```
                        C       INCLUDE IPORTS.INC
0004                    C           CIFHI EQU 4
0000                    C              PGPORT EQU 0;255;0
0005                    C              DR EQU 5; 197
0006                    C              SENSE2 EQU 6;198
0008                    C       PORT      EQU  8;+192     ;CTC output port value.
0006                    C       SNDPTO  EQU     6;+192
0003                    C       SNDPTI  EQU     3;+192
FFF0                    C       IJTABLE EQU 0FFF0H
00FF                    C       IJHIGH EQU 0FFH
00F0                    C       IJLOW EQU 0F0H
                        C       ;PORT EQU 0
                        C
                                ;
                                PUBLIC  DDBCM,SCODE,JPHL
                                PUBLIC  DIROUTP,LOADER,LOAD2
                                PUBLIC  CRLF,SNEW
                                PUBLIC  SOUT,SAUTO
                                ;
                                ;
0028                            JT      EQU     28H
                                ;
                                ;
0000  F3                        RST0:   DI
0001  AF                                XOR A
0002  21 4000                           LD HL,4000H
0005  C3 0194                           JP INITIAL
                                ;
0008  5E                        RST8:   LD E,(HL)
0009  23                                INC HL
000A  56                                LD D,(HL)
000B  23                                INC HL
000C  C9                                RET
000D  FF FF FF                          DB 0FFH,0FFH,0FFH
                                ;
0010  E3                        RST10:  EX (SP),HL
0011  F5                                PUSH AF
0012  7E                        SCRR1:  LD A,(HL)
0013  FE 40                             CP 64
                                ;       INC HL
0015  C3 06FA                           JP SCREEN
                                ;
0018  C3 3B74                   RST18:  JP OVERSP
001B                            CRLF:
001B  D7                        NEWLINE:RST SCRRST
001C  2D 0A                             DB 2DH,0AH
001E  C9                                RET
001F  00                                NOP
                                ;
0020  33                        RST20:  INC SP
0021  33                                INC SP
0022  C2 3C40                           JP NZ,SYFAIL
0025  C3 3C38                           JP SYSUCC
                                ;
0028  22 FD6F                   RST28:  LD (TMPHL),HL
002B  E1                                POP HL
```

```
002C  23                        INC HL          } Get and increment
002D  E5                        PUSH HL         } return address
002E  18 0B                     JR PAGE1
0030  E5        GETRST: RST30:  PUSH HL
0031  DD E5                    PUSH IX
0033  FD E5                    PUSH IY
0035  C3 24CC                  JP GETNEXT      - Page 0
0038  C3 FD4E             RST38: JP USRRST

003B  F5            PAGE1:  PUSH AF         }
003C  E3                    EX (SP),HL      } Save A
003D  22 FD71               LD (TMPA),HL    }
0040  E1                    POP HL          }
0041  2B                    DEC HL          } Get parameter
0042  7E                    LD A,(HL)       } byte
0043  CB 7F                 BIT 7,A         } If bit 7 set
0045  20 15                 JR NZ,CALSUB    } then jump table
0047  CB 77                 BIT 6,A         } If both bits 6 and 7
0049  CA FD54               JP Z,USERROR    } reset then jump to USERROR
004C  2A FAD2               LD HL,(PAGE)    } Save page
004F  E5                    PUSH HL         }
0050  CD 00CB               CALL SCENTR     - Page 1
0053  21 0089               LD HL,PEND      } Exit via page return routine
0056  E5                    PUSH HL         }
0057  21 2E6B               LD HL,P1JTBL    } Page 1 Jump
005A  18 0E                 JR CALSB2       } table
005C  CB 77         CALSUB: BIT 6,A         } If bit 6 reset then
005E  28 07                 JR Z,CALSB1     } skip reset
0060  21 0028               LD HL,28H       } Stack re-entry
0063  E5                    PUSH HL         } address.
0064  18 01                 JR CALSB1
0066  FF            NMI:   RST 38H
0067  21 00FC      CALSB1: LD HL,CALLTABLE - Jump table address
006A  E6 3F          CALSB2: AND 3FH         } Find and stack
006C  CD 0689                CALL FINDJP     } jump address
006F  E5                     PUSH HL
0070  2A FD71                LD HL,(TMPA)    } restore A and
0073  E5                     PUSH HL         } HL.
0074  F1                     POP AF          }
0075  2A FD6F                LD HL,(TMPHL)   }
0078  C9                     RET             - Jump.

0079  E5            KBD::  PUSH HL
007A  2A FAD2                LD HL,(PAGE)
007D  7D                     LD A,L
007E  E6 8F                  AND 8FH
0080  32 FAD2                LD (PAGE),A
0083  D3 00                  OUT (PGPORT),A
0085  E3                     EX (SP),HL
0086  CD 3618                CALL KBDP0
                      ;
                      ;
                      ;
0089  E3              PEND:EX (SP),HL
008A  F5                     PUSH AF
008B  7D                     LD A,L
008C  E6 70                  AND 70H
```

```
008E  67          LD H,A
008F  3A FAD2     LD A,(PAGE)
0092  E6 8F       AND 8FH
0094  B4          OR H
0095  32 FAD2     LD (PAGE),A
0098  D3 00       OUT (PGPORT),A
009A  F1          POP AF
009B  E1          POP HL
009C  C9          RET
009D  05          DB 5
009E  F7          SROM::  RST GETRST
009F  17                  RLA
00A0  17                  RLA
00A1  17                  RLA
00A2  17                  RLA
00A3  E6 70               AND 70H
00A5  32 FAD2             LD (PAGE),A
00A8  D3 00               OUT (PGPORT),A
00AA  C3 200C             JP 200CH
                  ;
                  ;
                  ;*******************************
00AD  3E BF       SHIF::  LD A,191
00AF  32 FD7E             LD (LASTDR),A
00B2  D3 05               OUT (DR),A ;Z IMPLIES )SHIFT
00B4  DB 05               IN A,(DR)
00B6  CB 77               BIT 6,A
00B8  C8                  RET Z
00B9  CB 47               BIT 0,A
00BB  C9                  RET
                  ;
                  ;
                  ;
00BC  E5          GETSTR::PUSH HL
00BD  2A FAD2             LD HL,(PAGE)
00C0  E3                  EX (SP),HL
00C1  E5                  PUSH HL
00C2  21 0089             LD HL,PEND
00C5  E3                  EX (SP),HL
00C6  E5                  PUSH HL
00C7  21 3A6F             LD HL,BSLOAD
                  ;********************* ADDRESS IN PAGE 1 OF GETSTR
00CA  E3                  EX (SP),HL
00CB  F5          SCENTR::        PUSH AF
00CC  3A FAD2             LD A,(PAGE)
00CF  E6 8F               AND 8FH
00D1  CB E7               SET 4,A
00D3  32 FAD2     SCENT0: LD (PAGE),A
00D6  D3 00               OUT (PGPORT),A
00D8  F1                  POP AF
00D9  C9                  RET
                  ;
                  ;Assembly line is set up with
                  ;length(2),linenumber(2),CODE,ditance
                  ;to labels(2),(LASTPG)
```

Restore ROM page saved on stack

- Display tracker in A

Select page 1

```
                        ;
                        ;
00DA  EB                SCODE:  EX DE,HL
00DB  2B                        DEC HL
00DC  2B                        DEC HL
00DD  2B                        DEC HL  ; HL-5
00DE  2B                        DEC HL
00DF  2B                        DEC HL
00E0  CD 1BE3                   CALL CODETEST
00E3  E5                        PUSH HL         ;New linepnt
00E4  21 FD81                   LD HL,BREAK
00E7  CB D6                     SET 2,(HL)
00E9  E5                        PUSH HL
00EA  2A FDC7                   LD HL,(BASEM)
00ED  CD 07B8                   CALL JPHL
00F0  E1                REDE:   POP HL
00F1  CB 96                     RES 2,(HL)
00F3  D1                        POP DE
                        ;
00F4  F5                SWITCHO:        PUSH AF
00F5  3A FAD2                   LD A,(PAGE)
00F8  E6 8F                     AND 8FH
00FA  18 D7                     JR SCENTO
                        ;
00FC  3D84              CALLTABLE:DW EVALAB
00FE  128F                      DW ACCSTACK
0100  12D1                      DW RETACC
0102  2391                      DW SQUA1
0104  129E                      DW OP1STACK
0106  12B6                      DW STACKACC
0108  1183                      DW EXCHANGE
010A  23A5                      DW LOG2X
010C  100B                      DW MULPLY
010E  23F2                      DW EXP2X
0110  109E                      DW DIVIDE
0112  0E86                      DW ADD
0114  112B                      DW FRAC
0116  23BC                      DW LOGNX
0118  12B9                      DW MOVDEOP1
011A  2000                      DW SET1ACC
011C  0E83                      DW SUBTRACT
011E  13C1                      DW POLY2
0120  13F3                      DW ALOP1
0122  0FFA                      DW REMAINDER
0124  120B                      DW PIOP1
0126  115D                      DW COMPAO
0128  1298                      DW STACKOP1
012A  13B2                      DW SINE
012C  137D                      DW COSINE
012E  100B                      DW SQUARE
0130  13D2                      DW POLYNOM
0132  12BF                      DW SAVEACC
0134  12A7                      DW MOVEDE
0136  11CB                      DW LARGEN
0138  11C5                      DW LARGE1
013A  1274                      DW DESTACK
```

} Page 4 Rom

```
013C  1317                    DW STAC
013E  14B9                    DW SYNPROC
0140  0FEA                    DW COMPACC
0142  0E3D                    DW DIV10
0144  2316                    DW REPDEC
0146  2FE9                    DW GETSPACE
0148  2FF3                    DW GETCHAR
014A  2FCD                    DW PUTSTACK
014C  2F90                    DW NEWSCREEN
014E  1429                    DW EDITOR
0150  1448                    DW EDEND
0152  297E                    DW MOREHEX
0154  0CAB                    DW PRINTX
0156  1A19                    DW FINDNSPC
0158  0679                    DW BSTPHL
015A  0671                    DW PGCHK
015C  06AC                    DW LDBSBT
015E  0647                    DW PGSLC1
0160  2350                    DW GETDE1
0162  1BDD                    DW MOVHL
0164  1A0C                    DW TEST1
0166  19E5                    DW GETLNE
0168  3C45                    DW AE
016A  3B96                    DW DIGIT
016C  2CFE                    DW ENDLIN
016E  3FE9                    DW FIND1$
0170  2A7A                    DW TINSERT
0172  3E7E                    DW EVALSE
0174  2A8D                    DW TEST2B
0176  3B87                    DW ALPHA
0178  3C91                    DW SE
017A  161F                    DW SINCOS
017C  C9              COPYBLK:        DB 0C9H
017D  C9                      DB 0C9H ;USER (last two bytes)
017E  00 00 00                DB 0,0,0  ;SETCALL
0181  01                      DB 1    ;IOPL
0182  FF                      DB 0FFH  ;Realby
0183  80                      DB 080H  ;alpha lock for KEY
0184  FA7A                    DW STKTOP  ;TOP OF CALC AREA
0186  FB4B                    DW START        ;SYSTOP
0188  FD48                    DW STACK  ;STACK BASE
018A  C9 00 00                DB 0C9H,0,0     ;USERINT
018D  C9 00 00                DB 0C9H,0,0     ;NODE
0190  C9 00 00                DB 0C9H,0,0 ;FEXPAND
0193  C9                      DB 0C9H ;USER NODDY
0194                   COPYBOT:
                       ;
                       ;
                       ;
0194                   INITIAL:
0194  32 FAD2                 LD (PAGE),A
0197  D3 00                   OUT (PGPORT),A
0199  08                      EX AF,AF
019A  06 08                   LD B,8
019C  70               FILL:  LD (HL),B
019D  23                      INC HL
```

RAM Sizing routine

```
019E  10 FC                    DJNZ FILL
01A0  01 0801                  LD BC,0801H
01A3  2B               TEST:   DEC HL  ;Test how many pages present.
01A4  7E                       LD A,(HL)
01A5  A9                       XOR C
01A6  20 09                    JR NZ,ZEROZ
01A8  0C                       INC C
01A9  10 F8                    DJNZ TEST
01AB  08               OK:     EX AF,AF
01AC  3C                       INC A
01AD  FE 10                    CP 10H
01AF  20 E3                    JR NZ,INITIAL
01B1  21 C000          ZEROZ:  LD HL,0C000H  ;Fill page 16 with zero.
01B4  11 C001                  LD DE,0C001H
01B7  01 3FFF                  LD BC,3FFFH
01BA  AF                       XOR A
01BB  77                       LD (HL),A
01BC  ED B0                    LDIR
01BE  08                       EX AF, AF
01BF  32 FA7A                  LD (LSTPG),A
01C2  DB 06            SENS:   IN A,(SENSE2)
01C4  E6 0C                    AND 0CH
01C6  1F                       RRA
01C7  1F                       RRA
01C8  32 FB45                  LD (CTYSLT),A
01CB  01 001B                  LD BC,COPYBOT-COPYBLK
01CE  21 017C                  LD HL,COPYBLK
01D1  11 FA8A                  LD DE,USER+1
01D4  ED B0                    LDIR
01D6  21 090A                  LD HL,CDATA
01D9  0E 1A                    LD C,26         ;26 bytes to move, B = 0
01DB  11 FD4B                  LD DE,RICHJL
01DE  ED B0                    LDIR
01E0  31 FD48                  LD SP,STACK
                       ;
01E3  3E 10                    LD A,10H
01E5  F5                       PUSH AF
01E6  F1               RMLP:   POP AF
01E7  C6 10                    ADD A,10H
01E9  FE 80                    CP 80H
01EB  28 18                    JR Z,BASIC5
01ED  32 FAD2                  LD (PAGE),A
01F0  D3 00                    OUT (PGPORT),A
01F2  F5                       PUSH AF
01F3  06 08                    LD B,8
01F5  21 1FFF                  LD HL,1FFFH
01F8  23               TSTLP:  INC HL
01F9  7E                       LD A,(HL)
01FA  B8                       CP B
01FB  20 E9                    JR NZ,RMLP
01FD  10 F9                    DJNZ TSTLP
01FF  CD 2010          RMLP2:  CALL RMNTRY
0202  18 E2                    JR RMLP
                       ;
2010                   RMNTRY  EQU     2010H
                       ;
```

Read Keyboard Configuration Switches

Initialize System Variables

Test for mounted ROMs

```
                        PUBLIC  SWITCHO
                        PUBLIC  INSERT
                        PUBLIC  DELETE
                        PUBLIC  NOGOOD
                        PUBLIC  BASENT
                        PUBLIC  BASIC
                        PUBLIC  STRHL
                        PUBLIC  STROUT
                        PUBLIC  LDBSBT

0008                    DEHL    EQU     8
0010                    SCRRST  EQU     10H
0030                    GETRST  EQU     30H
0080                    REM     EQU     128
00FF                    ENDMARK EQU     255
                        ;SWITCH ON INTERRUPTS
                        ;
0204    C9              RET ;SYNTAX BYTE
0205                    SNEW:
0205                    BASIC5:
0205                    BASICS::
0205    AF              XOR A
0206    32 FAD2         LD (PAGE),A
0209    D3 00           OUT (PGPORT),A
                        ;
020B    EF                      RST JT
020C    42                      DB 42H
                        ;CALL VDINIT
020D    AF                      XOR A
020E    32 FFAB                 LD (SCRN5+3),A
0211    3E 28                   LD A,40
0213    32 FFAD                 LD (SCRN5+5),A
0216    D7                      RST SCRRST      } select and clear VS5
0217    4D                      DB 4DH

0218    CD 093A         CALL KILSND             - Sound OFF
021B    01 0002         LD BC,2
021E    CD 27B7         CALL SETINIT            } sound buffers of size 2
                        ;
0221    CD 097F         NEWENT::        CALL IJINIT             ;******** CHANGED **********  - Interrupts OFF.
0224    3A FA7A                 LD A,(LSTP6)
0227    B7                      OR A
0228    21 4000                 LD HL,4000H
022B    22 FAAA                 LD (BASBOT),HL
022E    20 01                   JR NZ,MWNT1
0230    29                      ADD HL,HL
0231    22 FAA4         MWNT1:  LD (NBTOP),HL
0234    22 FAA7                 LD (BASTOP),HL
0237    22 FAAC                 LD (BASTPO),HL
023A    22 FD65                 LD (VAZERO),HL
023D    AF                      XOR A
023E    32 FAA6                 LD (NBTP6),A
0241    32 FAA9                 LD (BASTP6),A
                        ;CTC SOUND INT FLAG BASBOT ETC FOR NEW
0244                    BASIC1::        ;SET CODE POINTERS
```

} Initialize Pointers

```
02BD  C3 0274              JP NEXKEY
02C0  2A FD84        NOGOOD: LD HL,(ERRPOS)
02C3  ED 5B FA83             LD DE,(KBDBUF)
02C7  A7                     AND A
02C8  ED 52                  SBC HL,DE
02CA  4D                     LD C,L
02CB  CD 0744        EFFENT::      CALL STACKS
02CE  ED 7B FA96             LD SP,(SSTACK)
02D2  06 00                  LD B,0
02D4  D7                     RST SCRRST
02D5  60 23 E0 1E            DB 60H,23H,0E0H,30
                     ;
02D9  EF             EDENT:: RST JT
02DA  A9                     DB 0A9H
                     ;       CALL EDITOR
02DB  18 A6                  JR PROG
02DD  01 0000        ENDEDIT:      LD BC,0
02E0  ED 43 FD77             LD (AUTOINC),BC
02E4  C3 0250                JP BASIC2
02E7  EB             STRHL:  EX DE,HL
02E8  1A             STROUT: LD A,(DE)
02E9  FE FF                  CP 0FFH
02EB  C8                     RET Z
02EC  CB 7F                  BIT 7,A
02EE  13                     INC DE
02EF  CC 1B6A                CALL Z,CHOUT
02F2  28 F4                  JR Z,STROUT
02F4  E6 7F          SPACES: AND 7FH
02F6  47                     LD B,A
02F7  D7             SPC1:   RST SCRRST
02F8  81 20                  DB 81H,32
02FA  10 FB                  DJNZ SPC1
02FC  18 EA                  JR STROUT
                     ;
                     ;
02FE  ED 5B FA83     INSERT: LD DE,(KBDBUF)
0302  CD 121D                CALL INT64K   ;GET LINE NO. IN BC
0305  C5                     PUSH BC  ;STACK=LINE NO.DE=START
0306  62                     LD H,D   ;HL=DE=START
0307  6B                    LD L,E
0308  CD 1B43                CALL CALLEN ;BC=LENGTH
030B  E1                    POP HL
030C  C5                    PUSH BC
030D  E5                    PUSH HL
030E  03                    INC BC   ;STACK=LENGTH-4,LINE NO,LENGTH
030F  03                   INC BC
0310  03                    INC BC
0311  03                    INC BC
0312  C5                   INC BC   ;BC=LENGTH,HL=LINE NO,DE=START
0313  44                   PUSH BC  ;STACK= COPY,LINE NO,LENGTH
0314  4D                   LD B,H
0315  CD 06DE              LD C,L   ;BC=HL=LINE NO.,DE=START
0318  20 0B                CALL GETLN1
031A  CD 06CF              JR NZ,INS1  ;IF LINE EXISTS THEN DELETE IT
031D  C1                   CALL DELET1 ;DELETE LINE
031E  C1                   POP BC
                           POP BC
```

```
0244  21 000D          BASIC0::    LD HL,13
0247  22 FDB5                      LD (LABTABL),HL
024A  CD 00F4          BASIC4::    CALL SWITCH
024D  CD 27FB                      CALL KILLVAR
0250  CD 00F4          BASIC2::    CALL SWITCH
0253  CD 0744                      CALL STACKS
0256  CD 0996                      CALL INT125
0259  ED 7B FA96        BASIC::  LD SP,(SSTACK)
025D  AF               BASENT:  XOR A
025E  32 FAD2                   LD (PAGE),A
0261  D3 00                     OUT (PGPORT),A
0263  32 FD75                   LD (PRORPL),A
0266  D7                        RST SCRRST
0267  A3 1B 56 FF               DB 0A3H,1BH,'V',0FFH,6FH,0A5H,'Ready',68H,30
026B  6F A5 52 65
026F  61 64 79 68
0273  1E
0274  CD 0079        NEXKEY:  CALL KBD       ;GET FIRST KEY
0277  28 FB                   JR Z,NEXKEY
0279  FE 03                   CP 3          ;Break Key
027B  28 F7                   JR Z,NEXKEY
027D  D7                      RST SCRRST
027E  6F 40                   DB 6FH,40H
0280  CD 1424                 CALL EDIT1
0283  EF               PROG:    RST JT
0284  AA                     DB 0AAH
                       ;       CALL EDEND
0285  28 56                   JR Z,ENDEDIT
0287  2A FA83                 LD HL,(KBDBUF)
028A  E5                      PUSH HL
028B  CD 398C                 CALL SYN
028E  D7                      RST SCRRST
028F  61 23 20 12             DB 61H,23H,20H,18
0293  E1                      POP HL
0294  E5                      PUSH HL
0295  CD 24F3                 CALL OL1  ;PRINT OUT BASIC LINE
0298  D1                      POP DE
0299  2A FD82                 LD HL,(COMMAND)
                       ;IF COMMAND IS NOT DIRECT, THEN LINE WILL BE INSERTED
029C  CD 2CEC                 CALL RUN8
                       ;CLEAR EDIT SCREEN IS CONDITIONAL HERE IF REQUIRED
029F  ED 4B FD77       AUTOLIN:    LD BC,(AUTOINC)  ;BC = INC
02A3  78                       LD A,B
02A4  B1                       OR C
02A5  28 A9                    JR Z,BASIC2  ;RETURN IF AUTO OFF
02A7  D7                       RST SCRRST
02A8  6F 68 1E                 DB 6FH,68H,30
02AB  2A FD79                  LD HL,(AUTOST)  ;HL=LAST LINE
02AE  09                       ADD HL,BC
02AF  44                       LD B,H
02B0  4D                       LD C,L
02B1  22 FD79                  LD (AUTOST),HL
02B4  CD 0DD0                  CALL BCDEC
02B7  CD 02E8                  CALL STROUT
02BA  D7                       RST SCRRST
02BB  81 20                    DB 81H,32
```

Page 1
- Reset all variable pointers
- Page 1
- Reset Stacks
- Interrupts every 1/125th second
- Reset stack Pointer

ROM/RAM page 0
- output to screen

<ESC>V <+FF>
VS0 + CLS
Ready
VS0 + CLS
Cursor On

; Clear VS0, select VS1

Get an input line

- Jump if empty line
- HL to start of line
- Check Syntax and convert to tokens

VS1, <+01>, <+00>, <+1>

```
031F  C1                        POP BC
0320  18 DC                     JR INSERT
0322  C1              INS1:     POP BC  ;HL=STARTOF LINE
0323  C5                        PUSH BC
0324  D5                        PUSH DE
0325  CD 0351                   CALL MAKESP ;LINE DOESN'T EXIST.
                      ;RETURNS WITH HL SPACES,HL POINTS AT FIRST
0328  D1                        POP DE
0329  C1                        POP BC
032A  71                        LD (HL),C
032B  23                        INC HL
032C  70                        LD (HL),B
032D  23                        INC HL
032E  C1                        POP BC
032F  71                        LD (HL),C
0330  23                        INC HL
0331  70                        LD (HL),B
0332  23                        INC HL
0333  EB                        EX DE,HL
0334  C1                        POP BC
0335  ED B0                     LDIR
0337  E1                        POP HL
0338  CD 27FB                   CALL KILLVAR
033B  C9                 RET
                      ;
                      ;
033C  E5              FREESP::        PUSH HL
033D  D5                        PUSH DE
033E  CD 0344                   CALL TESTSP
0341  D1                        POP DE
0342  E1                        POP HL
0343  C9                        RET
                      ;
0344  EF              TESTSP::        RST JT
0345  AE                        DB 0AEH
                      ;         CALL BSTPHL ;HL=BASTPN
0346  CF                        RST DEHL  ;DE=(BASTPN)
0347  CD 0697         TSTP1:    CALL RMTPHL
034A  A7                        AND A
034B  ED 52                     SBC HL,DE  ;HL=FREE SPACE
034D  A7                        AND A
034E  ED 42                     SBC HL,BC
0350  C9                        RET
                      ;
                      ;MAKESP:        MAKES AVAILABLE A BLOCK OF FREE SPACE
                      ;IN PAGE A. IF NO SPACE IS AVAILABLE COPYDOWN IS CALLED
                      ;TO MAKE SPACE IN NEXT PAGE
                      ;HL POINTS AT POSITION WHERE SPACE IS REQUIRED
                      ;BC=AMMOUNT OF SPACE
                      ;SAVES HL
                      ;
0351  AF              MAKESP::        XOR A
0352  3A FAD1                   LD A,(BASLNP)
0355  32 FAD3                   LD (CRNTP6),A
0358  D5              MAKSP1:   PUSH DE
0359  F5                        PUSH AF
```

```
035A  E5                    PUSH HL
035B  C5                    PUSH BC
035C  3A FAA6               LD A,(NBTPG)
035F  3C                    INC A
0360  EF                    RST JT
0361  AF                    DB 0AFH
                      ;     CALL PGCHK
0362  38 14                JR C,MKSP1
0364  EF                    RST JT
0365  B0                    DB 0B0H
                      ;     CALL LDBSBT
0366  09                    ADD HL,BC
0367  EB                    EX DE,HL
0368  CD 06E7               CALL DETPN
036B  ED 53 FAA4            LD (NBTOP),DE
036F  3A FAD3               LD A,(CRNTPG)
0372  32 FAA6               LD (NBTPG),A ;SET (NBTOP)
0375  C3 0445               JP MKSP5
0378  CD 0344         MKSP1: CALL TESTSP
037B  D2 040E               JP NC,MKSP3
037E  3A FA7A               LD A,(LSTPG)
0381  EF                    RST JT
0382  AF                    DB 0AFH
                      ;CALL PGCHK
0383  20 02                JR NZ,MKSP2
0385  EF                    RST 28H   ;OUT OF SPACE IF LAST PAGE
0386  13                    DB 13H
0387  3A FAD1         MKSP2: LD A,(BASLNP)
038A  EF                    RST JT
038B  AF                    DB 0AFH
                      ;     CALL PGCHK
038C  C2 03FF               JP NZ,MKSP2E
038F  C1                    POP BC
0390  E1                    POP HL
0391  E5                    PUSH HL
0392  C5                    PUSH BC
0393  E5                    PUSH HL
0394  D5                    PUSH DE
0395  09                    ADD HL,BC
0396  EB                    EX DE,HL
0397  CD 0697               CALL RMTPHL
039A  EB                    EX DE,HL
039B  A7                    AND A
039C  ED 52                 SBC HL,DE  ;HL=START ADD+LEN -RAMTOP
039E  D1                    POP DE
039F  E1                    POP HL
03A0  38 5D                 JR C,MKSP2E
03A2  E5                    PUSH HL
03A3  A7                    AND A
03A4  EB                    EX DE,HL
03A5  ED 52                 SBC HL,DE  ;HL=(BASTPN)-START ADD
03A7  E5                    PUSH HL
03A8  C1                    POP BC   ;BC = BLOCK SIZE
03A9  D1                    POP DE   ;DE =START ADD
03AA  F5                    PUSH AF
03AB  CD 06F3               CALL INCPG
```

```
03AE  F1                    POP AF
03AF  28 0A                 JR Z,MKSPB
03B1  EF                    RST JT
03B2  B0                    DB 0B0H
                      ;     CALL LDBSBT ;HL=(BASBOT) OF NEXT PG.
03B3  A7                    AND A
03B4  D5                    PUSH DE
03B5  CD 0358               CALL MAKSP1 ;TRANSFER UPPER LINES
03B8  E1                    POP HL
03B9  18 17                 JR MKSPC
03BB  EB              MKSPB: EX DE,HL
03BC  3A FAA6               LD A,(NBTPG)
03BF  3C                    INC A
03C0  EF                    RST JT
03C1  AF                    DB 0AFH
                      ;     CALL PGCHK
03C2  20 0E                 JR NZ,MKSPC
03C4  32 FAA6               LD (NBTPG),A
03C7  E5                    PUSH HL
03C8  EF                    RST JT
03C9  B0                    DB 0B0H
                      ;     CALL LDBSBT
03CA  22 FAA4               LD (NBTOP),HL ;ADJUST (NBTOP) IF ON PG.
03CD  EB                    EX DE,HL
03CE  CD 06E7               CALL DETPN
03D1  E1                    POP HL
03D2  3A FAA9         MKSPC: LD A,(BASTPG)
03D5  3C                    INC A
03D6  EF                    RST JT
03D7  AF                    DB 0AFH
                      ;     CALL PGCHK
03D8  20 14                 JR NZ,MKSPD
03DA  ED 5B FAA7            LD DE,(BASTOP)
03DE  A7                    AND A
03DF  ED 52                 SBC HL,DE ;HL=START ADD -(BASTOP)
03E1  20 0B                 JR NZ,MKSPD
03E3  3A FAD3               LD A,(CRNTPG)
03E6  32 FAA9               LD (BASTPG),A
03E9  EF                    RST JT
03EA  B0                    DB 0B0H
                      ;     CALL LDBSBT
03EB  22 FAA7               LD (BASTOP),HL
03EE  C1              MKSPD: POP BC
03EF  E1                    POP HL
03F0  F1                    POP AF
03F1  D1                    POP DE
03F2  F5                    PUSH AF
03F3  EF                    RST JT
03F4  B0                    DB 0B0H
                      ;     CALL LDBSBT
03F5  F1                    POP AF
03F6  3A FAD3               LD A,(CRNTPG)
03F9  32 FAD1               LD (BASLNP),A
03FC  C3 0358               JP MAKSP1
                      ;
03FF  CD 047B         MKSP2E: CALL MVADR  ;DE=(BASTPN) BC = SPACE REQUIRED
```

```
                                ;DE= START ADDRESS OF BLOCK TO BE MOVED
                                ;BC=NUMBER OF BYTES IN BLOCK
0402  CD 06F3                  CALL INCPG
0405  EF                       RST JT
0406  B0                       DB 0B0H
                        ;       CALL LDBSBT ;HL=START OF NEXT PAGE
0407  A7                       AND A
0408  CD 0358                  CALL MAKSP1
040B  CD 06ED                  CALL DECPG
040E  C1                MKSP3: POP BC
040F  E1                       POP HL
0410  E5                       PUSH HL ;POSITION WHERE SPACE REQUIRED
0411  C5                       PUSH BC ;AMMOUNT OF SPACE
0412  09                       ADD HL,BC
0413  44                       LD B,H
0414  4D                       LD C,L ;BC =NEW POSITION OF LINE AT HL
0415  EF                       RST JT
0416  AE                       DB 0AEH
                        ;       CALL BSTPHL ;HL=BASTPN
0417  CF                       RST DEHL ;DE=(BASTPN)
0418  F1                       POP AF
0419  E1                       POP HL
041A  E5                       PUSH HL
041B  F5                       PUSH AF
041C  CD 1C73                  CALL IMOVE
041F  CD 06E7                  CALL DETPN
0422  CD 066E                  CALL PGCHK1
0425  20 04                    JR NZ,MKSP4
0427  ED 53 FAA4               LD (NBTOP),DE ;NEW (NBTOP) IF ON CURRENT PAGE
042B  3A FAA9           MKSP4: LD A,(BASTPG)
042E  EF                       RST JT
042F  AF                       DB 0AFH
                        ;       CALL PGCHK
0430  20 13                    JR NZ,MKSP5 ;JMP IF (BASTOP) NOT ON PAGE
0432  2A FAA7                  LD HL,(BASTOP)
0435  C1                       POP BC
0436  D1                       POP DE
0437  D5                       PUSH DE
0438  C5                       PUSH BC
0439  A7                       AND A
043A  ED 52                    SBC HL,DE
043C  38 07                    JR C,MKSP5
043E  2A FAA7                  LD HL,(BASTOP)
0441  09                       ADD HL,BC  ;BC = AMMOUNT OF SPACE
0442  22 FAA7                  LD (BASTOP),HL
0445  C1                MKSP5: POP BC
0446  E1                       POP HL
0447  F1                       POP AF
0448  D1                       POP DE
0449  C8                       RET Z                  ;RET IF NO TRANSFER
044A  D5                       PUSH DE
044B  E5                       PUSH HL
044C  C5                       PUSH BC
044D  AF                       XOR A
044E  CD 05DE                  CALL TRNSF1
0451  CD 11EC                  CALL COMPBC
```

```
0454  CD 05CD          CALL DDBCM1 ;(BASTPN-1 +BC
0457  3A FAD4          LD A,(PGN1)
045A  47               LD B,A
045B  3A FAA9          LD A,(BASTPG)
045E  B8               CP B
045F  C1               POP BC
0460  E1               POP HL
0461  D1               POP DE
0462  C0               RET NZ          ;RET IF (BASTOP) NOT ON SOURCE PAGE
0463  E5               PUSH HL
0464  2A FAA7          LD HL,(BASTOP)
0467  A7               AND A
0468  ED 52            SBC HL,DE  ;DE=START ADDRESS
046A  EB               EX DE,HL
046B  E1               POP HL
046C  C8               RET Z
046D  D8               RET C                   ;RET IF (BASTOP) LE START ADDRESS
046E  E5               PUSH HL
046F  3A FAD3          LD A,(CRNTPG)
0472  32 FAA9          LD (BASTPG),A ;(BASTOP) NOW ON DESTINATION PAGE
0475  19               ADD HL,DE ;HL=NEW BASTOP
0476  22 FAA7          LD (BASTOP),HL
0479  E1               POP HL
047A  C9               RET
047B  CD 0697    MVADR:  CALL RMTPHL
047E  A7                 AND A
047F  ED 42              SBC HL,BC
0481  44                 LD B,H
0482  4D                 LD C,L
0483  CD 048F             CALL LNSRCH
0486  EB                 EX DE,HL
0487  D5                 PUSH DE
0488  A7                 AND A
0489  ED 52              SBC HL,DE
048B  44                 LD B,H
048C  4D                 LD C,L
048D  D1                 POP DE
048E  C9                 RET
048F  D5         LNSRCH: PUSH DE
0490  CD 0644             CALL PGSLC2 ;SELECT CURRENT PAGE
0493  EF                  RST JT
0494  B0                 DB 0B0H
                   ;       CALL LDBSBT ;HL=(BASBOT)
0495  E5           NXTLN:  PUSH HL
0496  C5                   PUSH BC
0497  01 FFFF              LD BC,0FFFFH ;DUMMY
049A  CD 0609              CALL CMPLN
049D  C1                   POP BC
049E  A7                   AND A
049F  E5                   PUSH HL
04A0  ED 42                SBC HL,BC
04A2  E1                   POP HL
04A3  28 08                JR Z,SRCHY
04A5  30 03                JR NC,SRCHX
04A7  F1                   POP AF ;OLD HL FROM STACK
04A8  18 EB                JR NXTLN
```

Find next line

Compare BC with position of start of line

Jump if exact match or too low

repeat

HL = 1 + Position of end of last line below BC

```
04AA  E1              SRCHX:  POP HL      } Go back to last
04AB  D1                      POP DE      } line
04AC  C9                      RET
04AD  F1              SRCHY:  POP AF      } Carriage line OK
04AE  D1                      POP DE      }
04AF  C9                      RET
                      ;
04B0  3A FAD1         COPYDOWN::    LD A,(BASLNP)
04B3  32 FAD3                 LD (CRNTPG),A
04B6  E5              CPYNXT: PUSH HL   - Start of line ②
04B7  C5                      PUSH BC   - Length ②
04B8  E5                      PUSH HL   - Start ③
04B9  09                      ADD HL,BC - End
04BA  C1                      POP BC  ;BC = START ADDRESS OF DELETE LINE ②
04BB  E5                      PUSH HL   - End ①
04BC  EF                      RST JT
04BD  AE                      DB 0AEH   } (HL) = Top of current BASIC page
                      ;       CALL BSTPHL }
04BE  CF                      RST DEHL ;DE=(BASTPN) - Copy to DE
04BF  E1                      POP HL  - End of line ①
04C0  CD 0644                 CALL PGSLC2 - Select appropriate RAM page and set PGTOP
04C3  CD 1C73                 CALL IMOVE  - Move code downwards
04C6  CD 06E7                 CALL DETPN  - Set top of page to new position
04C9  C1                      POP BC  - Length ①
04CA  E1                      POP HL  - Start ②
04CB  D5                      PUSH DE - New top of BASIC page
04CC  3A FAA9                 LD A,(BASTPG) - A: Last basic page
04CF  EF                      RST JT
04D0  AF                      DB 0AFH
                      ;       CALL PGCHK  - Test for last basic page
04D1  20 50                   JR NZ,CPYDN1  - Jump if not
04D3  ED 5B FAA7              LD DE,(BASTOP)  DE = Top of basic
04D7  A7                      AND A
04D8  ED 52                   SBC HL,DE       }
04DA  30 47                   JR NC,CPYDN1 ;JMP UNLESS (BASTOP) LT START OF LINE
04DC  EB                      EX DE,HL        } HL = (BASTOP) - DE
04DD  A7                      AND A           } DE = Start of line
04DE  ED 42                   SBC HL,BC       }
04E0  AF                      XOR A           }
04E1  EF                      RST JT          } Jump if page ≠ 0.
04E2  AF                      DB 0AFH         }
                      ;       CALL PGCHK      }
04E3  28 3B                   JR Z,CPYDNB ;PG 0?
04E5  EB                      EX DE,HL    DE = new top of page
04E6  EF                      RST JT          } HL = Bottom of current page
04E7  B0                      DB 0B0H         }
                      ;       CALL LDBSBT     }
04E8  A7                      AND A
04E9  ED 52                   SBC HL,DE       } HL = New top of page
04EB  EB                      EX DE,HL        }
04EC  20 32                   JR NZ,CPYDNB ;NEW (BASTOP)=(BASBOT)? - Jump if different
04EE  CD 066E                 CALL PGCHK1 - Compare (NBTOPG) and (CRNTPG)
04F1  20 0A                   JR NZ,CPYDNA - Jump if different
04F3  2A FAA7                 LD HL,(BASTOP)  } Size of MOVE
04F6  ED 5B FAA4              LD DE,(NBTOP)   } region
04FA  A7                      AND A           }
```

```
04FB  ED 52              SBC HL,DE
04FD  F5           CPYDNA: PUSH AF
04FE  3A FAA9            LD A,(BASTPG)
0501  3D                  DEC A
0502  32 FAA9            LD (BASTPG),A
0505  EF                  RST JT
0506  AE                  DB 0AEH
                    ;     CALL BSTPHL
0507  2B                  DEC HL
0508  2B                  DEC HL
0509  CF                  RST DEHL
050A  ED 53 FAA7          LD (BASTOP),DE
050E  F1                  POP AF
050F  20 12               JR NZ,CPYDN1
0511  3A FAA9            LD A,(BASTPG)
0514  32 FAA6            LD (NBTPG),A
0517  2A FAA7            LD HL,(BASTOP)
051A  22 FAA4            LD (NBTOP),HL
051D  D1                  POP DE
051E  18 22               JR CPYDNZ
0520  22 FAA7      CPYDNB: LD (BASTOP),HL ;BASTOP=BASTOP-LINE LENGTH
0523  CD 066E      CPYDN1: CALL PGCHK1  ;CHECK IF NBTOP ON PAGE
0526  D1                  POP DE
0527  20 1C               JR NZ,CPYDN2
0529  EF                  RST JT
052A  B0                  DB 0B0H
                    ;     CALL LDBSBT
052B  A7                  AND A
052C  ED 52               SBC HL,DE
052E  20 0E               JR NZ,CPYDNC
0530  AF                  XOR A
0531  EF                  RST JT
0532  AF                  DB 0AFH
                    ;     CALL PGCHK
0533  28 09               JR Z,CPYDNC
0535  CD 06ED             CALL DECPG
0538  32 FAA6            LD (NBTPG),A
053B  EF                  RST JT
053C  AE                  DB 0AEH
                    ;     CALL BSTPHL
053D  CF                  RST DEHL
                    ;
053E  ED 53 FAA4   CPYDNC: LD (NBTOP),DE ;(NBTOP)=(BASTPN)
0542  C3 05C6      CPYDNZ: JP CPYDNX
0545  D5           CPYDN2: PUSH DE
0546  CD 0697              CALL RMTPHL
0549  A7                  AND A
054A  ED 52               SBC HL,DE
054C  EB                  EX DE,HL
054D  CD 06F3             CALL INCPG
0550  EF                  RST JT
0551  B0                  DB 0B0H
                    ;     CALL LDBSBT ;HL=(BASBOT)
0552  19                  ADD HL,DE ; HL=MAX TOP OFBLOCK TO BE MOVED
0553  E5                  PUSH HL
0554  EF                  RST JT
```

DE = top of current page

```
0555  AE                    DB 0AEH
                      ;     CALL BSTPHL
0556  CF                    RST DEHL
0557  E1                    POP HL
0558  E5                    PUSH HL
0559  A7                    AND A
055A  ED 52                 SBC HL,DE
055C  E1                    POP HL
055D  38 01                 JR C,CPYDN3
055F  EB                    EX DE,HL
```

HL = Minimum of top of Page or top of block to move

```
0560  44              CPYDN3: LD B,H
0561  4D                      LD C,L
0562  CD 048F                 CALL LNSRCH
0565  EB                      EX DE,HL
0566  EF                      RST JT
0567  B0                      DB 0B0H
                      ;CALL LDBSBT
0568  E5                      PUSH HL
0569  A7                      AND A
056A  EB                      EX DE,HL
056B  ED 52                   SBC HL,DE ;HL=BLOCK SIZE
056D  28 55                   JR Z,CPYDX2 ;FIRST LINE TOO LONG
056F  44                      LD B,H
0570  4D                      LD C,L
0571  D1                      POP DE ;DE=SOURCE ADDRESS
0572  E1                      POP HL ;HL= DEST ADDRESS
0573  CD 05DE                 CALL TRNSF1
0576  CD 05CD                 CALL DDBCM1
                      ;
                      ;
0579  D5                      PUSH DE
057A  3A FAA9                 LD A,(BASTPG)
057D  EF                      RST JT
057E  AF                      DB 0AFH
                      ;       CALL PGCHK
057F  20 1F                   JR NZ,CPYDN4
0581  EF                      RST JT
0582  B0                      DB 0B0H
                      ;       CALL LDBSBT
0583  ED 5B FAA7              LD DE,(BASTOP)
0587  EB                      EX DE,HL
0588  A7                      AND A
0589  ED 52                   SBC HL,DE
058B  C5                      PUSH BC
058C  D1                      POP DE
058D  EB                      EX DE,HL
058E  ED 52                   SBC HL,DE
0590  EB                      EX DE,HL
0591  38 0D                   JR C,CPYDN4
0593  D1                      POP DE
0594  D5                      PUSH DE
0595  19                      ADD HL,DE
0596  22 FAA7                 LD (BASTOP),HL
0599  3A FAD3                 LD A,(CRNTPG)
059C  3D                      DEC A
059D  32 FAA9                 LD (BASTPG),A
```

Transfer to BC

Find end of line in block

Amount to be actually moved

Jump if none

Copy block

Update top of page

```
05A0  CD 066E          CPYDN4: CALL PGCHK1
05A3  20 19                    JR NZ,CPYDN5
05A5  EF                       RST JT
05A6  B0                       DB 0B0H
                       ;       CALL LDBSBT
05A7  09                       ADD HL,BC
05A8  ED 5B FAA4               LD DE,(NBTOP)
05AC  A7                       AND A
05AD  ED 52                    SBC HL,DE
05AF  38 0D                    JR C,CPYDN5
05B1  E1                       POP HL
05B2  09                       ADD HL,BC ;HL=(BASTPN)+BLOCK SIZE
05B3  22 FAA4                  LD (NBTOP),HL
05B6  CD 06ED                  CALL DECPG
05B9  32 FAA6                  LD (NBTPG),A
05BC  18 08                    JR CPYDNX
                       ;
05BE  EF               CPYDN5: RST JT
05BF  B0                       DB 0B0H
                       ;       CALL LDBSBT ;HL=(BASBOT)
05C0  D1                       POP DE
05C1  C3 04B6                  JP CPYNXT
05C4  E1               CPYDX2: POP HL
05C5  E1                       POP HL
05C6  3A FAD1          CPYDNX: LD A,(BASLNP)
05C9  CD 064A                  CALL PGSLCT
05CC  C9                       RET
                       ;
05CD  EF               DDBCM1: RST JT
05CE  AE                       DB 0AEH
                       ;       CALL BSTPHL
05CF  2B                       DEC HL
05D0  2B                       DEC HL
05D1  E5               DDBCM:  PUSH HL
05D2  CF                       RST DEHL
05D3  E1                       POP HL
05D4  D5                       PUSH DE
05D5  EB                       EX DE,HL
05D6  09                       ADD HL,BC
05D7  EB                       EX DE,HL
05D8  73                       LD (HL),E
05D9  23                       INC HL
05DA  72                       LD (HL),D
05DB  23                       INC HL
05DC  D1                       POP DE
05DD  C9                       RET
                       ;
                       ;
05DE  3A FAD3          TRNSF1: LD A,(CRNTPG)
05E1  F5                       PUSH AF
05E2  3D                       DEC A
05E3  08                       EX AF,AF'  ;A'=(CRNTPG)-1
05E4  F1                       POP AF   ;A=CRNTPG
05E5  20 01                    JR NZ,TRN1  ;SOURCE PG
05E7  08                       EX AF,AF'
05E8  32 FAD4          TRN1:   LD (PGN1),A ;DEST PG
```

```
05EB  08          EX AF,AF'
05EC  32 FAD5     LD (PGN2),A
                  ;
05EF  C5          TRNSFR: PUSH BC
05F0  3A FAD4     TRNLP:  LD A,(PGN1)
05F3  CD 064A             CALL PGSLCT
05F6  1A                  LD A,(DE)
05F7  F5                  PUSH AF
05F8  3A FAD5             LD A,(PGN2)
05FB  CD 064A             CALL PGSLCT
05FE  F1                  POP AF
05FF  77                  LD (HL),A
0600  23                  INC HL
0601  13                  INC DE
0602  0B                  DEC BC
0603  78                  LD A,B
0604  B1                  OR C
0605  20 E9               JR NZ,TRNLP
0607  C1                  POP BC
0608  C9                  RET
                  ;
0609  CF          CMPLN:  RST DEHL  ;DE=LENGTH,HL=START+2
060A  D5                  PUSH DE  ;STACK=LENGTH
060B  CF                  RST DEHL  ;DE=LINE NO.HL=START+4
060C  EB                  EX DE,HL  ;HL=LINE NO. DE=START+4
060D  A7                  AND A   ;CLEAR CARRY
060E  ED 42               SBC HL,BC  ;HL =LINENO,BC=REQ LINE
0610  EB                  EX DE,HL  ;DE=?,HL=START+4
0611  D1                  POP DE   ;DE=LENGTH,HL=START+4
0612  F5                  PUSH AF  ;SAVE FLAGS
0613  19                  ADD HL,DE  ;DE=LENGTH,HL=START OF NEXT+4
0614  2B                  DEC HL
0615  2B                  DEC HL
0616  2B                  DEC HL
0617  2B                  DEC HL   ;HL=STARTOF NEXT,
0618  F1                  POP AF
0619  C9                  RET
                  ;
061A  AF          FINDLIN::       XOR A
061B  32 FAD1     FNDLN1: LD (BASLNP),A ;SET BASIC LINE PAGE NO
061E  CD 0647             CALL P6SLC1 ;SELECT PAGE IN A
0621  CD 06AC             CALL LDBSBT ;HL=(BASBOT)+4000 IF LSTP6
0624  22 FACF     FNDNXT: LD (BASELIN),HL
0627  CD 06BB             CALL TPCH1
062A  28 07               JR Z,NEXTP6 ;JUMP IF TOP
062C  CD 0609             CALL CMPLN  ;BC=LINE NO . HL = START
062F  C8                  RET Z
0630  D0                  RET NC
0631  18 F1               JR FNDNXT
0633  3A FAA9     NEXTP6: LD A,(BASTP6)
0636  EF                  RST JT  ;?
0637  AF                  DB 0AFH
                  ;       CALL PGCHK  ;CHECK IF LAST PAGE
0638  28 07               JR Z,NXTP1
063A  3A FAD3             LD A,(CRNTPG)
063D  3C                  INC A  ;INC PG NO.
```

Else
(PGN1) = (CRNTPG) - 1
(PGN2) = (CRNTPG) - 1

Copy BC bytes
from (DE) on PGN1
to (HL) on PGN2

On Entry
HL = Start Addr of line
BC = Required Line number
On Exit
HL = Start of next line
BC = Required Line Number
DE = Length of Line
Flags = Comparison of reqd and actual line No.

Search whole of memory for BASIC line with line number BC. Return with address of matching or next higher line numbe in (BASELIN) and (BASLNP)
Zero flag indicates match, Relevant RAM page is selected.

```
063E  C3 061B                 JP FNDLN1
0641  AF              NXTP1:  XOR A
0642  3C                      INC A
0643  C9                      RET
0644  3A FAD3         P6SLC2::       LD A,(CRNTP6)
0647  32 FAD3         P6SLC1::       LD (CRNTP6),A
064A  E5              P6SLCT::       PUSH HL
064B  6F                      LD L,A
064C  3A FAD2                 LD A,(PAGE)
064F  E6 F0                   AND 0F0H
0651  85                      ADD A,L
0652  32 FAD2                 LD (PAGE),A
0655  D3 00                   OUT (P6PORT),A
0657  3A FAA9                 LD A,(BASTP6)
065A  EF                      RST JT  ;?
065B  AF                      DB 0AFH
                        ;     CALL P6CHK
065C  20 05                   JR NZ,NBSTP
065E  2A FAA7                 LD HL,(BASTOP)
0661  18 06                   JR NBSTP1
0663  EF              NBSTP:  RST JT
0664  AE                      DB 0AEH  ;?
                        ;     CALL BSTPHL
0665  D5                      PUSH DE
0666  CF                      RST DEHL
0667  EB                      EX DE,HL
0668  D1                      POP DE
0669  22 FAD6         NBSTP1: LD (P6TOP),HL ;TOP=(BASTP0+2N)
066C  E1                      POP HL
066D  C9                      RET
066E  3A FAA6         P6CHK1::       LD A,(NBTP6)
0671  C5              P6CHK:: PUSH BC
0672  47                      LD B,A
0673  3A FAD3                 LD A,(CRNTP6)
0676  B8                      CP B
0677  C1                      POP BC
0678  C9                      RET
0679  D5              BSTPHL::       PUSH DE
067A  F5                      PUSH AF
067B  3A FAD3                 LD A,(CRNTP6)
067E  87                      ADD A,A
067F  5F                      LD E,A
0680  16 00                   LD D,0
0682  21 FAAC                 LD HL,BASTP0
0685  19                      ADD HL,DE
0686  F1                      POP AF
0687  D1                      POP DE
0688  C9                      RET
0689  E6 7F         FINDJP::       AND 7FH ;ASSUME POINTER IN A
068B  C5              FJP1:   PUSH BC
068C  4F                      LD C,A    ;FIND ADDRESS OF
068D  06 00                   LD B,0    ;WORD HANDLER
068F  09                      ADD HL,BC
0690  09                      ADD HL,BC
0691  4E                      LD C,(HL)
0692  23                      INC HL
```

Select RAM page A

Set PCTOP to top of current RAM page

Compare A with current page

Set (HL) to position of top of current basic page

```
0693  66                    LD H,(HL)
0694  69                    LD L,C
0695  C1                    POP BC
0696  C9                    RET
0697  21 BFF4         RMTPHL::    LD HL,0BFF4H
069A  3A FA7A                 LD A,(LSTPG)
069D  EF                      RST JT
069E  AF                      DB 0AFH
                      ;       CALL PGCHK
069F  C0                      RET NZ
06A0  2A FA92                 LD HL,(STKLIM)
06A3  D5                      PUSH DE
06A4  11 0080                 LD DE,80H
06A7  A7                      AND A
06A8  ED 52                   SBC HL,DE
06AA  D1                      POP DE
06AB  C9              RET
                      ;
06AC  2A FAAA         LDBSBT: LD HL,(BASBOT)
06AF  3A FA7A                 LD A,(LSTPG)
06B2  CD 0671                 CALL PGCHK  ;CHECK FOR LAST PAGE
06B5  C0                      RET NZ
06B6  7C                      LD A,H
06B7  C6 40                   ADD A,40H
06B9  67                      LD H,A
06BA  C9                      RET
06BB  E5              TPCH1:: PUSH HL
06BC  D5                      PUSH DE
06BD  ED 5B FAD6              LD DE,(PGTOP)
06C1  A7                      AND A
06C2  ED 52                   SBC HL,DE
06C4  D1                      POP DE
06C5  E1                      POP HL
06C6  C9                      RET
                      ;
                      ;
06C7  ED 5B FA83      DELETE: LD DE,(KBDBUF)
06CB  CD 06DB                 CALL GETLIN
06CE  C0                      RET NZ
06CF  E5              DELET1::        PUSH HL
06D0  4E                      LD C,(HL)
06D1  23                      INC HL
06D2  46                      LD B,(HL)
06D3  E1                      POP HL
06D4  CD 04B6                 CALL CPYNXT
06D7  CD 27FB                 CALL KILLVAR
06DA  C9               RET
                      ;
06DB  CD 121D         GETLIN::        CALL INT64K
06DE  D5              GETLN1::        PUSH DE
06DF  CD 061A                 CALL FINDLIN
06E2  2A FACF                 LD HL,(BASELIN)  ;HL = LINE, VALID IF Z FLAG
06E5  D1                      POP DE
06E6  C9                      RET
06E7  EF              DETPN:: RST JT
06E8  AE                      DB 0AEH
```

Load HL with Addr address from table

RMTPHL — Top of root page

Return if not last page

HL = (STKLIM) − 84

Top of available space in current page

Set HL to address of bottom of current page

Test HL against page top.

Load BC with length of line pointed to by HL

Find existing line, Select RAM page Start of line in HL.

```
                        ;       CALL BSTPHL
06E9  73                        LD (HL),E
06EA  23                        INC HL
06EB  72                        LD (HL),D
06EC  C9                        RET
06ED  3A FAD3           DECPG:: LD A,(CRNTPG)
06F0  3D                        DEC A
06F1  18 04                     JR NCP1
06F3  3A FAD3           INCPG:: LD A,(CRNTPG)
06F6  3C                        INC A
06F7  C3 0647           NCP1::  JP PGSLC1
                        ;
06FA  23                SCREEN: INC HL
06FB  CB 6F                     BIT 5,A
06FD  F5                        PUSH AF
06FE  30 06                     JR NC,SCR1
0700  E6 1F                     AND 1FH
0702  EF                SCREE2: RST JT
0703  AC                        DB 0ACH
                        ;       CALL PRINTX
0704  18 20                     JR ENDSCR
0706  FE 80             SCR1:   CP 128
0708  30 23                     JR NC,SCR2
070A  F5                        PUSH AF
070B  F5                        PUSH AF
070C  3E 1B                     LD A,27
070E  CD 00BC                   CALL GETSTR
0711  3E 5A                     LD A,'Z'
0713  CD 00BC                   CALL GETSTR
0716  F1                        POP AF
0717  E6 07                     AND 07
0719  CD 00BC                   CALL GETSTR
071C  F1                        POP AF
071D  CB 5F                     BIT 3,A
071F  28 05                     JR Z,ENDSCR
0721  3E 0C                     LD A,12
0723  CD 00BC                   CALL GETSTR
0726  F1                ENDSCR: POP AF
0727  C2 0012                   JP NZ,SCRR1
072A  F1                        POP AF
072B  E3                        EX (SP),HL
072C  C9                        RET
072D  FE C0             SCR2:   CP 192
072F  30 0D                     JR NC,SCRBC
0731  E6 1F                     AND 1FH
0733  C5                        PUSH BC  ;ONLY BC ON STACK
0734  47                        LD B,A
0735  7E                SCR21:  LD A,(HL)
0736  23                        INC HL
0737  EF                        RST JT
0738  AC                        DB 0ACH
                        ;       CALL PRINTX
0739  10 FA                     DJNZ SCR21
073B  C1                        POP BC
073C  18 E8                     JR ENDSCR
073E  79                SCRBC:  LD A,C
```

Store DE in top of current bank page

Select previous page

Select next page

```
073F  EF                    RST JT
0740  AC                    DB 0ACH
                      ;     CALL PRINTX
0741  78                    LD A,B
0742  18 BE                 JR SCREE2
                      ;
                      ;
                      ;
0744  2A FA7F         STACKS::      LD HL,(CALCBOT)
0747  22 FA81               LD (CALCST),HL
074A  21 FAD8               LD HL,GOSTACK
074D  22 FB41               LD (GOPTR),HL       ;Set gosub pointer to bottom of gosub stack
0750  21 FB4B               LD HL,START
0753  22 FA83               LD (KBDBUF),HL
0756  AF                    XOR A
0757  21 FB43               LD HL,GOSNUM
075A  77                    LD (HL),A           ;Zero gosub counter
075B  23                    INC HL
075C  77                    LD (HL),A     ;FORCOUNT
075D  32 FD69               LD (CONTFLG),A ;CONTFLG
0760  21 FFFE               LD HL,0FFFEH
0763  22 FD6A               LD (CONTAD),HL
0766  C5                    PUSH BC
0767  01 0000               LD BC,0
076A  CD 2C8D               CALL RESTENT
076D  C1                    POP BC
076E  C9                    RET
                      ;
                      ;

0010                  SCRRST  EQU     10H
0028                  ERRORRST        EQU     28H
0038                  FRONTRST        EQU     38H
0038                  MATHSRST        EQU     38H
                      ;
0010                  SCRRST  EQU      10H
                      ;
                      ;
                      ;
                      ;
                      ;
076F  21 FD7B         AUTORPT:      LD HL,AUTOCT
0772  7E                    LD A,(HL)
0773  FE B0                 CP 0B0H
0775  C8                    RET Z
0776  A7                    AND A
0777  28 03                 JR Z,ZERO
0779  AF                    XOR A
077A  35                    DEC (HL)
077B  C0                    RET NZ
077C  32 FD7C         ZERO: LD (LASTKY),A
077F  C9                    RET
                      ;
0780  F3              VDPINT: DI
0781  F5                    PUSH AF
0782  08                    EX AF,AF'
```

```
0783  F5              PUSH AF
0784  C5              PUSH BC
0785  D5              PUSH DE
0786  E5              PUSH HL
0787  FD E5           PUSH IY
0789  DD E5           PUSH IX
078B  DB 04           IN A,(CIFHI)
078D  CD 0924         CALL INTCLOCK
0790  3A FD5E         LD A,(INTFFF)
0793  EE 80           XOR 80H
0795  32 FD5E         LD (INTFFF),A
0798  FA 07AA   SMISS: JP M,INTFIN
079B  21 07B9         LD HL,INTADR
079E  CB 3F     INTLOOP:  SRL A
07A0  F5              PUSH AF
07A1  CF              RST 8
07A2  E5              PUSH HL
07A3  DC 07B7         CALL C,INTJMP
07A6  E1              POP HL
07A7  F1              POP AF
07A8  20 F4           JR NZ,INTLOOP
07AA  DD E1     INTFIN: POP IX
07AC  FD E1           POP IY
07AE  E1              POP HL
07AF  D1              POP DE
07B0  C1              POP BC
07B1  F1              POP AF
07B2  08              EX AF,AF'
07B3  F1              POP AF
07B4  FB              EI
07B5  ED 4D     RETT::  RETI
07B7  EB        INTJMP: EX DE,HL
07B8  E9        JPHL::  JP (HL)
                ;
                ;
                ;*********************
                ;INTURUPT WORDS GO HERE!!!!!
                ;
07B9  07D8      INTADR: DW INTRUP
07BB  09B0              DW BREAKKEY
07BD  076F              DW AUTORPT
07BF  07C7              DW VVVINT
07C1  FA98              DW USERINT
07C3  FA98              DW USERINT
07C5  FA98              DW USERINT
                ;
07C7  3A FAD2   VVVINT: LD A,(PAGE)
07CA  F5                PUSH AF
07CB  CD 00CB           CALL SCENTR
07CE  CD 3A55           CALL VINT
07D1  F1                POP AF
07D2  32 FAD2           LD (PAGE),A
07D5  D3 00             OUT (PGPORT),A
07D7  C9                RET
                ;
                ;
```

```
                        ;**************************
                        ;INTERUPT PART OF PROGRAM *
                        ;**************************
                        ;
07D8  3E 03             INTRUP: LD      A,3
07DA  47                INLOOP: LD B,A
07DB  08                        EX AF,AF'
07DC  78                        LD A,B
07DD  CD 07E6                   CALL MAINDO
07E0  08                        EX AF,AF'
07E1  3D                        DEC A
07E2  F2 07DA                   JP P,INLOOP
07E5  C9                        RET
                        ;
07E6  CD 0891           MAINDO: CALL    LOADER ;Load up pointers
07E9  E5                        PUSH    HL
07EA  D1                        POP     DE
07EB  23                        INC     HL
07EC  23                        INC     HL
07ED  23                        INC     HL
07EE  23                        INC     HL
07EF  CD 089E                   CALL    LOAD2 ;DE contains Address of Ctl Reg
07F2  E5                        PUSH    HL  ;SAVE Buffer Address.
07F3  D5                        PUSH    DE  ;Save ctrl reg Address
07F4  21 0009                   LD      HL,9  ;::::::::::::
07F7  19                        ADD     HL,DE ;HL points to DONE flag
07F8  7E                        LD      A,(HL) ;If the Done flag is not zero then
07F9  A7                        AND     A  ;go and finish gradiant.
07FA  20 29                     JR      NZ,DOENV ;::::::::::::
07FC  E5                        PUSH    HL  ;Save pointer to done flag.
07FD  11 FFFB                   LD      DE,-5  ;Point HL to out pointer
0800  19                        ADD     HL,DE ;address.
0801  C1                       POP     BC  ;Get pointer to done flag.
0802  D1                       POP     DE  ;Point DE to In pointer address.
0803  C5                       PUSH    BC  ;Put pointer to done flag back on stack.
0804  1A                      LD      A,(DE) ;Load in pointer in ACC
0805  BE                      CP      (HL)  ;If both pointers match ret
0806  28 1A                   JR      Z,FIN ;
0808  E1                NOTEQUL:POP   HL  ;Get pointer to Done flag.
0809  34                      INC     (HL)    ;Set Done flag
080A  EB                      EX      DE,HL
080B  E1                      POP     HL
080C  D5                      PUSH    DE
080D  E5                      PUSH    HL
080E  11 000A                 LD      DE,10
0811  19                      ADD     HL,DE
0812  7E                      LD      A,(HL)
0813  A7                      AND     A
0814  28 0C                   JR      Z,FIN
0816  E1                      POP     HL
0817  D1                      POP     DE
0818  1B                      DEC     DE
0819  1B                      DEC     DE
081A  1B                      DEC     DE
081B  1B                      DEC     DE
081C  01 0004                 LD      BC,4
```

```
                                        ; Copy 4 bytes from sound buffer
081F  ED B0                     LDIR    ; to control buffer
0821  C9                        RET     ;FINISHED
                        ;
0822  E1                FIN:   POP     HL  ;Balance Stack.
0823  E1                       POP     HL
0824  C9                       RET     ;FINISHED
                        ;
                        ;
0825  E1                DOENV: POP     HL      ;Get ctrl reg address (Base).
0826  11 0005                  LD      DE,5    ;Point it to frequency voltage.
0829  19                       ADD     HL,DE   ;----
082A  4E                       LD      C,(HL)  ;::::
082B  23                       INC     HL      ;Ld BC with value freq voltage.
082C  46                       LD      B,(HL)  ;::::
082D  2B                       DEC     HL      ;Restore pointer to freq volt.
082E  D1                       POP     DE      ;Get Snd buffer Base address.
082F  E5                       PUSH    HL      ;Save ctrl reg pointer pointing
                                               ;at frequency voltage.(1 ITEM ON STACK)
0830  13                       INC     DE
0831  13                       INC     DE
0832  13                       INC     DE      ;Point Snd buffer at freq gradiant.
0833  13                       INC     DE
0834  EB                       EX      DE,HL
0835  CF                       RST 8
0836  EB                       EX      DE,HL   ;DE holds pointer to volt gradiant.
0837  09                       ADD     HL,BC   ;Get new freq voltage.
0838  C1                       POP     BC      ;Get pointer to freq volt in ctrl reg.
0839  E5                       PUSH    HL      ;SAVE FREQ VOLTAGE ON STACK (note.is not included
                                               ;in stack count.
                                               ;
083A  D5                       PUSH    DE      ;Save pointer to volt gradiant (1 ITEM ON STACK)
083B  C5                       PUSH    BC
083C  EB                       EX      DE,HL   ;Put new volt freq in DE.
083D  E1                       POP     HL      ;HL cointains pointer to freq volt.
083E  73                       LD      (HL),E
083F  23                       INC     HL      ;Save freq voltage to freq volt address in ctrl buf.
0840  72                       LD      (HL),D
0841  23                       INC     HL      ;HL now points to volume voltage address in ctrl buf.
0842  4E                       LD      C,(HL)
0843  23                       INC     HL      ;BC now cointains volume voltage value and HL
0844  46                       LD      B,(HL)  ; is pointing to volume voltage in ctrl buffer.
0845  2B                       DEC     HL
0846  D1                       POP     DE      ;Get Snd buffer pointer to volume gradiant
0847  E5                       PUSH    HL      ;Save pointer to Volume voltage (1 ITEM ON STACK)
0848  EB                       EX      DE,HL
0849  CF                       RST 8
                                               ;DE cointans volume gradiant
                                               ;BC contains volume voltage
084A  EB                       EX      DE,HL   ;DE=HL & HL=DE
084B  09                       ADD     HL,BC   ;New volume voltage in HL.
084C  D5                       PUSH    DE
084D  C1                       POP     BC      ;BC now contains pointer to snd buffer at TIME.
084E  EB                       EX      DE,HL   ;DE=New volvolt::
084F  E1                       POP     HL      ;HL=Ctrl buffer pointing to volume voltage.(STACK=0)
0850  D5                       PUSH    DE      ;SAVE VOLUME VOLT ON STACK.NOT included in stack count
```

```
                                ;
                                ;
0851  73                        LD      (HL),E
0852  23                        INC     HL      ;Store volume voltage to volume voltage address.
0853  72                        LD      (HL),D
0854  23                        INC     HL      ;HL is now pointing to Done flag in Ctrl buffer.
0855  E5                        PUSH    HL      ;1 ITEM ON STACK.   HL : Ctrl + 4
0856  C5                        PUSH    BC
0857  E1                        POP     HL      ;HL=pointer to time in Snd buffer.   HL : Buff + 8
0858  CF                      RST 8                   HL = Buff + 10    DE = (Buff + 8)
0859  2B                        DEC  HL
085A  2B                        DEC     HL      HL = Buff + 8
085B  1B                        DEC     DE
085C  73                        LD      (HL),E  } Decrement and restore
085D  23                        INC     HL      } time
085E  72                        LD      (HL),D
085F  7B                        LD      A,E     } Test for zero time
0860  B2                        OR      D       }
0861  E1                        POP     HL          HL = Ctrl + 4
0862  20 0E                     JR      NZ,NOTDONE
0864  35                        DEC     (HL)  - Decrement done counter
0865  11 FFFB                   LD      DE,-5
0868  19                        ADD     HL,DE   HL = Ctrl + 4
0869  34                        INC (HL)        ;^^^NEW PART#########  } Increment table number
086A  7E                        LD      A,(HL)  }
086B  2B                        DEC     HL      } Test for last table
086C  BE                        CP      (HL)    }
086D  20 03                     JR      NZ,NOTDONE  } Back to beginning
086F  23                        INC     HL          } if so
0870  AF                        XOR     A           }
0871  77                        LD      (HL),A      }
                              ;
0872  08                     NOTDONE:EX     AF,AF'  }
0873  32 FE14                      LD      (CHAN),A } Channel number
0876  08                        EX      AF,AF'  }
0877  E1                        POP     HL      }
0878  29                        ADD     HL,HL   } VOL = Voltage / 64
0879  29                        ADD     HL,HL   }
087A  7C                        LD      A,H     }
087B  32 FE18                   LD      (VOL),A }
087E  E1                        POP     HL      }
087F  CB 3C                     SRL     H       }
0881  CB 1D                     RR      L       }
0883  CB 3C                     SRL     H       ;Xby8   } FRQ = Voltage / 8
0885  CB 1D                     RR      L       }
0887  CB 3C                     SRL     H       }
0889  CB 1D                     RR      L       }
088B  22 FE16                   LD      (FREQ),HL   }
088E  CD 08F6                   CALL    DIOUT0  - Output
                              ;
                              ;*********************MAIN ROUTINE************************
0891  E6 03                  LOADER::AND     3       ;From about here could be made
                                        ;into a routine.To ^^^^^^
                                        ;
0893  87                        ADD     A,A ;#####################################
0894  6F                        LD      L,A
```

```
0895  87          ADD   A,A ;MULT ACC BY No OF BYTES IN A CTL BUF  (10 bytes)
0896  87          ADD   A,A ;**************************************
0897  85          ADD   A,L
0898  21 FA52     LD    HL,CTRBADR
089B  85          ADD   A,L
089C  6F          LD    L,A
                  ;HL now contains Address of ctl buffer
089D  C9          RET
                  ;
089E  D5    LOAD2:PUSH  DE
089F  7E          LD    A,(HL) ;**********
08A0  6F          LD    L,A
08A1  26 00       LD    H,0
08A3  29          ADD   HL,HL ;Multiply buffer pointer
08A4  29          ADD   HL,HL ;by 12
08A5  54          LD    D,H
08A6  5D          LD    E,L
08A7  29          ADD   HL,HL
08A8  19          ADD   HL,DE ;**********
08A9  D1          POP   DE
08AA  D5          PUSH  DE
08AB  E5          PUSH  HL
08AC  EB          EX    DE,HL
08AD  23          INC   HL
08AE  CF    RST 8
                  ;
08AF  E1          POP   HL
08B0  19          ADD   HL,DE   ;Here HL is pointing to address
                  ;of buffer selected.
08B1  D1          POP   DE      ;DE is pointing to ctrl reg address.
08B2  C9          RET
                  ;
                  ;****************************
                  ;       OUTPUT TO SOUND CHIP
                  ;****************************
                  ;
                  ;CHANNEL IN MEMORY LOC 'CHAN'
                  ;FREQUENCY IN MEMORY LOC 'FREQ'
                  ;
                  ;VOLUME IN MEMORY LOC 'VOL'
                  ;
                  ;ALL REG SAVED
                  ;++++++++++++++++++++++++++++
                  ;
                  ;load channel from 'CHAN' &
                  ;convert to sound chip value.
                  ;
08B3  3A FE14 LDCHAN: LD    A,(CHAN)
08B6  E6 03       AND   3
08B8  47          LD    B,A
08B9  04          INC   B
08BA  21 08C1     LD    HL,CHANDT
08BD  23    LDCH0:INC   HL
08BE  10 FD       DJNZ  LDCH0
08C0  7E          LD    A,(HL)
08C1  C9    CHANDT:RET
```

```
08C2  80 A0 C0 E0          DB    128,128+32,128+64,128+64+32
                           ;.........................
                           ;
                           ;output freq value to sound
                           ;chip with channel number.
                           ;
08C6  CD 08B3              OFREQ:: CALL  LDCHAN
08C9  47                         LD    B,A
08CA  3A FE16                    LD    A,(FREQ)
08CD  E6 0F                      AND   0FH
08CF  B0                         OR    B
08D0  D3 06                      OUT   (SNDPTO),A
08D2  DB 03                      IN    A,(SNDPTI)
08D4  3A FE14                    LD    A,(CHAN)
08D7  E6 03                      AND   3
08D9  FE 03                      CP    3
08DB  C8                         RET   Z
                           ;
08DC  2A FE16                    LD    HL,(FREQ)
08DF  29                         ADD   HL,HL
08E0  29                         ADD   HL,HL
08E1  29                         ADD   HL,HL
08E2  29                         ADD   HL,HL
08E3  7C                         LD    A,H
08E4  E6 7F                      AND   127
08E6  D3 06                ONS1: OUT   (SNDPTO),A
08E8  DB 03                      IN    A,(SNDPTI)
08EA  C9                         RET
                           ;
                           ;
08EB  11 FE14              DIROUT: LD    DE,CHAN
08EE  21 FE04                      LD    HL,TEMP
08F1  01 0006                      LD    BC,6
08F4  ED B0                        LDIR
08F6  CD 08C6              DIOUT0: CALL OFREQ
                           ;output volume to sound chip
                           ;with channel number.
                           ;
08F9  CD 08B3              OVOL::  CALL LDCHAN
08FC  F6 10                        OR    16
08FE  47                           LD    B,A
08FF  3A FE18                      LD    A,(VOL)
0902  E6 0F                        AND   0FH
0904  EE 0F                        XOR   15
0906  B0                           OR    B
0907  18 DD                       JR    ONS1
0909  C9                          RET
                           ;
                           ;
                           ;**********************************
                           ;* REALTIME CLOCK res.1/64 sec  *
                           ;**********************************
                           ;Clock used with 1/64 sec inturupts.
                           ;From mem loc CLOCK is as follows:-
                           ;1/64 sec count
                           ;Sec's count. Min's count. Hours count.
```

```
                        ;
                        ;       22/9/83
                        ;
090A  C3                CDATA:  DB 0C3H
090B  1C11                      DW ENTERIN ;JP ENTERIN
090D  C3                        DB 0C3H
090E  1C03                      DW ENTER30 ;JP ENTER30
0910  C3                        DB 0C3H
0911  3622                      DW KEY1
0913  C3                        DB 0C3H
0914  18AF                      DW ERROR
0916  39 39 35 39              DB '99','59','59','0'+124
091A  35 39 AC
091D  0F                        DB 0FH  ;INTFFF
091E  40                        DB 40H                  ;TAPE RATE
091F  B0                        DB 0B0H                 ;MIDVAL
0920  2CF0                      DW RUN5         ;Return address 1
0922  2CDB                      DW RUN55                ;Return address 2
                      ;
                      ;
                      ;
0924  11 FD5D         INTCLOCK:       LD DE,CLOCK+6
0927  21 091C                 LD HL,CDATA+18
092A  06 07                 LD B,7
092C  1A              CLKLOOP:        LD A,(DE) ;Get count
092D  BE                        CP (HL)  ;reached it yet
092E  38 02                     JR C,FINCLK
0930  3E 2F                     LD A,'0'-1
0932  3C              FINCLK: INC A
0933  12                      LD (DE),A
0934  D8                      RET C   ;Not reached
0935  1B                       DEC DE
0936  2B                       DEC HL
0937  10 F3                    DJNZ CLKLOOP
0939  C9                       RET
                      ;
                      ;Routine to shut all
                      ;sound channel's off
                      ;including the noise
                      ;channel.
                      ;
                      ;ALL REG SAVED.
                      ;
093A  D5              KILSND::PUSH DE
093B  F5                      PUSH    AF
093C  AF                      XOR A
093D  32 FE18                 LD      (VOL),A
0940  21 FE14                 LD      HL,CHAN
0943  06 04                   LD      B,4
0945  77                      LD      (HL),A
0946  E5              KILSD0: PUSH HL
0947  C5                      PUSH BC
0948  CD 08F9                 CALL    OVOL
094B  C1                      POP  BC
094C  E1                      POP  HL
094D  34                      INC     (HL)
```

```
094E  10 F6              DJNZ  KILSD0
0950  F1                 POP   AF
0951  D1                 POP DE
0952  C9                 RET
                         ;
                         ;
                         ;Output a 'BELL' like
                         ;dong of a Note of
                         ;'G' of octave 6
                         ;
                         ;REG used:-'A' 'DE'
                         ;
0953  CD 093A            BELL::  CALL  KILSND  ;SHUT OFF SOUND CHAN
0956  01 0096                    LD    BC,150
0959  D5                FBELL:  PUSH DE
095A  3E 01                    LD    A,1
095C  32 FE14                  LD    (CHAN),A
095F  ED 43 FE16               LD    (FREQ),BC
0963  CD 08C6                  CALL  OFREQ
0966  3E 0F                    LD A,15
0968  32 FE18            BELL0:  LD    (VOL),A
096B  F5                       PUSH AF
096C  CD 08F9                  CALL  OVOL
096F  F1                       POP AF
0970  11 3000                  LD    DE,3000H
0973  1D                  BELL1:  DEC   E
0974  20 FD                    JR    NZ,BELL1
0976  15                       DEC   D
0977  20 FA                    JR    NZ,BELL1
0979  3D                       DEC A
097A  F2 0968                  JP    P,BELL0
097D  D1                       POP  DE
097E  C9                       RET
                         ;
                         ;
                         ;
                         ;
                         ;
                         ;       Init CTC and set all channels to
                         ;       disable inturupts and software reset.
097F  F3                 IJINIT::      DI
0980  ED 5E                    IM 2
0982  3E FF                    LD A,IJHIGH
0984  ED 47                    LD I,A
0986  3E F0                    LD A,IJLOW
0988  D3 0B                    OUT (PORT),A
098A  3E 03              KILLINT::     LD A,3
098C  D3 0B              KILL0:  OUT (PORT),A
098E  D3 09                    OUT (PORT+1),A
0990  D3 0A                    OUT (PORT+2),A
0992  D3 0B                    OUT (PORT+3),A
0994  ED 4D                    RETI
                         ;
                         ;Set up CTC channel 0 to cause inturupts
                         ;every 1/125 second.
                         ;
```

```
0996  F3              INT125::    DI
0997  E5                          PUSH HL
0998  21 0780                     LD HL,VDPINT  ;Get address of entry point
                                        ;of VDP inturupts
099B  22 FFF0                     LD (IJTABLE),HL ;and store to inturpt jump table.
099E  21 1C11                     LD HL,ENTERIN
09A1  22 FFF4                     LD (IJTABLE+4),HL
09A4  E1                          POP HL
09A5  3E A5                       LD A,0A5H
09A7  D3 0B                       OUT (PORT),A
09A9  3E 7D                       LD A,125
09AB  D3 0B                       OUT (PORT),A
09AD  FB                          EI
09AE  ED 4D                       RETI
                        ;
                        ;       The break key bits are as follows
                        ;
                        ;       BIT 0..set if break key depressed else reset.
                        ;       BIT 1..set if break key depressed at any time.Reset by moniter.
                        ;       BIT 2..if set and break key pressed will force a jump to
                        ;       routine 'ENTERIN'
                        ;
09B0  3A FD7E           BREAKKEY::      LD A,(LASTDR)
09B3  F5                        PUSH AF
09B4  3E FE                     LD A,0FEH
09B6  D3 05                     OUT (DR),A
09B8  DB 06                     IN A,(SENSE2)
09BA  CB 47                     BIT 0,A
09BC  3A FD81                   LD A,(BREAK)
09BF  20 07                     JR NZ,NOBREAK
                        ;********ALSO CHECK FOR SOFT SHIFT LOCK******
09C1  5F                        LD E,A
09C2  CD 00AD                   CALL SHIF
09C5  7B                        LD A,E
09C6  20 04                     JR NZ,BREK
09C8  CB 87             NOBREAK:        RES 0,A
09CA  18 02                     JR BRK
09CC  F6 03             BREK:   OR 3
09CE  32 FD81           BRK:    LD (BREAK),A
09D1  F1                        POP AF
09D2  D3 05                     OUT (DR),A
                        ;
09D4  3A FD81           ASBREAK:        LD A,(BREAK)
09D7  CB 47                     BIT 0,A
09D9  C8                        RET Z
09DA  CB 57                     BIT 2,A
09DC  C8                        RET Z
09DD  CB 8F                     RES 1,A
09DF  32 FD81                   LD (BREAK),A
09E2  D1                        POP DE  ;AF
09E3  D1                        POP DE  ;HL
09E4  D1                        POP DE
09E5  DD E1                     POP IX
09E7  FD E1                     POP IY
09E9  E1                        POP HL
09EA  D1                        POP DE
```

Interrupt jump table initialisation

A = 10100101 Timer : Interrupts on 256 pre-scale

Interrupts every 1/125th second

```
09EB  C1                    POP BC
09EC  F1                    POP AF
09ED  08                    EX AF,AF'
09EE  F1                    POP AF
09EF  C3 1C11               JP ENTERIN
                      ;
                      ;
09F2  E5              BREAKMON::PUSH HL
09F3  21 FD81               LD HL,BREAK
09F6  CB 46                 BIT 0,(HL)
09F8  E1                    POP HL
09F9  C9                    RET
                      ;
09FA  02 2C 02              DB 2,',',2
09FD  F7              SOUT: RST GETRST
09FE  C5                    PUSH BC
09FF  F7                    RST GETRST
0A00  C1                    POP BC
0A01  ED 79                 OUT (C),A
0A03  C9                    RET
                      ;
0A04  06 2C 06              DB 6,',',6
                      ;
0A07  F7              SAUTO: RST GETRST
0A08  C5                    PUSH BC
0A09  F7                    RST GETRST
0A0A  ED 43 FD77            LD (AUTOINC),BC
0A0E  E1                    POP HL
0A0F  ED 42                 SBC HL,BC
0A11  22 FD79               LD (AUTOST),HL
0A14  C9                    RET
                      ;
                      ;
                      ;
                      ;
                      END
```

A>

002B E1

```
                        .Z80
0000'                   ASEG
                        ORG     0A15H
                        ;
                        ;
FA81                    CALCST  EQU     0FA81H
11B9                    SMALL1  EQU     11B9H
                        ;
                        ;
0A15    54              SQUASH::    LD D,H
0A16    5D                          LD E,L
0A17    7E              SQ1:        LD A,(HL)
0A18    12                          LD (DE),A
0A19    FE FF                       CP 0FFH
0A1B    C8                          RET Z
0A1C    FE 20                       CP 32
0A1E    28 04                       JR Z,SQ3
0A20    23              SQ11:       INC HL
0A21    13              SQ2:        INC DE
0A22    18 F3                       JR SQ1
0A24    06 81           SQ3:        LD   B,81H
0A26    23              SQ4:        INC  HL
0A27    04                          INC B
0A28    28 08                       JR Z,SAVSQ
0A2A    BE                          CP (HL)
0A2B    28 F9                       JR Z,SQ4
0A2D    78                          LD   A,B
0A2E    3D                          DEC A
0A2F    12                          LD   (DE),A
0A30    18 EF                       JR   SQ2
0A32    78              SAVSQ:      LD   A,B
0A33    D6 03                       SUB 3
0A35    2B                          DEC HL
0A36    2B                          DEC HL
0A37    12                          LD   (DE),A
0A38    18 E7                       JR   SQ2
                        ;
                        ;
0A3A    CD 11B9         CHR$::  CALL SMALL1
0A3D    20 12                    JR NZ,CHRERR
0A3F    38 10                    JR C,CHRERR
0A41    0E 01           CHR1::  LD C,1
0A43    2A FA81         CHR0::  LD HL,(CALCST)
0A46    77                      LD (HL),A
0A47    23                      INC HL
0A48    71                      LD (HL),C
0A49    23                      INC HL
0A4A    36 00                   LD (HL),0
0A4C    23                      INC HL
0A4D    22 FA81                 LD (CALCST),HL
0A50    C9                      RET
0A51    EF              CHRERR: RST 28H
0A52    22                      DB 34
                        ;
                        ;
                        END
```

Compress string by tokenizing repeated spaces.

Put A, C, 0 on calculator stack