P5.ASM

bank & move module for PROF-180 CP/M 3 linked BIOS

CP/M RMAC ASSEM 1.1 #001 BANK & MOVE MODULE FOR PROF-180 CP/M 3 LINKED BIOS

title 'bank & move module for PROF-180 CP/M 3 linked BIOS'

; letzte Aenderung am:

; 12.12.1985 (Joachim)

cseg

public ?move,?xmove,?bank,dmatrans,adrtab

extrn @cbnk

maclib hd64180

maclib pdef

maclib psys

?xmove:

sbcd sbank ; lege source und dest. bank ab

0000+ED43 DB 0EDH,43H

0002+0A00 DW SBANK

0004 3EFF mvi a,0ffh ; setze flag fuer xmove

0006 320C00 sta x$flg ;

0009 C9 ret

000A 00 sbank db 0 ; source bank fuer xmove

000B 00 dbank db 0 ; destination bank fuer xmove

000C 00 x$flg db 0 ; flag fuer xmove aktiv

000D 0000 sadrs dw 0 ; zwischenspeicher fuer xmove

000F 0000 dadrs dw 0 ;

0011 0000 length dw 0 ;

?move:

0013 EB xchg ; we are passed source in DE and dest in HL

0014 3A0C00 lda x$flg ; xmove aktiv?

0017 3C inr a ;

jrz xmv ; sprung wenn ja

0018+2804 DB 28H,XMV-$-1

ldir ; use Z80 block move instruction

001A+EDB0 DB 0EDH,0B0H

001C EB xchg ; need next addresses in same regs

001D C9 ret

xmv: sded dadrs ; save destination

001E+ED53 DB 0EDH,53H

0020+0F00 DW DADRS

sbcd length ; save length

0022+ED43 DB 0EDH,43H

0024+1100 DW LENGTH

0026 3A0A00 lda sbank ; switch to source bank

0029 CDD700 call ?bank ;

002C 115700 lxi d,xmv$buff ; ziel ist buffer

ldir ; uebertrage

002F+EDB0 DB 0EDH,0B0H

0031 220D00 shld sadrs ;

0034 3A0B00 lda dbank ; switch to dest. bank

0037 CDD700 call ?bank ;

CP/M RMAC ASSEM 1.1 #002 BANK & MOVE MODULE FOR PROF-180 CP/M 3 LINKED BIOS

lded dadrs ;

003A+ED5B DB 0EDH,5BH

003C+0F00 DW DADRS

lbcd length ;

003E+ED4B DB 0EDH,4BH

0040+1100 DW LENGTH

0042 215700 lxi h,xmv$buff ;

ldir ;

0045+EDB0 DB 0EDH,0B0H

0047 3A0000 lda @cbnk ; orginal bank ein

004A CDD700 call ?bank ;

004D EB xchg ;

lded sadrs ;

004E+ED5B DB 0EDH,5BH

0050+0D00 DW SADRS

0052 AF xra a ;

0053 320C00 sta x$flg ;

0056 C9 ret

0057 xmv$buff: ds 128

?bank:

00D7 324BFF sta current$bank ; wichtig fuer cinit u. csysgen

00DA 0707070707 rlc ! rlc ! rlc ! rlc ! rlc ! rlc

00E0 E640 ani 40h

out0 bbr,a

00E2+ED3939 DB 0EDH,A\*8+1,BBR

00E5 C9 ret

dmatrans: ; dma transfer, source und destination

; daten stehen in @adrtab.

00E6 21F800 lxi h,adrtab ; zeiger auf datenliste.

00E9 012008 lxi b,sar0l+(8\*256) ; port und laenge angeben.

otimr ; spezial ausgabe hd64180

00EC+ED93 DB 0EDH,093H

00EE 7E mov a,m ; dcntl byte in akku

out0 dmode,a ; und ausgeben.

00EF+ED3931 DB 0EDH,A\*8+1,DMODE

00F2 3E60 mvi a,01100000b ; starte dma transfer

out0 dstat,a ;

00F4+ED3930 DB 0EDH,A\*8+1,DSTAT

00F7 C9 ret ;

;

adrtab: ;

00F8 0000000000 db 0,0,0,0,0,0,0,0 ; datenliste

0100 02 db 000000010b ;

0101 end