Расположение END в середине программы не означает, что команды ниже нее выполняться не могут; ее можно обойти, например, с помощью IF.

```
Ask F
If F="y"
End
EndIf
```

QUIT

Сброс системы. После инициализации запускается TR-DOS.

```
Quit
```

CHAIN <имя_файла:str>

Передать управление программе, записанной в файле с указанным именем. Управление вызывающей программе при окончании вызванной не возвращается. При передаче управления сохраняются значения всех переменных, установок команд KEY, BORDER, TEXTATTR, MODE, TEXTOUTSPACE, GRAPHOUTSPACE, ALT/ASCII, LINELEN, SOUND, BREAK, ERROR, а также содержимое экрана и позиция печати; кэш диска и оконный стек очищаются.

```
Chain "Manual"
Read N
Chain N
```

Демонстрация текста

Демонстрация текстов является, пожалуй, самой главной функцией системы PRESENTATION MASTER. Мощный встроенный демонстратор текстов позволяет создавать хорошо оформленные и иллюстрированные технические описания, рекламные каталоги и даже электронные журналы.

Текст для просмотра набирается в любом имеющемся текстовом редакторе (требования к текстовым файлам см. в главе "Технические подробности"). Для экономии памяти текст рекомендую набирать без левого отступа, т.к. отступ от края экрана можно сделать, установив соответствующее поле вывода. Количество символов в строке следует делать не больше, чем может уместиться в заданном поле вывода (см. также MODE), т.к. выходящий за правый край текст показан не будет (т.н. режим UnWarp).

Большие возможности по оформлению текста предоставляют встраиваемые в текст команды. Команды записываются в фигурных скобках, причем между открывающей скобкой и командой не должно быть пробела, например:

```
{Command "name" 4 R}
```

Если команда опознана и допустима к использованию внутри текста, она выполняется, а все, что находится внутри скобок и сами скобки на печать не выводятся; например, демонстратор, встретив строку

```
Любимая игра Алексея - {TextAttr 14}BUMPY{TextAttr OFF}.
```

напечатает: Любимая игра Алексея - BUMPY.

причем первые три слова и тире будут напечатаны текущими атрибутами, слово BUMPY - желтым по синему, а точка в конце - без атрибутов. Действие конкретной команды распространяется на весь текст после нее, вплоть до следующей такой же команды. Так, в данном примере для всего последующего текста будет установлен режим выключенных атрибутов. Если же команда неопознана или запрещена к использованию внутри текста, скобки и текст внутри них будут напечатаны как обычно; например строка

Для начала он {Михаил} решил разобраться с командой {PAUSE}

будет выведена на экран в том виде, как она есть. Однако следует учесть, что ошибка в параметрах выполняемой команды приведет к обычному сообщению об ошибке.

К использованию в тексте разрешено только три команды:

TEXTATTR - установка цветов печати;
 MODE - установка режима печати;
 PUTPIC - вставка иллюстрации.

Синтаксис и действие первых двух команд полностью идентично соответствующим внутренним (программным) командам (это, собственно, просто одни и те же команды). На последней же команде следует остановиться подробнее.

PUTPIC <имя_файла:str> <отступ:num>

Вставить в текст иллюстрацию, созданную утилитой PM-ILLUSTRATOR и записанную в файл с указанным именем. <отступ> задается в атрибутных знаках от левого края текущего поля вывода текста; иллюстрация по ширине должна полностью "влезать" в поле вывода. Начало иллюстрации располагается уже в строке, в которой расположена команда, поэтому в этой строке не должно быть печатного текста, за исключением случая, когда иллюстрация полностью затирает этот текст. Такое ограничение связано с тем, что демонстратор не гарантирует печать текста там, где есть иллюстрация (например, при просмотре текста снизу вверх). Эффект затирания текста изображением можно использовать, например, для вывода сообщения о загрузке иллюстрации, как это делается в демонстрационной программе:

```
{TextAttr 130}Идет загрузка...{TextAttr OFF}{PutPic "demo1i" 0}
```

Для того, чтобы не задерживать просмотр постоянными загрузками с диска, в системе применено кэширование диска. Это означает, что пока хватает свободной памяти, файлы (это относится не только к файлам иллюстраций) загружаются в буфер на свободное место, и только при нехватке памяти для загружаемого файла система начинает удалять из буфера самые "старые" файлы. Каждая команда загрузки проверяет, не находится ли уже в буфере требуемый файл, и если это так, файл загружается прямо из памяти. Сказанное вовсе не означает, что в памяти будет находиться две копии одного и того же файла - рабочая и сохраненная в буфере, просто команде загрузки будет возвращен указатель на находящийся в памяти файл.

В любом случае, в памяти полностью должны уместиться исполняемая программа, демонстрируемый текст, вся информация, сохраненная командой WINDOW (см. описание этой команды) и обе иллюстрации, фрагменты которых частично скрыты, а частично выступают с верхнего и нижнего края поля вывода в данный момент (если такая ситуация возможна в Вашем тексте). Иначе говоря, иллюстрация, соприкасающаяся с или выходящая за верхнюю или нижнюю границу поля, не может быть удалена из памяти для ее освобождения.

Это же относится и ко всей вышеперечисленной резидентной информации.

Если с логикой работы демонстратора Вам все понятно, перейдем к собственно просмотру текста. Перед вызовом демонстратора, командами ALT или ASCII установите нужную кодировку текста, установите начальные цвета, режим печати и поле вывода.

При просмотре можно перемещаться по тексту вверх и вниз на строку или страницу, а также в начало и в конец текста. Перемещение на строку выполняется клавишами ВВЕРХ и ВНИЗ (см. команду KEY); на страницу - ВЛЕВО и ВПРАВО. Переход в начало и в конец текста - клавиши В

НАЧАЛО и В КОНЕЦ. Выход из демонстратора и продолжение программы осуществляется при нажатии клавиш ВЫБОР или ВЫХОД. При выходе восстанавливаются начальные значения атрибута и режима печати; поле вывода не очищается.

Напоследок - один совет. Не злоупотребляйте переключением цветов и режимов. Не говоря о том, что пестрый текст хуже воспринимается (как, впрочем, и однотонный), команды смены режима и цвета чуть-чуть задерживают вывод текста, т.к. требуют времени на интерпретацию. Практически задержка не ощущается при использовании не более одной-двух команд в строке, да и то не в каждой.

Сообщения об ошибках

Все ошибки, определяемые системой PRESENTATION MASTER, можно разделить на два типа: ошибки интерпретатора и ошибки дисковой операционной системы (DOS). Любая ошибка ведет к останову программы и в конечном итоге к выходу из системы.

Генерацию сообщений об ошибках интерпретатора можно отключить с помощью команды ERROR. В этом случае компьютер будет сбрасываться сразу при обнаружении ошибки (это может понадобиться для того, чтобы не озадачивать ничего не понимающего пользователя какими-то мудреными сообщениями; эти сообщения нужны практически только составителю программы, а когда программа отлажена, генерацию сообщений лучше выключить). Если же печать сообщений включена, при ошибке будет показан фрагмент листинга программы с указанием команды, в которой произошла ошибка, появится соответствующее ситуации сообщение, а затем, после нажатия любой клавиши, компьютер все равно сбросится. Ниже перечислены все возможные сообщения об ошибках интерпретатора.

Синтаксическая ошибка

Возникает при невозможности определить смысл команды или параметра.

Недопустимый параметр

Появляется во всех случаях, когда указанный параметр не удовлетворяет условиям команды: не тот тип, выход за пределы диапазона и т.п.

Требуется команда

Возникает при попытке записи спецификатора или переменной на место команды.

Метка не найдена

Выводится, когда команда GOTO не может найти указанную метку.

Неопределенная переменная

Появляется при попытке использования неопределенной переменной там, где это недопустимо.

Несовместимые переменные

Возникает при попытке выполнения одной операции над параметрами различного типа - строкой и числом (в команде ADD или IF).

Недопустимый спецификатор

Имеет место при использовании спецификатора, не предусмотренного данной командой.

Требуется параметр

Выводится, если команде не хватает параметров.

Нет рамки на экране

Возникает при попытке использования команды DELBAR, когда на экране нет курсорной рамки меню.

Окно не открыто

Появляется при попытке закрытия окна командой DELWIN, когда на экране нет открытых окон.

Параметр не на месте

Возникает при расположении допустимого параметра в команде не там, где он должен находиться по правилам; например, в конструкции Pause Key.

Требуется переменная

Выводится при попытке использовать константу там, где требуется передавать параметр через переменную.

Слишком длинная строка

Имеет место при попытке определения строки длиннее 62 символов, или если такая строка получается в результате слияния строк командой ADD. Кроме того, это сообщение появляется, если при использовании команды внутри текста не закрыта фигурная скобка.

Нечисловая переменная

Возникает в некоторых случаях, когда через переменную передается строковой параметр вместо числового (например, в команде RANDOM).

Ошибка в записи числа

Возникает, когда обрабатываемая командой VAL строка не содержит в начале символьную запись числа.

Файл не найден

Выводится, если указанный в команде файл не найден на диске.

Мало памяти

Появляется при нехватке памяти для загружаемого файла или для сохраняемой командой WINDOW со спецификатором STORE информации, а также при попытке открыть более 14 окон.

Лишний параметр

Возникает, если в команде указано больше параметров, чем нужно.

Теперь перейдем к дисковым ошибкам. Основное их отличие от перечисленных ошибок интерпретатора в том, что они выводятся в отдельном окне без показа фрагмента листинга и что их вывод нельзя запретить командой ERROR. Кроме того, поскольку это, в принципе, ошибки не системы, а DOS (они просто перехватываются системой PM), соответствующие сообщения выводятся по-английски.

BREAK. Press any key to reset

Нажата клавиша BREAK. Нажмите любую клавишу для инициализации. (Функцию опроса клавиши CS/BREAK можно отменить, см. команду BREAK).

Drive X: disk not present

Отсутствует диск в дисковом X:

Disk error

Дисковая ошибка. Возникает при всех остальных ошибках DOS.

К дисковым ошибкам хоть и с натяжкой, но можно отнести предупреждение о физическом сбое диска. Главная особенность этой ошибки - это то, что после принятия одного из трех предоставляемых на выбор решений можно продолжить работу программы. Иначе говоря, эта ошибка не является фатальной.

Чтобы не перегружать оконную систему и не создавать побочных эффектов, выбор решения сделан в довольно неприглядной форме. При обнаружении сбоя диска раздается звуковой сигнал и в левом верхнем углу экрана появляются буквы A, R, I, что соответствует вариантам: A (Abort) - выход из системы, R (Retry) - повторить попытку чтения сбойного сектора, I (Ignore) - игнорировать сбойный сектор. Выбор производится клавишами ВВЕРХ и ВНИЗ как в обычном меню. После принятия решения затертое место экрана будет восстановлено.

Технические подробности

Распределение ресурсов

Требуемый объем ОЗУ компьютера, Кб.....	48
Область памяти, занимаемая системой PM.....	24576...42524
Количество свободной памяти, байт.....	23011
Режим прерываний.....	включены, IM 1

Характеристики системы PRESENTATION MASTER

Количество выполняемых команд.....	51
Количество служебных слов, воспринимаемых интерпретатором.....	91
Количество переменных.....	26
Количество типов данных.....	2
Максимальное количество одновременно открытых окон.....	14
Максимальное количество файлов, хранимых в кэш-памяти.....	25
Количество шрифтов.....	3 x 256 символов

Запуск системы и создание автостартующей демонстрации

Для запуска системы PRESENTATION MASTER запустите файл PM 1.0. После загрузки система ищет файл autoexex с текстом исполняемой программы и если его не находит, выдает соответствующее сообщение и завершает свою работу. Иначе сразу же начинается выполнение программы.

Если Вы хотите создать автостартующую демонстрацию на отдельном диске, перепишите туда файлы PM 1.0, autoexex и все файлы с текстами, иллюстрациями, картинками и т.п., используемые Вашей программой. Затем переименуйте PM 1.0 в boot - и демонстрационный диск готов.

Требования к текстовым файлам

Как уже не раз отмечалось, программу и демонстрируемый текст можно набрать в любом текстовом редакторе, использующем альтернативную кодировку текста или кодировку редактора TLW2+.

При кодировке TLW2+ символы с кодами 13 и 14 обозначают конец строки, 128 - переключение на русский знакогенератор, 129 - на латинский. Все остальные символы, коды которых не попадают в диапазон 32...127, игнорируются. Считается, что в начале текста установлен латинский знакогенератор.

При использовании альтернативной кодировки в текст можно включать любой из 256 символов, КРОМЕ символа с кодом 24, который ни в коем случае не должен встречаться ни в каком тексте.

Независимо от применяемой кодировки, признаком конца строки может быть как код 13, так и пара 13/10 (а при кодировке TLW2+ - еще и 14).

Система сама не определяет используемую кодировку текста, её нужно устанавливать командами ALT и ASCII (см.).

Несколько ограничений накладывается на файл, содержащий программу. Если в тексте программы применяется кодировка TLW2+, в первой же строке включите ее командой ASCII, поскольку при запуске включена альтернативная кодировка (о начальных установках см. ниже). Кроме того, в программе не должен встречаться признак конца строки 14 (может использоваться только 13), поэтому при наборе программы в TLW2+ не используйте автоматического перехода на новую строку (можно установить максимальное значение правого поля). И все же, при ошибке фрагмент листинга будет выводиться только в альтернативной кодировке.

Так что мой Вам совет: используйте альтернативную кодировку (по крайней мере, для программы) - и не будете знать проблем. Есть еще один немаловажный аргумент - текст в кодировке TLW2+ печатается чуть-чуть медленнее, т.к. он при этом перекодировается в альтернативную кодировку.

Начальные установки системы

При запуске системы устанавливаются такие начальные параметры, как будто была исполнена следующая программа:

```
CLS 56 (это значит: экран очищен, текущий атрибут - 56
      /чёрный по белому/, T
      extOutSpace 0 0 31 23 /во весь экран/,
      Locate 0 0 /конец комментария/ ) Border 7
GraphOutSpace 0 0 31 23
Key UP q i 9
Key DOWN a k 8
Key LEFT o j 6
Key RIGHT p l 7
Key FIRE 0 \32 \13
Key HOME Q \11
Key END A \10
Key QUIT \7 \14
TextAttr OFF
Mode 6
Alt
LineLen 62
Sound ON
Break ON
Error ON
```

На этом Ваше начальное знакомство с системой PRESENTATION MASTER будем считать оконченным. Смело приступайте к работе и не стесняйтесь экспериментировать!

УСПЕХОВ В РАБОТЕ!

Вы можете приобретать литературу "Инфоркома" по следующим адресам.

- г. Москва, ул. Новый Арбат, д. 2. 19-е отделение связи, 1-ый этаж операционного зала, с 10 до 17 ч., обед с 14 до 15. - Корпункт "ИНФОРКОМа".
- г. Москва, радиорынок в Митино, проезд до ст. метро Тушинская или поездом до пл. Трикотажная (Рижское направл.). Суббота, воскресенье, 9 - 14, место А-37.
- г. Москва, радиорынок в Царицыно, проезд до ст. метро Царицыно. Суббота, воскресенье, 9 - 14.
- г. Барнаул, ул. Э.Алексеевой, 2-а. Фирма "Комэл". Телефон: 76-42-98, факс: 76-42-25. г. Белгород, ул. Октябрьская, д.84, кв.103, "СТУДИЯ КОМПЬЮТЕР"
- г. Белгород, ул. Горького, д. 20, кв. 34, СТУДИЯ "СПЕКТР".
- г. Брянск, пр. Ленина, учебный корпус N2 Брянского строительного техникума, фойе, ТОО "ЗЕФА".
- г. Воронеж, Студия компьютерных игр SAN-SAN. Магазин-салон "ЭЛЕКТРОНИКА", тел. 14-00-73.
- г. Днепропетровск, ул. Шевченко, 34. Фирма "ЭКОС".
- г. Екатеринбург, магазин "СПЕКТРУМ", Главный проспект, 99.
- г. Екатеринбург, ул. Коминтерна, 5, общежитие РТИ, магазин "РАДИСТ".
- г. Екатеринбург, киоски "РАДИОДЕТАЛИ":
 - ул. Малышева, м-н "ОРБИТА",
 - ул. Белореченская, м-н "РАДИОТОВАРЫ",
 - ул. Блюхера, м-н "ОКРУЖНОЙ УНИВЕРСИТЕТ".
- г. Ижевск, Удмуртия. Радиорынок "БЕРЕЗОВАЯ РОЩА".
- г. Иркутск, ул. Урицкого, 2, магазин "ВСЕ ДЛЯ SPECTRUM".
- г. Йошкар-Ола, пл. Ленина, главный корпус МПИ "Модуль".
- г. Калининград (обл.), ул. Минская, д. 10, кв. 10, тел. 43-31-09
- г. Кемерово, магазин "ТЕХНИЧЕСКАЯ КНИГА", ул. Весенняя, 24.
- г. Кемерово, магазин "ОРБИТА", пр. Ленина, 133.
- г. Киров, "Дом науки и техн.", Магазин-салон "МАРС", ул. Производственная, д. 27.
- г. Красноярск, радиорынок, проезд до остановки "Затон", суббота, воскресенье.
- г. Набережные Челны, Татарстан, Новый Город, салон-магазин "ПРИНТЕР".
- г. Нижний Новгород, ТОО "КРИОНН", Магазин "ФОТОЛЮБИТЕЛЬ", ул. Горького, 146.
- г. Оренбург, магазин "ВОЕННАЯ КНИГА", ул. Советская.
- г. Пермь, Комсомольский проспект, 86, магазин "ЮНЫЙ ТЕХНИК", тел. 64-32-23.
- г. Прокопьевск, Кемеровской область, ул. Артема, 8, магазин фирмы "АРБАТ".
- г. Рыбинск, ул. Гоголя, 1. ТТЦ "ГНОМ".
- г. С.-Петербург, радиорынок "АВТОВО", киоск № 149, ("SCORPION-256").
- г. С.-Петербург, фирма "Скорпион", тел. 524-16-53, 248-47-69.
- г. Саратов, ДКР ("Сенной"), Радиоряды, суббота, воскресенье 9-18.
- г. Таганрог, радиорынок, суббота, воскресенье, проезд троллейбусом 1,5 до остановки "Центральный Рынок".
- г. Тула, Мясовкский рынок, суббота, воскресенье, фирма "ТИМКО".
- г. Тула, 300012, Тула-12, а/я 409, фирма "ТИМКО".
- г. Хабаровск, ул. Толстого, 22. Центр культуры и досуга профсоюзов. Суббота, Воскресенье с 9 до 12 часов.
- г. Челябинск, ул. Елькина, 45, магазин напротив Публичной Библиотеки.
- г. Ярославль, магазин "РАДИО-СПОРТ-ТУРИЗМ", отдел "Электроника", Ленинградский проспект.
- г. Ярославль, Московский пр-т, д.153, ТО маг-н "ГАММА".

Вниманию дистрибуторов!

Адреса наших оптовых покупателей публикуются бесплатно.

ЛР №063856 от 30.12.94 г.
Подписано а печать 18.10.95 г. Формат 84х108 1/32. Печать офсетная.
Тираж 5000 экз. Зак. 373.

ТОО «Формак», 121019, Москва, а/я 16.

Отпечатано с готового оригинал-макета в типографии ИПО Профиздат,
109044, Москва, Крутицкий вал, 18.
Плр № 050003 от 19.10.94 г.

