

CP/M Plus Patch mit dk'tronics RAM

Ein CPC 464 mit CP/M 3.0 und dk'tronics-RAM ergibt praktisch einen CPC 6128. Besagte Speichererweiterung ist vollständig kompatibel zu den zweiten 64KByte des 6128. Sie wird mit denselben Kommandos wie bei diesem Rechner über das Gate Array angesprochen (OUT &7F00, &C0 ... &C7).

CP/M Plus läßt sich mit der von dk'tronics gelieferten Software ohne Änderung sofort nach Eingabe folgender Befehle starten: `!emulate: !cpm` (vergl. Kapitel 3.15 des dk'tronics-Handbuchs). Meine Lösung, das EMS-File direkt zu patchen, bietet jedoch gegenüber der von dk'tronics vorgeschlagenen Methode drei Vorteile:

1. Es muß nicht nach jedem Booten durch ein Submitfile zeitaufwendig ein Patch-Programm gestartet werden, das eine korrekte Tastaturabfrage ermöglicht. Dies läßt sich durch die Einrichtung der KM-Indirection bei &BDF4 erreichen, die normalerweise auf dem 464 nicht existiert. Dazu ist der ROM-Bereich von &B1 bis &E7 ins RAM zu verlegen, damit der Call bei &D9 durch CALL &BDF4 ersetzt werden kann. CP/M Plus patched dann diese Indirection. Wenn sie auf dem 464 fehlt, führt dies dazu, daß die BIOS-Funktion 2 (Get Consol Status) nicht funktioniert. Beim veränderten EMS-File arbeitet sie dann korrekt.
2. Ein Reset mit SHIFT CTRL ESC führt nun wirklich zur Einschaltmeldung. Da beim 464 normalerweise keine Speichererweiterung angeschlossen ist, wird sie bei einem Reset vom Betriebssystem auch nicht initialisiert. Deshalb ist dann Block 4 der zweiten Bank anstelle des Bildschirmspeichers eingeblendet. Beim 6128 schaltet MC START PROGRAM Bank 0 ein. Beim gepatchten EMS-File wird nun die ROM-Routine KM TEST BREAK durch eine RAM-Routine ersetzt, die vor einem Reset die RAM-Bank 0 aktiviert.
3. Es ist eine schnellere Zeichenausgabe eingebaut. Dies wird durch eine sehr flinke Ersatzroutine für TXT WRITE CHAR ermöglicht. Allerdings wird jetzt immer der Zeichensatz aus dem ROM genommen, d.h., ein anderer oder inverse Schrift sind nicht mehr möglich. Deshalb läßt sich die schnellere Zeichenausgabe, die beim Booten aktiviert wurde, mittels des Dienstprogramms ONOFF.COM auch wieder zurücknehmen. ONOFF S schaltet den Patch ab, ONOFF F schaltet ihn an.

Dateien für automatische Tastaturbelegung

Datei:	PROFILE.SUB
Inhalt:	SETKEYS KEYS.DEF

Datei :	KEYS.DEF	
E 0	"DIR^M"	f0
E 1	"ERA "	f1
E 2	".BIN"	f2
E 3	"PIP "	f3
E 4	"REN "	f4
E 5	".COM"	f5
E 6	"A:"	f6
E 7	"TYPE "	f7
E 8	".*"	f8
E 9	"B:"	f9
E 10	".BAS"	f".
E 11	"PIP^M"	ctrl/f".
E 12	"DISCKIT3^M"	ctrl/enter
E 13	"SET "	shift/f".
E 14	"DIR B:^M"	shift/f0
E 18	"ERA *.BAK^M"	ctrl/del
E 19	"DIR A:^M"	ctrl/f0
E 20	"A:^M"	ctrl/f1
E 21	"B:^M"	ctrl/f2
E 22	"SUBMIT "	shift/enter
E 23	"DIR [FF,U=AL]"	shift/copy
79 C	"^146'"	CTRL-DEL => ERA *.BAK
6 N	"^13'"	
6 S	"^150'"	AUFRUF VON SUBMIT
6 C	"^140'"	AUFRUF VON DISCKIT3
9 S	"^151'"	AUFRUF VON DIR
9 N C	"^W"	COPY
15 S	"^142'"	
15 C	"^147'"	
7 S	"^141'"	
7 C	"^139'"	
14 S	"^130'"	
14 C	"^149'"	
13 S	"^148'"	
13 C	"^148'"	
20 S	"^149'"	
20 C	"^149'"	
0 N	"^#1F'"	CCP cursor UP
1 N	"^F"	cursor RIGHT
1 C	"^2'"	
2 N	"^#1E'"	cursor DOWN
8 N	"^A"	cursor LEFT
16 N S	"^G"	CLR
16 C	"^K"	
66 S	"^27'"	ESC
66 N	"^3'"	
66 C	"^252'"	
68 N	"^19'"	TAB => CTRL-S
68 S	"^17'"	SHIFT-TAB => CTRL-Q
79 S	"^X"	DEL

Für die Erzeugung von ONOFF.COM wird der Quellcode normal assembliert und anschließend als ONOFF.BIN abgelegt. Dann erfolgt mittels DDT

oder SID die Verschiebung nach 0100 und die Speicherung als ONOFF.COM. Das gemäß dem abgedruckten Quellcode gepatchte CP/M läuft allerdings nur auf dem 464. CP/M läßt sich jedoch auch auf dem 6128 und 664 vom veränderten EMS-File aus starten, wenn die Zeilen 23 bis 25 von Listing 1 zwischen die Zeilen 18 und 19 kommen. Das Label W1 bleibt an seinem Platz. Die Patches für Tastaturabfrage und Reset werden nur auf dem 464 initialisiert. Somit verfügt dann auch der 6128 über eine schnellere Zeichenausgabe unter CP/M.

Im folgenden finden Sie eine Anleitung für die Herstellung einer auf dem CPC 464 mit angeschlossener dk'tronics-Speichererweiterung lauffähigen CP/M-Plus-Version:

Beginnen Sie die Arbeit mit CP/M 2.2. Zunächst ist die A-Seite der CPC-6128-Systemdisketten mit dem DISCKIT2-Utility auf eine leere Diskette zu kopieren. Löschen Sie nun alle Files außer C10CPM3.EMS, und kopieren Sie SID.COM mit FILECOPY auf diese Diskette. Assemblieren Sie den CPM-Patch mit einem AMSDOS-Assembler (wenn nötig, Code an anderer Adresse ablegen) und speichern Sie den erzeugten Code als Binär-File CPM + PATC.BIN auf der Diskette ab. Dann sind, wie in Abschnitt 3.11.2 des dk'tronics-Handbuchs beschrieben, die Files BANK.BAS und RSX.BIN von der von dk'tronics mitgelieferten Cassette auf diese Diskette zu übertragen.

Geben Sie jetzt bitte folgende Zeilen ein, die Sie jeweils mit ENTER abschließen:

```
run "bank
ENTER
emulate: lcpm
sid cpm + patc.bin
m180, 280, a000
rc 10 cpm 3.ems
ma000, a043, 1260
ma044, a0dc, 6110
alda
mov a, h
cpi 1
jmp 01d60
ENTER
wcpmplus.ems, 100, 64ff
g0
era c10 cpm 3.ems
```

Nach einem Reset booten Sie nun erneut CP/M. Es müßte korrekt gestartet werden. Kopieren Sie ONOFF.COM auf die Diskette und lassen Sie nach ONOFF F und ONOFF S eine Directory ausgeben. Der Unterschied in der Geschwindigkeit und beim

großen O wird Ihnen auffallen. Führen Sie dann einen Reset mit SHIFT CTRL ESC durch. Es müßte die normale Einschaltmeldung auftauchen.

Leider bietet CP/M Plus nicht die Möglichkeit wie CP/M 2.2, direkt beim Booten Tastatur und Bildschirm einzustellen. Dies läßt sich jedoch einigermaßen über eine PROFILE.SUB-Batch-Datei erreichen.

Jürgen Weber

```

Programm: CP/M+ Patch
Computer: CPC 464/664/6128
Funktion: CP/M+ für alle CPCs
Listings: 2
Hardware: dk'tronics-RAM
Assembler: Schneider Devpac

```

Listing 1

Hisoft GENA3.1 Assembler. Page 1.

Pass 1 errors: 00

```

1 ; DIESEER PATCH FUER DAS EMS-FILE
2 ; ERMUEGLICHT DIE KORREKTE FUNKTION
3 ; DER BIOS-FUNKTION CONSOLE-STATUS
4 ; SOFERN EINEN KORREKTEEN RESET MIT
5 ; SHIFT/CTRL/ESC
6 ; FERNER EINE SCHNELLERE ZEICHENAUSGABE
7 ; BY JUERGEN WEBER 16.6.87
8
9          ORG #1D60
BD3A      10 ZIEL: EQU #BD3A          ; STATT BASICVECTOREN
BC65      11 ZIEL2: EQU #BC65        ; CAS-VECTOREN
6C10      12 LAGER: EQU #6110+#B00
13
14 START:
15          LD HL,LAGER          ; ERSATZ FUER TXT WRITE CHAR
16          LD DE,ZIEL2
17          LD BC,FINPRI-PRINT
18          LDIR
19          LD BC,ENDE-ANF        ; UPDATE KEY STATE
16E6      20          LD DE,ZIEL          ; & TST BREAK
1D71      21          LDIR
1D73      22          JR C,W1          ; 464
1D75      23          LD HL,#B7C4
1D78      24          LD (GOFF+DIST2+2),HL ; SCREEN OFFSET
1D7B      25          JR RETU
1D7D      26 W1:
27
28          LD A,#C3
1D7D      28          LD HL,#1BB7          ; UPDATE KEY
1D7F      29          LD (#BDF4),A          ; INDIREDICTION HERSTELLEN
1D82      30          LD (#BDF5),HL
1D85      31          LD HL,#KTEST+DIST
1D88      32          LD (#BDEF),HL          ; KM TEST BREAK
1D8B      33          LD HL,INTPAT+DIST          ; INTERRUPT-ENTRY PATCHEN
1D8E      34          LD (#B949),HL
1D91      35          LD HL,LAGER
1D94      36 RETU: LD HL,LAGER
1D97      37          LD DE,LAGER+1
1D9A      38          LD BC,LENTOT
1D9D      39          LD (HL),0
1D9F      40          LDIR          ; DATEN LOESCHEN
41
42
43          JP INIT+DIST          ; TXT WRITE CHAR PATCHEN
1DA1      43
44
45
46 ; AB HIER IN CAS-VECTOREN
9EC1      47 DIST2: EQU ZIEL2-$
48
49 PRINT:
1DA4      49          LD B,A          ; ZEICHEN
1DA4      50
51
52 ; *** ERSATZ FUER SCR CHAR POSITION ***
53
54          LD C,H          ; SPALTE
1DA5      54
55          LD H,0
1DA6      55
56          LD D,H
1DA8      56
57          LD E,L          ; DE=SPALTE
1DA9      57
58          ADD HL,HL
1DAA      58
59          ADD HL,HL
1DAB      59
60          ADD HL,DE
1DAC      60

```

TIPS + TRICKS

```

1DAD 29 61 ADD HL,HL
1DAE 29 62 ADD HL,HL
1DAF 29 63 ADD HL,HL
1DB0 29 64 ADD HL,HL ; HL=HL*80
1DB1 59 65 LD E,C
1DB2 19 67 ADD HL,DE
1DB3 ED5BC9B1 68 COFF: LD DE, (#B1C9) ; SCREEN OFFSET
1DB7 19 69 ADD HL,DE
1DB8 7C 70 LD A,H
1DB9 E607 71 AND 7
1DBB C640 72 ADD A, #40 ; HIGHBYTE SCREEN START
1DBD 4D 73 LD C,L ; AC=SCHIRMADRESSE
1DBE 68 75 LD L,B ; HL=ZEICHEN
1DBF 110038 76 LD DE, #3800 ; START MATRIXTABLE
1DC2 63 78 LD H,E ; H:=0
1DC3 29 79 ADD HL,HL
1DC4 29 80 ADD HL,HL
1DC5 29 81 ADD HL,HL ; HL=8*A
1DC6 19 82 ADD HL,DE ; HL=ANFANG ZEICHENMATRIX IM ROM
1DC7 EB 83 EX DE,HL ; DE=MATRIX
1DC8 67 84 LD H,A ; SPEICHERADRESSE
1DC9 69 85 LD L,C ; NACH HL
1DCA ED73A1BC 86 LD (STACK+DIST2), SP
1DCE F3 87 DI ; EIN INTERRUPT BRAUCHT SP
1DCF 310008 88 LD SP, #000 ; LINIENABSTAND
1DD2 0608 90 LD B,8 ; 8 RASTERZEILEN
1DD4 91 LOOP:
1DD4 1A 92 LD A, (DE) ; AUS ROM
1DD5 77 93 LD (HL), A ; IN SCHIRM
1DD6 39 94 ADD HL, SP ; HL:=HL+#800
1DD7 1C 95 INC E ; ZEICHEN NIE AN HIGH-WECHSEL
1DD8 10FA 96 DJNZ LOOP
1DDA ED7BA1BC 98 LD SP, (STACK+DIST2)
1DDE FB 99 EI
1DDF C9 100 RET
1DE0 101 STACKP:
1DE0 102 FINPRI:
1DE0 103

```

```

1DE0 104 ANF:
9F5A 105 DIST: EQU ZIEL-$
106
1DE0 107 INTPAT: ; IM ROM AB BIH
1DE0 2187B1 108 LD HL, #B187
1DE3 34 109 TMLOOP: INC (HL)
1DE4 23 110 INC HL
1DE5 28FC 111 JR Z, TMLOOP
1DE7 06F5 112 LD B, #F5
1DE9 ED78 113 IN A, (C)
1DEB 1F 114 RRA
1DEC 3008 115 JR NC, OVER
1DEE 2A8CB1 116 LD HL, (#B18C)
1DF1 7C 117 LD A,H
1DF2 B7 118 OR A
1DF3 C45301 119 CALL NZ, #153
1DF6 2A1EB8 120 OVER: LD HL, (#B81E)
1DF9 7C 121 LD A,H
1DFA B7 122 OR A
1DFB C45301 123 CALL NZ, #153
1DFE CD611F 124 CALL #1F61
1E01 2192B1 125 LD HL, #B192
1E04 35 126 DEC (HL)
1E05 C0 127 RET NZ
1E06 3606 128 LD (HL), 6
1E08 CDF4BD 129 CALL #BDF4 ; UPDATE KEY MAP
1E0B 2A90B1 130 LD HL, (#B190)
1E0E 7C 131 LD A,H
1E0F B7 132 OR A
1E10 C8 133 RET Z
1E11 2104B1 134 LD HL, #B104
1E14 CB06 135 SET 0, (HL)
1E16 C9 136 RET
137
1E17 21F3B4 138 KMYTEST: ; IM ROM AB 1C2FH
1E1A CB56 139 LD HL, #B4F3
1E1C C8 140 BIT 2, (HL)
1E1D 79 141 RET Z
1E1E EEA0 142 LD A,C
1E20 C2901C 143 XOR #A0
1E23 C5 144 JUMP: JP NZ, #1C90 ; KM BREAK EVENT
1E24 23 145 PUSH BC
1E25 060A 146 INC HL
1E27 8E 147 LD B, #A
1E28 2B 148 ADDI: ADC A, (HL)
1E29 10FC 149 DEC HL
1E2B C1 150 DJNZ ADDI
1E2C FE44 151 POP BC
1E2E 20F0 152 CP #A4
1E30 01C07F 153 JR NZ, JUMP
1E33 ED49 154 LD BC, #7FC0 ; BANK 0
1E35 C7 155 OUT (C), C
1E36 156 RST 0
157 INIT:
158
1E36 2165BC 159 LD HL, ZIEL2
1E39 22D4BD 160 LD ( #BDD4 ), HL
1E3C C9 161 RET
162
1E3D 163 ENDE:
164
BD90 165 INITIA: EQU INIT+DIST
0099 166 LEINTOT: EQU $-LEINT

```

```

Pass 2 errors: 00
Table used: 293 from 438
Pass 1 errors: 00

```

Listing 2

```

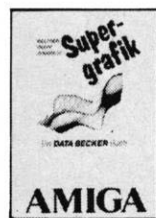
1 ; ONOFF 1987 BY JUERGEN WEBER
2
0100 3 ORG #100
0100 0E0C 4 LD C, 12
0102 CD0500 5 CALL 5 ; CP/M 3 ?
0105 7D 6 LD A, L
0106 FE31 7 CP #31
0108 C0 8 RET NZ
0109 218000 9 LD HL, #80
010C 7E 10 LD A, (HL)
010D 23 11 INC HL
010E B7 12 OR A
010F C8 13 RET Z ; KEIN ARGUMENT
0110 7E 14 SLOOP: LD A, (HL)
0111 FE20 15 CP ""
0113 23 16 INC HL
0114 28FA 17 JR Z, SLOOP
0116 E6DF 18 AND 223 ; UPPERS
0118 FE46 19 CP "F"
011A 2005 20 JR NZ, SLOW
011C CD1CFD 21 CALL #FD1C
011F 90BD 22 DEFW #BD90 ; SCHNELLE TEXTAUSGABE AN
0121 23 SLOW:
0121 FE53 24 CP "S"
0123 C0 25 RET NZ
0124 CD1CFD 26 CALL #FD1C
0127 51BB 27 DEFW #BB51 ; TXT RESET

```

Pass 2 errors: 00
Table used: 36 from 146

EDV-Buchversand Thomas Schlusenek

Wir haben das richtige Buch für Sie, überzeugen Sie sich selbst!



Bruno Jennich / Jens Trapp / Tobias Weltner
Amiga Supergrafik
Data Becker, 686 S. Geb.
Grafikprogrammierung mit den
vorhandenen BASIC-Befehlen,
Nutzung der Libraries, die Reg-
ister der Grafik-Chips, Aufbau
und Programmierung von
Screens, Windows, HAM, Half-
bits und Interlaces aus BASIC
und C, 1024 x 1024 Punkte
Superbitmap, gepufferte Multi-
tasking-Hardcopy-Routine -
zum Thema Grafik werden Sie
in Amiga Supergrafik nichts ver-
missen. 59,- DM



Frank Krenser / Jörg Koch
**Amiga-Programmier-
Handbuch für Amiga
500, 1000 und 2000**
Die wichtigsten System-
bibliotheken, Beispiele für
den Aufruf der Betriebssystem-
system-Routinen unter C, Auf-
ruf der DOS-Funktionen,
Programmieren von Win-
dows, Screens u. Gadgets,
Grafik u. Animation, Tips u.
Tools in C.
Markt & Technik, 1987,
390 S., inkl. Diskette. Pbd.
69,- DM



Dieter Quade
dBase III Plus für Einsteiger
Data Becker, 205 S. Geb.
Die Programmstruktur, der Pro-
grammgenerator, Programmie-
ren in dBase, Erstellen von Filter-
dateien, Memo-Variablen... bis
man endlich mit mehreren Daten-
banken arbeiten kann. Zahlreiche
praktische Übungsbeispiele, die
allesamt den typischen Alltagsan-
wendungen von dBase III Plus
entsprechen, helfen dabei, das
Beschreibene schnell zu verste-
hen. 49,- DM



Rudi Kost
**GEM-Anwenderhand-
buch Schneider PC**
Textverarbeitung, Grafik u.
Zeichnen mit den GEM-App-
likationen von Digital Re-
search. Eine ausführliche u.
leicht verständliche Anlei-
tung für die praktische
Arbeit. Mit vielen Tips für die
Verknüpfung der Pro-
gramme.
Markt & Technik, 1987,
350 S. Pbd. 49,- DM

BESTELLCOUPON

Stück	Titel	Preis
	Kostenloser Katalog	
	Amiga Supergrafik	
	Amiga-Programmier- Handbuch. für Amiga 500, 1000 und 2000	
	dBase III Plus für Einsteiger	
	GEM-Anwenderhand- buch Schneider PC	

EDV-Buchversand
Thomas Schlusenek 2724 Sottrum · Zevener Ring 10 · Tel. (042 64) 22 63 (gesch.)
Postgiroamt Hamburg (BLZ 200 100 20) Kto.-Nr. 602 723-208