

```

00000 '*****
*****
00010 ' * Befehlsweiterung ' H A
R D C O P Y ' (7.07.85) *
00020 ' * (c) 1985 by Helmut Tischer, Ast
ernstra#e 40, D-8052 Moosburg
00030 ' *****
*****
00040 'Gibt eine Kopie des Bildschirms a
n den Drucker f
00050 'Angepa"t f)r "NL0401" und kompati
ble Drucker
00060 'Anpassung f)r andere Drucker:
00070 'Tabelle in den Zeilen 20410(4.Byt
e)-20440 ab;ndern: Aufbau wie folgt:
00080 '<L(nge Gruppe 1>,<Steuerzeichen f
>r doppelte Dichte, 320 Punkte/Zeile>
00090 '<L(nge Gruppe 2>,<Steuerzeichen f
>r einfache Dichte, 320 Punkte/Zeile>
00100 '<L(nge Gruppe 3>,<Steuerzeichen f
>r Zeilenvorschub 7 Punkte>
00110 '<L(nge Gruppe 4>,<Steuerzeichen f
>r Zeilenvorschub 1/8 Zoll>
00120 'Pr)fsumme: Summe aller Zahlen in
einer Zeile
00130 'letzte Zeile mit '00' auff)llen u
nd Programml(nge in Zeile 20010 (ndern
00140 '
00150 'Speicherplatz reservieren
00160 PRINT"Wieviele Zeichen sollen frei
definierbar sein?"
00170 INPUT"SYMBOL AFTER ",sym%:SYMBOL A
FTER sym%
00180 OPENOUT"dunmy"Kassettenbuffer auf
Bereich oberhalb RAMTOP fixieren
00190 MEMORY HIMEM-1:CLOSEOUT:SPEED WRIT
E 1
00200 '
00210 'Programm einladen (mit Verschieb
eader)
00220 READ n,m:st=HIMEM-n+1:MEMORY st-1
00230 FOR i=0 TO n-1 STEP 8
00240 pruef=0
00250 FOR j=1 to i+7:READ d$:d=VAL("&"+d
$):POKE st+j,d:pruef=pruef+d:NEXT
00260 READ pr:IF pruef<>p THEN PRINT"Pr)f
summenfehler in Zeile"20030+i/8*10:END
00270 NEXT
00280 FOR i=1 to m
00290 READ d$:d=VAL("&"+d$):z=st+PEEK(st
+d)+256*PEEK(st+d+1)
00300 POKE st+d+1,INT(z/256):POKE st+d,z
-256*INT(z/256)
00310 NEXT
00320 '
00330 'Initialisierung
00340 CALL HIMEM+1
00350 END"Wenn Programm l(uft,durch 'NEW
' ersetzen
00360 '

```

```

20000 'L(nge, Anzahl der anzupassenden A
dressen
20010 DATA 336,27
20020 'Programm(lauff(hig ab Adresse #)
20030 DATA 01,09,00,21,17,00,C3,D1, 470
20040 DATA 0C,0E,00,C3,1B,00,4B,41, 561
20050 DATA 52,44,43,4F,50,D9,00,00, 593
20060 DATA 00,00,00,ED,73,31,01,3A, 460
20070 DATA C8,B1,87,FE,01,CE,00,87,1100
20080 DATA 32,29,01,0E,00,CD,13,01, 344
20090 DATA 0E,0A,CD,13,01,3E,02,CD, 510
20100 DATA 0A,00,3E,7F,32,20,01,2A, 460
20110 DATA C9,B1,7C,F6,C0,67,06,1D,1070
20120 DATA 05,20,05,3E,70,32,20,01, 315
20130 DATA 04,C5,CD,9F,00,3A,C8,B1,1000
20140 DATA FE,02,CC,9F,00,11,00,00, 644
20150 DATA A7,ED,52,7C,F6,3F,3C,28,1019
20160 DATA 08,11,00,3F,19,7C,F6,F0, 907
20170 DATA 67,0E,0A,CD,13,01,0E,00, 379
20180 DATA CD,13,01,E5,3E,42,CD,1E, 817
20190 DATA 0B,C2,23,01,E1,C1,10,C0,1043
20200 DATA 3E,03,21,33,01,4E,23,06, 269
20210 DATA 00,09,3D,20,F0,46,23,4E, 533
20220 DATA 23,CD,13,01,10,F9,C9,E5, 955
20230 DATA 21,33,01,3A,CB,01,FE,02, 776
20240 DATA 28,05,4E,23,0A,00,09,CD, 378
20250 DATA 95,00,E1,3A,CB,01,FE,02,1065
20260 DATA 06,28,28,02,06,50,C5,E5, 600
20270 DATA 11,2A,01,3E,07,ED,00,01, 527
20280 DATA FF,07,09,30,0A,01,50,C0, 602
20290 DATA 09,47,7C,E6,C7,67,78,3D, 917
20300 DATA 20,E8,21,CF,B1,3A,29,01, 704
20310 DATA 47,C5,11,2A,01,06,07,1A, 367
20320 DATA 13,06,FE,01,3F,CB,11,10, 739
20330 DATA F6,3A,20,01,A1,4F,CD,13, 809
20340 DATA 01,3A,C8,B1,A7,CC,13,01, 827
20350 DATA C1,23,10,DD,E1,C1,7C,E6,1237
20360 DATA FB,4F,23,7C,E6,07,01,67,1003
20370 DATA 10,AC,C9,79,CD,2B,0D,D8,1163
20380 DATA C5,E5,3E,42,CD,1E,0B,E1,1201
20390 DATA C1,28,F0,ED,7B,31,01,C9,1084
20400 DATA 00,00,00,00,00,00,00,00, 0
20410 DATA 00,00,00,04,1B,4C,40,01, 172
20420 DATA 04,1B,49,40,01,05,1B,41, 260
20430 DATA 07,1B,32,05,1B,41,0C,1B, 228
20440 DATA 32,00,00,00,00,00,00,00, 50
20450 'Bei Verschiebung anzupassende Adr
essen
20460 DATA 0001,0004,0009,000C
20470 DATA 001D,0029,002E,0033
20480 DATA 0038,003D,004E,0053
20490 DATA 0058,0074,0079,0082
20500 DATA 008B,009A,0001,0000
20510 DATA 00C1,00DE,00E3,00F2
20520 DATA 00F7,00FE,0125
20530 'ende

```

```

1  ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
2  ;*      H A R D C O P Y      für CPC464      17.07.83  *
3  ;*  (c) 1983 by Helmut Fischer  *
4  ;*      Asterstraße 40, D-8052 Rosenberg  *
5  ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
6  ;Gibt den Bildschirm auf den Drucker aus
7  ;
0000      org      $0000
0001      ;***** Betriebsystemadressen
0002      equ     equ      $b02b
0003      equ     equ      $b01e
0004      equ     equ      $b0c8
0005      equ     equ      $b0c9
0006      equ     equ      $b0cf
0007      equ     equ      $b0d1
0008      ;***** Routine in Betriebssystem einbinden
0009      ld      bc,eqs
000a      ld      hl,interf
000b      jp      kilast
000c      defb    defb    name
000d      jp      procpr
000e      defb    defb    "HARDCOPY"
000f      defb    defb    "C"-$00
0010      defb    defb    0
0011      defb    defb    4
0012      ;***** Vorinstellungen + Vorbereitung der Hardcopy
0013      procpr  ld      (data)sp      ; Stackpointer setzen
0014      ld      a,(medel)           ; Anzahl Punkte pro Byte?
0015      add     a,a
0016      cp      1
0017      adc     a,0                 ;Umwandlung 0 in 1
0018      add     a,a
0019      ld      (data)a
001a      ld      c,$0d              ;Zeilenvorschub
001b      call   print
001c      ld      c,$0a
001d      call   print
001e      ld      a,2
001f      call   tabsu               ;neue Zeilenhöhe
0020      ld      a,$7f              ;benötigte Bits=1
0021      (word)  ld      (word)a
0022      ld      hl,(screen)        ;Start des Bildspeichers
0023      ah      ah
0024      or      bc,0
0025      ld      b,a
0026      ld      b,$2f              ;Anzahl Graphicszeilen
0027      ;***** Bild ausgeben
0028      nextint dec     b
0029      jr      nz,weiter
002a      ld      a,$7f              ;Nur noch 4 Bits gültig
002b      (word)  ld      (word)a
002c      inc     b
002d      push   bc
002e      call   zeile              ;Zeile ausgeben
002f      ld      a,(medel)
0030      cp      02

```

005A	CC9F00	57	call	z-reile	zweite Hälfte	
005B	110000	58	ld	dx,#0000	Adresse nächste Zeile	
005C	A7	59	and	a	berechnen	
005D	8D02	60	shl	hl,dx		
005E	7C	61	ld	ax,b		
005F	F63F	62	or	ax,03F		
0060	3C	63	inc	a	nächste Zeilen-Gruppe?	
0061	2000	64	jr	z-ek1		
0062	11803F	65	ld	dx,#03F0	Korrektur	
0063	1F	66	add	hl,dx		
0064	7C	67	ld	ax,b		
0065	F63F	68	or	ax,03F	übertrag korrigieren	
0070	67	69	ld	bx,a		
0071	0E0A	70	shl:	ld	c,#0A	Zeilenverschiebung
0072	CB1301	71	call	print		
0073	0E00	72	ld	c,#0A		
0074	CB1301	73	call	print		
0075	E5	74	push	hl	Breaktaste?	
0076	3E42	75	ld	ax,#42		
0077	CB1E00	76	call	initalk		
0081	C22301	77	je	nz-ende		
0084	E1	78	pop	hl		
0085	C1	79	pop	bc		
0086	10C0	80	dinc	nextline	nächste Zeile	
0088	3E03	81	ld	ax,3	alte Zeilenhöhe	
		82	=====	Unterprogramm: Steuerzeichentabelle suchen		
008A	213301	83	tabaus:	ld	hl,steuer	hier stehen Steuer-codes
008D	4E	84	salc:	ld	c,(hl)	Tabelleinsatz (a) suchen
008E	23	85	inc	hl		
008F	0400	86	ld	b,0		
0091	09	87	add	hl,bc		
0092	3D	88	dec	a		
0093	20F8	89	jr	nc-satz		
0095	46	90	ausg1:	ld	b,(hl)	Anzahl Steuerzeichen
0096	23	91	inc	hl		Adresse
0097	4E	92	ausg1	ld	c,(hl)	Zeichen ausgeben
0098	23	93	inc	hl		
0099	CB1301	94	call	print		
009C	10F9	95	dinc	ausg		
009E	C9	96	ret			
		97	=====	Unterprogramm: Halb-Zeile ausgeben		
		98	graphikmode	ein-schalten		
009F	E5	99	zeile:	push	hl	Steuerzeichensatz
00A0	213301	100	ld	hl,steuer	für Graphikmode	
00A2	3AC001	101	ld	a,(mode)	ein-schalten suchen	
00A4	FE02	102	cp	2		
00A6	2005	103	jr	z-mode2		
00A8	4E	104	ld	c,(hl)		
00A9	23	105	inc	hl		
00AC	0400	106	ld	b,0		
00AE	09	107	add	hl,bc		
00AF	CD9500	108	mode2:	call	ausg1	gefundene Zeichen ausg.
00B2	E1	109	pop	hl		
		110	teile	ausgeben:		
00B3	3AC001	111	ld	a,(mode)		
00B4	FE02	112	cp	2		
00B6	0420	113	ld	b,#20		
00B8	2002	114	jr	z-char		
00BC	0450	115	ld	b,#50	8Bytes pro Zeile	
00BE	C5	116	char:	push	bc	
00BF	E5	117	push	hl		
		118	17	untereinanderliegende Bytes in Puffer		
00C0	112A01	119	ld	dx,puffer		
00C3	3E07	120	ld	ax,7	17 Pixelzeilen	
00C5	ED40	121	lines:	ldi		
00C7	01FF07	122	ld	bx,#0FF		
00CA	09	123	add	hl,bc		
00CB	300A	124	jr	nc-ek2		
00CD	0150C0	125	ld	bx,#0C0	Korrektur	
00D0	09	126	add	hl,bc		
00D1	47	127	ld	b,a	16-Register retten	
00D2	7C	128	ld	ax,b	16-Übertrag korrigieren	
00D3	E6C7	129	and	ax,7		
00D5	67	130	ld	b,a		
00D6	78	131	ld	ax,b		
00D7	30	132	shl:	shl	a	
00D8	20E9	133	jr	nz-line		
		134	puffer	zeichen ausgeben:		
00DA	21CF01	135	ld	hl,masken	zuständige Bits für Pkt	
00DB	3AC001	136	ld	a,(data)	Punkte pro Byte	
00E0	47	137	ld	b,a		
00E1	C5	138	notdot:	push	bc	
00E2	112A01	139	ld	dx,puffer		
00E5	0407	140	ld	b,7		
00E7	1A	141	notbit:	ld	a,(dx)	Byte aus Puffer
00E8	13	142	inc	dx		
00E9	A6	143	and	(hl)	irrelevante Bits filtern	
00EA	FE01	144	cp	01		
00EC	3F	145	ccf		16-Übertrag carry	
00ED	CB11	146	pl	c	16-Übertrag Drucker	
00EF	10F6	147	dinc	notbit		
00F1	3AC001	148	ld	a,(used)	zur 7 (4) bits	
00F4	A1	149	and	c		
00F5	4F	150	ld	c,a		
00F6	CB1301	151	call	print		
00F9	3AC001	152	ld	a,(mode)		
00FC	A7	153	and	a	Mode 0?	
00FD	CC1301	154	call	z-print	doppelte Ausgabe	
0100	C1	155	pop	bc		
0101	23	156	inc	hl		
0102	10D0	157	dinc	notdot		
		158	17	Schleife vorbereiten		
0104	E1	159	pop	hl		

Listing 2. Das Assembler-Listing (Fortsetzung)

```

0105 C1      161      pop    bc
0106 7C      161      ld     a,h           ;Korrektur vorbereiten
0107 86F8    162      and    #F8
0108 4F      163      ld     c,a
010A 23      164      inc    hl           ;nächstes Zeichen
010B 7C      165      ld     a,h           ;Übertrag korrigieren
010C 8607    166      and    #07
010E B1      167      or     c
010F 67      168      ld     b,a
0110 10AC    169      shr    char
0112 C9      170      ret

           171 ;===== Zeichen an Drucker senden
0113 79      172      print: ld     a,c           ;Zeichen an Drucker
           173
0114 CD28D0  173      call    scpr1
0117 B8      174      ret     c           ;Übertragung o.k.?
0118 C5      175      push   bc
0119 E5      176      push   hl
011A 3E42    177      ld     a,#42           ;ESC-Taste?
011C CD10B8  178      call    kstata
011F B1      179      pop    hl
0120 C1      180      pop    bc
0121 28F0    181      jr     z,print           ;Marfen
           182 ;===== Fehlerrückprung ins Basic
0123 ED753101 183      enter  ld     sp,oldskpt
0127 C9      184      ret
           185 ;===== Externe Register
0128         186      used1:  defs 1
0129         187      defat:  defs 1
012A         188      puffer:  defs 7
012B         189      oldskpt:  defs 2
012C         190 ;===== Tabelle der Steuerzeichen
012E 04184C40 191      steuer:  defs 4,#1b,"L",#40,#01           ;Doppelte Dichte,320 Pkt
0130 04184840 192      defs 4,#1b,"C",#40,#01           ;Einfache Dichte,320 Pkt
0132 05184107 193      defs 5,#1b,"A",#07,#1b,"2",Linefeed 7/72 Zeil
0134 0518410C 194      defs 5,#1b,"A",#0C,#1b,"2",Linefeed 1/8 Zeil
0137         195      end

```

Pass 2 errors: 00

```

ausg  0057  ausgal 0075  char  00BE
defa  0129  ende   0123  intern 0017
klist  0021  kstata 001E  line   00C5
masken 01CF  scpr1  002B  mode    01C8
ende2  00AF  name   000E  nextlin 0048
nab1b1 00E7  naldot 00E1  sk1     0071
sk2     0057  oldskp 0131  prncopy 0018
print  0113  puffer 012A  ras     0009
selt   008D  screen 01C9  steuer 0133
tabaus 008A  used    0128  weiter 0050
zeile   00FF

```

Table used: 385 free: 603

Listing 2. Das Assembler-Listing (Schluß)