

```

ORG    &HC000

XOR    A
LD     (MSG),A

LD     HL,FCB
CALL   OPEN
OR     A
JP     NZ,ERROR

LD     HL,(BLENG)      ;File length LSB
LD     (GAMSIZ),HL
LD     HL,(BLENG+2)    ;File length MSB
LD     (GAMSIZ+2),HL

LD     A,4             ;Load 1st 32k
CALL   &HF224
LD     DE,&H8000
LD     HL,&H4000
CALL   LOAD
OR     A
JP     NZ,ERROR
LD     A,5
LD     (PAGE),A
CALL   &HF224
LD     DE,&H8000
LD     HL,&H4000
CALL   LOAD
OR     A
JP     NZ,ERROR

LD     HL,&H8000        ;Convert until &HBF00
CALL   CONVRT
JP     NZ,FINAL

LOEP:  PUSH HL          ;Load next 16k
LD     A,(PAGE)
INC    A
LD     (PAGE),A
CALL   &HF224
LD     DE,&H8000
LD     HL,&H4000
CALL   LOAD
POP    HL
OR     A
JP     NZ,ERROR

CALL   CONVRT          ;Continue converting
JP     NZ,FINAL

JP     LOEP

FINAL: DI
IN     A,(&HA8)
AND    &HC0
LD     B,A
RRCA
RRCA
OR     B
RRCA
RRCA
OR     B
RRCA
RRCA
OR     B
OUT    (&HA8),A
LD     A,(&HFFFF)
CPL
AND    &HC0

```

```

LD      B,A
RRCA
RRCA
OR      B
RRCA
RRCA
OR      B
RRCA
RRCA
OR      B
LD      (&HFFFF),A
LD      H,4

MOVML:  LD      A,(PAGE)
        CP      H
        JP      C,MOVR
        LD      A,H
        OUT     (&HFE),A
        SUB     3
        CALL    MOVEMM
        INC     H
        JP      MOVML

MOVR:   LD      A,1
        OUT     (&HFC),A      ;1st 16k block
        INC     A
        OUT     (&HFD),A      ;2nd 16k block
        INC     A
        OUT     (&HFE),A      ;3rd 16k block
        XOR     A
        OUT     (&H99),A
        LD      A,&H80
        OUT     (&H99),A      ;No VDP interrupts
        LD      A,14
        OUT     (&HA0),A      ;Gamepad via PSG
        LD      A,&H00
        OUT     (&H99),A
        LD      A,&H81
        OUT     (&H99),A      ;No VDP interrupts
        JP      0             ;GO!

MOVEMM: OUT     (&HFD),A
        PUSH    HL
        LD      HL,&H8000
        LD      DE,&H4000
        LD      BC,&H4000
        LDIR
        POP     HL
        RET

CONVRT: DI
        LD      A,H
        SUB     &H40
        LD      H,A
        IN      A,(&HFE)
        PUSH    AF
        IN      A,(&HFD)
        PUSH    AF
        LD      A,(PAGE)
        OUT     (&HFE),A
        DEC     A
        OUT     (&HFD),A
        IN      A,(&HA8)
        PUSH    AF
        AND     &HC0
        LD      B,A
        RRCA
        RRCA
        OR      B
        RRCA

```

```

RRCA
OR    B
OUT   (&HA8),A
LD    A,(&HFFFF)
CPL
PUSH  AF
AND   &HC0
LD    B,A
RRCA
RRCA
OR    B
RRCA
RRCA
OR    B
LD    (&HFFFF),A

CONVR: LD    A,(HL)
        CP    &H32
        JP    Z,DLD
        CP    &HED
        JP    Z,INDIO
        CP    &HD3
        JP    Z,DOUT
        CP    &HDB
        JP    Z,DIN
        INC   HL
CONVR: LD    A,H
        CP    &HBF
        JP    NZ,CONVR1

        POP   AF
        LD    (&HFFFF),A
        POP   AF
        OUT   (&HA8),A
        POP   AF
        OUT   (&HFD),A
        POP   AF
        OUT   (&HFE),A
        LD    A,(MSG)
        AND   128
        EI
        RET

DLD:    INC   HL                ;32
        LD    A,(HL)
        INC   A
        JP    Z,DLD0
        INC   A
        JP    Z,DLD1
        INC   A
        JP    Z,DLD2
        INC   A
        JP    Z,DLD3
        JP    CONVR

DLD0:   INC   HL                ;32 FF
        LD    A,(HL)
        INC   A
        JP    NZ,CONVR
        LD    A,&HFE            ;32 FF FF
        JP    DLDE
DLD1:   INC   HL                ;32 FE
        LD    A,(HL)
        INC   A
        JP    NZ,CONVR
        LD    A,&HFD            ;32 FE FF
        JP    DLDE
DLD2:   INC   HL                ;32 FD
        LD    A,(HL)
        INC   A

```

	JP	NZ,CONVR	
DLDE:	LD	A,&HFC	;32 FD FF
	LD	(HL),A	
	DEC	HL	
	LD	A,&HD3	
	LD	(HL),A	
	DEC	HL	
	LD	A,&H3C	
	LD	(HL),A	
	INC	HL	
	INC	HL	
DLD3:	JP	CONVR	
	INC	HL	;32 FC
	LD	A,(HL)	
	INC	A	
	JP	NZ,CONVR	
	LD	A,(MSG)	
	OR	1	
	LD	(MSG),A	
	JP	CONVR	
INDIO:	INC	HL	;ED
	LD	A,(HL)	
	CP	&HA2	;ED A2 INI
	JP	Z,INDIOF	
	CP	&HA3	;ED A3 OUTI
	JP	Z,INDIOF	
	CP	&HB3	;ED B3 OTIR
	JP	Z,INDIOF	
	AND	&HC6	
	CP	&H40	;ED 41 I/O (C),r
	JP	Z,INDIOF	
	JP	CONVR	
INDIOF:	PUSH	HL	
	DEC	HL	
	DEC	HL	
INDIB:	LD	B,32	
	DEC	HL	;D3
	LD	A,(HL)	
	CP	&H0E	;0E LD C,**
	JP	Z,CLD	
	CP	&H01	;01 LD BC,**
	JP	Z,CLD	
	DJNZ	INDIB	
	LD	A,(MSG)	
	OR	16	
	LD	(MSG),A	
	POP	HL	
	JP	CONVR	
CLD:	CALL	DOUTS	
	POP	HL	
	JP	CONVR	
DOUT:	CALL	DOUTS	
	JP	CONVR	
DOUTS:	INC	HL	;D3
	LD	A,(HL)	
	CP	&H7E	
	JP	Z,PSG1	
	CP	&H7F	
	JP	Z,PSG2	
	CP	&HBE	
	JP	Z,VDP1	
	CP	&HBF	
	JP	Z,VDP2	
	CP	&HDE	
	JP	Z,KEY1	
	CP	&HDF	

	JP	Z,KEY2	
	CP	&HF0	
	JP	Z,FM1	
	CP	&HF1	
	JP	Z,FM2	
	RET		
DIN:	CALL	DINS	
	JP	CONVR	
DINS:	INC	HL	;DB
	LD	A,(HL)	
	CP	&H7E	
	JP	Z,PSG1	
	CP	&H7F	
	JP	Z,PSG2	
	CP	&HBE	
	JP	Z,VDP1	
	CP	&HBF	
	JP	Z,VDP2	
	CP	&HDC	
	JP	Z,PAD1	
	CP	&HDD	
	JP	Z,PAD2	
	CP	&HDE	
	JP	Z,PAD1	
	CP	&HDF	
	JP	Z,PAD2	
	CP	&HF2	
	JP	Z,FMDDET	
	RET		
PSG1:	LD	A,&H48	;D3/DB/0E/01 7E
	LD	(HL),A	
	RET		
PSG2:	LD	A,&H49	;D3/DB/0E/01 7F
	LD	(HL),A	
	RET		
VDP1:	LD	A,&H88	;D3/DB/0E/01 BE
	LD	(HL),A	
	RET		
VDP2:	LD	A,&H89	;D3/DB/0E/01 BF
	LD	(HL),A	
	RET		
PAD1:	LD	A,&HA2	;DB DC/DE
	LD	(HL),A	
	RET		
PAD2:	LD	A,(MSG)	;DB DD/DF
	OR	8	
	LD	(MSG),A	
	RET		
KEY1:	LD	A,(MSG)	;D3/0E/01 DE
	OR	2	
	LD	(MSG),A	
	RET		
KEY2:	LD	A,(MSG)	;D3/0E/01 DF
	OR	2	
	LD	(MSG),A	
	RET		
FM1:	LD	A,&H7C	;D3/0E/01 F0
	LD	(HL),A	
	LD	A,(MSG)	
	OR	4	
	LD	(MSG),A	
	RET		
FM2:	LD	A,&H7D	;D3/0E/01 F1
	LD	(HL),A	
	LD	A,(MSG)	
	OR	4	
	LD	(MSG),A	
	RET		

```

FMDET:  LD    A,(MSG)          ;DB F2
        OR    4
        LD    (MSG),A
        RET

ERROR:  JP     CLOSE
        LD    B,A
        LD    A,&H62
        JP    &HF37D

; ***** Read files *****
OPEN:   LD    HL,NAAM
        LD    DE,FCB+1
        LD    BC,11
        LDIR
        LD    DE,FCB
        LD    C,15
        CALL &HF37D
        OR    A
        RET   NZ
        LD    HL,1
        LD    (GROOT),HL
        DEC   HL
        LD    (BLOK),HL
        LD    (BLOK+2),HL
        RET

LOAD:   PUSH  DE
        EX    DE,HL
        LD    HL,(GAMSI2)
        OR    A
        SBC   HL,DE
        JP    C,CHKSIZ
        LD    (GAMSI2),HL
        EX    DE,HL
        JP    NZ,CHKSIZ
        LD    DE,(GAMSI2+2)
        LD    A,D
        OR    E
        JP    NZ,CHKSIZ
        JP    CHKSIZ4
CHKSIZ2: LD    B,H
        LD    C,L
        LD    HL,(GAMSI2+2)
        LD    A,L
        OR    H
        JP    Z,CHKSIZ3
        DEC   HL
        LD    (GAMSI2+2),HL
        LD    H,B
        LD    L,C
        LD    (GAMSI2),HL
        EX    DE,HL
        JP    CHKSIZ
CHKSIZ3: LD    HL,(GAMSI2)
CHKSIZ4: LD    A,(MSG)
        OR    128
        LD    (MSG),A

CHKSIZ: POP    DE
        PUSH   HL
        LD    C,26
        CALL &HF37D
        LD    DE,FCB
        POP    HL
        LD    C,39
        JP    &HF37D

CLOSE:  LD    DE,FCB
        LD    C,16

```

```

        CALL    &HF37D
        LD      A,1
        LD      (&HF1C1),A
        JP      &HFD9F

TEXT:    LD      A,(DE)
        CP      32
        JR      NC,TEXT1
        LD      A,32
TEXT1:   OUT     (&H98),A
        INC     DE
        DJNZ    TEXT
        RET

;***** Divide by 10 *****
DIV10:   LD      BC,&H080A      ;E=E/C, A=rest
        XOR     A
DIV10L:  SLA     E
        RL      A
        SUB     A,C
        JR      NC,DIV10N
        ADD     A,C
        DJNZ    DIV10L
        RET
DIV10N:  INC     E
        DJNZ    DIV10L
        RET

;***** Divide by 100 *****
DIV100:  LD      BC,&H0864      ;E=E/C, A=rest
        XOR     A
DI100L:  SLA     E
        RL      A
        SUB     A,C
        JR      NC,DI100N
        ADD     A,C
        DJNZ    DI100L
        RET
DI100N:  INC     E
        DJNZ    DI100L
        RET

;***** Divide 8 bits number *****
DIV8:    LD      B,16          ;DE=DE/C, A=rest
        XOR     A
DIV8L:   SLA     E
        RL      D
        RL      A
        SUB     A,C
        JR      NC,DIV8N
        ADD     A,C
        DJNZ    DIV8L
        RET
DIV8N:   INC     E
        DJNZ    DIV8L
        RET

;***** Divide *****
DIV16:   LD      A,B          ;DE=DE/BC, HL=rest
        CPL
        LD      B,A
        LD      A,C
        CPL
        LD      C,A
        INC     BC
        LD      HL,0
        LD      A,16
DIV16L:  SLA     E
        RL      D
        RL      L

```

```

        RL      H
        LD      (DIVDUM),HL
        ADD     HL,BC
        JR      NC,DIV16N
        INC     E
        DEC     A
        JR      NZ,DIV16L
        RET
DIV16N: LD      HL,(DIVDUM)
        DEC     A
        JR      NZ,DIV16L
        RET

```

;***** Divide *****

```

DIV32:  LD      A,B          ;IXDE=IXDE/BC, HL=rest
        CPL
        EXX
        LD      B,A
        EXX
        LD      A,C
        CPL
        EXX
        LD      C,A
        INC     BC
        LD      HL,0
        EXX
        LD      A,32
DIV32L: SLA     E
        RL      D
        RL      L
        RL      H
        EXX
        RL      L
        RL      H
        JR      NC,DIV32P
        ADD     HL,BC
        JR      DIV32O
DIV32P: LD      (DIVDUM),HL
        ADD     HL,BC
        JR      NC,DIV32N
DIV32O: EXX
        INC     E
        DEC     A
        JR      NZ,DIV32L
        RET
DIV32N: LD      HL,(DIVDUM)
        EXX
        DEC     A
        JR      NZ,DIV32L
        RET

```

;***** Multiply 8 bits number *****

```

MUL8:  LD      HL,0          ;HL=A*E
        LD      D,0
        LD      B,8
MUL8L: SLA     L
        RL      H
        RL      A
        JR      NC,MUL8N
        ADD     HL,DE
MUL8N: DJNZ    MUL8L
        RET

```

```

; B0 = RAM page accessed
; B1 = Keyboard address written
; B2 = FM PAC!
; B3 = 2nd keypad won't work!
; B4 = Indirect IO found but no setup for reg C
; B7 = Load finished
MSG:   DB      0

```

```
GAMSIZ: DW 0,0
DIVDUM: DW 0
PAGE: DW 0
FCB: DB 0,""
      DW 0
GROOT: DW 0
BLENG: DW 0
      DB 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
BLOK: DW 0
      DW 0
LOADED: DB 0
      ORG &HC600
NAAM: DB "HOMEBREWSMS"
      DB 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
```