practic 3/87, S. 111-112

Rainer Jänisch

Effektives Programmieren von Blockgrafik

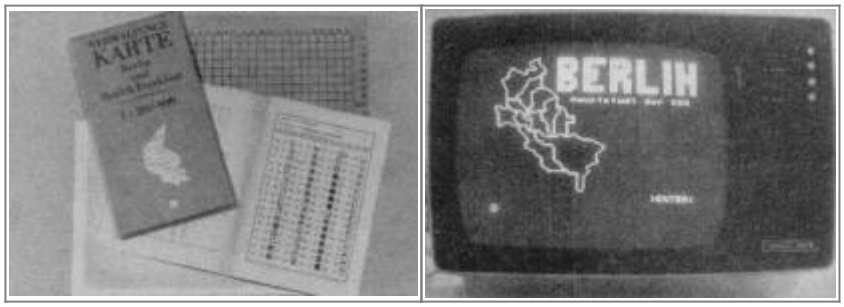
Zu Unrecht wird oft die Blockgrafik eines KC 87 oder Z 1013 als unzureichend angesehen, ohne alle Möglichkeiten auszunutzen.

Mit etwas Übung und gewissen Kompromissen lassen sich aber mit ihr viele grafische Probleme schnell lösen.

Unmittelbar auf dem Bildschirm zu konstruieren führt nicht so schnell zum Erfolg, wie es die Nutzung folgender Hilfsmittel ermöglicht. Entsprechend der Aufgabe, selbst etwas zu entwerfen oder ein vorgegebenes Bild umzusetzen, gibt es zwei Möglichkeiten:

Der eigene Entwurf

Man benutzt Papier mit quadratischen Kästchen. Wenn man selbst keine Linien ziehen will, klebt man mehrere Bogen Millimeterpapier bis zur gewünschten Größe zusammen. Zweckmäßig ist ein Tableau mit 32 Kästchen Breite und 32 Kästchen Höhe (mit jeweils 10 mm Kantenlänge). Nach der Kennzeichnung des „Bildschirmrandes“ werden noch die Bildschirmadressen (Anlage 8 des Handbuches zum Z 1013) an den Rand geschrieben und die schöpferische Phase kann beginnen.



Ist man mit dem Entwurf zufrieden, werden die Bildschirmadressen und aus der Tabelle der Grafiksymbole der entsprechende ASCII-Code des jeweiligen Zeichens (Kurve, Linie …) abgelesen.

Die Umsetzung einer vorgegebenen Zeichnung

In der benötigten Größe wird auf Transparentpapier oder Klarsichtfolie das Gitterraster gezogen und unverrutschbar auf der Zeichnung, Landkarte usw. befestigt.

Jetzt können die geeignetsten Grafiksymbole herausgesucht werden, die dem vorgegebenen Kurvenverlauf am besten entsprechen. Geringe Abweichungen müssen in Kauf genommen werden, da

nicht alle denkbaren Kurvenverläufe als Symbol vorhanden sind und es keine Sprünge im Linienverlauf geben darf.

Bei dem „Berlin-Quiz“ wurden z. B. die Balkenschriften mit der ersten Methode und die Landkarte mit der zweiten Methode programmiert **(Bild 1)**.

Die Programmierung von blinkenden Zeichen

Sollen in einem Programm Bildteile, Buchstaben oder andere Marken blinken, kann dieses mit wenigen Programmierschritten erreicht werden.

Mit einer FOR-NEXT-Schleife wird die Anzahl festgelegt, wie oft das betreffende Zeichen blinken soll.

```
10 FOR I = 0 TO 50
```

Soll das Zeichen zum Schluß stehenbleiben, wird zuerst ein „Space“ gesetzt und dann das Zeichen. Verfährt man umgekehrt, verschwindet das Blinkzeichen wieder.

Im Beispiel sollen drei auf der Spitze stehende Quadrate (ASCII: 138) gleichzeitig erscheinen und zum Schluß stehenbleiben.

```
20 PO.H.(ED85),32; PO.H.(EE28),32; PO.H.(EE2D),32
30 PO.H.(ED85),138;PO.H.(EE28),138; PO.H.(EE2D),138
40 NEXT I
```

Die Blinkfrequenz ist durch die Arbeitsgeschwindigkeit des Rechners festgelegt. Mit zusätzlichen FOR-NEXT-Schleifen lassen sich die Pausendauer

```
25 FOR I=0 TO 100; N.I
```

oder die Standzeit des Zeichens

```
25 FOR I = 0 TO 100;N.I
```

beliebig verlängern.

Rainer Jänisch

From:
https://hc-ddr.hucki.net/wiki/ - **Homecomputer DDR**

Permanent link:
https://hc-ddr.hucki.net/wiki/doku.php/z1013/literatur/practic-87-3-2

Last update: **2010/07/28 10:41**