MONITOR ROM

Daniel Dočekal

Popis základního programového vybavení SORD M5

```
;*************************************************************************
;*    SORD M5                                                            *
;*            MONITOR ROM                                                *
;*************************************************************************


; Zakladni provozni ROM pocitace SORD m5
;
        ORG     Ø


;Definice portu

CTCØ    EQU     Ø               ;Casovac Z8ØCTC

CTCCHØ  EQU     CTC
CTCCH1  EQU     CTC+1
CTCCH2  EQU     CTC+2
CTCCH3  EQU     CTC+3

VDP     EQU     1ØH             ;Videoprocesor

VDPD    EQU     VDP             ;Datovy port
VDPC    EQU     VDP+1           ;Prikazovy port

SGC     EQU     2ØH             ;Tonovy generator

KEYBRD  EQU     3ØH             ;Klavesnice

KEYMDØ  EQU     KEYBRD
KEYMD1  EQU     KEYBRD+1
KEYMD2  EQU     KEYBRD+2
KEYMD3  EQU     KEYBRD+3
KEYMD4  EQU     KEYBRD+4
KEYMD5  EQU     KEYBRD+5
KEYMD6  EQU     KEYBRD+6

JOY     EQU     KEYBRD+7

PRINTR  EQU     4ØH             ;Datovy port

RESET   EQU     5ØH
MOTOR   EQU     5ØH
STRBE   EQU     5ØH
ACMT    EQU     5ØH

SIO     EQU     6ØH             ;prip.pripojene SIO

PIO     EQU     7ØH             ;prip.pripojene PIO


; DEFINICE AKCNICH ZNAKU

CR      EQU     ØDH
.CR     EQU     .CR
.LEFT   EQU     1CH
.RIGHT  EQU     1DH
.UP     EQU     1EH
.DOWN   EQU     1FH
```

```
.DEL    EQU     8
.CANCEL EQU     18H
.TAB    EQU     9
.BEL    EQU     7

.HOME   EQU     ØBH
.CLS    EQU     ØAH

.SCUP   EQU     5
.SCDW   EQU     3
.SCLF   EQU     4
.SCRG   EQU     6

.SCRØ   EQU     15H
.SCRE   EQU     16H

.DSPE   EQU     19H
.DSPC   EQU     1AH

.GRI    EQU     13H
.GRII   EQU     12H
.MCLR   EQU     11H
.TEXT   EQU     14H

.INSRT  EQU     1ØH
.NORML  EQU     ØFH

.RETRN  EQU     ØDH

.CRLF   EQU     17H

.LF     EQU     ØAH

.NXT    EQU     ØEH

.STRT   EQU     2


SPACE   EQU     2ØH
NUL     EQU     Ø

; URCENI ROZMERU PAMETI V SYSTEMU

MAXRAM  EQU     -1

MINRAM  EQU     2ØØØH

ROMØ    EQU     MINRAM
ROM1    EQU     MINRAM+2ØØØH

EXROM   EQU     ROM1+2ØØØH

RAMSYS  EQU     7ØØØH
ENDSYS  EQU     7FFFH

EXRAM   EQU     ENDSYS+1


; DEFINICE POMERU V VRAM
```

```
;USPORADANI  I

NOUSE   EQU     Ø

SETSPR  EQU     2ØØØH

SETØ    EQU     28ØØH
SET1    EQU     SETØ+8ØØH

SCRØ    EQU     38ØØH
SCR1    EQU     SCRØ+4ØØH

ATRØ    EQU     3BØØH
ATR1    EQU     ATRØ+4ØØH

COLØ    EQU     3B8ØH
COL1    EQU     COLØ+4ØØH


;USPORADANI II
COLII   EQU     Ø

SPRCHR  EQU     COLII+18ØØH

SETII   EQU     2ØØØH

SCRØII  EQU     SCRØ
SCR1II  EQU     SCR1

ATRØII  EQU     ATRØ
ATR1II  EQU     ATR1

COLØII  EQU     COLØ
COL1II  EQU     COL1


VRMEND  EQU     3FFFH
```

```
;=========================================================================
;-------------------------------------------------------------------------
; 0000          Uplny start systemu po OFF/ON nebo RESET
;=========================================================================

RST0:   DI                      ;Zakaz preruseni
        IM      2               ;Preruseni bude typ vektor
        LD      SP,SYSTAK       ;MeziSP pro BOOT
        JR      M5BOT           ;Nahozeni systemu

;=========================================================================
; 0008          Vyber flagu aktivni obrazovky pro zpracovani
;=========================================================================
; OUT - A:=(DIFLGA)
;       HL:=DIFLGA

RST1:   LD      HL,DIFLGA       ;Flag aktivni obrazovky
        LD      A,(HL)          ; vyber DIFLGA
        RET

;-------------------------------------------------------------------------
; 000D          Uprava pro smer Dolu
;-------------------------------------------------------------------------
;IN:    E-souradnice Y
;OUT:   E-prictena +1

ADJSTD: INC     E               ; Y:=Y+1
        RET


; Tento byt tu je nevyuzit vzhledem k RST

        RET     Z

;=========================================================================
; 0010          Zapis bytu do VRAM    (HL)v:=A
;=========================================================================
; IN:   A - byt k zapsani
;       HL- adresa kam zapsat

PBVRID: DI                      ;Zakaz preruseni pro i/o operaci
        CALL    PBVRAM          ;Vlastni zapis bytu
        EI                      ;Zpetne povoleni
        RET

;-------------------------------------------------------------------------
;Uprava pro smer vLevo
;-------------------------------------------------------------------------
; IN:   D - souradnice X
; OUT:  D - souradnice -1

ADJSTL: DEC     D               ;X:=X-1
        RET

;=========================================================================
; 0018          Precteni bytu z VRAM    A:=(HL)v
;=========================================================================
; IN:   HL-adresa pro VRAM
; OUT:   A-byt z VRAM
```

```
GBVRID: DI                      ;Zakaz preruseni pro i/o operace
        CALL    GBVRAM          ;Vlastni precteni bytu
        EI                      ;Zpet povoleni preruseni
        RET

;-------------------------------------------------------------------------
;Uprava pro smer nahorU
;-------------------------------------------------------------------------
; IN:   E -souradnice Y
; OUT:  E -upravena -1

ADJSTU: DEC     E               ;Y:=Y-1
        RET

;=========================================================================
; 0020          Uzivatelsky restart do ROM od 2000H
;=========================================================================

RST4:   JP      2005H           ;Adresa v hlavicce ROM modulu

;-------------------------------------------------------------------------
;Uprava smeru pro vpRavo
;-------------------------------------------------------------------------
; IN:   D - souradnice x
; OUT:  D - upravena o +1

ADJSTR: INC     D               ;X:=X+1
        RET

;Tady je zase prebytecne misto kam se nic nedalo

        DB      0,0,0

;=========================================================================
; 0028          Uzivatelsky restart do ROM modulu
;=========================================================================

RST5:   JP      2008H           ;Skok do hlavicky ROM modulu

;-------------------------------------------------------------------------
;Pomocne instrukce pro ERRRET
;-------------------------------------------------------------------------

RETERR: SCF                     ;Nahod CY pro hlaseni chyby
        POP     HL              ;Vyber navratovou adresu
        EI                      ;Povol preruseni

;-------------------------------------------------------------------------
; 002E          Vyuzivan pro zruseni ucinku ruznych operaci
;-------------------------------------------------------------------------

EXRET:  RET                     ;A vrat se do volaci rutiny

;A zase prebyl jeden byt

        DB      0

;=========================================================================
; 0030          Uzivatelsky restart do RAM systemu
;=========================================================================
```

```
; !Skace do syst.prom v RAM - nutno zajistit vektor!

RST6:   JP      IVCTC6          ;Skok na vektor


;=============================================================================
; ØØ33          BOOT m5 z magnetofonu - nekonecna smycka
;=============================================================================

RBTCM:  CALL    BTCMT           ;Zavolej vlastni BOOT CMT
        JR      RBTCMT          ;Kdyz se vratil pokracuj
                                ;vratit se nesmi

;=============================================================================
; ØØ38          Uzivatelsky restart do RAM systemu
;=============================================================================
;!Skace do syst.prom v RAM -nutno zajistit vektor!

RST7:   JP      IVCTC7          ;Skok na vektor

;=============================================================================
; ØØ3B          Skokova tabulka pro rozlisovani typu chyb
;=============================================================================
; IN:   navratova adresa po CALL a z ni vybrany byt
; OUT:  neprimo - nahozene CY a v A kod chyby

ERRTBL:
        NOP     ;err7CH         neni definovan
.OVIEW: NOP     ;Err7DH         kurzor mimo okno
.CATRG: NOP     ;Err7EH         chyba V GTSTEP
.SPRDA: NOP     ;Err7FH         sprit je mimo obrazovku
.ACMIU: NOP     ;Err8ØH         neni definovanA
.CHKSM: NOP     ;Err81H         chyba kontr.souctu nebo VERIFY
.BREAK: NOP     ;Err82H         stisk RESET pri oper.s CMT
.FINUM: NOP     ;Err83H         jine jmeno pri VERIFY
.BUFFL: NOP     ;Err84H         pretekl KBUF
.DATOT: NOP     ;Err85H         preteceni buf pri OLD
                ;               cteni EOF bloku
                ;               nejsou data v KBUF
.ILGDM: NOP     ;Err86H         barveni znaku nelze
.TIMOT: NOP     ;Err87H         preteceni casu DWCNT
.SPRPOS:NOP     ;Err88H         chyba pozice spritu
.ILGCM: NOP     ;Err89H         moc EVENTu
                ;               chyba V RDCHR,VIEW
.SPNFD: NOP     ;Err8AH         neni definovana
.EVNRD: NOP     ;Err8BH         chyby spousteni DWCNT,UPCNT
.ELGEL: NOP     ;Err8CH         cte se jiny blok nez se melo
        NOP     ;Err8DH         neni definovano

;Vyhodnoceni cisla chyby a navrat

ERRRET: CALL    MTROF           ;Vypni motor byla chyba
        EX      (SP),HL         ;Navratova adresa do HL
        DEC     HL              ;Uprava na kod chyby
        DEC     HL              ;ktery je uschovan v
        LD      A,(HL)          ;adrese za CALL err
        SUB     ØBFH            ;Uprava na kod chyby
        JR      RETERR          ;A zaridit navrat

;=============================================================================
```

```
; 0058          Nahozeni systemu M5 po strance VDP,CTC,SGC,SYSPROM
;=============================================================================

M5BOT:  LD      A, 70H          ;Vektor 7000H pro CTC
        LD      I, A            ;do I registru
        LD      HL,DIFLGA       ;adresa obrazových sysv
        LD      BC,01CEH        ;delka prostoru
        CALL    NULPAD          ;zapl nulami
        LD      HL,SYTIDT       ;adresa tabulky sysprom
        LD      DE,7000H        ;adresa sysprom
        LD      BC,5DH          ;delka tabulky inic
        LDIR                    ;inicializace sysprom
        CALL    CTCINT          ;inicializace CTC
        EI                      ;TED JIZ BEZI PRERUSENI!
        CALL    SGINT           ;inicializace SGC
        CALL    VDPINT          ;inicializace VDP
        CALL    RCCSNM          ;Test na MONITOR ROM
        CALL    CHKRAM          ;Tes na RAM
        CALL    CHKRAM          ;najdi ROM a skoc
        JR      RBTCNT          ;nenasel - BOOT z CNT

;=============================================================================
; 0085          Konverze A na velka pismena
;=============================================================================
; IN:   A -  znak jez ma byt upraven
; OUT:  A -  pripadne upraveny znak

GTCAPC: CP      'a'             ;je to 'a'?
        RET     C               ;je    a  -  ven
        CP      '{'             ;je to '{'?
        RET     NC              ;je       -  ven
        SUB     '`'             ;konverze na velké
        RET

;=============================================================================
; 008E          Vyhledani zacatku a konce RAM + kontrola na OK?
;=============================================================================
; OUT:  nastaví syst. prom
        HL:=SYSTAK
CHKRAM: DI                      ;nutno zakazat!
        LD      HL,7000H        ;tady musi zacit RAM
LOOPT   DEC     HL              ;a pojedeme dolu
        CALL    TSBYT           ;test bytu na RAM/ROM
        JR      Z,LOOPT         ;je porad RAM
        INC     HL              ;byt nad je RAM
        LD      A,H
        AND     3               ;test na cele kilobyt
        OR      L
        JR      NZ,CHHALT       ;není kB -  hav.stav
        LD      (SMENTA) HL     ;uloz zacatek RAM
LOOPU   INC     HL              ;a pojedeme nahoru
        CALL    TSBYT           ;test bytu
        JR      Z,LOOPU         ;porad je RAM
        LD      A,H
        AND     3               ;test na cele kB
        OR      L
        JR      NZ,CHHALT       ;nenbyl cely kB
        LD      (SMENEA),HL     ;konec paměti RAM
        LD      (SUMNEA),HL     ;prvni byt neRAM
```

```
        LD      HL,7300H        ;Zacatek pameti
        LD      (SUMMTA),HL     ; uzivatele pevne
        EI                      ;muze INT
        RET

;-------------------------------------------------------------------------
; 00BC          Test bytu (HL) na RAM/ROM pomoci 5A a A5
;-------------------------------------------------------------------------

TSBYT:  LD      D,(HL)          ;testovany byt
        LD      A,01011010B     ;test poloviny
        LD      (HL),A          ;zkusmo zapis
        CP      (HL)            ;je tam to same?
        RET     NZ              ;neni, chyba nebo ROM
        CPL                     ;A:=10100101B
        LD      (HL),A          ;test druhe poloviny
        CP      (HL)            ;je tam?
        LD      (HL),D          ;obnov byt!
        RET                     ;ven Z=1 kdyz OK

;-------------------------------------------------------------------------
; 00C7          Test pameti MONITOR ROM na SUMA=0
;-------------------------------------------------------------------------

RCCSMM: LD      HL,0            ;test pameti od 0

;-------------------------------------------------------------------------
; 00CA          Test 8kB pameti od adresy HL na SUMA=0
;-------------------------------------------------------------------------

RCCSM:  LD      A,20H           ;Delka je 32*256by

;-------------------------------------------------------------------------
; 00CC          Test A*256bytu pameti od adresy HL
;-------------------------------------------------------------------------

CHSM:   ADD     A,H             ;vypocet vyssiho koncoveho
        LD      B,A             ;bytu tesu
        LD      C,0             ;kontr.soucet ini
LOOPC   LD      A,C             ;tvorba
        ADD     A,(HL)          ; kontrolniho
        LD      C,A             ;souctu
        INC     HL              ;nova adresa
        LD      A,H             ;
        CP      B               ;test na konec
        JR      NZ,LOOPC        ;jeste neni konec
        LD      A,C             ;kontr.soucet
        OR      A               ;test na nulovost
        RET     Z               ;je nula -> ven

;-------------------------------------------------------------------------
; 00DB          ***** havarijni stav BCOL8 , FCOL8 a HALT *****
;-------------------------------------------------------------------------

CHHALT: DI                      ;zakaz preruseni
        LD      A,88H           inic.kod
        OUT     (VDPC),A        ;zapis do VDP
        DEC     A               ;prikazovy kod
        OUT     (VDPC),A        ;zapis 87h
        HALT    ;BCOL8,FCOL8 a hal$
```

```
;-------------------------------------------------------------------------
; ØØE4          Nalezeni modulu ROM a skok do modulu
;-------------------------------------------------------------------------

CHKROM: LD      A,(2ØØØH)       ;ATTR ROMØ od 2ØØØH
        INC     A               ;test na Øffh - BUS
        LD      HL,EXRET        ;falesna navr.adresa
        PUSH    HL              ;navratova adresa
        JR      Z,TROM2         ;byla sbernice-test2
        LD      HL,(2ØØ1H)      ;adresa startu modulu
        EX      (SP),HL         ;na sp misto EXRET
        PUSH    AF              ;uschova ATTR
        LD      HL,(2ØØ3H)      ;adresa IPL startu
        CALL    JMPHL           ;GO TO IPL -vrati se
        POP     AF              ;attribut
        CP      2+1             ;je to 16k OD 2ØØØH?
        JR      Z,EXITC         ;ano je -> hop na ne
TROM2:  LD      A,(4ØØØH)       ;ATTR ROM2
        INC     A               ;test na Øffh-BUS
        JR      Z,EXITC         ;ano -skok na EXRET
                                ; nebo na (2ØØ1H)
        LD      HL,(4ØØ1H)      ;autostart ROM2
        EX      (SP),HL         ;vyber EXRET ci ROM1
        CALL    JMPHL           ;zavolej EXRET/ROM1
EXITC   POP     HL              ;start ROM2,1
        JP      (HL)            ;skoc do ni

;=========================================================================
; Ø1ØD          BOOT z CMT - vsechny .CM nahraje a potom prip.spusti
;=========================================================================

BTCMT:  LD      DE,FNDMV        ;adresa otazniku v pameti
        LD      HL,7262H        ;adresa bufferu pro IDT
        CALL    RFIDC           ;precti IDT
        JR      C,BTCMT         ;byla chyba ->znovu
        BIT     2,(HL)          ;test na .DT
        JR      NZ,BTCMT        ;je .DT -> znovu
BTCMTØ: PUSH    HL              ;adresa IDT schovana
        BIT     3,(HL)          ;je to .VM ?
        LD      A,' '           ;znak k tisku
        JR      NZ,PRESK        ;je to .VM
        LD      A,'*'           ;znak k tisku
PRESK:  CALL    DSPCH           ;vytiskni navesti
        INC     HL              ;adresa jmena
        LD      B,9             ;delka jmena
        CALL    DPSLN           ;vytisk jmena
        CALL    CRETL           ;vytisk CRLF
        POP     HL              ;adresa attr IDT
        BIT     3,(HL)          ;je to .VM?
        JR      NZ,BTCMT        ;ano .VM -> znovu
        XOR     A               ;je OLD
        CALL    BLOAD           ;precti data
        PUSH    AF              ;uschova CY
        CALL    C,BEL           ;chyba -> zazvon
        POP     AF              ;zpet CY
        JR      C,BTCMT         ;chyba -> znovu
        LD      HL,7262H        ;adresa IDT
        BIT     1,(HL)          ;je autostart?
        JR      Z,BTCMT         neni -> cti dal
```

```
        LD      DE,ØEH          ;offset na autostart
        ADD     HL,DE           ;v HL adr na Word
        LD      E,(HL)          ;
        INC     HL              ;vyber
        LD      D,(HL)          ;adresy
        EX      DE,HL           ;autostartu
        JP      (HL)            ;skoc na nahravku

;=======================================================================
; Ø151          Deleni   HL:= HL/A
;-----------------------------------------------------------------------

DIVIDS: LD      C,A             ;delime cislem A

;-----------------------------------------------------------------------
; Ø152          Deleni   HL:= HL/C
;-----------------------------------------------------------------------
; IN:   HL- delenec
        A - delitel (resp.C)
; OUT:  HL- podil
;       A - zbytek

DIVS5:  XOR     A               ;A:=Ø,CY:=Ø bit vysledku
        LD      B,16            ;pocet bitu vysledku
LOOPM:  ADC     HL,HL           ;rotace CY<-HL<-bit
        ADC     A,A             ;CY<-A<-HL<-bit
        JR      C,PRESA
        CP      C               ;porovnej ,dokud je mensi
        CCF                     ;nez A tak posouvej
        JR      NC,PRESB        ;porad dal
PRESA:  SUB     C               ;tento bit vysl.:=1
        SCF
PRESB:  DJNZ    LOOPM           ;pokracuj
        ADC     HL,HL           ;zarad posl.bit
        RET

;=======================================================================
; Ø165          Inicializacni tabulka nejdulezitejsich syst.promennych
;=======================================================================

SYTIDT:
        DW      IGNORI          ;CTC#Ø vektor
        DW      MPLAY           ;CTC#1 vektor
        DW      IGNORI          ;CTC#2 vektor
        DW      CTC3SP          ;CTC#3 vektor

        DB      ØC3H            ;vektor RST5
        DW      Ø
        DB      ØC3H            ;vektor RST6
        DW      Ø
        DW      SCCDT           ;adresa CTRL tab.
        DW      7ØØØH           ;zacatek RAM
        DW      8ØØØH           ;konec RAM
        DW      73ØØH           ;zacatek uziv.RAM
        DW      8ØØØH           ;konec uziv.RAM
        DB      Ø               ;prepinac obrazovek
        DB      33              ;prenosova rychlost
        DB      94H             ;flag klavesnice
        DW      KTKBD           ;adresa konv.tabulky
        DB      Ø,Ø,Ø,Ø         ;pozice JOY,ATTACK
```

```
        DW      EXRET           ;preruseni JOY
        DW      EXRET           ;preruseni ATTACK
        DW      IGNORB          ;preruseni SHIFT RESET
        DW      IGNORB          ;preruseni CTRL RESET
        DB      Ø,Ø,Ø           ;typy posl.tlacitek
        DB      5               ;citac omylu stisku
        DB      1EH,1EH         ;casy autorep.-pocatek
        DB      4,4             ;casy autor.-interval
        DW      KEYBUF          ;adresa KBUF
        DB      Ø,Ø             ;citace KBUF
        DB      3FH             ;delka KBUF
        DW      Ø               ;pocet period WAIT
        DB      32H             ;doba per.WAIT
        DB      Ø               ;TERMAL
        DB      4,2             ;zvuk klavesy

        DB      23H             ;EVENT flag
        DB      Ø               ;uziv.eventy Ø
        DW      Ø               ;adr.tabulky EVENTu
        DB      Ø               ;flag vedeni
        DB      32H,32H         ;perioda UPCNT
        DW      Ø               ;UPCN
        DB      32H,32H         ;perioda DWCNT
        DW      Ø               ;DWCNT
        DB      32H             ;perioda H:M:S
        DB      Ø,Ø,Ø           ;H,M,S
        DB      ØFFH,ØFFH       ;ALARM H,M
        DW      EXRET           ;preruseni ALARM
        DW      EXRET           ;preruseni EVHPRC
        DW      EXRET           ;obsluha spritu
        DB      8ØH             ;status spritu
        DB      2,1             ;krok spritu
        DB      Ø,Ø             ;citace zvuku
        DB      7               ;flag tiskarny
        DB      8Ø              ;pocet zn.tiskarny
        DB      Ø               ;pozice hlavy

;=========================================================================
; Ø1C2          Inicializace CTC- zapisy konstant
;-------------------------------------------------------------------------

CTCINT: XOR     A
        OUT     (CTCØ),A        ;zastaveni #Ø
        LD      HL,TCTCINT      ;adr.inic.tabulky
        LD      BC,4ØØH         ;B-pocet,C-port
LOOPI   LD      A,(HL)          ;byt
        OUT     (C),A           ;zapis
        INC     HL              ;nova adresa
        LD      A,(HL)          ;druhy byt
        OUT     (C),A           ;vyslat
        INC     HL              ;dalsi dvojice
        INC     C               ;novy port
        DJNZ    LOOPI           ;opakuj pro 4
        RET

; Ø1D7          Tabulka inic.kodu pro CTC#Ø-3
        DB      ØC7H,1          ;CTC#Ø
        DB      ØA7H,ØEH        ;CTC#1
        DB      57H,17H         ;CTC#2
        DB      ØC7H,1          ;CTC#3
```

```
;=============================================================================
;=============================================================================
; Ø1DF          Obslouzeni preruseni od CTC#3 - zajistuje beh systemu
;=============================================================================

CTC3SP: PUSH    AF
        PUSH    BC
        PUSH    DE
        PUSH    HL
        CALL    IGNORJ          ;falesne RETI
        IN      A,(VDPC)        ;STATUS VDP
        LD      (SPSTUS),A
        LD      HL,UEVMGF       ;flag EVENTu
        BIT     Ø,(HL)          ;flag vedeni
        SET     Ø,(HL)          ;EVENTY se vedou
        JR      NZ,CTC3EX       ;a jiz se vedly->ven
        EI                      ;ted uz povol
        LD      HL,EVMGFG
        BIT     Ø,(HL)          ;flag SPRITU
        LD      HL,(SPRPRC)     ;proces
        CALL    NZ,JMPHL        ;skok na proces
        LD      HL,EVMGFG
        BIT     1,(HL)          ;flag hodin
        CALL    NZ,CLOCKSP      ;proces hodin
        LD      HL,EVMGFG
        BIT     2,(HL)          ;flag UPCNT
        CALL    NZ,UPCTSP       ;proces UPCNT
        LD      HL,EVMGFG
        BIT     3,(HL)          ;flag DWCNT
        CALL    NZ,DWCTSP       ;proces DWCNT
        CALL    RSTSP           ;zjisti RESET-vykonej
;-----------------------------------------------------------------------------
; Ø21B          pouze JOY,KEY a USEREVENT

IGNORB: LD      HL,EVMGFG
        BIT     4,(HL)          ;flag JOY
        CALL    NZ,JOYSP        ;proces JOY
        LD      HL,EVMGFG
        BIT     5,(HL)          ;flag KEYBOARD
        CALL    NZ,KEYSP        ;proces klavesnice
        CALL    CVLOF           ;zajisti zvuk klavesnice
        LD      HL,EVMGFG
        BIT     6,(HL)          ;uzivatelske EVENTy
        CALL    NZ,UEVSP        ;proces uziv.eventu
        DI                      ;zakaz pro navrat
        LD      HL,UEVMGF
        RES     Ø,(HL)          ;EVENTy se mohou vest
; Ø23C          Zakonceni INT#3
;-----------------------------------------------------------------------------
CTC3EX: POP     HL
        POP     DE
        POP     BC
        POP     AF              ;zpet norm.sada
        EI                      ;povol.INT
        RET                     ;zpet do norm.behu

;-----------------------------------------------------------------------------
; Ø242          Obslouzeni prirustku H:M:S a ALARMU

CLOCKSP:LD      HL,CLKBTW       ;cas syst.hodin
        DEC     (HL)            ;zmensi citac
```

```
        RET     NZ              ;jeste ne
        LD      (HL),32H        ;inic.hodnota
        INC     HL              ;HL:=CLOCKS
        CALL    UPCLOCK         ;h:m:s +1
        RET     C               ;ven kdyz <1min
        CP      24              ;je den?
        LD      HL,(EVHPRC)     ;dnovy proces
        CALL    C,JMPHL         ;ano-vykonej
        LD      DE,(CLOCKM)     ;minuty
        LD      HL,(ALMTM)      ;minuty ALARMU
        OR      A               ;srovnej odectenim
        SBC     HL,DE           ;h:m := h:m Alarmu?
        LD      HL,(ALMPRC)     ;adr procesu
        RET     NZ              ;nenØ totez
        JP      (HL)            ;proces

;-------------------------------------------------------------------------
; Ø265          Obslouzeni UPCNT:=UPCNT+1

UPCTSP: LD      HL,UPCTBI       ;stand.cas UPCTN
        LD      A,(HL)
        INC     HL              ;HL:=UPCTBW citac
        DEC     (HL)            ;UPCTBW-1
        RET     NZ              ;jeste ne
        LD      (HL),A          ;inicializace
        LD      HL,(UPCNT)      ;vlastni UPCNT
        INC     HL              ;zvetsi
        LD      (UPCNT),HL
        RET

;-------------------------------------------------------------------------
; Ø275          Obslouzeni DWCNT:=DWCNT-1

DWCTSP: LD      HL,DWCTBI       ;stand.cas DWCNT
        LD      A,(HL)
        INC     HL              ;HL:=DWCTBW citac
        DEC     (HL)            ;zmensi citac
        RET     NZ              ;jeste ne
        LD      (HL),A          ;inic.
        LD      HL,(DWCNT)      ;vlastni DWCNT
        DEC     HL              ;-1
        LD      (DWCNT),HL
        LD      A,H             ;test
        OR      L               ; na DWCNT=Ø
        RET     NZ              ;nerovno
        LD      HL,EVMGFG
        SET     7,(HL)          ;flag docitani SET
        RES     3,(HL)          ;DWCNT OFF
        RET

;-------------------------------------------------------------------------
; Ø28F          Obslouzeni EVENTu uzivatele

UEVSP:  XOR     A               ;A:=Ø  ukoncovac
        LD      HL,(EVIFTA)     ;adr.inf.tabulky
CONTEV: EX      DE,HL           ; do DE
        LD      HL,EVMXNO       ;pocet EVENTu
        CP      (HL)            ;uz=A ->konec
        RET     Z
        INC     A               ;EVENT c.A je akt.
```

```
        LD      (UEVCT),A       ;c.akt.EVENTu
        LD      HL,6            ;delka polozky
        ADD     HL,DE           ;+offset
        LD      (UEVPT),HL      ;adresa pristi polozky
        EX      DE,HL           ;hl-adr.akt.polozky
        LD      E,(HL)
        INC     HL
        LD      D,(HL)
        BIT     7,D             ;DE:=WAIT ?
        JR      NZ,NXTEVN       ;ano WAIT -> dalsi
        DEC     DE              ;WAIT:=WAIT-1
        RES     7,D             ;EVENT neni OFF
        LD      (HL),D
        DEC     HL
        LD      (HL),E          ;obnov WAIT
        LD      A,D
        OR      E               ;test na WAIT=Ø
        JR      NZ,NXTEVN       ;neni ->dalsi
        INC     (HL)            ;um2le na +1
        INC     HL              ; ted jiz pujde vzdy
        INC     HL              ;adresa intervalu
        LD      A,(HL)          ;interval
        INC     HL              ;adresa citace
        DEC     (HL)            ;zmensi
        JR      NZ,NXTEVN       ;citac <>Ø
        LD      (HL),A          ;inic citac
        INC     HL
        LD      E,(HL)
        INC     HL
        LD      D,(HL)          ;adresa procesu
        EX      DE,HL
        CALL    JMPHL           ;skok na proces
NXTEVN: LD      HL,(UEVPT)      ;adresa noveho
        LD      A,(UEVCT)       ;cislo nove polozky
        JR      CONTEV          ;dalsi EVENT

;=============================================================================
; Ø2CF          Nastaveni H:M:S
;
; IN:   A - hodiny
;       HL- minuty  sekundy

STSCLK: DI
        LD      (CLOCKH),A      ;hodiny
        LD      (CLOCKS),HL     ;minuty a sekundy
        LD      HL,EVMGFG
        SET     1,(HL)          ;H:M:S ON

;-----------------------------------------------------------------------------
; Ø2DB          Zisk hodnot z H:M:S

; OUT:  A - hodiny
;       HL- minuty  sekundy

GTSCLK: DI
        LD      A,(CLOCKH)      ;hodiny
        LD      HL,(CLOCKS)     ;minuty a sekundy
        EI
        RET
```

```
;=============================================================================
; 02E4          Zvetseni casu v H:M:S
;-----------------------------------------------------------------------------

UPCLOCK:LD      A,59
        INC     (HL)            ;sec:=sec+1
        CP      (HL)            ;je uz 59sec?
        CCF
        RET     C               ;sec<=59 ->ven
        LD      (HL),0          ;sec:=0
        INC     HL              ;minuty
        INC     (HL)            ;min:=min+1
        CP      (HL)            ;min=59?
        RET     NC              ;min<=59 ->ven
        LD      (HL),0          ;min=0
        INC     HL              ;hodiny
        INC     (HL)            ;hod=hod+1
        LD      A,23
        CP      (HL)            ;hod=23?
        RET     NC              ;hod<=23 ->ven
        XOR     A
        LD      (HL),A          ;hod:=0
        RET

;=============================================================================
; 02FB          Spusteni UPCNT - dopredneho citace
;=============================================================================
; IN:   A - casova konst.UPCNT

STRTUC: LD      HL,EVMGFG
        BIT     2,(HL)          ;flag behu UPCNT
        JR      NZ,ERRUC        ;UPCNT jiz bezi->
        LD      H,A
        LD      L,A             ;inic.kod
        LD      (UPCTBI),HL     ;inic.UPCTBI,UPCTBW
        LD      HL,0
        LD      (UPCNT),HL      ;UPCNT:=0

;-----------------------------------------------------------------------------
; 030D          Znovuspusteni UPCNT
;-----------------------------------------------------------------------------

RSTRUC: LD      HL,EVMGFG
        SET     2,(HL)          ;UPCNT ON
        RET

;-----------------------------------------------------------------------------
; 0313          Zastaveni behu UPCNT
;-----------------------------------------------------------------------------

STOPUC: LD      HL,EVMGFG
        RES     2,(HL)          ;UPCNT OFF
        RET
        RET
```

```
;=============================================================================
; Ø319          Spusteni DWCNT - zpetneho citace
;=============================================================================
; IN:   A - konstanta DWCNT
;       HL- pocatecni hodnota

STRTDC: DI
        PUSH    HL              ;inic.hodnota
        LD      HL,EVMGFG
        BIT     3,(HL)          ;uz bezi DWCNT?
        POP     HL              ;zpet
ERRUC:  CALL    NZ,.EVNRD       ;chyba jiz bezi
        LD      (DWCNT),HL      ;inic.DWCNT
        LD      H,A
        LD      L,A             ;inic.kod citacu
        LD      (DWCTBI),HL     ;inic DWCTBI,DWCTBW
        LD      HL,EVMGFG
RUNDC:  OR      A               ;CY=Ø
        SET     3,(HL)          ;DWCNT ON
        RES     7,(HL)          ;DWCNT nedocital
        EI
        RET

;-----------------------------------------------------------------------------
; Ø336          Znovuspusteni DWCNT
;-----------------------------------------------------------------------------

RSTRDC: DI
        LD      HL,EVMGFG
        BIT     7,(HL)          ;je DWCNT u konce?
        JR      NZ,ERRUC        ;ano ->chyba
        JR      RUNDC           ;spust

;-----------------------------------------------------------------------------
;Zastaveni behu DWCNT
;-----------------------------------------------------------------------------
;
STOPDC: PUSH    HL
        LD      HL,EVMGFG
        RES     3,(HL)          ;DWCNT OFF
        POP     HL
        RET

;=============================================================================
; Ø348          nastaveni EVMGFG:=A bez bitu 7
;-----------------------------------------------------------------------------
; IN:   A - zadana hodnota
; OUT:  A - skutecna hodnota

STEVMF: LD      HL,EVMGFG
        ADD     A,A             ;CY=bit 7
        DI
        SLA     (HL)
        RRA                     ;konverze EVMGFG
        LD      (HL),A          ;se zachovanim b7

;-----------------------------------------------------------------------------
; Ø351          Vyber flagu Eventu - EVMGFG
;-----------------------------------------------------------------------------
; OUT:  A - hodnota v EMGFG
```

```
;       HL:=EVMGFG

GTEVMF: LD      A,(EVMGFG)      ;vyber EVMGFG
        EI
        RET

;===============================================================================
; Ø356          Inicializace uzivatelskych Eventu
;===============================================================================
; IN:   A - pozadovany pocet EVENTu
;       HL- adresa nove tabulky

ETREIT: CP      29H             ;pocet>29H?
        CALL    NC,.ILGCM       ;ano->chyba
        DI
        LD      (EVMXNO),A      ;inic.pocet EVENTu
        LD      (EVIFTA),HL     ;inic.pozice tabulky
        LD      HL,EVMGFG
        SET     6,(HL)          ;uziv.Eventy ON
        EI
        RET

;===============================================================================
; Ø369          Vypocet adresy spritu v VRAM
;===============================================================================
; IN:   A - cislo spritu
; OUT:  HL- adresa tabulky vlastnosti spritu

GTSATA: PUSH    DE
        LD      HL,(SATTLA)     ;tabulka spritu
        AND     1FH             ;c.spritu Ø-31
        ADD     A,A
        ADD     A,A             ;A:=A*4 -tbl po 4by
        LD      E,A
        LD      D,Ø             ;DE:=4*A  offset
        ADD     HL,DE           ;adresa spritu
        POP     DE
        RET

;===============================================================================
; Ø377          Prepocet souradnic spritu z LOG na FYZ
;===============================================================================
; IN:   DE - logicka pozice Y
;       HL - logicka pozice X

; OUT:  D  - fyzicka pozice X
;       E  - fyzicka pozice Y
;       B  - ECB bit

GTSPPC: LD      B,Ø
        LD      A,(GRFLA)       ;velikost spritu
        AND     38H             ;maska
        LD      C,A             ;BC=8,16,32
        LD      A,E
        INC     D
        JR      Z,YNEG          ;Y je zaporna
        DEC     D               ;oprava Y zpet
        JR      NZ,CHYSPR       ;je + a >256->chyba
        CP      ØD1H
        JR      NC,CHYSPR       ;Y>ØD1H ->chyba
```

```
        CP      ØCØH
        JR      NC,GTCØ         ;vetsi ØCØH
        JR      LTCØ            ;mensi
YNEG:   CP      ØD2H
        JR      C,CHYSPR        ;<-46 chyba
        NEG                     ;udelej kladnou
        CP      C               ;test s rozmerem
        JR      C,LTCØ          ;presahuje do obrazovky
GTCØ:   SET     Ø,B             ;flag ze neni videt
        LD      A,7FH           ;kod chyby .SPRDA
LTCØ:   DEC     E               ;korekce pro VDP
        LD      D,L
        INC     H               ;je x zaporne?
        JR      Z,SPRNEG        ;ano-LOC dle praveho rohu
        SRL     B               ;RES Ø,B a CY=1nebyl videt
        DEC     H               ;oprava x
        RET     Z               ;je x<255 prepocteno
CHYSPR: LD      E,ØD1H
        CALL    .SPRPOS         ;chyba
SPRNEG: SET     7,B             ;x je zaporne - flag
        LD      A,ØE0H          ;a:=-32
        CP      L
        JR      NC,CHYSPR       ;x<-32 chyba
        LD      A,L
        ADD     A,2ØH           ;x:=x+32
        LD      D,A             ;d:=x+32 korekce
        LD      A,L
        NEG                     ;a=ABS(x)
        CP      C               ;jeste presahuje?
        JR      NC,CHBSPR       ;nepresahuje chyba
        OR      A               ;je videt CY=Ø
        BIT     Ø,B             ;byl videt pro Y?
        RES     Ø,B             ;Ted je videt!
        RET     Z               ;a byl i predtim
CHBSPR: CALL    .SPRDA          ;chyba

;=============================================================================
; Ø35C          Smazani spritu z obrazovky
;=============================================================================
; IN:   A - cislo spritu

DELSPR: CALL    GTSATA          ;zjisteni adresy
        LD      A,ØD0H          ;nova pozice Y
        RST     PBVRID          ;zapis pozici Y
        RET

;-----------------------------------------------------------------------------
; Ø3CC          Smazani spritu z obrazovky
;-----------------------------------------------------------------------------
; IN:   A - cislo spritu

ERSSPR: LD      E,ØD2H          ;y sour.kam dat po ERASE

;=============================================================================
; Ø3CE          Lokalizace spritu
;-----------------------------------------------------------------------------
; IN:   A - cislo spritu
;       DE- logicka pozice Y
;       HL- logicka pozice X
```

```
; OUT:  E - fyzicka pozice Y
;       D - fyzicka pozice X
;       B - ECB v bitu 7

MVSPA:  PUSH    HL              ;uschova xLOG
        PUSH    HL
        CALL    GTSATA          ;adresa spritu
        EX      (SP),HL         ;xLOG do Hl
        CALL    GTSPPC          ;prepocet na fyzicke
        POP     HL              ;adr tabulky
        PUSH    AF              ;cislo spritu
        DI
        CALL    STVWAD          ;priprava adres VDP
        OUT     (C),E           ;zapis Y do VRAM
        PUSH    AF
        POP     AF              ;casovani
        OUT     (C),D           ;zapis X do VRAM
        INC     HL
        INC     HL
        INC     HL              ;adresa barvy
        RST     GBVRID          ;vyber barvy
        AND     ØFH             ;orezani na 4nizsi
        OR      B               ;dopln o B - bit'
        RST     PBVRID          ;zapis barvu
        POP     AF
        POP     HL
        RET

;=============================================================================
; Ø3EE          Prepocet souradnic FYZ->LOG
;=============================================================================
; IN:   B - ECB v bitu 7
;       D - fyzicka pozice X
;       E - fyzicka pozice Y

; OUT:  DE- logicka pozice Y
;       HL- logicka pozice X

GTSPLC: LD      H,Ø
        LD      L,D             ;HL:=fyz.x
        BIT     7,B             ;test ECB
        JR      Z,ECBØ          ;kdyz Ø preskoc
        DEC     H               ;H=-1
        LD      A,D             ;A:=D+32
        SUB     2ØH             ;uprava pro levy roh
        LD      L,A             ;HL pozice
ECBØ:   LD      D,Ø             ;DE:=y pozice
        INC     E               ;korekce dle VDP
        LD      A,ØD1H          ;priznak ze neni v obr.
        CP      E               ;test
        JR      Z,CHASPR        ;neni v obrazovce
        RET     NC              ;je to ok  pozice<D2
        INC     A               ;test na D2
        CP      E
CHASPR: CALL    Z,.SPNFD        ;byl na D2->chyba
        DEC     D               ;vice ja D2
        OR      A               ;tedy je to zaporne
        RET                     ;d=ØFFh=-1

;=============================================================================
```

```
;=============================================================================
; Ø4ØB          Premisteni spritu, ale ne plynuly pohyb
;=============================================================================
; IN:   A - cislo spritu
;       B - prirustek v X
;       C - prirustek V Y

; OUT:  DE- logicka pozice Y
;       HL- logicka pozice X
;       B - ECB v bitu 7

MVSPR:  PUSH    AF              ;cislo spritu
        PUSH    BC
        CALL    GTSPOS          ;vyber inform o spritu
        POP     BC
        CALL    NC,ADDVCT       ;nebyla-li chyba vypocitej
        POP     AF              ;pozici
        PUSH    DE
        CALL    MVSPA           ;sprit na HL,DE
        POP     DE              ;zpet y
        RET

;-----------------------------------------------------------------------------
; Ø41B          Pripocet vektoru pohybu spritu
;-----------------------------------------------------------------------------
; IN:   HL- logicka pozice X
;       DE- logicka pozice Y
;       B - vektor X
;       C - vektor Y

; OUT:  HL- logicka pozice X
;       DE- logicka pozice Y

ADDVCT: PUSH    BC
        CALL    VEKTOR          ;pricti vektor z C
        POP     BC
        LD      C,B             ;C=vektor z B
VEKTOR: LD      B,Ø             ;BC=vektor z C
        BIT     7,C             ;test na zaporny
        JR      Z,VPLUS         ;je kladny OK
        DEC     B               ;a ted je zaporny
VPLUS:  EX      DE,HL           ;zamena pozice
        ADD     HL,BC           ;pricti posuv
        RET

;=============================================================================
; Ø42B          Vyber informaci o spritu
;=============================================================================
; IN:   A - cislo spritu
; OUT:  B - barva
;       C - kod predlohy
;       DE- logicka pozice Y
;       HL- logicka pozice X

GTSPOS: CALL    GTSATA          ;adresa tabulky
        DI
        CALL    GBVRAM          ;vyber y
        LD      E,A             ;E:=y
        INC     HL
        CALL    GBVRAM
```

```
        LD      D,A             ;D:=x
        INC     HL
        EI
        RST     GBVRID          ;C:=predloha
        LD      C,A
        INC     HL
        RST     GBVRID
        LD      B,A             ;B:=barva s ECB
        CALL    GTSPLC          ;prepocet FYZ->LOG
        RES     7,B             ;ECB:=Ø
        RET

;=============================================================================
; Ø445          Nastaveni barvy spritu
;=============================================================================
; IN:   A - cislo spritu
;       B - pozadovana barva

STSCOL: CALL    GTSATA          ;adresa tabulky
        INC     HL
        INC     HL
        INC     HL
        DI
        CALL    GBVRAM          ;precteni barvy z+3
        AND     8ØH             ;bit 7 - ECB
        OR      B               ;dovnitr barvu
        RST     PBVRID          ;zapsat
        RET

;-----------------------------------------------------------------------------
; Ø454          Nastaveni predlohy spritu
;-----------------------------------------------------------------------------
; IN:   A - cislo spritu
;       C - kod pozadovane predlohy

STSCOD: CALL    GTSATA          ;adresa tabulky
        INC     HL
        INC     HL
        LD      A,C             ;predloha
        RST     PBVRID          ;zapsat na +2
        RET


;=============================================================================
; Ø45C          Zvoleni velikosti spritu
;=============================================================================
; IN:   A - nastavovana velikost

MAGFY:  AND     3               ;velikost na 3bity
        LD      C,A
        LD      B,Ø             ;do BC
        LD      HL,SPRSIZ       ;tabulka velikosti
        ADD     HL,BC           ;adresa polozky
        RRCA
        RRCA                    ;velikost*4
        OR      (HL)            ;presun vyssich peti
        LD      HL,GRFLA
        LD      C,A             ;flag do C
        LD      A,(HL)          ;graficky flag
        AND     7               ;zachovani spodnich 3
```

```
        OR      C               ;dodat nove
        LD      (HL),A          ;zaznamenat

;=============================================================================
; Ø471          Podle toho ktera je DISPLAY provede VREGI
;=============================================================================

VREGRI: LD      A,(SVSSW)
        BIT     1,A             ;ktera je display?
        CALL    Z,VREGI         ;kdyz procesni tak inic
        RET


;-----------------------------------------------------------------------------
; Ø47A          Pomocne vypocty pro pocitani vektoru spritu
;-----------------------------------------------------------------------------

CALSTV: PUSH    BC
        LD      A,(SPSTPC)      ;krok spritu v x
        INC     A               ;+1
        LD      B,A
MULTDM: ADD     HL,HL           ;dm:=dm*2na(krok+1)
        DJNZ    MULTDM          ;vypocet nove souradnice
        ADD     HL,BC           ;sour+c   dm*krok+d2
        LD      A,C
        CALL    DIVIDS          ;HL:=HL/dy
        SRL     H
        RR      L               ;HL:=HL/2
        POP     BC              ;    dm*2na(krok)
        RET                     ;HL:=------------+Ø.5
                                ;          dy

;-----------------------------------------------------------------------------
; Ø48E          Pomocny vypocet pro vypocet kroku spritu
;-----------------------------------------------------------------------------

CHSTV:  LD      A,L             ;spodni byt posunuti
        OR      A               ;CY=Ø; vstup HL=-d
        INC     H               ;test na zaporne
        JR      Z,DLTNEG        ;je zaporne posunuti
        RES     Ø,B             ;flag ze SGN(d)=-
        DEC     H               ;oprava posunuti
        RET     Z               ;kdyz =256
        SCF                     ;neni mensi -neumim chyba
        RET

DLTNEG: NEG                     ;kladne
FNDMV:  CCF                     ;test na kudy prisel
        SET     Ø,B             ;flag ze SGN(d)=+
        RET

;=============================================================================
; Ø49F          Vypocet vektoru pro posuny spritu
;=============================================================================
; IN:   B - planovana pozice X
;       C - planovana pozice Y
;       HL- logicka pozice X
;       DE- logicka pozice Y

; OUT:  B - vektor X   (+-2)
```

```
;       C - vektor Y   (+-2)

GTSTEP: PUSH    AF
        PUSH    BC
        LD      C,B             ;v C je x
        XOR     A               ;a:=Ø
        LD      B,A             ;BC:=y
        SBC     HL,BC           ;skutecne-plan
        EX      DE,HL           ;DE:=dx posunuti
        POP     BC              ;cilova
        LD      B,A             ;b:=Ø  BC:=y
        OR      A               ;CY=Ø
        SBC     HL,BC           ;HL-dy
        CALL    CHSTV           ;test polarity -dy
        JR      C,ILGCM         ;abs dy>255 ->chyba
        LD      C,A             ;C:=abs dy
        EX      DE,HL           ;HL:=dx
        SLA     B               ;vysledek testu dy bit 1
        CALL    CHSTV           ;test polarity dx
        JR      C,ILGCM         ;abs dx>255 ->chyba
        LD      E,B             ;vysledek testu
        LD      B,A             ;B:=abs dx
        CP      C               ;test na dy=dx
        LD      A,(SPSTEP)      ;stand.krok spritu
        LD      D,A
        JR      C,GTD           ;dy>dx
        CP      B               ;A=dy?
        JR      NC,LTMIN        ;dx<min.krok
        LD      L,C             ;l:=dy
        LD      H,Ø             ;HL:=dy
        LD      C,B             ;C,B:=dx
        CALL    CALSTV          ;vypocet
        LD      C,L             ;c:=vysledek
        LD      B,D             ;krok v ose x'standard
        JR      CONTGT
GTD:    CP      C               ;test
        JR      NC,LTMIN        ;dy<minim.krok chyba
        LD      L,B
        LD      H,Ø             ;HL:=dx
        CALL    CALSTV          ;vypocet
        LD      B,L             ;b:=vysledek
        LD      C,D             ;C:=krok y standard
CONTG:  SET     2,E
LTMIN:  SRL     E               ;CY:=SGN dx
        JR      C,PLUSD         ;ke +    .
        LD      A,B             ;je -
        NEG
        LD      B,A             ;dx=-dx
PLUSD:  SRL     E               ;CY=SGN dY
        JR      C,DPLUS         ;d je +
        LD      A,C
        NEG
        LD      C,A             ;dy:=-dy
DPLUS:  POP     AF
        SRL     E
        CALL    NC,.CATRG       ;obe d <nez krok
        OR      A               ;CY:=Ø
        RET
;-------------------------------------------------------------------------------
;Nahrada chyboveho hlaseni .ILGCM
ILGCM:  POP     AF
```

```
        LD      A,89H           ;chybove hlaseni
        SCF                     ;CY:=1
        RET

;=============================================================================
;Tabulka bodovych velikosti spritu

SPRSIS: DB      ØØ8H,Ø1ØH,Ø1ØH,Ø2ØH
;=============================================================================

;=============================================================================
; Ø4FF          Nastaveni DISPLAY MODE 1
;=============================================================================
; IN:   B - informace o FUNC,SHIFT,CTRL

STDM1:  PUSH    HL
        LD      HL,DIFLGA
        SET     1,(HL)          ;CTRL se tiskne
        BIT     2,B             ;opravdu?
        JR      NZ,EXITST       ;ano->ven
        BIT     3,B             ;opravdu?  druhy SHIFT
        JR      NZ,EXITST       ;ano-> ven
        RES     1,(HL)          ;CTRL se vykona
EXITST: POP     HL
        RET

;=============================================================================
;-----------------------------------------------------------------------------
; Ø511          Obsluzna rutina klavesnice napojena na INT#CTC3
;=============================================================================

KEYSP:  LD      HL,LKYADR       ;adr.posl.schv.tl.
        LD      A,(HL)          ;do reg.A
        OR      A               ;bylo vubec nejake?
        JR      Z,NOKEY         ;nebylo ->
        DEC     A
        LD      C,A
        AND     7
        INC     A
        LD      B,A             ;B:=poradi bitu v portu
        LD      A,C
        RRCA
        RRCA
        RRCA
        AND     1FH
        ADD     A,3ØH
        LD      C,A             ;C:=cislo portu
        IN      A,(C)           ;test rady
TSTBIT: RRCA
        DJNZ    TSTBIT          ;test bitu B
        JR      C,KYPUSH        ;je stisknuto->
NOKEY:  CALL    SCNKB           ;prohledej klavesnici
        LD      (PHSKAD),A      ;adresa stiskleho tl.
        INC     (HL)
        DEC     (HL)            ;test na nulovost posl.schv.
        DEC     HL
        LD      (HL),B          ;LSKYST:=B
        INC     HL
        JR      Z,ONOKEY        ;posl.schv.=Ø ->
        CP      (HL)            ;posl.sch.=posl.stisk.?
```

```
        INC     HL
        JR      Z,LKYPHS        ;ano->
        OR      A               ;stiskle=Ø?
        JR      NZ,PSHNOZ       ;ne->
        DEC     (HL)            ;ano -zmensi citac omylu
        JR      NZ,CHCNOZ       ;jeste nenulovy
PSHNOZ: DEC     HL              ;jiz nulovy
ONOKEY: LD      (HL),A          ;LKYADR:=stis.tl.
        INC     HL
        LD      (HL),5          ;inic.citace omylu
        INC     HL
        LD      C,(HL)
        INC     HL
        LD      (HL),C          ;inic autorep.start
        INC     HL
        INC     HL
        LD      (HL),1          ;inic.citac.pro interv.
        JR      ISOKIN          ; ale pouze pro prvni

KYPUSH: IN      A,(3ØH)         ;test FUNC&spol
        AND     ØFH
        LD      B,A
        LD      (LSKYST),A      ;LSKYST:=func&spol
        INC     HL
LKYPHS  LD      (HL),5          ;citac omylu inic
CHCNOZ: INC     HL
        INC     HL
        DEC     (HL)            ;zmensi citac autor.
        RET     NZ              ;jeste <>Ø
        INC     (HL)            ;pro priste na +1
        INC     HL
        LD      C,(HL)          ;interval
        INC     HL
        DEC     (HL)            ;zmensi citac interv.
        RET     NZ              ;jeste nenulovy
        LD      (HL),C          ;inic interval
ISOKIN: LD      A,(LKYADR)      ;adresa tlacitka
        CALL    DECAD           ;vydekoduj
        RET     C
        PUSH    BC
        PUSH    HL
        PUSH    AF
        LD      A,(KINFLG)      ;flag kbd
        OR      A
        CALL    M,BELK          ;bit7=1 =>BELK
        POP     AF
        POP     HL
        POP     BC
        BIT     1,B             ;FUNC?
        JP      Z,PTKDT         ne-uloz znak
        RES     1,B             ;ano uloz func !neFUNC
        LD      C,Ø             ;prednast.citace
        LD      A,(FKMGFG)      ;flag FUNC
        ADD     A,A             ;test na typ
        JR      C,NOCNT         ;->bez citace
        LD      C,(HL)          ;vyber citac
        INC     HL              ;nova adresa
        LD      A,C             ;test na citac
        OR      A               ;=Ø -> ven
        RET     Z
```

```
NOCNT:  LD      A,(HL)
        OR      A               ;je znak =Ø?
        RET     Z               ;ano->ven
        PUSH    HL
        CALL    PTKDT           ;uloz nak
        POP     HL
        RET     C
        LD      A,(HL)          ;dalsi znak
        INC     HL
        CP      CR              ;je to CR?
        RET     Z               ;ano ->ven
        OR      A               ;je to ØØ?
        DEC     C               ;zmensi citac
        JR      NZ,NOCNT        ;do c=Ø
        RET

;=============================================================================
;-----------------------------------------------------------------------------
; Ø5A5          Obslouzeni JOY napojene na CTC#3
;=============================================================================

JOYSP:  LD      B,2             ;typ JOY 1=Hi Ø=1Ø
        LD      A,(EVMGFG)
        LD      C,A             ;C:=flag EVENTu
LOOPJO: PUSH    BC
        IN      A,(JOY)         ;test JOY
        BIT     Ø,B             ;je to spodni?
        JR      NZ,LOWER        ;ano
        RRCA
        RRCA
        RRCA
        RRCA
LOWER:  AND     ØFH             ;orezani
        LD      E,A             ;inform z JOY
        LD      D,82H           ;kod chyby 82 (asw2)
        AND     ØAH             ;test na bity 3,1
        CP      ØAH             ;je tam 1Ø1Ø?
        JR      Z,CHAJOY        ;ano
        DEC     D               ;DE=81 kod chyba (ASW1)
        LD      A,E
        AND     5
        CP      5               ;test na bity 2,Ø
        JR      NZ,OKJOY        ;ne Ø1Ø1
CHAJOY: LD      A,D
        LD      D,Ø             ;d=Ø
        JR      OKASW
OKJOY:  LD      D,Ø             ;D=Ø
        LD      HL,JOYDTB       ;adr.konv.tabulky
        ADD     HL,DE
        LD      A,(HL)          ;konverze dle tabulky
OKASW:  LD      HL,ASWNØ1       ;adr.sysvar
        LD      E,B
        DEC     E
        ADD     HL,DE
        ADD     HL,DE           ;vypocet adresy sysvar
```

```
        BIT     7,A             ;test na ASW
        RES     7,A             ;neni ASW
        JR      Z,ISJOY         ;nebylo ASW->je JOY
        CP      (HL)            ;testuj obsah nyni
        JR      Z,ISIS          ;uz tam je-preskoc
        LD      (HL),A          ;zapis novou
        LD      HL,(ASWPRC)     ;proces ASW
JUMPJO: BIT     4,C             ;je aktivovan EVENT?
        CALL    NZ,JMPHL        ;ano-vykonej
ISIS:   POP     BC              ;zpet citac     POP     BC
        DJNZ    LOOPJO          ;dokoncit
        RET
ISJOY:  LD      (HL),Ø          ;ASW=Ø
        INC     HL              ;adresa JOY
        CP      (HL)            ;uz je tam?
        JR      Z,ISIS          ;ano-preskoc
        LD      (HL),A          ;zapis novy
        LD      HL,(JOYPRC)     ;proces JOY
        JR      JUMJO           ;a prip.tam skoc

;=============================================================================
; Ø5FE          Rutina skoku na adresu HL , vyuzivana jako CALL JMPHL
;-----------------------------------------------------------------------------

JMPHL:  JP      (HL)

;=============================================================================
; Ø5FF          Zjisteni a skok na stiskle CTRL ci SHIFT RESET
;-----------------------------------------------------------------------------

RSTSP:  LD      HL,KINFLG
        LD      B,(HL)          ;flag kbd do B
        RES     5,(HL)          ;RESET=EI
        IN      A,(5ØH)         ;precti RESET
        OR      A
        RET     P               ;nedrzi->zpet
        SET     5,(HL)          ;RESET=DI
        IN      A,(3ØH)         ;test FUNC&spol
        LD      HL,(HLTPRC)     ;proces HALT
        BIT     Ø,A             ;je CTRL?
        JR      NZ,RSTGO        ;ano->
        AND     ØCH             ;byl alespon jeden sHIFT?
        RET     Z               ;nebyl->ven
        LD      HL,(RSTPRC)     ;proces RESET
RSTGO:  BIT     5,B             ;puvodni flag RESET?
        RET     NZ              ;byl DI ->ven
        EX      (SP),HL         ;fale3na navr.adresa
        CALL    STOPDC          ;zastavit DWCNT
        DI
        LD      HL,UEVMGF
        RES     Ø,(HL)          ;zastav uziv.EVEnty
        JR      ERSCU           ;zrus kurzor a skoc

;=============================================================================
; Ø629          Zabezpeceni blikani kurzorem
;-----------------------------------------------------------------------------

BLNKC:  LD      HL,(LNBUF)
        DEC     HL              ;citac akt.kurozru -1
        LD      (LNBUF),HL
```

```
        LD      A,H
        OR      L
        RET     NZ              ;jeste <>Ø ->ven
        RST     RST1            ;DIFLGA
        BIT     3,A             ;kurzor status
        JR      Z,DSPCU ;je-videt ->zviditelni

;-------------------------------------------------------------------------
; Ø638          Zruseni kurzoru z obrazovky
;-------------------------------------------------------------------------

ERSCU:  RST     RST1            ;DIFLGA
        BIT     4,A             ;je kursor ve VIEW
        RET     NZ              ;ne->ven
        CALL    RDSCHA          ;znak z pozice kurzoru
        CP      1BH             ;je to kurzor?
        RET     NZ              ;ne->nebudeme menit
        LD      A,(CCUCRA)      ;znak na pozici kurzoru
        RST     PBVRID          ;zapis znak
        RST     RST1            ;DIFLGA
        RES     3,(HL)          ;kurzor neni videt
        LD      A,(ERSECA)      ;cas pro stridani
        JR      INITIM          ; kdy neni inic

;-------------------------------------------------------------------------
; Ø64E          Zviditelneni kurzoru v obrazovce
;-------------------------------------------------------------------------

DSPCU:  RST     RST1            ;DIFLGA
        BIT     4,A             ;je-mimo VIEW?
        RET     NZ              ;ano->ven
        CALL    RDSCHA          ;znak na pozici kurzoru
        LD      (CCUCRA),A      ; uschovat
        LD      A,1BH           ;znak kurzoru
        RST     PBVRID          ; zapis
        RST     RST1            ;DIFLGA
        SET     3,(HL)          ;kurzor je videt
        LD      A,(DISPCA)      ;cas kdy je videt
INITIM: LD      H,A
        LD      L,Ø
        LD      (LNBUF),HL      ;inic.citace kurzoru
        RET

;=========================================================================
; Ø668          Vstup dat z klavesnice
;=========================================================================
; IN:   B - pocet znaku
;       HL- adresa buf v RAM
; OUT:  A - posledni znak
;       B - pocet znaku
;       HL- posledni adresa+1

EDTLN:  PUSH    BC
        PUSH    HL
        CALL    EDTST           ;EDIT&INPUT-opraveni obr.
        POP     HL
        POP     BC

;-------------------------------------------------------------------------
; Ø66F          Presun textu z VRAM do RAM
;-------------------------------------------------------------------------
```

```
; IN:   B - pocet znaku
;       DE- posice v obrazovce X,Y
;       HL- adresa bufferu
; OUT:  A - posledni znak
;       B - pocet znaku
;       HL- posledni adresa+1

ACEST:  CALL    RDSTM           ;precte B zn. zD,E do HL

;-------------------------------------------------------------------------
; Ø672          kurzor na zacatek dalsi radky
;-------------------------------------------------------------------------

MVSNXL: PUSH    AF
        PUSH    BC
        PUSH    HL
        CALL    SNTOS           ;kurzor na zacatek radky
        PUSH    AF
        CALL    MVACS
        POP     AF
        LD      A,CR
        CALL    C,DSPCHA
        CALL    STOPDC          ;zastav DWCNT
        POP     HL
        POP     BC
        POP     AF
        RET

;=========================================================================
; Ø689          Vstup dat z klavesnice
;=========================================================================
; OUT:  D - zacatek textu  X
;       E - zacatek textu  Y

EDTST:  CALL    STRTKT          ;start DWCNT
        LD      A,(SVSSW)       ;flag obrazovek
        PUSH    AF
        LD      DE,(CURSYA)     ;posice kurzoru
LOOPED: PUSH    DE
NEXTC:  CALL    ACECHI          ;vstup jednoho znaku
        JR      NC,NOERRK       ;nebyla-li chyba
        POP     DE
        POP     HL
        RET                     ;navrat pri chybe
;Testovaci rutina pro EDTST
NOERRK: CP      CR              ; ?CR
        JR      Z,ISCR          ;ano je->
        CP      ØCH             ;je CTRL L?
        JR      NZ,NOTFF        ;ne->
        LD      A,B
        AND     ØCH             ;test na SHIFTy
        JR      Z,NEXTC         ;sjou nulove->
        LD      A,ØCH           ;A:=CTRL L
NOTFF:  LD      HL,KINFLG
        RES     6,(HL)          ;rolovani ne
        CALL    DSPCHK          ;tisk dle KINFLG a nast.modu
        CP      3               ;je to CTRLC?
        JR      C,ISCTC         ;je >CTRLC->
        CP      7
        JR      C,ISROL         ;je <'G'->
```

```
ISCTC:  BIT     6,(HL)          ;flag ROLL?
        JR      Z,NEXTC         ;ne ->
        LD      A,5             ;CTRL E
ISROL:  POP     DE
        PUSH    DE              ;kurzor
        INC     E               ;y:=y+1
        JR      Z,NEXTC         ;je=0-> byl-1
        POP     DE              ;DE=kurzor
        LD      HL,TMOVCS       ;adresa tabulky pohybu kur.
        SUB     3               ;orezani na typ pohybu
        CALL    EXTBL           ;skok na prepocet
        CALL    CCUROT          ;umisteni kurzoru
        JR      NC,LOOPED       ;bez chyby->
        LD      E,-1            ;y=-1
        JR      LOOPED          ;dal
ISCR:   LD      DE,(CURSYA)     ;kurzor
        POP     HL              ;byvaly kurzor
        POP     BC
        LD      A,(SVSSW)       ;flag obrazovek
        XOR     B
        RRCA                    ;nejn.bit-psaci
        JR      C,ISSET         ;je=1 ->
        EX      DE,HL
        CALL    CMPCUR          ;novy=byvaly?
        EX      DE,HL
        JR      C,ISSET         ;novy<byvaly ->
        PUSH    HL
        CALL    SCTOS           ;na konec radky
        POP     HL
        CALL    CMPCUR          ;porovnej nyni
        CCF
        RET     NC              ;byvaly>novy ->
        EX      DE,HL           ;DE:=stara(byvala)
        OR      A
        RET                     ;zpet s CY:=0
ISSET:  CALL    SCTOS           ;presun na konec radky
        OR      A
        RET                     ;zpet a CY=0

;=============================================================================
; 06FF          Tabulka adres rutin pro prepocet pohybu kurzoru

TMOVCS: DW      ADJSTD
        DW      ADJSTL
        DW      ADJSTU
        DW      ADJSTR
;-----------------------------------------------------------------------------

;=============================================================================
; 0707          Definice kurzoru podle (DIFLGA)
;=============================================================================

CHGCP:  RST     RST1            ;DIFLGA
        LD      A,(KINFLG)      ;typ vstupu
        AND     3               ;v A
        LD      B,(HL)
        RR      B               ;CY=1 kdyz INSERT
        RL      A               ;a:=2*typ+INSERT
        ADD     A,A
        ADD     A,A
```

```
        ADD     A,A             ;8*A tj.offset predlohy
        LD      C,A
        LD      B,Ø             ;BC offset predlohy
        LD      A,1             ;znakovy generator
        BIT     5,(HL)          ;rezim displeje
        JR      Z,NOGII         ;neni GII
        LD      A,(CURSYA)      ;Y sour kursoru
        RRA
        RRA
        RRA                     ;vydeleno (tretiny)
        AND     3
        INC     A               ;generator 1,2,3
NOGII:  LD      HL,CURPTB       ;adresa predloh kurzoru
        ADD     HL,BC           ;+offset
        CALL    STOCUR          ;definice vzhledu
        ADD     A,3             ;budeme mit barvu
        LD      HL,CURCOL
STOCUR: LD      BC,Ø11BH        ;jeden od 1BH
        PUSH    DE
        PUSH    AF
        CALL    STCHR           ;definice
        POP     AF
        POP     DE
        OR      A               ;CY:=Ø
        RET

;=============================================================================
; Ø73F          Zmena rezimu vstupu klavesnice
;=============================================================================
; IN:   A - Ø1 Letter
;           2 Caps
;           3 Graphics
;       B - inform o FUNC&spol

CHKYM:  BIT     7,B             ;spec.CTRL key flag
        SCF                     ;CY:=1  chyba
        RET     Z               ;kdyz Ø tak zpet
        CP      4
        RET     NC              ;>3 ven - nebude se
        AND     3
        RET     Z               ;=Ø ven - nebude se
        DEC     A               ;konverze na tvar v DIFLGA
        LD      B,A
        LD      HL,KINFLG
        LD      A,(HL)
        AND     ØFCH
        OR      B               ;pridat nove urceni vstupu
        SET     2,A             ;byla zmena modu
        LD      (HL),A
        RET

;=============================================================================
; Ø756          Precteni klavesy z KBUF nebo cekani na stisk
;=============================================================================
; OUT:  A - ASCII kod znaku
;       B - Inform.o FUNC&spol je-li A<2ØH

WTKDTC: CALL    CHGCP           ;definice kurzoru
ZNOVU:  CALL    BLNKC           ;vypis/nevypis kurzoru
        PUSH    DE
```

```
        LD      HL,KIBFLG
        BIT     2,(HL)          ;zmena modu?
        RES     2,(HL)          ;ne!
        CALL    NZ,CHDCP        ;kdyz zmo -def.kurzoru
        CALL    GTKDT           ;precti znak z KBUF
        POP     DE
        RET     NC              ;zpet pri nechybe
        LD      A,(KVHGFG)
        ADD     A,A             ;test DHDHFG
        JR      NC,ZHOVU        ;nebyl konec znovu
        CALL    KRSCU           ;smaz kurzor
        CALL    STOPDC          ;zastav DMCHT
        CALL    .TIMOT          ;chyba preteceni

;-------------------------------------------------------------------------------
; 077B          Smazani obsahu buff klavesnice
;-------------------------------------------------------------------------------

CLKBF:  DI
        LD      A,(KDTGPT)      ;pridavani je
        LD      (KDTPPT),A      ;stejne s vybiranim
        EI                      ;tj.KBUF je prazdny
        RET

;===============================================================================
; 0784          Porovnani dvou souradnic
;-------------------------------------------------------------------------------
; IN:   HL - souradnice c.1
;       DE - souradnice c.2
; OUT:  CY=1 Z=0  s1<s2
;       CY=0 Z=1  s1=s2
;       CY=0 Z=0  s1>s2

CMPCUR: LD      A,L     ;minul
        CP      E       ; =minul?
        RET     NZ      ;nerovna->ven
        LD      A,H     ;vynul
        CP      D       ; =vynul?
        RET

;-------------------------------------------------------------------------------
; 078A          Podprogram smazani znaku po CTRL DEL
;-------------------------------------------------------------------------------

ACKDEL: INC     C               ;citac delky v buf
        DEC     C
        SCF                     ;zadne znaky
        RET     Z
        DEC     HL              ;zmensit adresu v buf
        LD      (HL),0          ;zapsat prazdno
        INC     B               ;pocet bytu buf
        PUSH    HL
        CALL    RDSCHA          ;znak z pozice kurzoru
        PUSH    AF
        PUSH    BC
        CALL    LFTAW           ;kurzor vlevo
        POP     BC
        POP     AF
        OR      A
        JR      Z,ZHDUL         ;znak=0
```

```
        LD      A,' '
ZNNUL:  LD      HL,(CURADA)     ;adresa kurzoru
        RST     PBVRID          ;zapis znaku
        POP     HL
        DEC     C               ;zmenseno
        RET

;=============================================================================
; Ø7A9          Vstup dat z klavesnice
;=============================================================================
; IN:   B - delka bufferu
;       HL- adresa bufferu
; OUT:  A - posledni znak
;       B - zbyly pocet znaku
;       C - skutecny pocet znaku
;       HL- posl znak+1

ACELN:  PUSH    DE
        LD      C,Ø
        CALL    STRTKT
LOOPAC: PUSH    BC
        PUSH    HL
NOCTRY: CALL    ACECHI          ;vstup znaku
        JR      NC,GOACE        ;bez chyby->
        POP     HL
        POP     BC
        POP     DE
        RET
;Podprogram ACELN pro upravu znaku
GOACE:  CP      CR              ;?CR
        JR      Z,JACR          ;ano->
        CP      ' '
        JR      NC,NOCTRL       ;>' ' ->
        CALL    STDM1           ;stav tisku CTRL
        JR      NZ,NOCTRL       ;byl shift
        LD      HL,TUKONC
        LD      BC,Ø
        CPIR                    ;najdi v tabulce
        JR      Z,NASEL
        CP      18H             ;?CTRLX
        JR      Z,ISCTRX
        CP      ØBH             ;?CTRLH
        JR      Z,ISCTRH
        CP      7               ;?CTRLG
        JR      Z,ISCTRG
        CP      19H             ;?CTRL Y
        JR      NZ,NOCTRY       ;ne-dalsi znak->
ISCTRG: CALL    DSPCH           ;vytsik CTRL Y,G
        JR      NOCTRY          ;dalsi znak
ISCTRH: POP     HL              ;buffer
        POP     BC              ;delka
        CALL    ACEDEL          ;smaz
        JR      LOOPAC
ISCTRX: POP     HL
        POP     BC
LOOPDL: CALL    ACEDEL          ;smaz znaky
        JR      NZ,LOOPDL       ;az do konce
        JR      LOOPAC          ;nove cteni

NOCTRL: CALL    DSPCH           ;vytiskni znak
```

```
        POP     HL              ;adresa
        POP     BC              ;delka a pocet
        LD      (HL),A          ;zapis znak
        INC     HL              ;nova adresa
        INC     C               ;pocet znaku
        DEC     B               ;pocet bytu v bufferu
        JR      NZ,LOOPAC       ;jeste <>0
        CALL    STOPDC          ;zastav DWCNT
        POP     DE
        CALL    .BUFFL          ;chyba pretekl buffer
JACR:   CALL    DSPCH           ;vytisk CR
NASEL:  CALL    STOPDC          ;zastav DWCNT
        LD      B,00H   ;< >
        CALL    STDM1           ;CTRL se vykonava
        POP     HL
        POP     BC
        LD      (HL),A          ;zapis posl.znaku
        INC     HL
        INC     C               ;+1 znak
        DEC     B               ;-1 pozice v buf
        OR      A
        LD      (TERMAL),A      ;zapis jako ukoncovac
        POP     DE
        RET

;Tabulka moznych ukoncovacu rutiny ACELN

TUKONC: DB      1EH             ;CTRL sipka nahoru
        DB      1FH             ;CTRL sipka dolu
        DB      1CH             ;CTRL sipka vlevo
        DB      1DH             ;CTRL sipka vpravo
        DB      1BH             ;CTRL [
        DB      0AH             ;CTRL J
        DB      17H             ;CTRL W
        DB      0BH             ;CTRL K

;=============================================================================
; 0827          Vstup jednoho znaku, z KBUF,nebo ceka s kurzorem
;=============================================================================
; IN:   A=0 znak nevytisknout
;         <>0 znak vytisknout
; OUT:  A - ASCII kod
;       B - prip.inform.o FUNC&spol

ACECH:  PUSH    DE
        PUSH    BC
        PUSH    HL
        LD      D,A             ;uschova A
        CALL    STRTKT          ;spust DWCNT
        CALL    ACECHI          ;vstup znak
        JR      C,VENAC         ;chyba->ven
        CALL    STOPDC          ;zastav DOWNCNT
        LD      A,D
        OR      A               ;je tisk vstoupleho?
        LD      A,E
        JR      Z,VENAC         ;ne->
        OR      A
        CALL    NZ,DSPCHK       ;vytiskne dle byv.CTRL
        POP     HL
        LD      D,B
```

```
        POP     BC
        LD      B,D             ;FCTN&spol
        POP     DE
        RET

;=============================================================================
; Ø845          Vstup znaku jako ACECH, ale nespousti DWCNT a netiskne
;=============================================================================
; OUT:  A - ASCII kod
;       B - prip.inform.o FUNC&spol

ACECHI: LD      A,(KINFLG)
        BIT     4,A             ;je zvuk?
        CALL    Z,CLKBF         ;ano - zazvuc
        LD      HL,EVMGFG
        SET     5,(HL)          ;KEYSW ON
        CALL    DSPCU           ;vypis kurzor
        CALL    WTKDTC          ;precti z KBUF
        RET     C               ;chyba ->
        LD      E,A
        CALL    ERSCU           ;smaz kurzor
        LD      A,E
        OR      A               ;CY:=Ø
        RET

;=============================================================================
; Ø860          Zaplneni useku pameti ØØ

NULPAD: XOR     A               ;kod je ØØ

;-----------------------------------------------------------------------------
; Ø861          Zaplneni useku pameti kodem
;-----------------------------------------------------------------------------
; IN:   HL - adresa odkud
;       BC - delka !!>1
;        A - pripadny kod

PAD:    LD      D,H
        LD      E,L             ;adresa ulozeni
        INC     DE              ;do DE
        LD      (HL),A          ;uloz kod
        DEC     BC              ;delka
        LDIR                    ;zaplneni presunem
        RET

;=============================================================================
; Ø869          Ulozeni dat do KBUF
;=============================================================================
; IN:   A - ASCII kod znaku
;       B - prip.inform.o FUNC&spol

PTKDT:  CALL    CHKYM           ;nastaveni rezimu
        RET     NC
        PUSH    BC
        PUSH    AF
        LD      A,(KBFSIZ)      ;delka KBUF
        LD      C,A
        LD      HL,(KBUFTA)     ;adresa KBUF
        LD      A,(KDTPPT)      ;citac zapisu
        LD      E,A
```

```
        LD      D,Ø             ;do DE
        ADD     HL,DE           ;adresa tl.v KBUF
        POP     AF
        LD      (HL),A          ;uloz znak
        CALL    UPPTP           ;otestovani pomeru
        JR      Z,ENDKBF        ;chyba ->konec KBUF
        LD      A,(HL)          ;vyber znak
        CP      ' '             ;? space
        JR      NC,ENDPTK       ;>' ' ->konec ukladani
        LD      HL,(KBUFTA)     ;adresa kBUF
        ADD     HL,DE
        LD      (HL),B          ;uloz informaci o FCTN&spol
        CALL    UPPTP           ;otestuj pomery
        SCF
        CCF                     ;CY:=Ø
        JR      NZ,ENDPTK       ;je OK ->uz ven
        DEC     E               ;zmensi citac
ENDKBF: DEC     E               ; nebude ulozeno
        LD      A,84H           ;kod chyby
        SCF                     ;k tomu CY=1
ENDPTK: PUSH    AF
        LD      A,C             ;delka KBUF
        AND     E               ;orez.na patr.delku
        LD      (KDTPPT),A      ;novy zapisovaci citac
        POP     AF
        POP     BC
        RET

;-------------------------------------------------------------------------
; Pomocny podprogram - test pomeru v KBUF

UPPTP:  INC     E               ;zvetseni citace znaku
        LD      A,C             ;delka KBUF
        AND     E               ;orezani s citacem
        LD      E,A             ;vysledek
        LD      A,(KDTGPT)      ;citac vystupu
        CP      E               ;porovnej
        RET

;=========================================================================
; Ø8AC          Vyber informaci z buf klavesnice
;=========================================================================
; OUT:  A - ASCII znak
;       B - prip.inform.o FUNC&spol

GTKDT:  LD      B,Ø
        LD      A,(KBFSIZ)      ;delka KBUF
        LD      C,A
        LD      HL,KDTGPT       ;citac cteni
        LD      A,(HL)
        DEC     HL
        CP      (HL)            ;porovnej s citacem vystupu
        CALL    Z,.DATOT        ;chyba cte se vstup!
        LD      E,A
        LD      D,Ø             ;citac cteni
        LD      HL,(KBUFTA)     ;adresa
        ADD     HL,DE           ;znaku
        LD      A,(HL)          ;znak
        OR      A               ;CY=Ø a test na Ø
        PUSH    AF
```

```
        INC     E               ;citac zvetsit
        CP      ' '             ;je to SPACE
        JR      NC,ISCHRN       ;je to norm.znak
        LD      A,C             ;delka
        AND     E               ;orezej
        LD      E,A             ;citac znaku
        LD      HL,(KBUFTA)     ;adresa
        ADD     HL,DE           ; znak
        LD      B,(HL)          ; FCTN&spol
        INC     E               ;zvetsi citac
ISCHRN: LD      A,C             ;rozmer KBUF
        AND     E               ;orezat
        LD      (KDTGPT),A      ;novy citac vystupu
        POP     AF
        RET

;=============================================================================
; 08DA          Vydekodovani tlacitka
;=============================================================================
; IN:   A - adresa tlacitka
;       B - inf.o FUNC&spol
; OUT:  A - ASCII
;       B - inf.o FUNC&spol
:       HL- bylo-li FUNC tak adr

DECAD:  CP      7               ;je-li
        RET     C               ;<7 -> ven
        LD      C,B             ;FCTN & spol
        SRL     C               ;bylo CTRL?
        JR      C,DECTR         ;ano->
        SRL     C               ;bylo FUNC?
        JR      C,DECFN         ;ano->
        LD      B,0             ;flag SHIFT
        JR      Z,NESHFT
        INC     B               ;shift
NESHFT: LD      C,A             ;adresa tl.
        LD      A,(KINFLG)      ;flag KBD
        AND     3               ;rezim vstupu
        ADD     A,A
        ADD     A,B             ;A:=A*2+B
        LD      E,A
        LD      A,C             ;adresa

;Vydekodovani Letter,Caps ci Graphics

LKUKT:  LD      D,0             ;DE:=(2*mod.vst.)+B
        LD      HL,(KBCTET)     ;konverzni tabulka kbd
        ADD     HL,DE
        ADD     HL,DE           ;HL:=HL+2*(2*modvst+B)
        LD      E,(HL)
        INC     HL
        LD      H,(HL)          ;HL:=adresa konkretni tbl
        LD      L,E
        SUB     7               ;zmenseni na 0 az
        LD      E,A
        LD      A,D             ;dE:=offset , A:=0
        RET     C               ;kdyz preteklo ->chyba
        ADD     HL,DE           ;adresa polosky
        LD      A,(HL)          ;vydek.znak
        OR      A
```

```
        RET     NZ              ;zpet pri nenulovem
        SCF
        RET                     ;znak=Ø a CY:=1 chyba

;Vydekodovani CTRL

DECTR:  LD      E,Ø
        CALL    LKUKT           ;vydek jako Letter
        RET     C               ;chyba -> ven
        CALL    GTCAPC          ;konverze na CAPS
        RES     1,B             ;B:=Ø
        LD      C,A             ;znak
        LD      HL,TBCTRN       ;tabulka
        LD      A,(HL)          ;znak
FINDC:  CP      C               ;je to on?
        INC     HL
        LD      A,(HL)          ;vyber ekvivalent
        INC     HL
        RET     Z               ;byl to on ->ven
        LD      A,(HL)          ;vyber znak
        OR      A
        JR      NZ,FINDC        ;nebyl=Ø pokracuj
        LD      A,C             ;znak do a
        AND     ØBFH            ;orezani na CTRL
        RET

; Tabulka prevodu znaku na CTRL

TBCTRN: DB      '@',1EH
        DB      ':',1Ch
        DB      ';',1DH
        DB      '/',1FH
        DB      '\',8
        DB      Ø

; Vydekodovani jako FUNC

DECFN:  LD      E,Ø
        CALL    LKUKT           ;vydek.jako Letter
        RET     C               ;chyba ->
        RES     1,B
        SET     7,B             ;B:=80h
        SUB     'Ø'             ;orez.na Ø-4Fh
        RET     C               ;chyba->
        CP      ØAH             ;je to 1Ø?
        CCF
        RET     NC              ;mensi 1Ø -ven
        SET     1,B
        RES     7,B             ;B:=2
        CP      31H             ;kdyz velka pismena
        RET     C               ;ven
        CP      4BH             ;kdyz vetsi nez mala
        CCF
        RET     C               ;taky ven
        SUB     31H             ;korekce na Ø-1FH
        LD      E,A
        LD      A,(FKMGFG)      ;pocet FCTN key
        AND     1FH             ;orezani
        SCF
        RET     Z               ;kdyz Ø ven
```

```
        DEC     A
        CP      E               ;test meze
        RET     C               ;vetsi -> ven
        LD      HL,(FKIFTA)     ;adresa tabulky
        ADD     HL,DE
        ADD     HL,DE           ;adresa adresy tlacitka
        LD      E,(HL)
        INC     HL
        LD      D,(HL)          ;adresa tlacitka
        EX      DE,HL           ; do HL
        RET

;=============================================================================
; Ø966          Prohledani klavesnice
;=============================================================================
; OUT:  A - adresa stiskleho
        B - FUNC&spol

SCNKB:  LD      BC,0636H        ;B:=pocet portu C:=adresa
LOOPSC: IN      A,(C)           ;test portu
        JR      NZ,STISTL       ;pri stisklem
        DEC     C               ;novy port
        DJNZ    LOOPSC          ;dalsi
        IN      A,(C)           ;test posl.portu
        AND     ØCØH            ;orezani na CR a Space
STISTL: PUSH    AF
        IN      A,(3ØH)         ;test FCTN&spol
        AND     ØFH             ;orezani
        LD      B,A             ;stav FCTN
        POP     AF

;=============================================================================
; Ø97B          Vypocet adresy tlacitka
;=============================================================================
; IN:   A - precteny port
;       C - cislo portu
; OUT:  A - adresa tlacitka

CALKAD: PUSH    BC
        LD      B,7
LOOPCL: ADD     A,A
        JR      C,ISBIT         ;test bitu v A
        DJNZ    LOOPCL
ISBIT   LD      A,C
        SUB     3ØH
        ADD     A,A
        ADD     A,A
        ADD     A,A
        ADD     A,B
        INC     A               ;a=((port-3Ø)*8)+bit)+1
        POP     BC
        CP      7
        RET     NC              ;kdyz >7 tak ven
        LD      A,Ø
        RET                     ;ven a chyba

;=============================================================================
; Ø992          Spusteni citace klavesnice
;-----------------------------------------------------------------------------

STRTKT: PUSH    HL
        LD      HL,EVMGFG
```

```
        RES     7,(HL)          ;konec DWCNT
        LD      HL,(KINWTM)     ;pocet period pro WAIT
        LD      A,H
        OR      L
        JR      Z,NESTRT        ;je-li nula tak nic
        LD      A,(KINWTM)      ;perioda pro WAIT
        CALL    STRTDC          ;start DWNCNT
NESTRT: POP     HL
        RET

;===============================================================================
; Konverzni tabulka pro JOYPAD - konverze smeru

JOYDTB: DB      Ø,3,1,2,7,Ø,8,Ø
        DB      5,4,Ø,Ø,6,Ø,Ø,Ø
;-------------------------------------------------------------------------------

Predlohy kurzoru vsech typu

CURPTB: DB      ØFFH            ;XXXXXXXX
        DB      ØBFH            ;X XXXXXX
        DB      ØBFH            ;X XXXXXX
        DB      ØBFH            ;X XXXXXX
        DB      ØBFH            ;X XXXXXX
        DB      ØBFH            ;X XXXXXX
        DB      83H             ;X     XX
        DB      ØFFH            ;XXXXXXXX

        DB      ØFFH            ;XXXXXXXX
        DB      ØCFH            ;XX  XXXX
        DB      ØEFH            ;XXX XXXX
        DB      ØEFH            ;XXX XXXX
        DB      ØEFH            ;XXX XXXX
        DB      ØEFH            ;XXX XXXX
        DB      ØEFH            ;XXX XXXX
        DB      ØFFH            ;XXXXXXXX

        DB      ØFFH            ;XXXXXXXX
        DB      ØC7H            ;XX   XXX
        DB      ØBBH            ;X XXX XX
        DB      ØBFH            ;X XXXXXX
        DB      ØBFH            ;X XXXXXX
        DB      ØBBH            ;X XXX XX
        DB      ØC7H            ;XX   XXX
        DB      ØFFH            ;XXXXXXXX

        DB      ØFFH            ;XXXXXXXX
        DB      ØFFH            ;XXXXXXXX
        DB      ØFFH            ;XXXXXXXX
        DB      ØC3H            ;XX    XX
        DB      ØBFH            ;X XXXXXX
        DB      ØBFH            ;X XXXXXX
        DB      ØC3H            ;XX    XX
        DB      ØFFH            ;XXXXXXXX

        DB      ØFFH            ;XXXXXXXX
        DB      ØC7H            ;XX   XXX
        DB      ØBBH            ;X XXX XX
        DB      ØBFH            ;X XXXXXX
        DB      ØA3H            ;X X   XX
```

```
        DB      ØBBH            ;X XXX XX
        DB      ØC3H            ;XX    XX
        DB      ØFFH            ;XXXXXXXX

        DB      ØFFH            ;XXXXXXXX
        DB      ØFFH            ;XXXXXXXX
        DB      ØC3H            ;XX    XX
        DB      ØBBH            ;X XXX XX
        DB      Ø83H            ;X     XX
        DB      ØFBH            ;XXXXX XX
        DB      Ø87H            ;X    XXX
        DB      ØFFH            ;XXXXXXXX
;=============================================================================

; Konverzni tabulka klavesnice

KTKBD:  DW      KAALPH          ;Letter
        DW      KAALPS          ;Letter+Shift
        DW      KAKANA          ;Caps
        DW      KAKANS          ;Caps+Shift
        DW      KAGRPH          ;Graphics
        DW      KAGRPS          ;Graphics+Shift
;=============================================================================

;tabulky konverzi jednotlivych rezimu

KAALPH: DB      ' ',CR,'12345678'
        DB      'qwertyuiasdfghj'
        DB      'kzxcvbnm,ØØ-',5EH
        DB      './_\op@[l;:]'

KAALPS: DB      ' ',CR,'!"#$%&'('
        DB      'QWERTYUIASDFGHJ'
        DB      'KZXCVBNM<)',Ø,'='
        DB      '~>?¦OP`{L+*}'

KAKANA: DB      ' ',CR,'12345678'
        DB      'QWERTYUIASDFGHJ'
        DB      'KZXCVBNM,ØØ-',5EH
        DB      './_\OP@[L;:]'

KAKANS: DB      ' ',CR,'!"#$%&'('
        DB      'qwertyuiasdfghj'
        DB      'kzxcvbnm<)',Ø,'='
        DB      '~>?¦op`{l+*}'

KAGRPH: DB      ' ',CR,'12345678'
        DB      8CH,86H,8EH,87H,9ØH,91H,92H,93H,8AH
        DB      8BH,89H,88H,94H,95H,98H,97H,8DH,85H
        DB      8FH,96H,9AH,9BH,99H,',ØØ-',5EH,'.'
        DB      81H,84H,'\',82H,83H,9CH,9EH,8ØH,';'
        DB      9DH,9FH

KAGRPS: DB      ' ',CR,'!"#$%&'('
        DB      ØECH,ØE8H,ØEEH,ØE7H,ØFØH,ØF1H,ØF2H
        DB      ØF3H,ØEAH,ØEBH,ØE9H,ØE8H,ØF4H,ØF5H
        DB      ØF6H,ØF7H,ØEDH,ØE5H,ØEFH,ØF6H,ØF9H
        DB      ØFAH,ØFBH,'<)',Ø,'=~>',ØE1H,E4H
        DB      '¦',ØE2H,ØE3H,ØFCH,ØFEH,ØEØH,'+'
        DB      ØFDH,ØFFH
```

```
;=============================================================================
; ØB1F          Nastaveni aktivni obrazovky do rezimu Graphic II
;=============================================================================

GMODE:  LD      A,(DIFLGP)
        AND     6ØH .
        SCF                     ;neni GII jiz v druhe?
        RET     NZ              ;ano ->chyba
        RST     RST1
        BIT     5,A             ;rezim dipleje
        SCF                     ;chyba
        RET     NZ              ;jiz je GII
        CALL    CLRSS           ;CLS+ERASE
        LD      HL,(CPATLP)     ;adresa predloh I skryte
        PUSH    HL
        LD      HL,18ØØH        ;delky generatoru
        LD      (CPATLP),HL     ;nova tab predloh I
        LD      (SPATLP),HL     ;nova tabulka predloh spritu
        EX      DE,HL           ;DE:=18ØØh
        POP     HL              ;byvale HL
        CALL    BLKMV2          ;presun generatoru znaku
        LD      HL,SVSSW
        SET     3,(HL)          ;uspor.je II
        CALL    PTINTD          ;inic.akcni SCR
        RST     RST1
        LD      (HL),2ØH        ;je GII
        LD      HL,2ØØØH
        LD      (CPATLA),HL     ;predlohy I
        LD      H,Ø
        LD      (CCOTLA),HL     ;tabulka barev od Ø
        LD      H,18H
        LD      (SPATLA),HL     ;sprity od 18ØØH
        CALL    VREGI           ;inic VDP

;=============================================================================
; ØB5E          Inicializace barev a predloh na standard GII
;=============================================================================

FNTRST: LD      BC,18ØØH        ;delka barv.tabulky
        CALL    ITG2M           ;zaplneni b.t. stand.barvou

;-----------------------------------------------------------------------------
; ØB64          Inicializace predloh pro GII
;-----------------------------------------------------------------------------

LCPATG: CALL    LCPAT           ;definice znaku

;-----------------------------------------------------------------------------
;Definice znaku GII do tretin 2,3 podle tretiny 1
;-----------------------------------------------------------------------------

LCPTGC: LD      HL,2ØØØH        ;adresa 1 trerin
        LD      DE,28ØØH        ;kam
        LD      BC,Ø8ØØH        ;delka generatoru
        PUSH    BC
        CALL    BLKMV           ;druha jako prvni
        POP     BC              ;jeste jednu tretinu

;=============================================================================
; ØB75          Presun bloku VRAM->VRAM
;=============================================================================
```

```
BLKMV:  RST     GBVRID          ;precti znak
        EX      DE,HL
        RST     PBVRID          ;zapis znak
        EX      DE,HL
        INC     DE
        INC     HL              ;nove adresy
        DEC     BC
        LD      A,B
        OR      C               ;uz BC=Ø?
        JR      NZ,BLKMV        ;jeste ne->
        RET

;=============================================================================
; ØB81          Presun bloku VRAM->VRAM o delce 2kB
;=============================================================================

BLKMV2: LD      BC,Ø8ØØH        ;delka 2kB
        JR      BLKMV           ;obyc.presun

;=============================================================================
;Definice znaku dle kompr.tvaru v ROM
;=============================================================================

LCPATB: LD      (7227H),DE
        LD      E,C
        XOR     A
        LD      D,A             ;DE:=C
        PUSH    HL
        LD      HL,(CPATLA)     ;adresa predloh
        EX      DE,HL
        ADD     HL,HL
        ADD     HL,HL
        ADD     HL,HL
        ADD     HL,DE           ;8*hl
        EX      DE,HL
        POP     HL
        LD      A,8
        LD      (LNBUF),A
LOOPBL: PUSH    DE
        PUSH    BC
        LD      A,(7227H)
        LD      C,A
        ADD     A,A
        ADD     A,A
        ADD     A,A
        SUB     C
        LD      C,A
        LD      B,Ø
        LD      DE,7231H
        PUSH    DE
        LDIR
        POP     DE
        POP     BC
        LD      C,8
        EX      (SP),HL
        EX      DE,HL
NOZERC: PUSH    BC
        PUSH    DE
        LD      BC,(LNBUF)
        LD      DE,7229H
```

```
        CALL    UPACP
        LD      A,C
        LD      (LNBUF),A
        LD      B,8
        EX      (SP),HL
        LD      A,(7228H)
        LD      C,A
LOOPLB: LD      A,(DE)
        BIT     6,C
        JR      Z,ISZER6
        SRL     A
ISZER6: BIT     7,C
        JR      Z,ISZER7
        CPL
ISZER7: RST     PBVRID
        INC     HL
        INC     DE
        DJNZ    LOOPLB
        EX      DE,HL
        POP     HL
        POP     BC
        DEC     B
        JR      Z,END00
        DEC     C
        JR      NZ,NOZERC
        POP     HL
        JR      LOOPBL
END00:  POP     HL
        RET

;-------------------------------------------------------------------------
;Vypocty pro LCPATB
;-------------------------------------------------------------------------

UPACP:  PUSH    BC
        PUSH    DE
        LD      B,8
        XOR     A
LOOPWR: LD      (DE),A
        INC     DE
        DJNZ    LOOPWR
        POP     DE
        POP     BC
LOOPDJ: PUSH    HL
        PUSH    BC
        PUSH    DE
        LD      B,7
LOOPDJ: SRL     (HL)
        EX      DE,HL
        RR      (HL)
        EX      DE,HL
        INC     HL
        INC     DE
        DJNZ    LOOPJD
        POP     DE
        POP     BC
```

```
        DEC     C
        JR      Z,ENDTYP
        POP     HL
        JR      TODJN
ENDTYP  POP     AF
        LD      C,8
TODJN:  DJNZ    LOOPDJ
        RET

;===============================================================================
; ØC14 Uprava adres sysvar pri zmene obrazovek
;===============================================================================

PTINTD: DI

PTINT:  RST     RST1            ;DIFLGA
        EX      DE,HL
        LD      HL,INICSCT      ;inic tabulka
        LD      BC,1CH          ;delka
        LDIR                    ;presun tabulky

; ØC1F ---------------------------------------------------------------------
PAGET:  LD      A,(SVSSW)
        PUSH    AF
        BIT     Ø,A             ;psaci?
        POP     BC
        LD      A,B             ;flag SCR
        RET     Z               ;kdyz psaci=Ø tak OK ->ven

; ØC28 ---------------------------------------------------------------------
PAGEM:  PUSH    AF
        LD      HL,CODTLA       ;scrtbl+1
        LD      B,4             ;vse je o Ø4ØØh dal
        LD      C,B
LOOPCO: INC     HL
        LD      A,C
        ADD     A,(HL)
        LD      (HL),A          ;konverze vyssich bytu
        INC     HL              ;adres tabulek VDP
        DJNZ    LOOPCO          ;4* tj CODTLA,CCOTLA,CPATLA,SATTLA
        LD      A,C
        LD      HL,7Ø9AH        ;predlohyI vyssi byt
        ADD     A,(HL)
        LD      (HL),A          ;toto je o Ø8ØØh dal
        LD      A,C
        LD      HL,7ØA9H        ;vyssi cast adresy kurzoru
        ADD     A,(HL)          ;taktez o Ø4ØØh dal
        LD      (HL),A
        POP     AF              ;a=svssw
        RET

;===============================================================================
; ØC44          Inicializace obrazovky do rezimu MULTICOLOR
;===============================================================================

MMODE:  RST     RST1            ;DIFLGA
        BIT     5,A             ;je GII?
        CALL    NZ,CHNGLC       ;ano ->zmen II na I
        LD      A,(SVSSW)
        BIT     3,A             ;usporadani?
```

```
        SCF
        RET     NZ              ;uspor II ->ven chyba
        CALL    CLRSS           ;CLS+ERASE
        CALL    PTINTD          ;inic scrtable
        RST     RST1
        LD      (HL),40H        ;je MCOLOR
        JR      VREGI           ;inic VDP

;=============================================================================
; ØC5C          Umisti diplej i proces obrazovku na stranku Ø
;=============================================================================

NRMSC:  XOR     A
        CALL    WRTSC           ;psaci je=Ø
        XOR     A               ;displej je Ø
        DB      Ø1              ;falesne LD BC,23EH

;=============================================================================
; ØC62          Provede vymenu diplej obrazovek = CTRL Y
;=============================================================================

RVDSPP: LD      A,2             ;priznak prevraceni

;=============================================================================
; ØC64          Umisteni displej obrazovku
;=============================================================================
; IN:   A - Ø stranka Ø
;           1 stranka 1
;           2 opacna stranka

DSPSC:  LD      B,A             ;cislo scr pro displej
        LD      A,(SVSSW)
        LD      C,A             ;flag obrazovek
        RRA
        RRA
        AND     Ø1H             ;zachovani DSPPGE
        CP      B               ;je jeko zadana?
        RET     Z               ;ano-> ven
STOSSW: LD      A,C             ;SVSSW
        XOR     6               ;negace-jsou to ty druhe
        LD      (SVSSW),A       ;obnovit
        JR      VREGI           ;inic.

;=============================================================================
; ØC77          Vyber displej obrazovky
;=============================================================================
; IN:   A - Ø do procesni
;           1 do skryte

FRMSC:  LD      B,A             ;cislo disp
        LD      A,(SVSSW)
        LD      C,A
        RRA
        AND     1               ;DSPTBL
        CP      B               ;je to taz?
        JR      NZ,STOSSW       ;ne-neni
        RET

;=============================================================================
; ØC83          Nastaveni barvy papiru obrazovky FCOL
;=============================================================================
```

```
; IN:   B - cislo barvy

STFCOL: LD      HL,BDCOLA       ;barva pozadi
        LD      A,(HL)          ;do A
        AND     ØFH             ;orezani spodku
        LD      C,A             ;do C
        LD      A,B             ;zadana barva
        RRCA
        RRCA
        RRCA
        RRCA                    ;nahoru
        OR      C               ;do BDCOLA
        LD      (HL),A
        RST     RST1            ;DIFLGA
        BIT     7,(HL)          ;je TEXT?
        RET     Z               ;neni
        JR      VREGI           ;je - obarvit hned

;=============================================================================
; ØC97          Obarveni barvy pozadi obrazovky  BCOL
;=============================================================================
; IN:   B - cislo barvy

STBCOL: LD      HL,BDCOLA       ;barvy FCOL a BCOL
        LD      A,(HL)
        AND     ØFØH            ;ponechat vrchni FCOl
        LD      C,A
        LD      A,B
        AND     ØFH             ;zadane spodni
        OR      C
        LD      (HL),A          ;zapsat novou fcol+bcol

;=============================================================================
; ØCA3          Inicializace registru VDP dle obsahu sysprom proc.obr.
;=============================================================================

VREGI:  LD      A,(SVSSW)
        CALL    GTVDMD          ;vypocty regØ,1
        LD      (LNBUF),DE      ;uschova pro potom
        LD      DE,LNBUF+2
        INC     HL
        INC     HL              ;CODTLA(p)+1  vyssi
        LD      A,(HL)          ;vyssi byt tab predloh II
        RRCA
        RRCA
        AND     ØFH             ;konverze do reg.
        LD      (DE),A          ;(LNBUF+2):=A
        INC     DE              ;LNBUF+3
        PUSH    DE
        INC     HL
        LD      E,(HL)
        INC     HL
        LD      D,(HL)          ;DE:=CCOTLA(p)
        EX      DE,HL           ;do HL
        ADD     HL,HL
        ADD     HL,HL           ;o2 bity nahoru
        LD      A,H             ;pouze vrchni
        POP     HL
        BIT     5,B             ;je gII?
        JR      Z,NOGII         ;ne->
```

```
        OR      7FH             ;Ø111 1111
NOGII:  LD      (HL),A          ;do LNBUF+4
        INC     HL
        INC     DE              ;CPATLA+1
        INC     DE
        LD      A,(DE)          ;vyssi byt
        RRCA
        RRCA
        RRCA                    ;dolu
        AND     Ø7H             ;orezat
        BIT     5,B             ;je GII?
        JR      Z,GIINO         ;ne->
        OR      3               ;dodat
GIINO:  LD      (HL),A          ;do LNBUF+5
        INC     HL              ;LNBUF+5
        INC     DE              ;SATTLA
        LD      A,(DE)
        RLA
        INC     DE
        LD      A,(DE)          ;SATTLA+1
        RLA
        AND     7FH             ;orezani pro VDP
        LD      (HL),A          ;LNBUF+5
        INC     HL              ;LNBUF+6
        INC     DE
        INC     DE              ;SPATTLA+1
        LD      A,(DE)
        RRCA
        RRCA
        RRCA
        AND     Ø7H
        LD      (HL),A          ;orezani a LNBUF+6
        INC     HL              ;LNBUF+7
        INC     DE              ;BDCOLA
        LD      A,(DE)
        LD      (HL),A          ;LNBUF+7 -hezky za sebou
        LD      HL,LNBUF+7
        LD      BC,Ø811H        ;8bytu a port 11h
        DI
LOOPOU: OUTD                    ;vyslat byt
        LD      A,B             ;cislo reg
        SET     7,A             ;a vyslat
        OUT     (C),A           ;jako povelovy mod
        JR      NZ,LOOPOU       ; a do B=Ø
        EI
        RET

;=============================================================================
; ØDØ4          Inicializace proc.obrazovky do rezimu TEXT
;=============================================================================

TMODE:  RST     RST1
        OR      A
        RET     M               ;TEXT jiz je->ven
        BIT     5,A             ;je GII
        CALL    NZ,CHNGLC       ;ano-uspor.II->I
        CALL    PTINTD          ;inic SCRTBL
        PUSH    AF
        RST     RST1            ;DIFLGA
        LD      (HL),8ØH        ;je TEXT
```

```
        LD      A,28H
        LD      (WIDTVA),A      ;VIEW Ø,Ø,39,23
        LD      (WIDTDA),A      ;rozmer je 4Øzn
        POP     AF
        BIT     3,A             ;jake je usporadani
        JR      Z,LAYI          ;je I ->
        LD      HL,18ØØH
        LD      (CPATLA),HL     ;inic.predlohy kdyz II
LAYI:   LD      HL,(CPATLA)
        LD      (SPATLA),HL     ;sprity na znaky
        CALL    CLRSC           ;CLS

;-------------------------------------------------------------------------
; ØD2F          Inicializace videoprocesoru+definice znaku

IVDPCH: CALL    VREGI           ;inic VDP
        JR      LCPAT           ;definice znaku

;-------------------------------------------------------------------------
; ØD34          Pomocny podprogram vypoctu inic.hodnot regØ,1 VDp
;-------------------------------------------------------------------------

GTVDMD: BIT     1,A             ;?obr.displeje
        RST     RST1            ;DIFLGA
        JR      Z,PROCES        ; je procesni->
        LD      L,ØB8H          ;skryta  HL:=DIFLGP
PROCES: PUSH    HL
        LD      E,1BH
        XOR     A
        LD      D,A             ;DE:=1bH
        ADD     HL,DE           ;HL=GRFL.
        LD      C,(HL)          ;C:=GRFL.
        POP     HL
        LD      B,(HL)          ;B:=DIFLG.
        BIT     5,B             ;je GII
        JR      Z,GIINOU        ;ne->
        SET     1,A             ;ØØØØ ØØ1Ø
GIINOU: LD      E,A             ;E je Ø nebo "
        INC     E               ;bud 1 nebo 3
        LD      A,38H           ;ØØ11 1ØØØ
        SLA     C               ;CY:=velikost sprite
        RLA                     ;Ø111 ØØØs
        SLA     C               ;CY:=magnify
        RLA                     ;1110 ØØsm
        LD      C,A             ;C:=1110 ØØsm
        LD      A,B             ;a:=DIFLG.
        RRCA
        RRCA
        RRCA                    ;765 na 432
        AND     18H             ;a:=ØØØM MØØØ
        OR      C               ;111M MØsm
        LD      D,A
        RET

;=========================================================================
; ØD5E          Zmena usporadani II na uspor.I a definice znaku
;=========================================================================

CHNGLC: LD      A,(SVSSW)
        RRCA                    ;CY:=psaci stranka
```

```
        LD      HL,2800H        ;predlohy Ø
        JR      C,PAGØ          ;stranka0->
        LD      HL,3000H        ;predlohy1
PAGØ:   PUSH    HL
        LD      (70BDH),HL      ;tabulka predloh I
        LD      HL,2000H
        LD      (70C1H),HL      ;tabulka predloh spritu
        LD      HL,1800H
        POP     DE
        CALL    BLKMV2          ;definice znaku dle spritu
        LD      HL,1800H
        LD      DE,2000H
        CALL    BLKMV2          ;definice novych spritu jako znaku
        LD      HL,SVSSW
        RES     3,(HL)          ;uspor.I

;=============================================================================
; ØD89          Definice znaku dle systemove normy
;=============================================================================

LCPAT:  LD      BC,041CH        ;4znaky od 1ch
        LD      HL,GPATAR       ;predlohy sipek
        CALL    STCCHR          ;definice
        LD      DE,4005H
        LD      BC,6020H
        CALL    LCPATB
        LD      BC,2080H        ;20znkau od 80h
        CALL    STCCHR          ;definice
        LD      DE,0007H
        LD      BC,40A0H
        CALL    LCPATB
        LD      BC,20E0H        ;20 zn od e0h
        CALL    STCCHR          ;definice
        LD      HL,(CPATLA)
        LD      BC,RST1
        XOR     A
        CALL    PADVRM
        LD      HL,(CPATLA)
        PUSH    HL
        LD      BC,RST1
        ADD     HL,BC
        EX      DE,HL
        POP     HL
        LD      BC,0208H
        ADD     HL,BC
        LD      BC,00D0H

;=============================================================================
; ØDCB          Presun bloku VRAM->VRAM inverzne
;=============================================================================
; IN:   BC- delka bloku
;       DE- adresa cile
;       HL- adresa zdroje

BLKMVC: RST     GBVRID          ;byt
        EX      DE,HL
        CPL                     ;negace bytu
        RST     PBVRID          ;zapsat
        EX      DE,HL
        INC     HL
```

```
        INC     DE              ;nove adresy
        DEC     BC
        LD      A,B
        OR      C               ;test na BC=Ø
        JR      NZ,BLKMVC       ;nerovno ->
        RET

;=============================================================================
; ØDD8          Inicializace proc.obrazovky do rezimu Graphics I
;=============================================================================

CMODE:  RST     RST1
        AND     ØEØH            ;maska na rezim
        SCF
        RET     Z               ;chyba GI uz je
        BIT     5,A             ;je GII?
        CALL    NZ,CHNGLC       ;ano - zmena uspor.
        CALL    CLRSS           ;CLS + ERASE
        CALL    PTINTD          ;druha obrazovka je v GII?
        BIT     3,A
        JR      Z,TWOGNO        ;neni->
        LD      HL,18ØØH        ;je-predlohy
        LD      (SPATLA),HL     ;spritu
        LD      (CPATLA),HL     ;i znaku od 18ØØH
TWOGNO: CALL    IVDPCH          ;inic vdp i chr

;=============================================================================
; ØDF8          Inic.tabulky barev na normu v rezimu GII
;-----------------------------------------------------------------------------

STCTBL: LD      BC,2ØH          ;delka tabulky barev GI

;-----------------------------------------------------------------------------
; ØDFB          Inic tabulky barev na standard
;-----------------------------------------------------------------------------
; IN:   BC- pocet bytu tabulky barev

ITG2M:  LD      A,(BDCOLA)      ;impl.barva
        LD      HL,(CCOTLA)     ;tabulka barev

;=============================================================================
; ØE01          Zaplneni useku VRAM hodnotou
;=============================================================================
; IN:   A - hodnota
;       HL- adresa odkud
;       BC- pocet bytu

PADVRM: LD      D,A
        RST     PBVRID          ;zapis bytu
        INC     HL              ;nova adresa
        DEC     BC
        LD      A,B
        OR      C               ;citac uz =Ø?
        LD      A,D             ;obnova bytu
        JR      NZ,PADVRM       ;jest <>Ø
        RET

;=============================================================================
; ØE0B          Inicializace systemu zobrazeni
;=============================================================================
VDPINT: LD      A,40H           ;pasivni i aktivni
```

```
        LD      (DIFLGA),A      ;na falesny MCOLOR
        LD      (DIFLGP),A
        CALL    SCRDEF          ;definice
        LD      HL,(70BDH)      ;znaky
        LD      DE,(70C1H)      ;sprity
        LD      BC,0800H        ;delka
SCRDEF: CALL    BLKMV           ;definice dle SCR1
        CALL    CMODE           ;gI
        CALL    CLRSC           ;CLS
        CALL    ERSPRA          ;ERASE
        JR      RVWRTP          ;zamena obrazovek

;=============================================================================
; 0E2E          Provede zamenu diplej i proces.obrazovky    CTRLV
;=============================================================================

REVSC:  CALL    RVDSPP          ;zamena displej

;-----------------------------------------------------------------------------
; 0E31          Provede zamenu procesni obrazovky      CTRLZ
;-----------------------------------------------------------------------------

RVWRTP: LD      A,02H           ;kod zameny

;-----------------------------------------------------------------------------
; 0E33          Urceni procesni obrazovky
;-----------------------------------------------------------------------------
; IN:   A - 0 stranka 0
;           1 stranka 1
;           2 opacna stranka

WRTSC:  LD      B,A             ;uschova
        LD      A,(SVSSW)
        LD      C,A
        AND     1               ;cislo procesni obrazovky
        CP      B               ;je uz to ona?
        LD      A,C
        RET     Z               ;ano -> ven
        XOR     3               ;zamena proces a zobr.
        LD      (SVSSW),A
        RST     RST1            ;HL:=DIFLGA
        LD      DE,DIFLGP       ;DE:=DIFLGP
        LD      BC,24h          ;delka SCRTBL

;=============================================================================
; 0E49          Zamena obsahu dvou oblasti pameti RAM
;-----------------------------------------------------------------------------
; IN:   HL- zacatek bloku1
;       DE- zacatek bloku2
;       BC- delka bloku

        DI
EXCLOP: LD      A,(DE)          ;byt
        LDI                     ;dat tam jiny
        DEC     HL
        LD      (HL),A          ;puvodni dat na presunuty
        INC     HL
        JP      PE,EXCLOP       ;dle BC skoc
        EI
        RET
```

```
;=============================================================================
; ØE55          Definice do setu Ø1
;=============================================================================
; IN:   B - pocet znaku
;       C - pocatecni znak
;       HL- adresa predloh v RAM

;       A - pro dalsi rutiny  set kam

STCCHR: LD      A,Ø1H
        DB      16H             ;falesne LD D,ØAFH

;=============================================================================
; ØE58          Definice predlohy spritu
;=============================================================================

STSCHR: XOR     A               ;A:=Ø   nebo D:=ØAFH

;=============================================================================
; ØE59          Definice predlohy do vsech setu
;=============================================================================

STCHR:  PUSH    HL
        CALL    GFCCP           ;vypocet dat definice
        EX      DE,HL
        POP     HL
        JR      C,ILGCM         ;bylo chybne cislo->

;=============================================================================
; ØE61          Presun bloku dat RAM -> VRAM
;-----------------------------------------------------------------------------
; IN:   BC- pocet bytu
;       DE- adresa VRAM
;       HL- adresa RAM

CVTIR:  DI
        EX      DE,HL           ;HL=kam do VRAM
        PUSH    BC
        CALL    STVWAD          ;priprav VDP pro zapis
        POP     BC
        EX      DE,HL           ;Hl je odkud
DALSTO: LD      A,(HL)          ;byt
        OUT     (VDPD),A        ;zapis do VDP
        INC     DE
        INC     HL              ;nove adresy
        DEC     BC
        LD      A,B
        OR      C               ;citac=Ø?
        JR      NZ,DALSTO       ;jeste ne->
        EI
        RET

;=============================================================================
; ØE75          Precteni predlohy
;=============================================================================
; IN:   A - set odkud
;       B - pocet znaku
;       C - pocatecni znak
;       HL- adresa kam ulozit RAM

RDCHR:  PUSH    HL
```

```
        CALL    GFCCP           ;parametry predlohy
        POP     DE
ILGCM:  CALL    C,.ILGCM        ;pri chybe

;=============================================================================
; ØE7D          Presun pameti VRAM -> RAM
;-----------------------------------------------------------------------------
; IN:   BC- pocet bytu
;       DE- adresa RAM
;       HL- adresa VRAM

VCTIR:  DI
        LD      A,L
        OUT     (VDPC),A        ;zapis
        LD      A,H
        OUT     (VDPC),A        ;adresy
        INC     HL
        DEC     BC
        LD      A,B
        OR      C               ;test na nulovost
        IN      A,(VDPD)        ;data z VDP
        EI
        LD      (DE),A          ;uloz byt
        INC     DE
        JR      NZ,VCTIR        ;jeste BC<>Ø
        RET

;-----------------------------------------------------------------------------
; ØE9Ø          Podprogram zisku adresy a delky predloh
;-----------------------------------------------------------------------------
; IN:   A - set odkud
;       B - pocet znaku
;       C - pocatecni znak
; OUT:  HL- adresa pocatku predloh
;       BC- nutna delka predloh

GFCCP:  CP      7               ;test cisla setu
        CCF
        RET     C               ;chyba-moc velke
        LD      D,A
        RST     RST1
        BIT     5,A             ;test na GII
        JR      NZ,GIIANO       ;je GII->
        LD      A,D
        CP      2               ;ted jenom Ø,1 set
        CCF
        RET     C               ;moce velky-> ven
GIIANO: CALL    GENER           ;HL:=adresa generatoru
        EX      DE,HL
        LD      A,C             ;kod prvni predlohy
        LD      L,B
        LD      H,Ø             ;pocet predloh
        INC     B
        DEC     B               ;test na nulovost
        JR      NZ,NO256        ;je Ø=256
        INC     H               ;256
NO256:  ADD     HL,HL
        ADD     HL,HL
        ADD     HL,HL           ;pocet zn*7=pocet bytu
        LD      B,H
```

```
        LD      C,L             ;do BC
        LD      L,A
        LD      H,Ø             ;kod predlohy
        ADD     HL,HL
        ADD     HL,HL
        ADD     HL,HL           ;*8
        ADD     HL,DE           ;HL:=adresa predlohy
        RET

GENER:  LD      A,D             ;cislo generatoru
        LD      HL,(SPATLA)     ;predlohy spritu
        OR      A
        RET     Z               ;je spritu - HL je OK
        DEC     A
        LD      HL,(CPATLA)     ;predlohy znaku
        CP      3
        JR      C,LT3           ;je to jeste porad znak->
        INC     A               ;je barvovy.
        LD      HL,(CCOTLA)     ;predlohy barev
LTØ3:   AND     3
        ADD     A,A
        ADD     A,A
        ADD     A,A
        ADD     A,H             ;HL:=HL+8ØØ*a
        LD      H,A
        RET                     ;HL:=poc.adresa generatoru

;=========================================================================
; ØED3          Urceni barvy skupiny osmi znaku v GI
;=========================================================================
; IN:   B - cislo skupiny
;       C - barva skupiny

STICOL: RST     RST1            ;DIFLGA
        AND     ØE0H
        CALL    NZ,.ILGDM       ;neni GI -> chyba
        LD      A,B             ;cislo skupiny
        AND     1FH             ;orezat maskou
        LD      E,A
        LD      D,Ø             ;DE:=cislo skupiny
        LD      HL,(CCOTLA)     ;adresa tabulky
        ADD     HL,DE           ;+offset
        OR      A
        BIT     7,B             ;test na druh operace
        JP      NZ,GBVRID       ;cteni a konec
        LD      A,C
        RST     PBVRID          ;zapis barvy
        RET

;=========================================================================
; ØEEC          Vstup znaku z obrazovky
;=========================================================================
```

```
; IN:   B - pocet znaku
;       D - pocatecni pozice X
;       E - pocatecni pozice Y
;       HL- adresa bufferu RAM

; OUT:  A - posledni znak
;       B - zbyly neprecteny pocet
;       HL- posl.znak+1

RDSTM:  CALL    ACSCH           ;znak z displeje
        LD      (HL),A          ;do bufferu
        INC     HL
        DEC     B               ;dalsi
        CP      CR              ;je to CR?
        RET     Z               ;ano->ven
        LD      A,D
        OR      E
        LD      A,C             ;zpet znak
        RET     Z               ;x,y=Ø -> ven
        INC     B
        DEC     B               ;test na nulovost
        JR      NZ,RDSTM        nenulove tak dal
        CALL    .BUFFL          ;chyba -buffer full bez CR

;=========================================================================
; ØFØØ          Precteni znaku z obrazovky a posun kurzoru
;=========================================================================
; IN:   DE- pozice X,Y
; OUT:  A - znak z pozice

ACSCH:  CALL    CCUROT          ;test spravnosti kurzoru
        JR      C,EXVIEW        ;kurzor utek
        PUSH    BC              ;pocet
        PUSH    HL              ;adresa
        PUSH    DE              ;pozice
        CALL    RDSCH           ;znak z D,E
        POP     DE
        POP     HL
        POP     BC
        OR      A               ;test znaku
        JR      NZ,VALID        ;platny znak
EXVIEW: LD      A,CR            ;nahrada neplatnych
VALID:  LD      C,A             ;znak
        CALL    STPCUR          ;kurzor na dalsi znak
        LD      A,C             ;znak zpet do A
        RET

;=========================================================================
; ØF19          Zruseni resimu INSERT
;=========================================================================

STOVRM: RST     RST1            ;DIFLGA
        RES     Ø,(HL)          ;zrus INSERT
        JR      STOKNF          ;zapis zmenu rezimu

;-------------------------------------------------------------------------
; ØF1E          Nastaveni rezimu INSERT
;-------------------------------------------------------------------------

STINSM: RST     RST1            ;DIFLGA
```

```
        SET     Ø,(HL)          ;nastav INSERT
STOKNF: LD      HL,KINFLG
        SET     2,(HL)          ;zapis zmeny rezimu
        RET

;=============================================================================
; ØF27          doplni radek s kurzorem je-li treba mezerami
;-----------------------------------------------------------------------------

FILSP:  LD      A,(CURSXA)      ;x-souradnice
        OR      A
        RET     Z               ;je Ø a ven
        LD      B,A             ;B=x = kolik chybi do Ø
        LD      HL,(CURADA)     ;adresa kurzoru
LOOPPS: DEC     HL              ;o jeden zpet
        RST     GBVRID          ;vyber znak
        OR      A               ; je to Ø
        RET     NZ              ;neni-konec->
        LD      A,' '
        RST     PBVRID          ;zapis mezeru
        DJNZ    LOOPPS          ;az do zacatku radku
        RET

;=============================================================================
; ØF3A          Vsune jeden znak a zbytek posune
;-----------------------------------------------------------------------------
; IN:   A - vsouvany znak
;       D - pozice X
;       E - pozice Y
; OUT:  D - pozice konce radku X
;       E - pozice konce radku Y

SIFTR:  LD      (LNBUF),A       ;vsouvany kod na prvni pozici
        LD      (CURPSV),DE     ;uschova DE
        CALL    MVACS           ;kursor
SIFTCR: LD      HL,(CURADA)     ;adresa kurzoru
        CALL    CLCSFR          ;C:=delka do konce radku
        JR      Z,ENDLIN        ;je=Ø=je na konci radku
        EX      DE,HL           ;adr.kur v DE
        LD      HL,LNBUF+1      ;adresa kam ukladat
LOPGET: CALL    GETVRM          ;cti znak a posun kurzor
        JR      Z,CHARØØ        ;byl to znak Ø->
        DJNZ    LOPGET          ;docti cely text
ENDLIN: IN      A,(VDPD)        cteni pos.znaku
        LD      D,A             ;uschova kodu
        INC     B               :b:=1
CHARØØ: EI
        DEC     B               ;odpocti posledni znak
        LD      A,C             ;a:=pocet zn.do konce radky
        SUB     B               ;a:=pocet nenactenych znaku
        LD      B,A
        LD      HL,(CURADA)     ;adresa kurzoru
        PUSH    DE
        CALL    WDVLB           ;tisk celeho bufferu
        POP     DE
        DEC     HL              ;Hl:=adresa posl tisk.znaku
        XOR     A
        OR      (HL)
        LD      A,D             ;kod znaku co nebyl v LNBUF
        LD      (LNBUF),A       ;uloz na prvni pozici
```

```
        JR      Z,LSTCHR        ;byl-li nulovy->
        CALL    CCURML          ;test kur na y=23
        JR      C,LSTCHR        ;dal uz neni text
        CALL    CRETL           ;kurzor na dalsi radku
        LD      A,(LNBUF)       ;neprecteny znak
        OR      A
        JR      NZ,SIFTCR       ;neni nula-vsun znak(y)
        LD      A,(CURSYA)      ;je to ØØ - vsun radku
        LD      E,A
        CALL    SIFTD           ;vsun jednu radku od kursoru
        XOR     A
        LD      B,A             ;B,A:=Ø
LSTCHR: PUSH    AF
        LD      A,(CURSXA)      ;a:=x
        ADD     A,B             ;A:=x+prip.delka textu
        LD      HL,WIDTVA       ;test na VIEW
        CP      (HL)
        JR      NZ,NOEXVW       ;neni mimo
        LD      HL,CURSYA       ;je presne na kraji
        INC     (HL)            ;o radek dolu =LF
        XOR     A               ;a odradkovani=CR
NOEXVW: LD      D,A             ;D=Ø nebo konec textu
        LD      A,(CURSYA)      ;A:=y
        LD      E,A             ;D,E=x,y
        PUSH    DE
        LD      DE,(CURPSV)     ;ven
        CALL    MVACS           ;kurzor na zacatek
        POP     DE              ;konecna pozice
        POP     AF
        LD      A,(LNBUF)       ;posledni znak
        RET

;=============================================================================
; ØFAB          Kurzor vlevo a DEL
;-----------------------------------------------------------------------------

DELTC:  RST     RST1            ;DIFLGA
        BIT     4,A             ;kurzor ve VIEW
        RET     NZ              ;ne->ven
        LD      A,(CURSXA)      ;pozice x
        OR      A               ;=Ø?
        JR      NZ,SXØN         ;neni->
        LD      A,(CURSYA)      ;pozice Y
        OR      A               ;je nula?
        JR      Z,SYØ           ;ano->
        DEC     A               ;Y:=Y-1
        LD      E,A             ;E:=y
        LD      A,(WIDTVA)      ;pravy rozmer VIEW
        DEC     A               ;zmensit =x
        LD      D,A             ;DE=x,y
        CALL    RDSCH           ;precti znak
        OR      A
        JR      Z,SYØ
SXØN:   CALL    RDSCHA          ;znak z kurzoru
        OR      A               ;test na nulu
        CALL    Z,LFTAW         ;ano-> kurzor -->
SYØ:    LD      DE,(CURSYA)     ;DE:=x,y

;=============================================================================
; ØFD3          Smaze jeden znak a zbytek posune
;-----------------------------------------------------------------------------
```

```
;       E - pozice znaku Y

SIFTL:  LD      (CURPSV),DE     ;uschova
        CALL    MVACS           ;umisti kurzor
LOPING: LD      HL,(CURADA)     ;adresa kurzoru
        INC     HL              ;+1
        CALL    CLSSFR
        EX      DE,HL
        LD      HL,LNBUF
        JR      Z,NOCHAR
TOEND:  CALL    GETVRM
        JR      Z,ISØØIS
        DJNZ    TOEND
NOCHAR: PUSH    BC
        CALL    CCURML
        LD      A,Ø
        JR      C,NOTTXT
        LD      D,A
        LD      A,(CURSYA)
        LD      E,A
        INC     E
        PUSH    HL
        CALL    RDSCH
        POP     HL
NOTTXT: LD      (HL),A
        POP     BC
ISØØIS: EI
        LD      A,C
        SUB     B
        LD      B,A
        LD      HL,(CURADA)
        CALL    WDVLB
        DEC     HL
        XOR     A
        OR      (HL)
        JR      Z,EXITDC
        CALL    CRETL
        JR      LOPING
EXITDX: LD      DE,(CURPSV)
        JP      MVACS

;Podprogram  inicializace VDP pro READ a vypocet delky
;  od kursoru do konce radku

CLCSFR: DI
        CALL    STVRAD          ;priprava VDP pro operaci
        LD      A,(CURSXA)      ;pozice x
        LD      B,A             ;do B
        LD      A,(WIDTVA)      ;pozice VIEW
        SUB     B               ;test je-li vevnitr
        LD      C,A             ;delka od kurzoru do konce radku
        LD      B,A
        DEC     B
        RET

;-------------------------------------------------------------------------
;Podprogram precteni bytu z VRAM a ulozeni do BUF

GETVRM: IN      A,(VDPD)        ;cti znak z VRAM
        INC     DE              ;dalsi znak
```

```
        PUSH    BC
        EX      DE,HL
        CALL    STVRAD          ;priprava na cteni
        EX      DE,HL
        POP     BC              ;obnova citace
        LD      (HL),A          ;umisti znak do buf
        INC     HL              ;dalsi misto v buf
        OR      A               ;test na nulovost
        RET

;=============================================================================
; 103C          Posuv casti obrazovky dolu o radek
;-----------------------------------------------------------------------------
; IN:   E - radek odkud

SIFTD:  LD      HL,(UPRMVA)
        LD      A,(HEITVA)
        LD      H,A
        PUSH    HL
        LD      A,E
        ADD     A,L
        LD      (UPRMVA),A
        LD      A,H
        SUB     E
        LD      (HEITVA),A
        CALL    SCRDW
        POP     HL
        LD      A,H
        LD      (HEITVA),A
        LD      A,L
        LD      (UPRMVA),A
        RET

;=============================================================================
; 105B          Presun textu na obrazovku s ukoncovacem,CTRL tiskne
;=============================================================================
; IN:   HL - adresa textu
; OUT:  A  - posledni znak
;       HL - posledni adresa+1

DSPLTB: DB      6               ;TO je   LD B,6  a  NOP

;=============================================================================
; 105C          Presun textu na obrazovku s ukoncovacem a CTRL se vykonava
;=============================================================================

DSPLTA: LD      B,0

;=============================================================================
; 105E          Presun textu na obrazovku a ukoncen 0D - CTRL podle B
;=============================================================================

DSPLTK: CALL    STDM1           ;nastav rezim tisku CTRL
        LD      B,0             ;prevence max 256 znaku

;=============================================================================
; 1063          Presun B znaku na obrazovku ,pripadne i ukoncovace nutno uvazo
t
;=============================================================================
; IN:   HL- adresa textu
;        B- max.pocet znaku
```

```
; OUT:    A - posl. znak
;        HL- adr. posl. znaku+1
DSPLN:   LD       A, (HL)          ; znak
         INC      HL               ; nova adresa
         OR       A                ; je Ø?
         RET      Z                ; ano ven
         DEC      B                ; zmensi citac
         CP       CR               ; je CR?
         JR       Z, DSPCHA        ; vytiskni a konec
         CALL     DSPCH            ; vytiskni znak
         INC      B                ; oprav citac zpet
         DJNZ     DPSLN            ; a b■ opakuj
         RET
;=============================================================================
; 1Ø73             Skok na prislusne CTRL
;-----------------------------------------------------------------------------
CNTLC:   LD       HL, (SCCDTA)     ; tabulka adres CTRL
;=============================================================================
; 1Ø76             Skok podle tabulky skoku
;=============================================================================
; IN:     A - cislo polozky
;         HL- adresa tabulky
EXTBL:   ADD      A,A              ; 2■a - po dvou bytech
         PUSH     DE
         LD       E,A
         LD       D, Ø             ; DE je offset
         ADD      HL,DE            ; do tabulky
         POP      DE
         LD       A, (HL)
         INC      HL
         LD       H, (HL)          ; adresa kam skocit
         LD       L,A
         JP       (HL)             ; skoc
;=============================================================================
; 1Ø82             Tisk znaku a posuv kurzoru - CTRL tiskne
;=============================================================================
DSPCHB:  DB       6                ; TO je LD B,6 a NOP
;=============================================================================
; 1Ø83             Tisk znaku a posuv kurzoru - CTRL se vykonava
;=============================================================================
DSPCHA:  LD       B,Ø
;=============================================================================
; 1Ø85             Tisk znaku a posuv kurzoru ▪ CTRL podle B
;=============================================================================
DSPCHK:  CALL     STDM1            ; nastaveni rezimu tisku
;=============================================================================
; 1Ø88             Tisk znaku a posuv kurzoru
;=============================================================================
```

```
; IN:   A - znak k tisku

DSPCH:  PUSH    AF
        PUSH    BC
        PUSH    DE
        PUSH    HL
        LD      HL,CTC3EX       ;falesna navr.adresa
        PUSH    HL
        LD      HL,DIFLGA
        BIT     1,(HL)          ;je exekuce CTRL
        JR      NZ,DISCTR       ;ne je displej
        CP      ' '             ;test na CTRL
        JR      C,CNTLC         ;ano-> vykonej
DISCTR: BIT     Ø,(HL)          ;je psani ci INsERT
        JR      Z,OVRWC         ;je psani

;Pokracovani DSPCH - tisk pri INSERT

INSTC:  LD      HL,DIFLGA
        BIT     4,(HL)          ;kurzor ve VIEW
        JR      NZ,RGTAW        ;kurzor vpravo
        PUSH    AF
        CALL    FILSP           ;dopln mezerami
        LD      DE,(CURSYA)     ;sour x,y
        POP     AF
        CALL    SIFTR           ;vsun znak
        JR      NC,RGTAW        ;kurzor vpravo a konec
        RST     RST1
        BIT     2,A             ;pohyb nebo uzavreno
        JR      NZ,RGTAW        ;uzavreno-kurzor vpravo a konec
        LD      HL,KINFLG
        SET     6,(HL)          ;ano roluje se
        CALL    SCRUP           ;odroluj nahoru
        LD      A,(CURSYA)      ;y souradnice
        OR      A               ;je=Ø?
        CALL    NZ,UPRAW        ;ne kurzor nahoru
        JR      RGTAW           ;ano=Ø kurzor vpravo

;Pokracovani DSPCH - tisk primo do obrazovky

OVRWC:  LD      HL,DIFLGA
        BIT     4,(HL)          ;kurzor ve VIEW
        JR      NZ,RGTAW        ;mimo-kurzor vpravo
        LD      HL,(CURADA)     ;adresa kurzoru
        RST     PBVRID          ;zapis znak
        CALL    FILSP           ;dopln mezery
        LD      DE,(CURSYA)     ;sour. x,y
        CALL    STPCUR          ;kurzor vpravo
        RST     RST1
        BIT     2,A             ;pohyb ci uzavreno
        JR      NZ,RGTAW        ;uzavreno - tak vpravo
        LD      A,D
        OR      E               ; je x,y=Ø,Ø?
        JR      NZ,RGTAW        ;ne - vpravo
        LD      HL,KINFLG
        SET     6,(HL)          ;roluje se

;=============================================================================
; 1ØED          Tisk CR a LF
;=============================================================================
CRETL:  CALL    CRET            ;tisk CR
```

```
;=============================================================================
; 10F0          Tisk LF
;=============================================================================

LFEED:  CALL    CCURML
        JP      C,SCRUP         ;chyba-scroll nahoru

;=============================================================================
; 10F6          Posuv kurzoru smerem dolu
;-----------------------------------------------------------------------------

LWRAW:  LD      A,03H
        DB      1               ;znamy for s  LD BC,23EH

;-----------------------------------------------------------------------------
; 10F9          Posuv kurzoru nahoru
;-----------------------------------------------------------------------------

UPRAW:  LD      A,2
        DB      1

;-----------------------------------------------------------------------------
; 10FC          Posuv kurzoru vpravo
;-----------------------------------------------------------------------------

RGTAW:  LD      A,1
        DB      6

;-----------------------------------------------------------------------------
; 10FF          Posuv kurzoru vlevo
;-----------------------------------------------------------------------------

LFTAW:  XOR     a

;=============================================================================
; 1100          Posun kurzoru smerem A
;=============================================================================
; IN:   A - 0 vlevo
;           1 vravo
;           2 nahoru
;           3 dolu

STPCU:  LD      DE,(CURSYA)     ;sour. x,y
        LD      HL,JMPCUR       ;skokova adresa
        PUSH    HL
        OR      A
        JR      Z,STPCUL        ;A=0 -> vlevo
        DEC     A
        JR      Z,STPCUR        ;A=1 -> vpravo
        DEC     A
        JR      Z,STPCUU        ;A=2 -> nahoru
        JR      STPCUD          ;A=. -> dolu

;Navratova rutina pro posuny kurzoru

JMPCUR: JP      MVACS

;=============================================================================
;Vypocet pro pohyb kurzoru vpravo
;-----------------------------------------------------------------------------
```

```
; IN:   DE- pozice X,Y
; OUT:  DE- nova pozice X,Y
;    toto plati i pro nasledujici

STPCUR: LD      A,(DIFLGA)
        INC     D               ;X:=x+1
        BIT     4,A             ;je ve VIEW
        RET     NZ              ;je tam
        CALL    CURODX          ;test chyby v x
        RET     NC              ;neni chyba
        LD      D,Ø             ;je chyba x=Ø
        INC     E               ;y=y+1
        JR      TESTY           ;test y

;-------------------------------------------------------------------------
; 1126          Vypocet pro pohyb kurzoru dolu
;-------------------------------------------------------------------------

STPCUD: LD      A,(DIFLGA)
        INC     E
        BIT     4,A             ;je ve VIEW?
        RET     NZ              ;je-tam
TESTY:  CALL    CUROEY          ;testuj chybu Y
        RET     NC              ;neni
        LD      E,Ø             ;y=Ø konec
        RET

;=========================================================================
; 1134          Vypocet pro pohyb kurzoru vlevo
;=========================================================================

STPCUL: LD      A,(DIFLGA)
        DEC     D               ;x:=x-1
        BIT     4,A
        RET     NZ              ;je ve VIEW
        CALL    CURODX
        RET     NC              ;chyba v x neni
        LD      A,(WIDTVA)      ;max.hodnota x+1
        DEC     A               ;spravna hodnota x
        LD      D,A             ;maximalne vpravo
        DEC     E               ;y:=y-1
        JR      TESTYY

;-------------------------------------------------------------------------
; 1147          Vypocet pro pohyb kurzoru nahoru
;-------------------------------------------------------------------------

STPCUU: LD      A,(DIFLGA)
        DEC     E
        BIT     4,A
        RET     NZ              ;je ve VIEW
TESTYY: CALL    CUROEY          ;test chyby v Y
        RET     NC              ;zadna neni
        LD      A,(HEITVA)
        DEC     A               ;maximalni Y
        LD      E,A
        RET                     ; aven
```

```
;=========================================================================
; 1158          Tabulace mezerami
;=========================================================================

TABLT:  LD      A,' '           ;Tabulacni znak

;=========================================================================
; 115A          Tabulace
;-------------------------------------------------------------------------
; IN:   A - tabulacni znak

TABLAT: PUSH    AF
        LD      A,(CURSXA)      ;pozice x
        CPL
        AND     7
        INC     A
        LD      B,A             ;vypocet doplnku k nasobku osmi
        POP     AF
TABLOP: CALL    DSPCH           ;znak
        DJNZ    TABLOP          ;a to b*
        RET

;=========================================================================
; 116A          Vydani zvuku klavesnice
;-------------------------------------------------------------------------

BELK:   LD      HL,(BELKF)      ;L=frekv. H=delka
        LD      A,H
        LD      (BELKC),A       ;delka do citace
        LD      DE,ØCØDØH       ;D=kanal 3 e=3kan hlas
        JR      BEEP            ;zacni piskat

;-------------------------------------------------------------------------
; 1176          Vydani zvuku zvonku BEL=7 ASCII
;-------------------------------------------------------------------------

BEL:    DI
        LD      HL,(BELFA)      ;L=frekv. H=delka
        LD      A,H
        LD      (BELC),A        ;zapis delky
        LD      DE,ØAØBØH       ;inic kody SGC - druhy kanal

;=========================================================================
; 1181          Vystup D,L,E na SGC

;=========================================================================
BEEP:   LD      C,SGC           ;port SGC
        OUT     (C),D           ;postupne tri byty
        OUT     (C),L
        OUT     (C),E
        EI
        RET

;-------------------------------------------------------------------------
; 118B          Obslouzeni trvani zvuku BEL a BELK
;-------------------------------------------------------------------------

CVLOF:  LD      HL,BELKC        ;citac zvuku
        LD      A,ØDFH          ;zvuk BEL - zeslabeni OFF#3
        CALL    JEDEN           ;zaobstarej BEL
```

```
        INC     HL              ;bELK zvuk citac
        LD      A,ØBFH          ;BELK zvuk OFF#2
JEDEN:  INC     (HL)
        DEC     (HL)            ;test na nulu
        RET     Z               ;citac jiz nula-ven
        DEC     (HL)            ;zmensi citac
        RET     NZ              ;jeste neni nula-nevypinej
        OUT     (SGC),A         ;ted ho vypni
        RET

;=============================================================================
; 118B          Hleda zacatek posloupnosti znaku v obrazovce
;=============================================================================
; OUT:  DE- pozice X,Y

SCTOS:  CALL    MVCURE          ;kurzor na 1 znak radky
ISQUE:  LD      A,(CURSYA)      ;a:=y
        OR      A
        JR      Z,YSØ           ;y rovno Ø
        CALL    UPRAW           ;kurzor nahoru
        RST     GBVRID          ;znak
        OR      A
        JR      NZ,ISQUE        ;byl tam znak
        CALL    LWRAW           ;kurzor zpet dolu
YSØ:    CALL    CRET            ;CR
        RST     GBVRID          ;cti znak
        OR      A
        JR      NZ,QUEIS        ;je tam znak
        SCF
QUEIS:  JR      EXTTOS          ;zakoncit

;Podprogram da kursor na konec TV radky v ktere byl

MVCURE: LD      HL,(CURSYA)     ;souradnice x,y
        LD      (CURPSV),HL     ;uschovat
        LD      A,(WIDTVA)      ;max x VIEW
        LD      D,A             ;d:=VIEW x
        DEC     D               ;korekce
        LD      E,L             ;E=Y kurzoru
        JP      MVACS           ;kurzor umistit

;=============================================================================
; 11CA          Hleda zacatek dalsi posloupnosti
;=============================================================================
; OUT:  DE- pozice X,Y

SNTOS:  CALL    MVCURE          ;kurzor na prvni znak radky
        CALL    CCURML          ;test je-li dalsi radka
        JR      C,EXTOS         ;neni dalsi
        RST     GBVRID          ;znak z dalsi
        OR      A
        PUSH    AF
        CALL    LWRAW           ;kurzor dolu
        POP     AF
        JR      NZ,NEXTE        ;je-tam znak - dalsi
EXTTOS: PUSH    AF
        LD      D,Ø
        LD      A,(CURSYA)      ;a:=Y
        LD      E,A             ;DE:=x,y a x=Ø
        PUSH    DE
```

```
        LD      DE,(CURPSV)     ;stara pozice
        CALL    MVACS           ;obnov kurzor
        POP     DE
        POP     AF
        RET

;=============================================================================
; 11ED          Tisk na displej v matici
;=============================================================================
; IN:   B - rozmer X
;       C - rozmer Y
;       HL- adresa DAT

DSPMX:  EX      DE,HL           ;DE:=adr.dat
        LD      A,B
        LD      L,C
        CALL    MLTAL           ;HL:=L*A:=x*y -plocha
        ADD     HL,DE           ;adresa konce dat
        CALL    UPRCUR          ;test kurzoru a matice
        JR      C,EXMTRX        ;matice mimo obraz->ven
        CALL    GFACMX          ;upravy
LOPPMX: PUSH    DE
        CALL    CURUPR
        CALL    WDVPM
        POP     DE
        LD      A,D
        LD      D,ØØH
        ADD     HL,DE
        DEC     A
        LD      D,A
        JR      NZ,LOPPMX
        JR      EXMTRX

;=============================================================================
; 120E          Cteni z dipleje v matici
;=============================================================================
; IN:   B - rozmer X
;       C - rozmer Y
;       HL- adresa bufferu

RDSMX:  PUSH    HL
        EX      DE,HL
        LD      A,B
        LD      L,C
        CALL    MLTAL
        EX      DE,HL
        LD      (HL),ØØH
        INC     HL
        DEC     DE
        LD      A,D
        OR      E
        JR      NZ,L1216
        POP     DE
```

```
        CALL    UPRCUR
        JR      C,EXMTRX
        CALL    GFACMX
L1227:  PUSH    DE
        CALL    CURUPR
        EX      DE,HL
        CALL    RDVPM
        EX      DE,HL
        POP     DE
        LD      A,D
        LD      D,Ø
        ADD     HL,DE
        DEC     A
        LD      D,A
        JR      NZ,L1227
EXMTRX: LD      HL,(SCRDFR)
        RET
UPRCUR: LD      (SCRDFR),HL     ;adresa konce textu
        EX      DE,HL           ;DE:=konec , HL:=plocha
        LD      DE,(CURSYA)
        JP      CCURMM          ;otestuj kurzor a matici
CURUPR: LD      E,C
        LD      D,Ø
        ADD     HL,DE
        PUSH    HL
        PUSH    BC
        LD      DE,(CURPSV)
        LD      A,C
        ADD     A,D
        LD      D,A
        CALL    GCURSA
        EX      DE,HL
        POP     BC
        LD      HL,CURPSV
        INC     (HL)
        POP     HL
        RET

;-------------------------------------------------------------------------
; 126Ø          Upravy pro tisky a cteni v matici
;-------------------------------------------------------------------------

GFACMX: LD      A,(HEITVA)
        SUB     E
        CP      C
        JR      C,L1268
        LD      A,C
L1268:  BIT     7,E
        JR      Z,L1285
        PUSH    DE
        PUSH    HL
        LD      L,B
        LD      A,E
        NEG
        CALL    MLTAL
        POP     DE
        ADD     HL,DE
        POP     DE
        LD      A,C
        ADD     A,E
```

```
        LD      E,Ø
        PUSH    HL
        LD      HL,HEITVA
        CP      (HL)
        JR      C,L1284
        LD      A,(HL)
L1284:  POP     HL
L1285:  LD      (CURPSV),DE
        PUSH    HL
        PUSH    AF
        LD      H,Ø
        LD      A,(WIDTVA)
        SUB     D
        CP      B
        JR      C,L1295
        LD      A,B
L1295:  BIT     7,D
        JR      Z,L12A8
        LD      A,D
        NEG
        LD      H,A
        LD      A,B
        ADD     A,D
        PUSH    HL
        LD      HL,WIDTVA
        CP      (HL)
        JR      C,L12A7
        LD      A,(HL)
L12A7:  POP     HL
L12A8:  LD      L,A
        LD      A,B
        SUB     H
        SUB     L
        LD      E,A
        POP     AF
        LD      D,A
        LD      C,H
        LD      B,L
        POP     HL
        RET

;=============================================================================
; 12B3          Rolovani obrazovky smerem nahoru
;=============================================================================

SCRUP:  LD      HL,(WIDTDA)
        LD      H,Ø
        LD      (SCRDFR),HL
        LD      D,H
        LD      E,H
        JR      ROLVER

;=============================================================================
; 12BF          Rolovani obrazovky smerem dolu
;=============================================================================

SCRDW:  LD      H,ØFFH
        LD      A,(WIDTDA)
        CPL
        LD      L,A
```

```
        INC     HL
        LD      (SCRDFR),HL
        LD      D,Ø
        LD      A,(HEITVA)
        LD      E,A
        DEC     E
ROLVER: CALL    GCURSA
        LD      A,(HEITVA)
        LD      B,A
        DEC     B
        JR      Z,L12EF
L12DB:  PUSH    BC
        LD      D,H
        LD      E,L
        LD      BC,(SCRDFR)
        ADD     HL,BC
        LD      A,(WIDTVA)
        LD      B,A
        PUSH    HL
        CALL    BLKTRS
        POP     HL
        POP     BC
        DJNZ    L12DB
L12EF:  LD      A,(WIDTVA)
        LD      B,A
        XOR     A
L12F4:  RST     PBVRID
        INC     HL
        DJNZ    L12F4
        RET

;=============================================================================
; 12F9          Rolovani obrazovky vpravo
;=============================================================================

SCRRG:  LD      HL,1
        LD      (SCRCOR),HL
        DEC     HL
        DEC     HL
        LD      (SCRDFR),HL
        LD      D,Ø
        LD      A,32
        JR      ROLHOR

;=============================================================================
; 130A          Rolovani obrazovky vlevo
;=============================================================================

SCRLF:  LD      A,(WIDTVA)
        DEC     A
        LD      D,A
        CPL
        LD      L,A
        LD      H,ØFFH
        INC     HL
        LD      (SCRCOR),HL
        LD      HL,1
        LD      (SCRDFR),HL
        XOR     A
ROLHOR: LD      (SCRLC),A
```

```
        LD      A,(HEITVA)
        PUSH    AF
        LD      E,A
        DEC     E
        CALL    GCURSA
        POP     BC
L132B:  PUSH    BC
        PUSH    HL
        LD      DE,(SCRCOR)
        ADD     HL,DE
        LD      D,H
        LD      E,L
        LD      BC,(SCRDFR)
        ADD     HL,BC
        LD      A,(WIDTVA)
        LD      B,A
        DEC     B
        CALL    NZ,BLKTRS
        POP     HL
        POP     BC
        LD      A,(SCRLC)
        RST     PBVRID
        LD      D,Ø
        LD      A,(WIDTDA)
        LD      E,A
        OR      A
        SBC     HL,DE
        DJNZ    L132B
        RET

;=============================================================================
; 1353          Nastaveni systemoveho VIEW
;-----------------------------------------------------------------------------

VIEWRS: LD      BC,(HITDA)
        LD      HL,Ø
        JR      STOVEW

;=============================================================================
; 135C          NastAveni rozmeru VIEW - okna
;-----------------------------------------------------------------------------
; IN:   H - levy okraj
;       L - pravy okraj
;       D - horni okraj
;       E - dolni okraj

VIEWP:  LD      BC,(HITDA)
        LD      A,E
        CP      C
        JR      NC,L136B
        SUB     D
        JR      C,L136F
        INC     A
        LD      C,A
        LD      A,L
        CP      B
L136B:  CALL    NC,.ILGCM
        SUB     H
L136F:  CALL    C,.ILGCM
        INC     A
```

```
        LD      B,A
        LD      L,D
STOVEW: LD      (HEITVA),BC
        LD      (UPRMVA),HL
        JR      HOMEP

;=============================================================================
; 137E          Ma zrusit obrazovku-VIEWa obsah - a smazat sprity
;ale neudela to pro chybu!!!!!
;=============================================================================

CLRSS:  RST     RST1            ;diflga
        ADD     A,A             ;JE TEXT rezim?
        RET     NC              ;tady MA BYT   RET C
        CALL    VIEWRS          ;normalini VIEW
        CALL    CLRSC           ;smaZ obrazovku

;=============================================================================
; 1387          Smaze vsechny sprity pomoci ERSSPR
;=============================================================================

ERSPRA: LD      A,1FH           ;pocet spritu
L1389:  PUSH    AF
        CALL    ERSSPR          ;ERASE A
        POP     AF
        OR      A
        RET     Z               ;A=Ø ->ven
        DEC     A
        JR      L1389           ;opakuj A*

;=============================================================================
; 1393          Smazani obrazovky znakem ØØ
;=============================================================================

CLRSX:  XOR     A               ;mazaci znak

;-----------------------------------------------------------------------------
; 1394          Smazani obrazovku libovolnym znakem
;-----------------------------------------------------------------------------
; IN:   A - mazaci znak

CLRSCX: LD      HL,LNBUF        ;radkova pamet
        LD      B,4Ø            ;pocet znaku max.radky
L1399:  LD      (HL),A
        INC     HL              ;vytvoreni radky znaku
        DJNZ    L1399
        LD      DE,Ø
        CALL    GCURSA          ;kurzor
        LD      BC,(HEITVA)
L13A7:  PUSH    HL
        CALL    WDVLB           ;radku na obrazovku
        POP     HL
        LD      A,(WIDTDA)
        LD      E,A
        LD      D,Ø
        ADD     HL,DE
        DEC     C
        JR      NZ,L13A7        ;vytisk n* LNBUF - smazano

;=============================================================================
```

```
;=============================================================================
; 13B6          Umisteni kurzoru do pozice Ø,Ø
;-----------------------------------------------------------------------------

HOMEP:  LD      DE,Ø            ;sour.kurzoru Ø,Ø
        JR      MVACS

;=============================================================================
; 13BB          Smazani znaku vpravo od kurzoru do konce radky
;=============================================================================

CANCL:  LD      DE,(CURSYA)     ;kurzor
CONTCN: PUSH    DE
        CALL    RDSCH           ;znak
        POP     DE
        RET     C               ;nebyl zadny tak uz ven
        RET     Z               ;byla nula-konec
        XOR     A
        RST     PBVRID          ;zapis novy - prazdny
        CALL    STPCUR          ;posuv kurzoru
        JR      CONTCN          ;opakuj

;=============================================================================
; 13CD          Vytisk znaku CR - navrat kursoru na zacatek radky
;-----------------------------------------------------------------------------

CRET:   LD      A,(CURSYA)      ;pozice y kurzoru
        LD      E,A
        LD      D,Ø             ;DE=x,y=Ø,y
        JR      MVACS           ;umisti kurzor

;=============================================================================
;Presun kurzoru na zacatek posloupnosti znaku
;-----------------------------------------------------------------------------

SCTOSD: CALL    SCTOS           ;najdi zacatek posloupnosti
        JR      MVACS           ;dej tam kurzor

;-----------------------------------------------------------------------------
;Presun kurzoru na zacatek dalsi posloupnosti znaku
;-----------------------------------------------------------------------------

SNTOSD: CALL    SNTOS           ;najdi zacatek dalsi a umisti

;=============================================================================
; 13DD          Umisteni kurzoru
;=============================================================================
; IN:   D - souradnice X
;       E - souradnice Y

MVACS:  LD      (CURSYA),DE     ;sour.kurzoru
        CALL    CCUROT          ;test spravnosti
        RST     RST1            ;DIFLGA
        SET     4,(HL)          ;kurzor mimo VIEW
        JR      C,L13EB         ;ano-je mimo
        RES     4,(HL)          ;kurzor ve VIEW
        PUSH    AF
        CALL    GCURSA          ;vypocet adresy
        LD      (CURADA),HL     ;adresa
        POP     AF
```

```
        CALL    C,.OVIEW        ;pry chybe->
        RET

;=============================================================================
; 13F7          Vypocet adresy kurzoru
;-----------------------------------------------------------------------------
; IN:   DE- souradnice kurzoru
; OUT:  HL- adresa v VRAM

GCURSA: LD      A,D
        XOR     80H
        LD      D,A
        LD      A,E
        XOR     80H
        LD      E,A
        LD      A,(WIDTDA)
        CP      40
        LD      BC,0EB80H
        JR      Z,L140C
        LD      BC,0EF80H
L140C:  LD      L,E
        CALL    MLTAL
        ADD     HL,BC
        LD      E,D
        LD      D,0
        ADD     HL,DE
        PUSH    HL
        LD      HL,(UPRMVA)
        LD      E,H
        LD      A,(WIDTDA)
        CALL    MLTAL
        LD      D,0
        ADD     HL,DE
        LD      DE,(CODTLA)
        ADD     HL,DE
        EX      DE,HL
        POP     HL
        ADD     HL,DE
        RET

;=============================================================================
; 142C          Nasobeni HL:=HL*DE
;=============================================================================

MULTHD: PUSH    BC
        LD      B,H
        LD      C,L
        LD      A,16            ;pocet bitu vysledku
        LD      HL,0            ;vysledek
L1434:  ADD     HL,HL
        RL      E
        RL      D
        JR      NC,L143C
        ADD     HL,BC
L143c:  DEC     A
        JR      NZ,L1434
        POP     BC
        RET

;-----------------------------------------------------------------------------
```

```
;-------------------------------------------------------------------------
; 1441          Nasobeni HL:=L*A
;-------------------------------------------------------------------------

MLTAL:  PUSH    DE
        LD      H,Ø             ;HL=L
        LD      D,H
        LD      E,A             ;DE:=A
        CALL    MULTHD          ;HL:=HL*DE=L*A
        POP     DE
        RET

;=========================================================================
; 144B          Presun VRAM->VRAM s delkou urcenou pouze B
;-------------------------------------------------------------------------

RDVPM:  PUSH    BC
        LD      C,B
        LD      B,Ø             ;BC:=B
;-------------------------------------------------------------------------
        CALL    VCTIR           ;normalini presun
        POP     BC
        RET

; 1454          Presun VRAM->RAM na adresu LNBUF
;-------------------------------------------------------------------------

BLKTRS: PUSH    DE
        LD      DE,LNBUF        ;adresa kam
        CALL    RDVPM           ;presun RAM->VRAM
        POP     HL

;-------------------------------------------------------------------------
; 145C          Presun RAM->VRAM z adresy LNBUF
;-------------------------------------------------------------------------

WDVLB:  EX      DE,HL
        LD      HL,LNBUF        ;adresa odkud

;-------------------------------------------------------------------------
; 146Ø          Presun MEM->VRAM s delkou urcenou pouze B
;-------------------------------------------------------------------------

WDVPM:  PUSH    BC
        LD      C,B
        LD      B,Ø             ;BC:=B
        CALL    CVTIR           ;normalni presun
        POP     BC
        RET

;-------------------------------------------------------------------------
; 1469          Pomocny podprogram - testy pozic kurzoru
;-------------------------------------------------------------------------

CCURMM: CALL    CURODX          ;test x
        JR      NC,L1475
        BIT     7,A
        RET     Z
        ADD     A,B
        DEC     A
```

```
        SCF
        RET     M
L1475:  CALL    CUROEY
        BIT     7,A
        RET     Z
        ADD     A,C
        DEC     A
        SCF
        RET     M
        OR      A
        RET

;-------------------------------------------------------------------------
; 1481          Test na pritomnost kurzoru ve VIEW
;-------------------------------------------------------------------------

CCUROT: CALL    CUROEY          ;test y
        RET     C               ;chyba ->ven

;-------------------------------------------------------------------------
; 1485          Test na souradnici x kurzoru

CURODX: LD      A,D             ;x-ova souradnice

;-------------------------------------------------------------------------
; 1486          Test souradnice x v A registru

CCUROX: PUSH    HL
        LD      HL,WIDTVA       ;testovana sysprom
        JR      L1497

;-------------------------------------------------------------------------
; 148C          Podprogram kurzoru - test
;-------------------------------------------------------------------------

CCURML: LD      A,(CURSYA)      ;souradnice y
        INC     A               ;+1
        JR      L1493           ;dokonci test

;-------------------------------------------------------------------------
; 1492          Test souradnice Y kurzoru

CUROEY: LD      A,E
L1493:  PUSH    HL
        LD      HL,HEITVA       ;testovana sysvar
L1497:  OR      A
        JP      M,L149C         ;kdyz kurzor zaporny
        CP      (HL)            ;porovnej a nastav F
L149C:  POP     HL
        CCF
        RET

;=========================================================================
; 149F          Priprava adresy VDP pro zapis dat
;=========================================================================
; IN:   HL - adresa VRAM

STVWAD: SET     6,H             ;priznak zapisu

;-------------------------------------------------------------------------
```

```
;-------------------------------------------------------------------------
; 14A1          Priprava adresy VDP pro cteni dat

STVRAD: LD      C,VDPC          ;povelovy port
        OUT     (C),L
        OUT     (C),H           ;zapis adresy
        RES     6,H             ;zpet HL na adresu
        LD      C,VDPD          ;priprava C pro priste
        RET

;=========================================================================
; 14AC          Cteni z pozice kurzoru jednoho znaku
;=========================================================================
; OUT:  A - znak
;       HL- adresa kurzoru

RDSCHA: LD      HL,(CURADA)     ;adresa kurzoru
        JR      L14BA

;=========================================================================
; 14B1          Cteni znaku z obrazovky z urcene pozice
;=========================================================================
; IN:   DE-souradnice X,
; OUT:  A - znak
;       HL- adresa znaku

RDSCH:  CALL    CCUROT          ;test kurzoru
        CALL    C,.OVIEW        ;chyba->je mimo
        CALL    GCURSA          ;zjisti adresu
L14BA:  RST     GBVRID          ;precti byt
        OR      A               ;otestuj ho a CY=0
        RET

;=========================================================================
; 14BD          Zapis bytu do pameti VRAM
;=========================================================================
; IN:   A - byt
;       HL- adresa

PBVRAM: PUSH    BC
        CALL    STVWAD          ;priprava pro zapis
        OUT     (VDPD),A        ;zapis bytu
        POP     BC
        RET

;=========================================================================
; 14C5          Precteni bytu z VRAM
;=========================================================================
; IN:   HL- adresa
; OUT:  A - byt

GBVRAM: PUSH    BC
        CALL    STVRAD          ;priprava pro cteni
        IN      A,(VDPD)        ;precti byt
        POP     BC
        RET

;=========================================================================
; 14CD          Tabulka adres procedur pro vykonani CTRL+klavesa
```

```
SCCDT:  DW      DSPCH3          ;NOP
        DW      DSPCH3          ;NOP
        DW      SCTOSD          ;kurzor na zacatek radku
        DW      SCRDW           ;rolovani dolu
        DW      SCRLF           ;rolovani vlevo
        DW      SCRUP           ;rolovani nahoru
        DW      SCRRG           ;rolovani vpravo
        DW      BEL             ;zvonek
        DW      DELTC           ;smaz znak
        DW      TABLT           ;tabulace
        DW      LFEED           ;dalsi radka
        DW      HOMEP           ;kurzor HOME
        DW      CLRSC           ;smaz obrazovku
        DW      CRETL           ;RETURN
        DW      SNTOSD          ;kurzor na zac.dalsi radky
        DW      STOVRM          ;psani
        DW      STINSM          ;insert
        DW      MMODE           ;MCOLOR
        DW      GMODE           ;GII
        DW      CMODE           ;GI
        DW      TMODE           ;TExT
        DW      NRMSC           ;navrat do stranky Ø
        DW      REVSC           ;vymena stranky
        DW      CRETL           ;CR+LF
        DW      CANCL           ;smazani do konce radky
        DW      RVDSPP          ;pohled prohodit
        DW      RVWRTP          ;kurzor prohodit
        DW      DSPCH3          ;NOP
        DW      RGTAW           ;kurzor vlevo
        DW      LFTAW           ;kurzor vlevo
        DW      UPRAW           ;kurzor nahoru
        DW      LWRAW           ;kurzor dolu
;-------------------------------------------------------------------------

; 15ØD          Inicializacni tabulka pro SCREEN TABLE

INISCT:
        DB      Ø               ;DIFLGA
        DW      38ØØH           ;CODTLA predlohyII
        DW      3B8ØH           ;cCOTLA barvy
        DW      28ØØh           ;CPATLA predlohy U
        DW      3BØØH           ;SATTLA sprity attr
        DW      2ØØØH           ;SPATLA sprity predlohy
        Db      ØE1H            ;BDCOLA barvy
        DB      Ø,Ø,23,31       ;VIEW Ø,Ø,23,31
        DB      23,31           ;rozmery obrazovky
        DB      Ø,Ø             ;x,y kurzoru
        DW      38ØØH           ;adresa kurzoru
        DB      Ø               ;znak na kurzoru
        DB      16,16           ;casy kurzoru
        DB      8,14            ;udaje BEL
        DB      9               ;GRFLA flag grafiky

;=========================================================================
; 1529          Nahrani IDT bloku a podle nej i dat
;=========================================================================
; IN:   HL- adresa IDT

BSAVE:  DI
        CALL    VDFAC           ;vypnuti SGC
```

```
        PUSH    HL              ;adresa IDT
        CALL    WTFID           ;nahraj IDT
        POP     HL
        RET     C               ;pri chybe ->ven
        CALL    LFTBC           ;HL:=od BC:=kolik
        CALL    BSAVD           ;nahraj data
        JR      EXTOFF          ;motor OFF a ven

;===============================================================================
; 153B          Precteni souvislych dat podle IDT
;===============================================================================
; IN:   A - Ø OLD
;           1 VERIFY
;       HL- adresa buf IDT

BLOAD:  OR      A               ;je OLD/VERIFY
        JR      Z,ISOLD         ;je OLD->
        PUSH    HL
        LD      HL,SYSFMT
        SET     1,(HL)          ;je VERIFY
        POP     HL
ISOLD:  DI
        CALL    LFTBC           ;HL:=od  BC:=kolik
        CALL    BLODD           ;precist data
        PUSH    AF
        CALL    SGINTS          ;inic.SML
        POP     AF
EXTOFF: CALL    MTROF           ;vypni motor
        EI
        RET                     ;a ven s CY!

;===============================================================================
; 1556          Vyber informaci z IDT
;===============================================================================
; IN:   HL- adresa IDT
; OUT:  HL- data od
;       BC- data delka

LFTBC:  LD      DE,1Ø           ;offset v hlavicce
        ADD     HL,DE           ;adresa od
        LD      E,(HL)
        INC     HL
        LD      D,(HL)          ;vyber od
        INC     HL
        LD      C,(HL)
        INC     HL
        LD      B,(HL)          ;vyber delky
        EX      DE,HL
        RET

;===============================================================================
```

```
;=============================================================================
; 1563          Nahrani dat dle SySFMT v blocich na magnetofon
;=============================================================================
; IN:   HL- data od
;       BC- data delka

BSAVD:  LD      A,B
        OR      C               ;je delka dat=Ø?
        RET     Z               ;ano->ven
        OR      A               ;je <256?
        LD      A,C             ;delka bloku
        JR      Z,NOFULL        ;delka >256->
        XOR     A               ;A:=256
NOFULL: PUSH    BC
        LD      B,A             ;delka bloku
        LD      C,2             ;IBG pisk
        CALL    WATBL           ;ven blok
        POP     BC
        RET     C               ;pri chybe  ven
        DEC     B               ;BC:=BC-256
        JP      P,NOFULL        ;opakuj kdyz BC>256
        RET

;=============================================================================
; 1579          Precteni bloku dat z magnetofonu dle SYSFMT
;=============================================================================
; IN:   HL- data kam
;       BC- data delka

BLODD:  LD      A,B
        OR      C
        RET     Z               ;delka dat=Ø ->ven
        PUSH    BC
        CALL    RATBL           ;precti blok
        POP     BC
        RET     C               ;chyba->ven
        DEC     B               ;delka -256
        JP      P,BLODD         ;jeste cist
        RET

;=============================================================================
; 1587          Precte IDT a zkontroluje jmeno
;=============================================================================
; IN:   HL- buffer pro IDT
;       DE- adresa jmena

RFIDC:  PUSH    HL              ;adr.buf
        PUSH    DE              ;adr.jmena
        CALL    RDFID           ;precti IDT
        POP     DE
        POP     HL
        RET     C               ;chyba ->ven
        INC     HL              ;adresa jmena
        CALL    CPFNM           ;porovnej
        DEC     HL              ;vrat zpet
        CALL    C,.FINUM        ;neni stejne->chyba
        RET

;=============================================================================
; 1598          Precte IDT blok a nastavi SYSFMT
```

```
;=============================================================================
; IN:   HL- adresa bufferu

RDFID:  DI
        CALL    VDFAC           ;vypnuti SGC
        CALL    MTRON           ;zapnuti motoru
        PUSH    HL
        CALL    SRLAD           ;pockej na uvodni pisk
        POP     HL
        JR      C,BREAK         ;byla chyba
        XOR     A
        LD      (SYSFMT),A      ;MEM OLD
        LD      A,'H'           ;cte se HEADER
        PUSH    HL
        CALL    LOADC           ;precti blok
        POP     HL
        JR      NC,NOERTP       ;nebyla chyba->
        CP      8CH             ;bylo .ILGEL?
        JR      Z,RDFID         ;ano -znovu
        SCF                     ;chyba
        RET

NOERTP: XOR     A
        BIT     3,(HL)          ;jr to CPU?
        JR      Z,ISCPU         ;ano->
        INC     A               ;je to VRAM
ISCPU:  LD      (SYSFMT),A      ;uloz typ
        RET

;=============================================================================
; 15C3          Zapise IDT a nastavi SYSFMT
;=============================================================================
; IN:   HL- adresa bufferu

WTFID:  CALL    MTRON           ;zapnout motor
        LD      BC,0E00H        ;zpozd.konstanta
DLYFID: IN      A,(50H)         ;test RESET
        RLCA
        JR      C,BREAK         ;je RESET
        XOR     A
DLYWTF: DEC     A
        JR      NZ,DLYWTF       ;zpozdeni
        DEC     BC
        OR      B
        JR      NZ,DLYFID       ;zpozdeni B*
        LD      (SYSFMT),A      ;MEM OLD
        LD      A,'H'           ;blok HEADER
        LD      BC,1F1FH        ;delka 1FH a pisk tez
        PUSH    HL
        CALL    STORC           ;zapis bloku
        POP     HL
BREAK:  CALL    C,.BREAK        ;chyba
        JR      NOERTP          ;zpet a ini SYSFMT

;=============================================================================
; 15E8          Zapis bloku typu FILE
;-----------------------------------------------------------------------------

WATBLF: LD      A,'F'
        DB      11              ;falesne LD  DE,443EH
```

```
;-------------------------------------------------------------------------
; 15EB          Zapis bloku typu DATA
;-------------------------------------------------------------------------
; IN:   HL- adresa ulozeni bloku
;       B - pocet bytu bloku
;       C - delka IBG pisku

WATBL:  LD      A,'D'

;-------------------------------------------------------------------------
; 15ED          Zapis bloku libovolneho typu
;-------------------------------------------------------------------------
; IN:   zaklad viz WATBL a WATBLF
;       A - znak kodu typu
; OUT:  HL- posl.byt+1
        LD      DE,443EH

STROC:  CALL    MTRON           ;zapni motor
        LD      D,A             ;znak urceni typu
        PUSH    BC
        LD      B,C             ;delka IBG v BC
TOBEP:  SCF
        CALL    OUTPS           ;jednickovy puls
        DEC     BC
        LD      A,B
        OR      C
        JR      NZ,TOBEP        ;vysilej IBG
        POP     BC
        CALL    STBYT           ;zapis ident.bloku
        LD      D,B
        CALL    STBYT           ;zapis poctu bytu bloku
        LD      C,Ø             ;kontr.soucet je nula
STOREB: IN      A,(5ØH)
        RLCA                    ;RESET?
        JR      C,BREAK         ;by
        LD      A,(SYSFMT)      ;typ pameti
        BIT     Ø,A
        CALL    NZ,GBVRAM       ;bude VRAM - vyber
        JR      NZ,ISVRM        ;uz neber PCU
        LD      A,(HL)          ;vyber CPU
ISVRM:  LD      D,A
        LD      A,C
        ADD     A,D
        LD      C,A             ;kontr.soucet
        CALL    STBYT           ;zapis byt na CMT
        INC     HL
        DJNZ    STOREB          ;a to vse B*
        LD      D,C
        CALL    STBYT           ;zapis kontr.soucet
        OR      A               ;CY=Ø
        RET

;=========================================================================
; 1626          Zapis jednoho bytu na CMT
;-------------------------------------------------------------------------
; IN:   D - zapisovany byt

STBYT:  PUSH    BC
        LD      B,8             ;pocet bitu
        OR      A
```

```
        CALL    OUTPS           ;start bit=Ø
STOPLS: RRC     D
        CALL    OUTPS           ;vysli bit
        DJNZ    STOPLS          ; 8:
        SCF
        CALL    OUTPS           ;stop bit=1
        POP     BC
        RET

;=============================================================================
; 163A          Vyslani pozadovaneho pulsu na magnetofon
;=============================================================================
; IN:   CY- bit

OUTPS:  LD      A,3             ;nahozeni
        CALL    BITS            ;vysli
        LD      A,2             ;shozeni
BITS:   OUT     (5ØH),A         ;vysli
        CALL    NC,HALFNO       ;podle CY dvakrat tolik
HALFNO: LD      A,(STDLY)       ;zpozdeni
STDLYD: DEC     A
        JR      NZ,STDLYD       ;realizace zpozdeni
        RET

;=============================================================================
; 164D          Precteni jednoho bloku typu FILE
;-----------------------------------------------------------------------------

RATBLF: LD      A,'F'
        DB      1               ;falesne  LD    BC,443Eh

;=============================================================================
; 165Ø          Precteni jednoho bloku typu DATA
;=============================================================================
; IN:   HL- adresa kam
; OUT:  HL- posl.byt+1

RATBL:  LD      A,'D'

;=============================================================================
; 1652          Precteni jednoho bloku typu daneho uzivatelelm
;-----------------------------------------------------------------------------
; IN:   zaklad viz RATBLF a RATBL
;       A - znak zadaneho typu

LOADC:  CALL    MTRON           ;spust motor
        PUSH    AF
        CALL    WAITST          ;cekej na zavadecku
        CALL    LDBYT1          ;precti ident bloku
        POP     DE
        JR      C,BREAK2        ;chyba->
        CP      'E'             ;je to EOF?
        CALL    Z,.DATOT        ;ano->
        CP      D
        CALL    NZ,.ILGEL       ;cteme jiny blok!
        PUSH    DE
        CALL    LDBYT           ;precti byt
        POP     DE
        LD      B,A             ;pocet bytu
        LD      (AVILMT),A      ;citac do AVILMT
```

```
        LD      A,D
        CP      'F'             ;je to FILE?
        JR      NZ,NOCHCK       ;ne-nekontroluj
        LD      A,(BSIZMT)      ;delka buf
        CP      B
        CALL    C,.BUFFL        ;buffer je moc kratky->
NOCHCK: LD      D,Ø             ;kontr.soucet
GETBLK: PUSH    DE
        CALL    LDBYT           ;precti byt
        POP     DE
BREAK2: CALL    C,.BREAK        ;->chyba
        LD      E,A
        LD      A,(SYSFMT)
        BIT     1,A             ;typ pameti
        JR      Z,OLDMEM        ;je to OLD
        BIT     Ø,A             ;typ pameti
        CALL    NZ,GBVRAM       ;precti VRAM
        JR      NZ,VRMIS        ;a preskoc RAM
        LD      A,(HL)          ;vyber RAM
VRMIS:  CP      E               ;je to stejne?
        JR      NZ,CHKSM        ;ne-neni->chyba VERIFY
        JR      CONTRA          ;opakuj a zmensi citac
OLDMEM: BIT     Ø,A             ;typ pameti
        LD      A,E
        CALL    NZ,PBVRAM       ;zapis do VRAM
        JR      NZ,CONTRA       ;preskoc RAM
        LD      (HL),A          ;zapis do RAM
CONTRA: ADD     A,D
        LD      D,A             ;kontr.soucet
        INC     HL              ;nova adresa
        DJNZ    GETBLK          ;cely blok
        LD      B,D
        CALL    LDBYT           ;kontr.soucet
        CP      B
        CALL    NZ,.CKSM        ;neni stejny->
        RET

;=============================================================================
; 16B3          Precteni jednoho bytu z CMT
;=============================================================================

LDBYT:  CALL    RDCNT           ;precti start bit

;-----------------------------------------------------------------------------
; 16B6          Precteni jednoho bytu bez start bitu
;-----------------------------------------------------------------------------
; OUT:  A - precteny byt

LDBYT1: PUSH    BC
        PUSH    HL
        LD      B,8             ;pocet bitu
        LD      H,Ø             ;vysledek
        LD      A,(7257H)
        LD      L,A             ;delka pulzu
LOPLDB: CALL    RDCNT           ;puls
        JR      C,ERRLDB        ;chyba->
        CP      L               ;je stejny s L?
        RR      H               ;zaroluj ho do H
        DJNZ    LOPLDB          ;8*
        CALL    RDCNT           ;stop bit
```

```
        LD      A,H             ;byt
ERRLDB: POP     HL
        POP     BC
        RET

;=============================================================================
; 16D1          Urceni rychlosti na zaklade uvodniho piskani
;=============================================================================

SRLAD:  CALL    MTRON           ;spust motor
        LD      HL,Ø            ;vysledek
        LD      (AVRHL),HL      ;zatim Ø
        LD      B,H
        LD      C,L
        CALL    RDCNT           ;precti puls
        LD      (AVRHL+2),A     ;celkova delka
LOPSRL: CALL    RDCNT           ;puls
        RET     C               ;chyba->ven
        PUSH    AF
        LD      A,D
        LD      D,Ø
        ADD     HL,DE
        PUSH    HL
        LD      HL,(AVRHL)
        LD      E,A
        ADD     HL,DE
        LD      (AVRHL),HL
        POP     HL
        POP     DE
        LD      A,(AVRHL+2)
        SUB     D
        ADD     A,1Ø
        CP      2Ø
        JR      NC,SRLAD
        LD      A,D
        LD      (AVRHL+2),A
        DJNZ    LOPSRL
        LD      A,H
        SRL     A
        ADD     A,H
        LD      HL,(AVRHL)
        LD      L,A
        LD      A,H
        SRL     A
        ADD     A,H
        LD      H,A
        LD      (AVRHL),HL
        RET

;=============================================================================
; 1717          Ceka po IBG a doladuje rychlost
;-----------------------------------------------------------------------------

WAITST: PUSH    HL
        LD      C,Ø
        CALL    RDCNT
        LD      HL,(AVRHL)
NEXTPS: CALL    RDCNT           ;puls
        JR      C,ERRWTS        ;chyba->
        LD      A,E
```

```
        CP      L
        JR      C,NEXTPS
        LD      A,D
        CP      H
        JR      NC,ENDWTS
        LD      C,2
        CALL    RDCNT
ENDWTS: LD      A,H
        ADD     A,L
        LD      (AVRHL+2),A
ERRWTS: POP     HL
        RET

;=============================================================================
; 1739          Rutina zmereni delky pulsu na magnetofonu
;=============================================================================
; IN:   C - Ø0 puls je -_
;           Ø2 puls je _-
; OUT:  A - celkova delka
;       D - delka prve casti
;       E - delka druhe casti

RDCNT:  IN      A,(5ØH)         ;test RESET
        RLCA
        RET     C               ;je tam->
        AND     2               ;test vstupu
        XOR     C               ;C=Ø0 if _   c=Ø2 if -
        JR      NZ,RDCNT        ;je to on - cekej
        IN      A,(1)           ;cas CTC#1
        LD      D,A             ;D:=cas nabehu
WAIPUL: IN      A,(5ØH)         ;test reset
        RLCA
        RET     C               ;bylo->
        AND     2               ;test bitu
        XOR     C
        JR      Z,WAIPUL        ;je to on-cekej
        IN      A,(1)           ;cas CTC#1
        LD      E,A             ;doba behu
        LD      A,7
        OUT     (1),A
        XOR     A
        OUT     (1),A           ;znovu inic.CTC - bezi dalsi puls
        LD      A,D
        NEG
        LD      D,A             ;D=-d
        LD      A,E
        NEG
        PUSH    AF              ;A=-E
        SUB     D
        LD      E,A             ;A=rozdil
        POP     AF
        OR      A               ;CY=Ø
        RET

;=============================================================================
; 1765          porovna shodnost dvou textu pokud nenajde znak ?
;-----------------------------------------------------------------------------
; IN:   DE - adresa textu1
;       HL - adresa textu2
; OUT:  Z  - Z/NZ  shoda/neshoda
```

```
CPFNM:  PUSH    HL              ;adresa jmena
COMPNM: LD      A,(DE)          ;prvy znak
        CP      '?'             ;je to ?
        JR      Z,EXCFNM        ;ano->ven
        CP      (HL)            ;je stejne
        SCF
        JR      NZ,EXCPNM       ;neni->ven
        INC     DE
        INC     HL              ;nove adresy
        OR      A
        JR      NZ,COMPNM       ;do konce textu
EXCPNM: POP     HL
        RET

;=============================================================================
; 1776          Spusteni motiru
;=============================================================================

MTRON:  PUSH    AF
        XOR     A               ;inic.byt
        OUT     (40H),A         ;printer=00
        LD      A,2             ;MOTOR ON
        JR      STOMTR          ;zapis

;=============================================================================
; 177E          Zastaveni motoru
;=============================================================================

MTROF:  PUSH    AF
        XOR     A
STOMTR: OUT     (50H),A         ;zapis motor
        POP     AF
        RET                     ;ven

;=============================================================================
; 1784          Test tiskarny na konec povol.poctu znaku
;-----------------------------------------------------------------------------

PCLMX:  PUSH    HL
        LD      HL,PMXCLM       ;pocet znaku
        LD      A,(HL)          ;do A
        INC     HL
        CP      (HL)            ;porovnej s citacem vystouplych
        POP     HL
        RET     NZ              ;jeste vporadku

;=============================================================================
; 178D          Vysle CRLF je-li povoleno
;-----------------------------------------------------------------------------

POTNL:  BIT     1,E             ;je autom CRLF ?
        RET     Z               ;ne->ven
        LD      A,CR            ;je CR
        JR      POTCH           ;a bude i LF

;=============================================================================
; 1794          Vysle LF je-li povoleno
;-----------------------------------------------------------------------------

POTLF:  BIT     0,E             ;je povoleno LF?
```

```
        RET     Z               ;ne-navrat
        LD      A,LF            ;znak LF

;=============================================================================
; 1799          Vystup jednoho znaku na tiskarnu
;=============================================================================
; IN:   A - znak

POTCH:  PUSH    DE
        PUSH    HL
        LD      D,A             ;znak
RDY?:   IN      A,(50H)         ;test READY
        BIT     1,A
        JR      Z,RDY?          ;cekej na READY od tiskarny
        LD      A,D
        OUT     (40H),A         ;znak
        LD      A,1
        OUT     (50H),A         ;vysli STROBE
        LD      A,11            ;zpozd.konst.
WAISTB: DEC     A
        JR      NZ,WAISTB       ;STROBE drzi nahore
        OUT     (50H),A         ;STROBE spadlo
        LD      A,D             ;znak
        LD      HL,PHDPOS
        INC     (HL)            ;hlava o znak+
        CP      CR
        JR      NZ,ENDPRT       ;nebyl CR ukonci to
        LD      A,(POUTFG)
        LD      E,A
        CALL    POTLF           ;vyzkousej autom.CRLF
        LD      A,CR            ;obnoveni CR
        LD      (HL),0          ;reset hlavy
ENDPRT: POP     HL
        POP     DE
        RET

;=============================================================================
; 17C7          Vystup textu na tiskarnu
;=============================================================================
; IN:   HL- adresa textu
;       B - pocet bytu
; OUT:  A - posl.znak
;       B - zbyla delka
;       HL- posL.znak+1

POTLN:  LD      A,(POUTFG)
        LD      E,A             ;flag tiskarny
NXTPRT: LD      A,(HL)          ;znak
        INC     HL              ;citac
        OR      A
        RET     Z               ;nuly se netisknou
        CP      TAB
        JR      Z,PRTTAB        ;tiskneme TAB
        CP      ' '
        JR      NG,GTSPCE       ;je to znak
        CP      CR
        JR      NZ,NOCR         ;neni to CR
GTSPCE: CALL    POTCH           ;vypis znak ci CR
        CP      CR              ;je tam?
        JR      NZ,CRNOU        ;neni
```

```
        DEC     B               zmensi citac
        RET                     ;a ven

CRNOU:  CALL    PCLMX           ;kontrola poctu
NOCR:   DEC     B               ;citac
        JR      NZ,NXTPRT       ;nenulovy->dal
        LD      A,D             ;posl znak
        RET

PRTTAB: BIT     2,E             ;je povoleno TAB?
        JR      Z,NOCR          ;neni
        LD      A,(PHDPOS)      ;hlava
        CPL
        AND     07H
        INC     A               ;pocet znaku pro tabulaci
        LD      C,A
LOPTAB: LD      A,' '           ;tabulacni znak
        CALL    POTCH           ;vysli jej
        CALL    PCLMX           ;test poctu
        DEC     C               ;zmensi citac TAB
        JR      NZ,LOPTAB
        JR      NOCR            ;pokracuj ve vypisu

;=============================================================================
; 1805          Tisk textu na tiskarnu
;=============================================================================
; IN:   HL- adresa textu
;       BC- pocet bytu
; OUT:  A - posl.znak
;       HL- posl.znak+1

POTBL:  LD      A,(HL)          ;znal
        CALL    POTCH           ;tisk
        DEC     BC
        LD      A,B
        OR      C               ;citac
        LD      A,(HL)          ;znak
        INC     HL              ;adresa+
        JR      NZ,POTBL        ;do BC<>0
        RET

;=============================================================================
; 1811          Inicializace kopletniho SML
;=============================================================================

SGINT:  DI
        LD      HL,IGNSML
        LD      (SEXTA),HL      ;uziv.rutina SML je ignore
        LD      HL,NASTR
        LD      (SEXPA),HL      ;inic.tabulky nastroju

SGINTS: DI
        LD      HL,1F1FH
        LD      (TMPOD),HL      ;inic TMPOD,TMPOC
        XOR     A
        LD      (MSCAL),A       ;nulova transpozice
        LD      A,3             ;budou tri SGC
```

```
        LD      HL,SGPBF1
        LD      DE,SGSYT1
LOPSGI: EX      DE,HL
        LD      (HL),E
        INC     HL
        LD      (HL),D
        INC     HL
        EX      DE,HL
        PUSH    HL
        LD      HL,INISYT
        LD      BC,12
        LDIR
        POP     HL
        LD      BC,32
        ADD     HL,BC
        DEC     A
        JR      NZ,LOPSGI
        CALL    VØFAC           ;vsechno vypnout
        LD      A,ØA7H
        OUT     (Ø1H),A         ;inicializace
        LD      A,ØEH
        OUT     (Ø1H),A         ;CTC
        EI
IGNSML: RET

;=========================================================================
; 1855          Tabulka inic.kodu pro tabulku kanalu SML

INISYT: DB      32              ;delka vyr.pameti
        DB      Ø,Ø             ;citace i/o
        DB      7               ;cas pro HOLD
        DB      Ø               ;hlasitost
        DB      2               ;oktava
        DB      16              ;impl.delka tonu
        DW      Ø               ;adresa dat
        DB      16              ;vyst citac nastroje
        DB      Ø               ;citac delky tonu
        DB      Ø               ;citac casu HOLD
;=========================================================================

;=========================================================================
; 1861          Obsluha SML napojena na kanal #3 CTC
;=========================================================================

MPLAY:  PUSH    AF
        PUSH    HL
        LD      HL,TMPOC        ;zmensi citac tempa
        DEC     (HL)
        CALL    Z,PLAY          ;uz nula-zahraj
        POP     HL
        POP     AF

;-------------------------------------------------------------------------
; 186C          Pouze EI+RETI - lze vyuzit

;-------------------------------------------------------------------------
IGNORI: EI                      ;navrat po INT z RETI

;-------------------------------------------------------------------------
; 186D          Pouze RETI - vyuziva se jako falesne
;-------------------------------------------------------------------------
```

```
IGNORJ: RETI                    ;pouziva se jako falesne

;=============================================================================
; 186F          Samotna obsluha SML
;=============================================================================

PLAY:   PUSH    BC
        PUSH    DE
        PUSH    IX
        LD      B,2             ;tri kanaly
CONTPL: CALL    GTAST           ;zjisti adresu
        PUSH    HL
        POP     IX              ;adresa tabulky SML
        LD      A,(IX+ØCH)      ;citac delky tonu
        OR      A
        JR      Z,LENTØ         ;je nula->
        DEC     (IX+ØCH)        ;zmensit
        JR      NZ,NOLENT       ;jeste nenulovy
        CALL    STDVOL          ;vypnoutkanal B
LENTØ:  LD      C,(IX+Ø4H)      ;vstupni citac
        CALL    COMAN           ;vykona prip.prikaz
        JR      C,NOCMD         ;prikaz nebyl
        PUSH    AF
        LD      HL,(SEXTA)
        CALL    JMPHL           ;skok na uziv.rutinu
        POP     AF
        OR      A
        JR      Z,LENTØ         ;kdyz byl nevykonny
NOCMD:  LD      A,(IX+ØAH)
        OR      A
        JR      Z,L18A7
        CALL    L18C2
        JR      L18B3

L18A7:  LD      A,(IX+ØCH)      ;citac delky tonu
        OR      A
        JR      Z,L18B3         ;je rovenØ->
        DEC     (IX+ØDH)        ;zmensi HOLD
        CALL    Z,STDVOL        ;vypnout pri HOLD=Ø
L18B3:  DEC     B               ;novy kanal
        JP      P,CONTPL        ;az do Ø
        LD      A,(TMPOD)
        LD      (TMPOC),A       ;obnova citace tempa
        POP     IX
        POP     DE
        POP     BC
        RET

L18C2:  LD      A,(IX+ØBH)      ;vyst.citac
        CP      16              ;je to REST?
        RET     Z               ;ano->
        DEC     (IX+ØDH)        ;zmensit HOLD
        JR      NZ,HOLDNØ       ;nenulovy->
        LD      (IX+ØBH),Ø8     ;vyst citac=8
        JR      L18D6

HOLDNØ: CP      8               ;je vyst citac 8/
        RET     Z               ;ano ven
L18D6:  LD      L,(IX+Ø9H)
        LD      H,(IX+ØAH)      ;adresa dat
```

```
        LD      E,(IX+ØBH)      ;vyst citac
        SRL     E               ;/2 - jsou po dvou v bytu
        PUSH    AF
        LD      D,Ø
        ADD     HL,DE           ;adresa bytu
        LD      D,(HL)          ;vyber bytu
        POP     AF
        LD      A,D             ;prubeh do A
        JR      C,LICHE         ;kdyz liche
        RRCA
        RRCA
        RRCA
        RRCA                    ;dej to kam patri
LICHE:  CALL    STVOL           ;nastav hlasitost
        INC     (IX+ØBH)        ;zvetsi citac prubehu
        RET

COMMAM: PUSH    BC
        CALL    OPKOD           ;vyber bytu -Op.kod
        JR      C,NOOPK         ;neni op.kod
        LD      C,A
        RRCA
        RRCA
        RRCA
        RRCA
        AND     7               ;v A je urceni operace
        LD      HL,CMDTBL       ;tabulka adres
        CALL    EXTBL           ;skok na operaci
NOOPK:  POP     BC
        RET

;-------------------------------------------------------------------------------
; 190A          Tabulka adres prikazu Sound Music Language

CMDTBL: DW      NOTA            ;obyc.nota
        DW      NOTAP           ;nota na pianu
        DW      OKTAVA          ;nastaveni oktavy
        DW      VOLUME          ;nast.hlasitosti
        DW      NASTR           ;volba nastroje
        DW      SHOLD           ;nastaveni HOLD
        DW      IMLEN           ;impl.delka tonu
        DW      TEMPO           ;volba tempa
;-------------------------------------------------------------------------------

;===============================================================================
; 191A          Prikaz SML  -  zahraj notu
;===============================================================================

NOTA:   LD      A,C
        AND     ØFH             ;cislo noty
        JR      Z,REST          ;kdyz nula tak Pausa
        DEC     A               ;zmensit na Ø~
        PUSH    BC
        LD      C,(IX+Ø7H)      ;oktava
        CALL    STFRQ           ;nastav frekvenci
        POP     BC
        JR      SETVOL          ;nastav hlasitost noty

REST:   CALL    CHNOFF          ;vypni kanal
        LD      (IX+ØBH),16     ;vyst citac je 16
```

```
        RET                     ;a tim je REST a ven

;=============================================================================
; 1932          Prikaz SML - Nota na pianu podle poradi
;=============================================================================
NOTAP:  CALL    OPKOD           ;vyber dalsiho bytu
        LD      C,A             ;cislo noty
        PUSH    BC
        RES     7,A
        DEC     A
        LD      DE,12
DIVS12: INC     D
        SUB     E
        JR      NC,DIVS12       ;vydeleni dvanacti
        DEC     D
        ADD     A,E
        LD      C,D
        CALL    STFRQ           ;nastav frekvenci
        POP     BC
SETVOL: LD      A,(IX+ØAH)      ;adresa dat
        OR      A
        LD      A,(IX+Ø6H)      ;hlasitost
        JR      Z,ISSØ          ;je sØ->
CHNOFF: LD      A,15            ;hlasitost OFF
ISSØ:   LD      D,C
        CALL    STVOL           ;nastav hlasitost
        BIT     7,D             ;bylo to i udanim delky?
        LD      A,(IX+Ø8H)      ;impl.delka tonu
        CALL    NZ,OPKOD        ;ano bylo vyber delku
        LD      (IX+ØCH),A      ;citac delky tonu
        LD      L,A
        LD      H,Ø
        LD      D,H
        LD      E,L
        LD      A,(IX+Ø5H)      ;cas pro HOLD
        OR      A
        JR      NZ,L197Ø        ;neni nula
        LD      L,A
        JR      L1976

L197Ø:  DEC     A
        JR      Z,L1976
        ADD     HL,DE
        JR      L197Ø

L1976:  LD      B,3
L1978:  SRL     H
        RR      L
        DJNZ    L1978
        INC     L
        LD      (IX+ØDH),L      ;citac casu hold
        LD      (IX+ØBH),Ø      ;vyst citac prubehu
        XOR     A
        DEC     A
        RET

;-----------------------------------------------------------------------------
; 1989 -        Tabulka frekvenci not

TABNOT: DW      7Ø35H
```

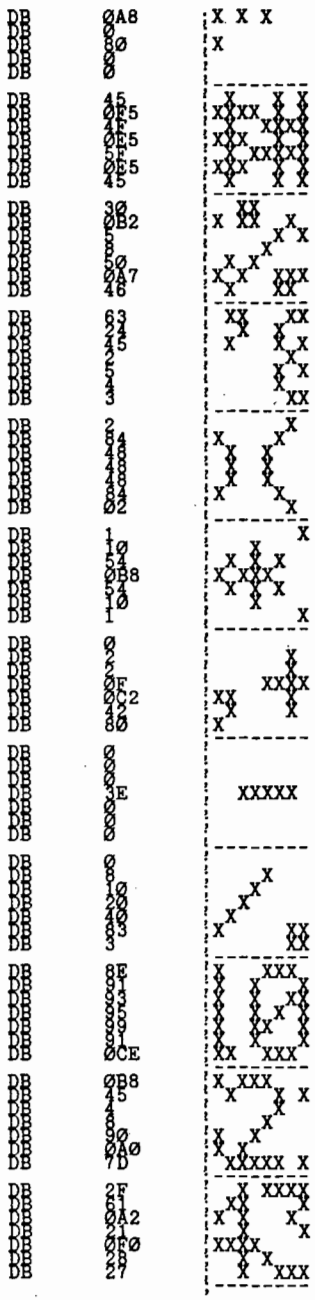
```
        DW      7Ø32H
        DW      ØAØ2FH
        DW      ØFØ2CH
        DW      7Ø2AH
        DW      1Ø28H
        DW      ØDØ25H
        DW      ØBØ23H
        DW      ØBØ21H
        DW      ØCØ1FH
        DW      ØØ1EH
        DW      1C5ØH
;-------------------------------------------------------------------------

;=========================================================================
; 19A1          Prikaz SML - nastaveni oktavy nebo transpozice
;-------------------------------------------------------------------------

OKTAVA: LD      A,C             ;opkod
        BIT     7,A             ;test na vetvici bit
        JR      NZ,TRANS        ;nastavime transpozici
        AND     7               ;orezani
        LD      (IX+Ø7H),A      ;zapis oktavu
        XOR     A
        RET

TRANS:  AND     15              ;orezani
        LD      (MSCAL),A       ;transpozici zapsat
        XOR     A
        RET

;=========================================================================
; 19B4          Prikaz SML - nastaveni hlasitosti
;-------------------------------------------------------------------------

VOLUME: LD      A,C             ;opkod
        AND     ØFH             ;hlasitost
        SUB     ØFH
        NEG                     ;a otocit ji pro SGC
        LD      (IX+Ø6H),A      ;zapsat
        XOR     A
        RET

;=========================================================================
; 19CØ          Prikaz SML - nastaveni nastroje
;-------------------------------------------------------------------------

NASTR:  LD      A,C             ;opkod
        AND     7               ;nastroj
        LD      HL,Ø            ;nastroj sØ
        JR      Z,NSØ           ;opravdu sØ zapis
        DEC     A               ;kod nastroje
        ADD     A,A
        ADD     A,A
        ADD     A,A             ;8bytu na nastroj
        LD      E,A
        LD      D,Ø             ;offset v tabulce
        LD      HL,(SEXPA)      ;tabulka nastroju
        ADD     HL,DE           ;adresa nastroje
NSØ:    LD      (IX+Ø9H),L
        LD      (IX+ØAH),H      ;zapis adresy nastroje
```

```
        XOR     A
        RET

;===============================================================================
; 19DB          Prikaz SML - nastaveni HOLD
;-------------------------------------------------------------------------------

SHOLD:  LD      A,C             ;opkod
        AND     15              ;orezani
        LD      (IX+Ø5H),A      ;zapis HOLD
        XOR     A
        RET

;===============================================================================
; 19E3          Prikaz SML - nastaveni impl.delky not
;-------------------------------------------------------------------------------

IMLEN:  CALL    OPKOD           ;vyber delku not
        LD      (IX+Ø8H),A      ;zapis delku not
        XOR     A
        RET

;===============================================================================
; 19EB          prikaz SML - nastaveni tempa
;-------------------------------------------------------------------------------

TEMPO:  CALL    OPKOD           ;tempo
        LD      E,A
        LD      D,Ø             ;do DE
        LD      HL,1D4CH
        ADD     HL,DE
        CALL    DIVIDS
        LD      A,L
        SRL     H
        RRA                     ;upravy
        LD      (TMPOD),A       ;zapis
        XOR     A
        RET

;===============================================================================
; 1AØ1          Podprogram SML - nastaveni frekvence
;-------------------------------------------------------------------------------

STFRQ:  LD      D,A             ;KOD NOTY
        LD      A,(MSCAL)       ;transpozice
        ADD     A,D             ;uprava
        CP      12              ;je pres notu b?
        JR      C,NOOKT         ;neni pres oktavu
        INC     C               ;o oktavu vyse
        SUB     12              ;!a nota je tez posunuta
NOOKT:  ADD     A,A             ;polozky po dvou bytech
        LD      E,A
        LD      D,Ø             ;offset
        LD      HL,TBLNOT       ;tabulka not
        ADD     HL,DE
        LD      D,(HL)
        INC     HL
        LD      E,(HL)          ;hodnota noty
        LD      A,C
        AND     7               ;oktava
```

```
        JR      Z,STFRQD
L1A1D:  SRL     D
        RR      E
        DEC     A
        JR      NZ,L1A1D

;Pokracovani STFRQ - nastaveni frekvence jiz v SGC

STFRQD: LD      A,E
        RRCA
        RRCA
        RRCA
        RRCA
        LD      C,80H
        CALL    STVFC
        LD      A,D
        JR      EXOUT

;===============================================================================
; 1A31          Vypnuti vsech tri kanalu SGC
;===============================================================================

VOFAC:  LD      B,3             ;tri kanaly
OFFCHB: CALL    STDVOL          ;vypni kanal B
        DJNZ    OFFCHB          ;b*

;===============================================================================
; 1A38          Vypne B-ty kanal SGC
;-------------------------------------------------------------------------------

STDVOL: LD      A,0FH           ;hlasitost=15=OFF

;===============================================================================
; 1A3A          Nastaveni hlasitosti kanalu B SGC
;-------------------------------------------------------------------------------

STVOL:  LD      C,90H           ;zeslabeni=1001B

;-------------------------------------------------------------------------------
; 1A3C          Nastaveni frekvence kanalu B
;-------------------------------------------------------------------------------

STVFC:  AND     0FH
        LD      E,A
        LD      A,B
        RRCA
        RRCA
        RRCA
        AND     60H
        OR      E
        OR      C               ;kompletni ridici kod
EXOUT:  OUT     (20H),A
        RET

;-------------------------------------------------------------------------------
; 1A4A          Vypocet adresy kanalu SGC
;-------------------------------------------------------------------------------

GTAST:  LD      A,B             ;cislo bufferu
        ADD     A,A
```

```
        ADD     A,A
        ADD     A,A
        ADD     A,A             ;A:=a*16
        SUB     B
        SUB     B               ;A:=A*14
        LD      E,A
        LD      D,Ø             ;offset tabulku
        LD      HL,SGSYT1
        ADD     HL,DE           ;adresa tabulky
        RET

;-------------------------------------------------------------------------
; 1A59          Podprogram interpretru SML - zik oper.kodu
;-------------------------------------------------------------------------

OPKOD:  PUSH    BC
        PUSH    IX
        POP     HL
        LD      E,(HL)
        INC     HL
        LD      D,(HL)          ;adresa bufferu
        INC     HL
        LD      B,(HL)          ;delka bufferu
        INC     HL
        LD      A,(HL)          ;vystupni citac
        INC     HL
        CP      (HL)            ;je roven vstupnimu?
        SCF
        JR      Z,EXITOP        ;ano->ven chyba
        LD      C,(HL)          ;vyst.citac
        LD      A,C
        INC     A
        CP      B
        JR      C,VCTOK         ;vst+1<vystupni
        XOR     A
        LD      (HL),A          ;vynulovat vst.citac
VCTOK:  LD      B,Ø
        EX      DE,HL
        ADD     HL,BC           ;adresa polozky
        LD      A,(HL)          ;polozka
        OR      A               ;CY=Ø
EXITOP: POP     BC
        RET
```

```
;=============================================================================
;Cely nasledujici prostor jsou pouze tabulky
        .RADIX  16              ;vse je sestnactkove
;=============================================================================
;Tabulka prubehu systemovych nastroju
NASTR:  DB      01,23,45,67
        DB      89,0AB,0CD,0EF  ;nastroj S1

        DB      0FE,0DC,0BA,98
        DB      76,54,32,1F     ;nastroj S2

        DB      01,23,45,67
        DB      76,54,32,1F     ;nastroj S3

        DB      76,54,32,10
        DB      12,34,56,7F     ;nastroj S4

        DB      02,46,8A,0CE
        DB      24,68,0AC,0EF   ;nastroj S5

        DB      0FC,84,01,23
        DB      45,67,89,0AF    ;nastroj S6

        DB      0FC,84,01,23
        DB      45,43,21,0F     ;nastroj S7
;=============================================================================
;Cely dalsi prostor jsou pouze predlohy znaku

CPATAR: DB      0               ;--------
        DB      8               ;    X
        DB      4               ;   X
        DB      0FE             ;XXXXXXX
        DB      4               ;   X
        DB      8               ;    X
        DB      0               ;
        DB      0               ;--------

        DB      0               ;
        DB      20              ;  X
        DB      40              ; X
        DB      0FE             ;XXXXXXX
        DB      40              ; X
        DB      20              ;  X
        DB      0               ;
        DB      0               ;--------

        DB      10              ;   X
        DB      38              ;  XXX
        DB      54              ; X X X
        DB      10              ;   X
        DB      10              ;   X
        DB      10              ;   X
        DB      10              ;   X
        DB      0               ;--------

        DB      10              ;   X
        DB      10              ;   X
        DB      10              ;   X
        DB      10              ;   X
        DB      54              ; X X X
        DB      38              ;  XXX
        DB      10              ;   X
        DB      0               ;--------

;Zde jsou predlohy ASCII od 20H
;jejich tvar ale neni uplne normalni
;bylo pouzito komprimacniho programu

CPATA:  DB      80              ;--------
        DB      80              ;X
        DB      80              ;X
        DB      80              ;X
        DB      0               ;
        DB      0               ;
        DB      80              ;X
                                ;--------
        DB      28              ;  X X
        DB      28              ;  X X
```

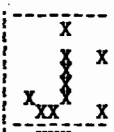

DB ØA8
DB Ø
DB 8Ø
DB Ø
DB Ø
DB 45
DB ØE5
DB 45
DB ØE5
DB 45
DB ØE5
DB 45
DB 3Ø
DB ØB2
DB 5
DB 5Ø
DB ØA7
DB 46
DB 63
DB 24
DB 45
DB 2
DB 5
DB 4
DB 3
DB 2
DB 84
DB 48
DB 48
DB 48
DB 84
DB Ø2
DB 1
DB 1Ø
DB 54
DB ØB8
DB 54
DB 1Ø
DB 1
DB Ø
DB 5
DB Ø
DB ØF
DB ØC2
DB 42
DB 8Ø
DB Ø
DB Ø
DB Ø
DB 3E
DB Ø
DB Ø
DB Ø
DB Ø
DB 8
DB 1Ø
DB 2Ø
DB 4Ø
DB 83
DB 3
DB 8E
DB 91
DB 93
DB 95
DB 99
DB 91
DB ØCE
DB ØB8
DB 45
DB 4
DB 8
DB 8Ø
DB ØAØ
DB 7D
DB 2F
DB 51
DB ØA2
DB 21
DB ØFØ
DB 28
DB 27

```
        DB      0BEH
        DB      20H
        DB      3CH
        DB      85H
        DB      43H
        DB      42H
        DB      9CH

        DB      0F9H
        DB      0AH
        DB      14H
        DB      27H
        DB      44H
        DB      41H
        DB      43H

        DB      0CEH
        DB      31H
        DB      31H
        DB      0EEH
        DB      31H
        DB      31H
        DB      8EH

        DB      1H
        DB      32H
        DB      32H
        DB      1H
        DB      30H
        DB      30H
        DB      1H

        DB      20H
        DB      48H
        DB      88H
        DB      0H
        DB      86H
        DB      42H
        DB      24H

        DB      0H
        DB      0H
        DB      3EH
        DB      81H
        DB      3EH
        DB      0H
        DB      0H

        DB      74H
        DB      8AH
        DB      9H
        DB      10H
        DB      21H
        DB      21H
        DB      24H

CPATC:  DB      0CEH
        DB      31H
        DB      2DH
        DB      0F5H
        DB      35H
        DB      2EH

        DB      79H
        DB      0C6H
        DB      46H
        DB      7AH
        DB      47H
        DB      0C6H
        DB      7AH

        DB      0C7H
        DB      28H
        DB      18H
        DB      18H
        DB      18H
        DB      28H
        DB      0C7H

        DB      0FFH
        DB      21H
        DB      21H
        DB      0BDH
        DB      21H
        DB      21H
        DB      3FH
```

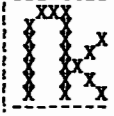
```
DB      77      ;  XXX XXX
DB      8C      ; X   X  X
DB      84      ; X      X
DB      ØBF     ; X XXXXXX
DB      8C      ; X      X
DB      8C      ; X      X
DB      7C      ;  XXXXX
DB      ØD1     ; XX X    X
DB      91      ; X  X    X
DB      91      ; X  X    X
DB      9F      ; X  XXXX X
DB      91      ; X  X    X
DB      91      ; X  X    X
DB      ØD1     ; XX X    X
DB      9D      ; X  XX   X
DB      8       ;    X
DB      8       ;    X
DB      8       ;    X
DB      8       ;    X
DB      8       ;    X
DB      48      ;  X  X
DB      ØB1     ; X XX    X
DB      8       ;    X
DB      9       ;    X  X
DB      ØA      ;    X X
DB      ØC      ;    XX
DB      ØA      ;    X X
DB      9       ;    X  X
DB      ØF8     ; XXXXX
DB      63      ;  XX   XX
DB      77      ;  XXX XXX
DB      6B      ;  XX X XX
DB      6B      ;  XX X XX
DB      ØE3     ; XXX   XX
DB      63      ;  XX   XX
DB      63      ;  XX   XX
DB      74      ;  XXX X
DB      8C      ; X   XX
DB      8E      ; X   XXX
DB      8D      ; X   XX X
DB      8C      ; X   XX
DB      8C      ; X   XX
DB      74      ;  XXX X
DB      ØDE     ; XX XXXX
DB      31      ;   XX   X
DB      3E      ;   XXXXX
DB      ØB0     ; X XX
DB      30      ;   XX
DB      ØB0     ; X XX
DB      ØF9     ; XXXXX  X
DB      18      ;    XX
DB      18      ;    XX
DB      7A      ;  XXXX X
DB      ØD2     ; XX X  X
DB      ØCA     ; XX  X X
DB      45      ;  X   X X
DB      ØF7     ; XXXX XXX
DB      18      ;    XX
DB      18      ;    XX
DB      17      ;    XXX
DB      10      ;    X
DB      10      ;    X
DB      1F      ;    XXXX
DB      63      ;  XX   XX
DB      62      ;  XX   X
DB      62      ;  XX   X
DB      62      ;  XX   X
DB      62      ;  XX   X
DB      ØA2     ; X X   X
DB      1C      ;    XXX
DB      8C      ; X   XX
DB      8C      ; X   XX
DB      8C      ; X   XX
DB      ØAC     ; X X XX
DB      ØAC     ; X X XX
DB      ØAA     ; X X X X
DB      51      ;  X X   X
```

```
DB      31          |   XX   X  |
DB      31          |   XX   X  |
DB      2A          |  X X X X  |
DB      44          | X   X   X |
DB      8A          |X   X X   X|
DB      91          |X  X   X  X|
DB      91          |X  X   X  X|
DB      7E          |  XXXXXX   |
DB      6           |     XX    |
DB      0A          |    X X X  |
DB      11          |   X   X  X|
DB      20          |  X        |
DB      40          | X         |
DB      7C          | XXXXX     |
DB      7           |     XXX   |
DB      4           |     X     |
DB      84          |X    X     |
DB      44          | X   X     |
DB      27          |  X  XXX   |
DB      14          |   X X     |
DB      7           |     XXX   |
DB      38          |  XXX      |
DB      39          |X  XXX   X |
DB      48          | X  X      |
DB      8           |    X      |
DB      8           |    X      |
DB      8           |    X      |
DB      38          |  XXX      |
DB      1           |       X   |
DB      2           |      X    |
DB      4           |     X     |
DB      0           |           |
DB      0           |           |
DB      0           |           |
DB      0F8         | XXXXX     |
DB      8           |    X      |
DB      4           |     X     |
DB      0C2         | XX    X   |
DB      20          |  X        |
DB      0E0         | XXX       |
DB      20          |  X        |
DB      0E0         | XXX       |
DB      40          | X         |
DB      40          | X         |
DB      0F1         | XXXX   X  |
DB      48          | X  X      |
DB      49          | X  X  X   |
DB      4A          | X  X X X  |
DB      0F1         | XXXX   X  |
DB      20          |  X        |
DB      20          |  X        |
DB      0E7         | XXX  XXX  |
DB      28          |  X X      |
DB      28          |  X X      |
DB      28          |  X X      |
DB      0E7         | XXX  XXX  |
DB      80          |X          |
DB      0           |           |
DB      1C          |   XXX     |
DB      0A3         | X X   XX  |
DB      3F          |  XXXXXX   |
DB      21          |  X    X   |
DB      1C          |   XXX     |
DB      0           |           |
DB      71          |  XXX   X  |
DB      91          |X  X  X  X |
DB      93          |X  X  X  XX|
DB      71          |  XXX   X  |
DB      11          |   X    X  |
DB      61          |  XX    X  |
DB      90          |X  X  X    |
DB      10          |   X       |
DB      9C          |X  XXX     |
DB      92          |X  X  X    |
DB      92          |X  X  X    |
DB      92          |X  X  X    |
DB      0D2         | XX X  X   |
```

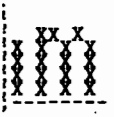
```
DB      8
DB      0
DB      9
DB      8
DB      8
DB      8
DB      18
DB      31
```



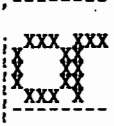
```
DB      0C8
DB      48
DB      49
DB      4A
DB      4C
DB      48
DB      49
```



```
DB      0
DB      0
DB      34
DB      0AA
DB      0AA
DB      0AA
DB      0AA
```

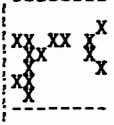


```
DB      0
DB      0
DB      77
DB      6C
DB      6C
DB      6C
DB      74
```

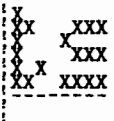


```
DB      0
DB      0DC
DB      52
DB      52
DB      0DC
DB      50
DB      50
```

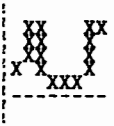
```
DB      0
DB      1
DB      0DA
DB      62
DB      41
DB      0C0
DB      40
```



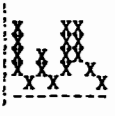
```
DB      0
DB      80
DB      0C7
DB      88
DB      81
DB      0A0
DB      0CF
```



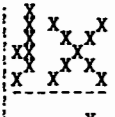
```
DB      0
DB      0
DB      63
DB      62
DB      62
DB      0A2
DB      1C
```



```
DB      0
DB      0
DB      6C
DB      6C
DB      0AC
DB      0AA
DB      51
```

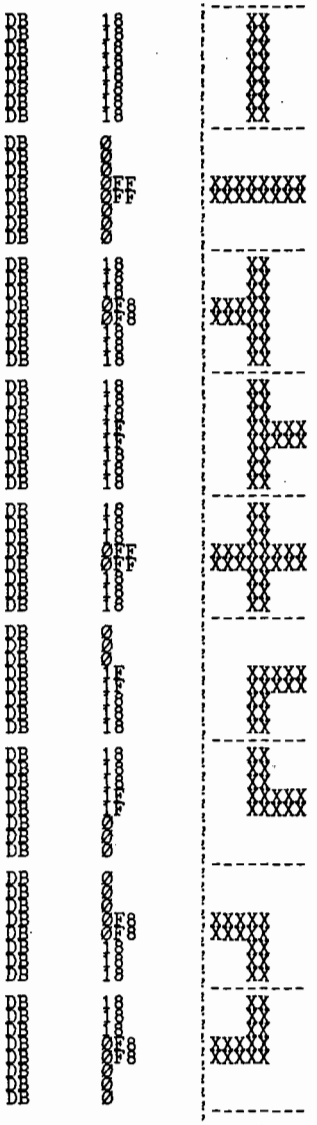


```
DB      0
DB      40
DB      51
DB      4A
DB      0C4
DB      4A
DB      91
```



```
DB      0
DB      2
DB      7E
DB      0A
DB      11
DB      20
DB      7D
```

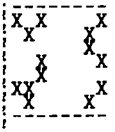
```
        DB      41
        DB      42
        DB      42
        DB      4
        DB      42
        DB      42
        DB      41

        DB      10
        DB      8
        DB      8
        DB      44
        DB      28
        DB      08
        DB      10

        DB      8
        DB      18
        DB      2A
        DB      4D
        DB      38
        DB      18
        DB      8

CPATGØ: DB      ØFF
        DB      ØFF
        DB      ØFF
        DB      ØFF
        DB      ØFF
        DB      ØFF
        DB      ØFF
        DB      ØFF

        DB      38
        DB      44
        DB      82
        DB      82
        DB      82
        DB      44
        DB      38
        DB      Ø

        DB      10
        DB      38
        DB      7C
        DB      ØFE
        DB      ØFE
        DB      Ø54
        DB      10
        DB      38

        DB      10
        DB      38
        DB      38
        DB      ØD6
        DB      ØFE
        DB      54
        DB      10
        DB      38

        DB      Ø
        DB      Ø
        DB      60
        DB      80
        DB      Ø
        DB      Ø
        DB      Ø
        DB      Ø

        DB      18
        DB      18
        DB      18
        DB      ØFF
        DB      ØFF
        DB      Ø
        DB      Ø
        DB      Ø

        DB      Ø
        DB      Ø
        DB      Ø
        DB      ØFF
        DB      ØFF
        DB      18
        DB      18
        DB      18
```


DB 18
DB 18
DB 18
DB 18
DB 18
DB 18
DB 18
DB 18
DB 0
DB 0
DB 0FFH
DB 0FFH
DB 0
DB 0
DB 0
DB 18
DB 18
DB 18
DB 0F8H
DB 0F8H
DB 18
DB 18
DB 18
DB 18
DB 18
DB 18
DB 1FH
DB 1FH
DB 18
DB 18
DB 18
DB 0
DB 0
DB 0
DB 1FH
DB 1FH
DB 18
DB 18
DB 18
DB 18
DB 0
DB 0
DB 0
DB 0
DB 0F8H
DB 0F8H
DB 18
DB 18
DB 18
DB 18
DB 0F8H
DB 0F8H
DB 0
DB 0
DB 0

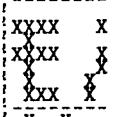
```
        DB      Ø               ;--------
        DB      Ø               ;
        DB      Ø               ;
        DB      Ø               ;
        DB      ØFF             ;XXXXXXXX
        DB      ØFF             ;XXXXXXXX
        DB      Ø               ;
        DB      Ø               ;--------
        DB      Ø               ;
        DB      Ø               ;
        DB      Ø               ;
        DB      Ø               ;
        DB      Ø               ;
        DB      Ø               ;
        DB      ØFF             ;XXXXXXXX
        DB      ØFF             ;XXXXXXXX
        DB      Ø               ;--------
        DB      Ø               ;
        DB      Ø               ;
        DB      Ø               ;
        DB      Ø               ;
        DB      ØFF             ;XXXXXXXX
        DB      ØFF             ;XXXXXXXX
        DB      ØFF             ;XXXXXXXX
        DB      ØFF             ;XXXXXXXX
        DB      Ø               ;--------
        DB      Ø               ;
        DB      Ø               ;
        DB      ØFF             ;XXXXXXXX
        DB      ØFF             ;XXXXXXXX
        DB      ØFF             ;XXXXXXXX
        DB      ØFF             ;XXXXXXXX
        DB      ØFF             ;XXXXXXXX
        DB      ØFF             ;XXXXXXXX
        DB      3Ø              ;  XX
        DB      3Ø              ;  XX
        DB      3Ø              ;  XX
        DB      3Ø              ;  XX
        DB      3Ø              ;  XX
        DB      3Ø              ;  XX
        DB      3Ø              ;  XX
        DB      3Ø              ;  XX
        DB      ØCØ             ;XX
        DB      ØCØ             ;XX
        DB      ØCØ             ;XX
        DB      ØCØ             ;XX
        DB      ØCØ             ;XX
        DB      ØCØ             ;XX
        DB      ØCØ             ;XX
        DB      ØCØ             ;XX
;Tato predloha se tez pouziva jako barva kurzoru
CURCOL: DB      ØFØ             ;XXXX
        DB      ØFØ             ;XXXX
        DB      ØFØ             ;XXXX
        DB      ØFØ             ;XXXX
        DB      ØFØ             ;XXXX
        DB      ØFØ             ;XXXX
        DB      ØFØ             ;XXXX
        DB      ØFØ             ;XXXX
        DB      ØFC             ;XXXXXX
        DB      ØFC             ;XXXXXX
        DB      ØFC             ;XXXXXX
        DB      ØFC             ;XXXXXX
        DB      ØFC             ;XXXXXX
        DB      ØFC             ;XXXXXX
        DB      ØFC             ;XXXXXX
        DB      ØFC             ;XXXXXX
        DB      1               ;       X
        DB      3               ;      XX
        DB      7               ;     XXX
        DB      ØF              ;    XXXX
        DB      1F              ;   XXXXX
        DB      3F              ;  XXXXXX
        DB      7F              ; XXXXXXX
        DB      FF              ;XXXXXXXX
        DB      ØFF             ;XXXXXXXX
```

```
        DB      0FF
        DB      7E
        DB      3C
        DB      18
        DB      0
        DB      0
        DB      0
        DB      0

        DB      0
        DB      0
        DB      0
        DB      0
        DB      18
        DB      3C
        DB      7E
        DB      0FF

        DB      80
        DB      0C0
        DB      0E0
        DB      0F0
        DB      0F8
        DB      0E0
        DB      0C0
        DB      80

        DB      3
        DB      0C
        DB      10
        DB      20
        DB      40
        DB      40
        DB      80
        DB      80

        DB      80
        DB      80
        DB      40
        DB      40
        DB      20
        DB      10
        DB      0C
        DB      03

        DB      0C0
        DB      30
        DB      8
        DB      4
        DB      2
        DB      2
        DB      1
        DB      1

        DB      1
        DB      1
        DB      2
        DB      2
        DB      4
        DB      8
        DB      30
        DB      0C0

;Dalsi predlohy jsou opet zmatene - jde o znaky
; mezinarodni abecedy A0-BF

CPATL:  DB      8
        DB      8
        DB      8
        DB      8
        DB      8
        DB      8
        DB      8
        DB      8

        DB      5
        DB      0A
        DB      16
        DB      8
        DB      0F
        DB      8
        DB      89

        DB      0C1
        DB      7
        DB      0C4
        DB      87
        DB      84
        DB      7
```

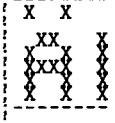
```
DB      0A1
DB      42
DB      3
DB      51
DB      50
DB      0C1
DB      42
```



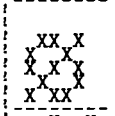
```
DB      0
DB      0F1
DB      40
DB      0F1
DB      41
DB      42
DB      72
```



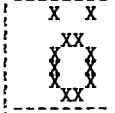
```
DB      48
DB      0
DB      31
DB      49
DB      79
DB      49
DB      49
```



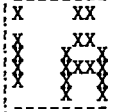
```
DB      0
DB      0
DB      34
DB      48
DB      24
DB      24
DB      58
```



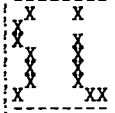
```
DB      12
DB      0
DB      0C
DB      12
DB      12
DB      12
DB      0C
```



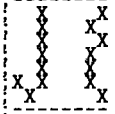
```
DB      86
DB      0
DB      86
DB      89
DB      8F
DB      89
DB      9
```



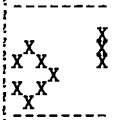
```
DB      44
DB      80
DB      84
DB      44
DB      44
DB      44
DB      83
```



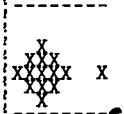
```
DB      21
DB      22
DB      21
DB      22
DB      22
DB      0A2
DB      41
```



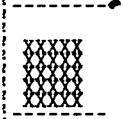
```
DB      0
DB      1
DB      41
DB      0A1
DB      10
DB      0A0
DB      40
```



```
DB      0
DB      0
DB      20
DB      70
DB      0F9
DB      70
DB      20
```

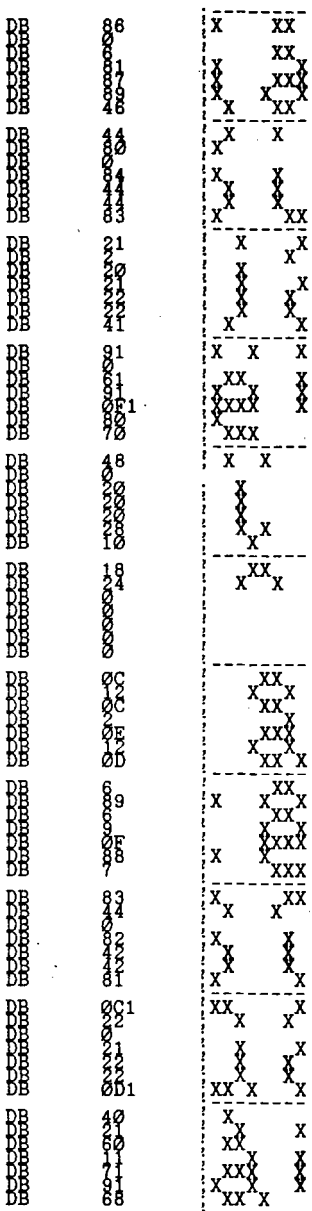


```
DB      0
DB      0
DB      7C
DB      7C
DB      7C
DB      7C
DB      7C
```



```
DB      8
DB      Ø
DB      Ø
DB      8
DB      8
DB      8
DB      8

DB      87
DB      8
DB      6
DB      9
DB      6
DB      81
DB      ØE

DB      41
DB      82
DB      45
DB      44
DB      47
DB      4C
DB      4B

DB      84
DB      84
DB      4
DB      4
DB      23
DB      ØC4

DB      Ø
DB      1Ø
DB      2Ø
DB      41
DB      82
DB      2
DB      51

DB      88
DB      51
DB      2Ø
DB      ØF8
DB      2Ø
DB      21
DB      2Ø

DB      1C
DB      14
DB      1Ø
DB      1Ø
DB      1Ø
DB      5Ø
DB      7Ø

DB      8
DB      8
DB      8E
DB      8
DB      8
DB      8
DB      8

DB      86
DB      8
DB      8
DB      ØE
DB      9
DB      9
DB      16

DB      ØC4
DB      Ø
DB      Ø
DB      Ø
DB      Ø
DB      Ø
DB      Ø

DB      ØC3
DB      Ø
DB      Ø
DB      Ø
DB      Ø
DB      Ø
DB      Ø
```

```
DB      10          |    X    |
DB      10          |    X    |
DB      10          |    X    |
DB      0F0         |  XXXX   |
DB      90          |  X  X   |
DB      90          |  X  X   |
DB      90          |  X  X   |
-----------------------------------
DB      88          |  X   X  |
DB      88          |  X   X  |
DB      88          |  X   X  |
DB      0F8         |  XXXXX  |
DB      20          |    X    |
DB      20          |    X    |
DB      20          |    X    |
-----------------------------------
DB      0           |         |
DB      0           |         |
DB      0           |         |
DB      10          |    X    |
DB      28          |   X X   |
DB      44          |  X   X  |
DB      7C          |  XXXXX  |
-----------------------------------
DB      0           |         |
DB      0           |         |
DB      0           |         |
DB      0           |         |
DB      0           |         |
DB      0           |         |
DB      80          | X       |
-----------------------------------
DB      5           |     X X |
DB      0A          |    X X  |
DB      6           |     XX  |
DB      81          | X      X|
DB      87          | X    XXX|
DB      9           | X   X  X|
DB      88          | X   XX  |
-----------------------------------
DB      1           |       X |
DB      82          | X     X |
DB      3           |      XX |
DB      4           |     X   |
DB      87          | X    XXX|
DB      4           |     X   |
DB      0A3         | X X   XX|
-----------------------------------
DB      0A0         | X X     |
DB      41          |  X     X|
DB      2           |     X   |
DB      42          |  X    X |
DB      0A1         | X X    X|
DB      0A1         | X X    X|
DB      0A2         | X X   X |
-----------------------------------
DB      0           |         |
DB      1           |       X |
DB      0D0         | XX X   X|
DB      29          |  X X X  |
DB      0F8         | XXXXX   |
DB      0A0         | X X     |
DB      78          |  XXXX   |
-----------------------------------
DB      48          |  X  X   |
DB      0           |         |
DB      30          |   XX    |
DB      8           |    X    |
DB      38          |   XXX   |
DB      48          |  X  X   |
DB      34          |   XX X  |
-----------------------------------
DB      0           |         |
DB      0           |         |
DB      0           |         |
DB      34          |   XX X  |
DB      26          |  X  XX  |
DB      14          |   X X   |
DB      2C          |  X XX   |
-----------------------------------
DB      0           |         |
DB      12          |    X  X |
DB      0           |         |
DB      0C          |     XX  |
DB      12          |    X  X |
DB      12          |    X  X |
DB      8C          | X   XX  |
-----------------------------------
```


DB 86
DB 0
DB 0
DB 81
DB 87
DB 89
DB 46
DB 44
DB 80
DB 0
DB 84
DB 44
DB 44
DB 63
DB 21
DB 2
DB 20
DB 21
DB 22
DB 41
DB 91
DB 0
DB 61
DB 81
DB 0F1
DB 90
DB 70
DB 48
DB 0
DB 20
DB 20
DB 20
DB 18
DB 10
DB 18
DB 24
DB 0
DB 0
DB 0
DB 0
DB 0
DB 0C
DB 12
DB 0C
DB 2
DB 0E
DB 12
DB 0D
DB 6
DB 89
DB 6
DB 9
DB 0F
DB 88
DB 7
DB 83
DB 44
DB 0
DB 82
DB 42
DB 42
DB 81
DB 0C1
DB 32
DB 0
DB 31
DB 32
DB 32
DB 0D1
DB 40
DB 31
DB 60
DB 11
DB 71
DB 91
DB 68

```
        DB      20
        DB      10
        DB      30
        DB      48
        DB      78
        DB      40
        DB      38

        DB      10
        DB      8
        DB      0
        DB      10
        DB      10
        DB      14
        DB      8

        DB      8
        DB      4
        DB      0
        DB      0C
        DB      12
        DB      12
        DB      8C

        DB      4
        DB      82
        DB      80
        DB      89
        DB      49
        DB      9
        DB      86

        DB      3
        DB      80
        DB      43
        DB      44
        DB      83
        DB      0
        DB      0C7

        DB      80
        DB      1
        DB      82
        DB      82
        DB      81
        DB      0A0
        DB      43

        DB      0
        DB      0
        DB      0
        DB      1
        DB      0
        DB      0
        DB      0

        DB      0
        DB      50
        DB      50
        DB      68
        DB      40
        DB      40
        DB      40

        DB      8
        DB      10
        DB      0
        DB      0
        DB      0
        DB      0
        DB      0

CPATG1: DB      33
        DB      33
        DB      0CC
        DB      0CC
        DB      33
        DB      33
        DB      0CC
        DB      0CC

        DB      38
        DB      7C
        DB      0FE
        DB      0FE
        DB      0FE
        DB      7C
        DB      38
        DB      0
```

```
DB      6C
DB      0FE
DB      0FE
DB      0FE
DB      0FE
DB      7C
DB      38
DB      10
```



```
DB      10
DB      38
DB      7C
DB      0FE
DB      0FE
DB      7C
DB      38
DB      10
```



```
DB      10
DB      20
DB      20
DB      10
DB      8
DB      4
DB      4
DB      8A
```



```
DB      1
DB      2
DB      4
DB      8
DB      10
DB      20
DB      40
DB      80
```



```
DB      80
DB      40
DB      20
DB      10
DB      8
DB      4
DB      2
DB      1
```



```
DB      81
DB      41
DB      24
DB      18
DB      24
DB      42
DB      81
```



```
DB      0
DB      0
DB      0FF
DB      0
DB      0
DB      0FF
DB      0
DB      0
```



```
DB      18
DB      18
DB      0F8
DB      18
DB      18
DB      0F8
DB      18
DB      18
```



```
DB      18
DB      18
DB      1F
DB      18
DB      18
DB      1F
DB      18
DB      18
```



```
DB      18
DB      18
DB      0FF
DB      18
DB      18
DB      0FF
DB      18
DB      18
```

```
DB      1
DB      3
DB      7
DB      0F
DB      1F
DB      3F
DB      7F
DB      0FF

DB      0FF
DB      7F
DB      3F
DB      1F
DB      0F
DB      07
DB      03
DB      1

DB      80
DB      C0
DB      0E0
DB      0F0
DB      0F8
DB      0FC
DB      0FE
DB      0FF

DB      0FF
DB      0FE
DB      0FC
DB      0F8
DB      0F0
DB      0E0
DB      0C0
DB      80

DB      0
DB      0
DB      0FF
DB      0FF
DB      0
DB      0
DB      0
DB      0

DB      0FF
DB      0FF
DB      0
DB      0
DB      0
DB      0
DB      0
DB      0

DB      0FF
DB      0FF
DB      0FF
DB      0FF
DB      0
DB      0
DB      0
DB      0

DB      0FF
DB      0FF
DB      0FF
DB      0FF
DB      0FF
DB      0FF
DB      0
DB      0

DB      0C
DB      0C
DB      0C
DB      0C
DB      0C
DB      0C
DB      0C
DB      0C

DB      3
DB      3
DB      3
DB      3
DB      3
DB      3
DB      3
DB      3
```

```
DB      0E
DB      0F
DB      0F
DB      0F
DB      0F
DB      0F
DB      0F
DB      0F

DB      3F
DB      3F
DB      3F
DB      3F
DB      3F
DB      3F
DB      3F
DB      3F

DB      0F0
DB      0F0
DB      0F0
DB      0F0
DB      0F
DB      0F
DB      0F
DB      0F

DB      0E
DB      0F
DB      0F
DB      0F
DB      0F0
DB      0F0
DB      0F0
DB      0F0

DB      0FF
DB      07E
DB      3C
DB      18
DB      18
DB      3C
DB      7E
DB      0FF

DB      81
DB      0C3
DB      0E7
DB      0FF
DB      0FF
DB      0E7
DB      0C3
DB      81

DB      3
DB      0E
DB      1F
DB      3F
DB      7F
DB      7F
DB      0FF
DB      0FF

DB      0FF
DB      0FF
DB      07F
DB      7F
DB      3F
DB      1F
DB      0F
DB      3

DB      0C0
DB      0F0
DB      0F8
DB      0FC
DB      0FE
DB      0FF
DB      0FF
DB      0FF

DB      0FF
DB      0FF
DB      0FE
DB      0FE
DB      0FC
DB      0F8
DB      0F0
DB      0C0
```

```
;A to je konec znaku a zaroven konec pameti
;Pouze zbyly tri byty kam se uz nic nevesio
        DB      0,0,0
```

```
;=============================================================================
; 7000          Definice potrebnych systemovych promennych MONITOR ROM
;=============================================================================

        ORG     07000H

;-----------------------------------------------------------------------------
; 7000          SYSTEM TABLE
;-----------------------------------------------------------------------------

IVCTC0: DW      IGNORI          ;CTC#0
IVCTC1: DW      MPLAY           ;CTC#1
IVCTC2: DW      IGNORI          ;CTC#2
IVCTC3: DW      CTC3SP          ;CTC#3

IVCTC6  JP      0               ;vektor RST  30H
IVCTC7  JP      0               ;vektor RST  38H

SCCDTA  DW      SCCDT           ;adresa CTRL tabulky

SMEMTA: DW      7000H           ;zacatek RAM
SMEMEA: DW      8000H           ;konec ram
SUMMTA: DW      7300H           ;zacatek uziv.RAM
SUMMEA: DW      8000H           ;konec uziv.RAM

SVSSW:  DB      0               ;prepinac

STDLY   DB      21H             ;prenosova rychlost

;-----------------------------------------------------------------------------
; 701A          KEYBOARD INFORMATION TABLE
;-----------------------------------------------------------------------------

KINFLG  DB      94H             ;flag klavesnice
KBCTET  DW      09E7H           ;konverzni tabulka

ASWN01: DB      0               ;data z ATTACK1
JOYDR1  DB      0               ;data z JOY1
ASWN02  DB      0               ;data z ATTACK1
JOYDR2  DB      0               ;data z JOY2

JOYPRC  DW      EXRET           ;preruseni JOY
ASWPRC  DW      EXRET           ;preruseni ATTACK

RSTPRC  DW      IGNORB          ;preruseni RESET
HLTPRC  DW      IGNORB          ;preruseni HALT

PHSKAD  DB      0               ;adr.posl.stisk.tl.
LSKYST  DB      0               ;posl.stis.FUNC&spol
LKYADR  DB      0               ;adr.posl.sch.tl.

CHATIT  DB      5               ;citac omylu

ARPSTI  DB      1EH             ;autor.start
ARPSTW  DB      1EH             ;autor.start
ARPITI  DB      4               ;autor.interval
ARPITW  DB      4               ;autor.interval

KBUFTA  DW      KEYBUF          ;buf.klavesnice
```

```
KDTPPT  DB      Ø               ;vystupni citac
KDTGPT  DB      Ø               ;vstupni citac

KBFSIZ  DB      3FH             ;delka KBUF

KINWTM  DW      Ø               ;doba WAIT
KINWTH  DB      32              ;interv.WAIT

TERMAL  DB      Ø               ;ukoncovac

BELKF   DB      4               ;frekvence stisku
BELKFL  DB      2               ;delka zvuku

;-------------------------------------------------------------------------
; 7Ø3C          EVENT CONTROL TABLE
;-------------------------------------------------------------------------

EVMGFG  DB      Ø               ;event flag
EVMXNO  DB      Ø               ;pocet eventu
EVIFTA  DW      Ø               ;adr.tabulky

UEVMGF  DB      Ø               ;flag#3

UPCTBI  DB      32              ;interv.
UPCTBW  DB      32              ;interv.
UPCNT   DW      Ø               ;dopredny citac

DWCTBI  DB      32              ;interval
DWCTBW  DB      32              ;interval
DWCNT   DW      Ø               ;zpetny citac

CLKBTW  DB      32              ;interv.
CLOCKS  DB      Ø               ;hodiny
CLOCKM  DB      Ø
CLOCKH  DB      Ø

ALMTM   DB      -1              ;ALARM
ALMTH   DB      -1

ALMPRC  DW      EXRET           ;proces ALARMU
EVHPRC  DW      EXRET           ;proces dne

SPRPRC  DW      EXRET           ;proces spritu

SPSTUS  DB      8ØH             ;status spritu

SPSTEP  DB      2               ;krok spritu
SPSTPC  DB      1

BELKC   DB      Ø               ;citac zvuku
BELC    DB      Ø

;-------------------------------------------------------------------------
; 7Ø5A          POUT TABLE
;-------------------------------------------------------------------------

POUTFG  DB      7               ;flag tiskarny
PMXCLM  DB      5Ø              ;pocet znaku
PHDPOS  DB      6               ;pozice hlavy
```

```
;-------------------------------------------------------------------------
; 705D          MUSIC TABLE
;-------------------------------------------------------------------------

SEXTA   DW      IGNSML          ;proces uziv.

SEXPA   DW      NASTR           ;nastroje

TMPOD   DB      1FH             ;tempo
TMPOC   DB      1FH

MSCAL   DB      0               ;transpozice

SGSYT1  DS      14              ;udaje interpretru
SGSYT2  DS      14
SGSYT3  DS      14

;-------------------------------------------------------------------------
; 7082          ACMT TABLE
;-------------------------------------------------------------------------

SYSFMT  DB      0               ;flag CMT

PUTPMT
GETPMT  DB      0               ;i/o citac

AVILMT  DB      0               ;delka datx

BSIZMT  DB      -1              ;delka buf

RWBFMT  DW      0FF00H          ;adresa buf

;-------------------------------------------------------------------------
; 7094          PROCESS SCREEN TABLE
;-------------------------------------------------------------------------

DIFLGA  DB      0               ;flag

CODTLA  DW      3800H           ;predlohy II
CCOTLA  DW      3B80H           ;barvy znaku
CPATLA  DW      2800H           ;predlohy I
SATTLA  DW      3B00H           ;sprity vlastnosti
SPATLA  DW      2000H           ;sprity predlohy

BDCOLA  DB      0E1H            ;barva obrazovky

UPRMVA  DB      0               ;VIEW
LFTMVA  DB      0
HEITVA  DB      18H
WIDTVA  DB      20H

HITDA   DB      18H             ;rozmer obrazovky
WIDTDA  DB      20H

CURSYA  DB      0               ;kurzor
CURSXA  DB      0

CURADA  DW      0               ;adresa kurzoru
CCUCRA  DB      0               ;znak na kurzru
DISPCA  DB      10H             ;cas videni
```

```
ERSECA  DB      10H             ;cas schovani

BELFA   DB      8               ;frekv.BEL
BELFLA  DB      0EH             ;delka BEL

GRFLA   DB      9               ;gr.flag

GCURYA  DB      0               ;gaf.x,y
GCURXA  DB      0

GPLPRA  DW      0               ;proces GII
GIMPRA  DW      0               ;proces uziv.GII
MPLPRA  DW      0               ;proces MCOLOR

;-------------------------------------------------------------------------
; 70B8          SPARE SCREEN TABLE
;-------------------------------------------------------------------------

DIFLGP  DB      0
CODTLP  DW      0
CCOTLP  DW      0
CPATLP  DW      0
SATTLP  DW      0
SPATLP  DW      0
BDCOLA  DB      0
UPRMVP  DB      0
LFTMVP  DB      0
HEITVP  DB      0
WIDTVP  DB      0
HITDP   DB      0
WIDTDP  DB      0
CURSYP  DB      0
CURSXP  DB      0
CURADP  DW      0
CCUCRP  DB      0
DISPCP  DB      0
ERSECP  DB      0
BELFP   DB      0
BELFLP  DB      0
GRFLP   DB      0
GCURYP  DB      0
GCURXP  DB      0
GPLPRP  DW      0
GIMPRP  DW      0
MPLPRP  DW      0

;-------------------------------------------------------------------------
; 70DC          FUNCTION KEY TABLE
;-------------------------------------------------------------------------

FKMGFG  DB      0               ;flag FCTN
FKIFTA  DW      0               ;adresa tabulky

;-------------------------------------------------------------------------
; 70DF          SYSTEM BUFFER
;-------------------------------------------------------------------------

KEYBUF  DS      64              ;klavesnice

ACMTBF  DS      64              ;magnetofon
```

```
SGPBF1  DS      32              ;SML1
SGPBF2  DS      32              ;SML2
SGPBF3  DS      32              ;SML3

;-------------------------------------------------------------------------
; 71BF          SPRITE MANAGEMENT TABLE
;-------------------------------------------------------------------------

MXPSND  DB      ØC              ;pocet pohyb.se
SPIFTA  DW      SPIFTB          ;adresa pohybu

SPLINK  DS      32              ;spojovaci mapa

;-------------------------------------------------------------------------
; 71E2          SPRITE TABLE
;-------------------------------------------------------------------------

SPIFTB  DS      12*5            ;pohyby


;-------------------------------------------------------------------------
; 721E          nedefinovane prostory
;-------------------------------------------------------------------------

NDEF1   DB      Ø
NDEF2   DB      Ø
NDEF3   DB      Ø
NDEF4   DB      Ø
NDEF5   DB      Ø

;-------------------------------------------------------------------------
; 7223          MONITOR WORK TABLE
;-------------------------------------------------------------------------

UEVCT   DB      Ø               ;citac eventu
UEVPT   DB      Ø               ;adresa eventu

LNBUF   DS      4Ø              ;vyrovn.pamet

SCRDFR  DW      Ø               ;krok rolovani
SRDR    DW      Ø               ;data rolovani

CURPSV  DW      Ø               ;pozice kurzoru

SCRLCH  DB      Ø               ;znak rolovani

AVRHL   DB      Ø,Ø,Ø           ;rychlost cteni

ACHTBF  DS      32              ;buf pro IDT

NOTUSE  DS      72

SYSTKL  DS      64              ;prostor pro SP

SYSTAK                          ;prostor uzivatele
```

Vydala 602. ZO Svazarmu, Wintrova 8, 160 41 Praha 6 pro potřeby vlastního aktivu.
Autor: Daniel Dočekal. Publikace neprošla jazykovou úpravou. Náklad 500 výtisků.
Neprodejné! Praha, prosinec 1986