

# Mini-Grafik VC 20

## Diese kleine Grafik-Hilfe läuft auf der Grundversion des VC 20.

Wie der Name schon sagt, handelt es sich um eine Grafik-Erweiterung im kleinen Stil. Damit wird Grafik-Programmierung auch ohne große Speichererweiterung oder (teures) Grafik-Modul möglich. Da in Maschinensprache geschrieben, ist diese Basic-Erweiterung überdies sehr schnell. Minutenlange Wartezeiten auf die Fertigstellung einer Bildschirmgrafik entfallen damit.

Zur Anwendung der Grafikhilfe: Man nehme das Programm-Listing, tippe es in den VC 20 und erfreue sich folgender Befehle und Funktionen:

@ON	schaltet den Bildschirm auf Grafikdarstellung um, wobei die momentane Zeichenfarbe zur Plotfarbe wird
@CLR	löscht den Grafik-Speicher
@f,x,y	setzt (f ≠ 0) oder löscht (f=0) einen Punkt mit der Koordinate (x,y). Für f gilt: 0 ≤ f < 256, für x wie für y gilt: 0 ≤ x < 128. f, x und y können beliebige Variable oder Formeln sein

```

100 REM MG10 BY W.WIRTH <150>
105 POKE 55,0:POKE 56,21:CLR:POKE 648,22 <106>
110 SYS 58651:POKE 36879,14:PRINT" {WHITE}" <164>
115 DEF FN B4(X)=X-48+(X>57)*7 <065>
120 FOR I=0 TO 255:READ BY$:PRINT" {HOME}"I <066>
, BY$ <130>
125 BY=16*FN B4(ASC(BY$))+FN B4(ASC(RIGHT$ <193>
(BY$,1)) <195>
130 CS=CS+BY:POKE 5376+I,BY:NEXT <100>
135 FOR I=0 TO 3:READ BY:CS=CS+BY <110>
140 POKE 6140+I,BY:NEXT <161>
145 IF CS=29186 THEN SYS 5389:SYS 58238
150 PRINT" {CLR,DOWN,SPACE}CHECKSUMMENFEHLE <139>
R !" <199>
155 PRINT" {DOWN,SPACE}DATABLOCK PRUEFEN ." <155>
:END <218>
160 DATA 19,15,83,C4,7C,C5,1A,C7,29,15,EF <143>
165 DATA 15,00,A2,0B,BD,00,15,9D,00,03,CA <180>
170 DATA 10,F7,60,8A,4B,AD,05,90,C9,DE,D0 <205>
175 DATA 03,20,18,E5,68,4C,3B,C4,20,73,00 <186>
180 DATA 08,C9,40,F0,04,28,4C,E7,C7,28,20 <250>
185 DATA 3C,15,4C,AE,C7,20,73,00,C9,91,D0 <031>
190 DATA 55,20,73,00,A2,04,BD,FB,17,9D,01 <233>
195 DATA 90,CA,D0,F7,AD,86,02,9D,00,96,EB <234>
200 DATA D0,FA,A9,7F,AA,9D,00,16,E9,08,E9 <237>
205 DATA 00,29,7F,CA,10,F4,60,20,9E,D7,8A <014>
210 DATA 10,F9,4C,48,D2,A2,00,86,FD,20,6A <038>
215 DATA 15,48,4A,4A,4A,4A,66,FD,09,18,85 <171>
220 DATA FE,20,FD,CE,20,6A,15,A8,68,29,07 <009>
225 DATA AA,A9,00,38,6A,CA,10,FC,60,C9,9C <234>
230 DATA D0,19,20,73,00,A2,18,86,23,A0,00 <005>
235 DATA 84,22,98,91,22,C8,D0,FB,E6,23,E8 <220>
240 DATA E0,20,90,F4,60,C9,8E,D0,06,20,73 <035>
245 DATA 00,4C,18,E5,20,6A,15,08,20,FD,CE <253>
250 DATA 20,73,15,28,F0,04,11,FD,D0,04,49 <082>
255 DATA FF,31,FD,91,FD,60,20,73,00,20,FA <250>
260 DATA CE,20,73,15,31,FD,F0,02,A9,01,20 <225>
265 DATA 94,D7,20,B4,DF,4C,F7,CE,A9,00,85 <083>
270 DATA 0D,20,73,00,C9,40,F0,DD,20,79,00 <016>
275 DATA 4C,8D,CE,144,17,0,222 <085>

```

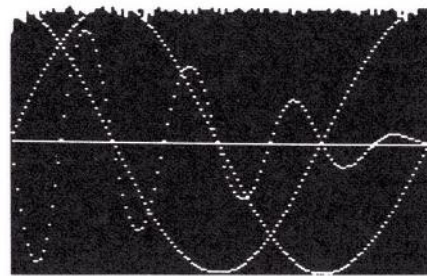
© 64'er

Listing 1. »Mini-Grafik VC 20« (Basic-Lader)

@(x,y)	ist eine Funktion, welche testet, ob der Punkt mit der Koordinate (x,y) gesetzt (Ergebnis = -1) oder nicht gesetzt (Ergebnis = 0) ist. Beispiel: »PRINT @(5,7)«
@RETURN	schaltet den Bildschirm schließlich wieder auf normalen Print-Modus

Das Programm implementiert sich nach dem Start des einzugebenden Basic-Laders (Listing 1) von selbst und ist absolut absturzsicher. Es hat, man lese und staune, eine Länge von nur 256 Byte. Für eine Auflösung von 128 mal 128 Punkten sind 2 KByte RAM unerlässlich. Somit verbleiben in der Grundversion 1279 Byte und mit 3 KByte Erweiterung sage und schreibe 4351 Byte Basic-RAM. Zugegeben, das ist nicht die Welt, aber für den Anfang durchaus ausreichend, und dafür wurde die Routine schließlich konzipiert. Überdies wird der Kassetten-Puffer nicht benutzt, und man kann von einem Programm aus mit »LOAD« ein weiteres nachladen. Listing 2 ist ein kleines Demo-Programm, um die Fähigkeiten der »Mini-Grafik« einmal vorzuführen (siehe Bild).

(Wolfgang Wirth/ev)



Die »Mini-Grafik« erzeugt ein kleines Grafik-Fenster

```

10 @ON:@CLR:DIM P(127) <236>
20 FOR I=0 TO 127 <254>
30 @1,0,I <096>
40 NEXT <170>
50 FOR I=1 TO 127 <029>
60 @1,I,64 <184>
70 NEXT <200>
80 FOR X=0 TO 127 <073>
90 @1,X,(1-SIN(X*PI/64))*63.5 <074>
95 @1,X,(1-COS(X*PI/64))*63.5 <078>
100 NEXT <230>
180 FOR X=0 TO 127 <173>
190 @1,X,(SIN(X*PI/16))*(127-X)/2+63.5 <070>
200 NEXT <074>
250 FOR I=1 TO 127 <229>
260 @1,I,0 <071>
265 @0,0,I <075>
270 NEXT <145>
280 REM <168>
281 Y=RND(TI)*128 <245>
285 P(Y)=P(Y)+1:IF P(Y)>127 THEN 400 <228>
286 @1,Y,P(Y) <059>
300 GOTO 280 <080>
400 FOR I=6144 TO 8191 <083>
420 POKE I,255-PEEK(I):NEXT <016>
900 @RETURN <085>

```

© 64'er

Listing 2. Demo-Programm zur »Mini-Grafik«