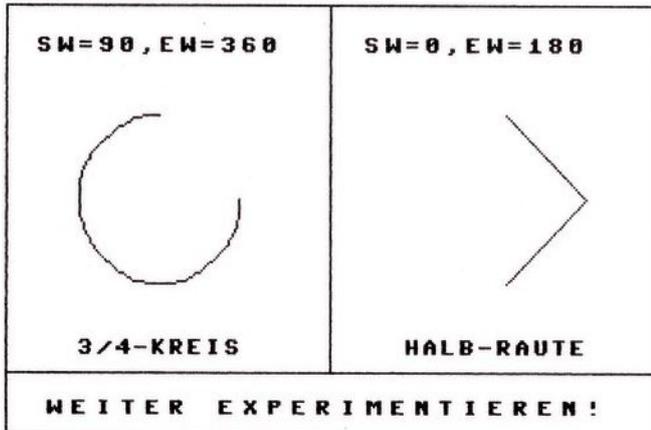


Und zuletzt beginnen wir, die START- und END-Winkel-Zahlen zu verändern! Denn: START 0 und END 360 ergibt ja eine voller Figur. Was aber, wenn Sie einen Halbkreis oder eine Halb-Raute wollen? Obere oder untere Hälfte? Waagrecht oder senkrecht »zerschnitten«? Bitte ändern Sie unser Programm ein letztes Mal:

```
5 INPUT "START-WINKEL";SW
6 INPUT "END-WINKEL";EW
25 ARC 199,90,SW,EW,12,35,35,1
```

ABSTAND 12 ergibt wieder einen Kreis, zwei gleiche Radien bedeuten einen »richtigen« Kreis. Wenn Sie jetzt ein Gefühl für

Bild 5.  $\frac{3}{4}$  Kreis + Halbe Raute/ARC



die Möglichkeiten, welche Ihnen die START- und END-Winkel-Zahlen bieten, gewonnen haben( siehe Bild 4), dann:

- Verändern Sie den ABSTAND wieder (siehe Bild 5)
- Verändern Sie den x-Radius (x1)
- Verändern Sie den y-Radius (y1)

Dadurch, daß Sie immer nur einen Parameter verändern, bekommen Sie für jeden einzelnen dieser Parameter ein »Ge-



Bild 6. Einladung zu einer Party

fühl«; somit werden Sie den ARC-Befehl durchs Ausprobieren wirklich in den Griff bekommen!

In Verbindung mit REC für den Rahmen und Text-Befehlen, kann mit ARC in Null-Komma-Nix sogar »ernstlich« gearbeitet werden: Visitenkarten, Briefbögen, Übersichten, Einladungen (Bild 6) und so weiter. Wenn Sie einen Drucker haben, der den COPY-Befehl des Simons Basic unterstützt, oder aber mit einem Maschinenprogramm per SYS-Befehl Ihre Bildschirme auf Papier überträgt, dann steht Ihren »Fotosatz«-Experimenten nichts mehr im Wege!

(Vera F. Birkenbihl/aa)

## Die RS232-Schnittstelle am VC 20

**Der VC 20 enthält im Betriebssystem bereits alle notwendigen Routinen zur Verwaltung einer RS232-Schnittstelle. Für den Anschluß von Peripheriegeräten wird auf den User-Port nur noch eine Erweiterungsplatine zur Anpassung der Signalpegel gesteckt.**

Auf dem Markt erhältlich sind sowohl Schaltungen nach der V.24-Norm als auch nach der TTY-Norm (20 mA). Nach dem richtigen elektrischen Verbinden der Geräte ist die Einstellung der Parameter für die Zeichenübertragung im Computer dem angeschlossenen Gerät entsprechend vorzunehmen.

Eingestellt werden müssen die Baudrate, die Anzahl der Datenbits, die Anzahl der Stoppbits, die Art der Paritätsüberprüfung und das Rückmeldeverhalten. Im VC 20, wo diese Schnittstelle unter der Gerätenummer 2 verwaltet wird, müssen beim Öffnen des Übertragungskanal in zwei Registern, dem Kontrollregister und dem Befehlsregister, Zahlen zur Einstellung der Betriebsart übergeben werden. Es ist stets eine mühsame und fehlerträchtige Arbeit, derartige Zahlen aus Tabellen zusammensuchen und richtig zu verknüpfen.

Das hier vorgestellte Programm nimmt dem Anwender die Suche nach den Zahlen und ihre Verknüpfung ab. In einem Bildschirmdialog werden mögliche Einstellungen der Betriebsart angeboten und ausgewählt. Dabei werden die aktuellen Werte der beiden Register stets angezeigt. Nach der Einstellung beginnt ein Übertragungstest. Zuerst werden die Zeichen »Leerzeichen« bis »Rückpfeil« auf den Bildschirm geschrieben und an das Gerät gesendet. Dann wird noch ein Testsatz übertragen. Danach kann mit über die Tastatur eingegebenen Zeichen weiter getestet werden. In das Einstellungsmenü wird durch Drücken der » - «-Taste zurückgesprungen.

Soll das Programm erst einmal ohne ein Peripheriegerät getestet werden, so genügt es, an den User-Port einen Teststecker anzuschließen. Dieser muß nur eine Brücke zwischen Pin »M« (Senden) und den Pins »C« und »B« (Empfangen) enthalten. Beim Öffnen des Files für Adresse zwei werden automatisch ein Sende- und ein Empfangspuffer mit einer Kapazität von je 256 Zeichen bereitgestellt. Über die Kurzschlußbrücke werden die gesendeten Daten in den Empfangspuffer übertragen. Dort können sie ausgelesen und weiter verarbeitet werden. In diesem Programm werden sie einfach auf den Bildschirm geschrieben.

### So arbeitet des Testprogramm

In Zeile 140 werden Anfangswerte in die Variable K(I) für die veränderlichen Parameter der Schnittstelle eingegeben. Die Datazeilen 150 bis 240 enthalten Textblöcke für den Bildschirmdialog. Nach dem Einschalten der Repeat-Funktion für alle Tasten in Zeile 250 beginnt der Dialogabschnitt in Zeile

### Testprogramm zum Initialisieren einer seriellen Schnittstelle

```

100 REM RS 232 SCHNITTSTELLE
110 REM 13.02.84
120 REM (C) H.-J. KELLERMANN
130 REM ANFANGSWERTE FESTLEGEN
140 K(0)=0:K(1)=6:K(2)=0:K(3)=0:K(4)=0:K(5)=0:K(6)=0
150 DATA6,"","TEST-ROUTINE","BAUD-RATE",
"DATENWORT-LAENGE","STOP-BITS"
160 DATA"HANDSHAKE","UEBERTRAGUNGSART","
PARITAETSPRUEFUNG"
170 DATA10,"BAUD"," ANW. "," 50 "," 7
5 "," 110 "," 134.5"," 150 "
180 DATA" 300 "," .600 "," 1200 "," 1800
"," 2400 "
190 DATA3," BITS",8,7,6,5
200 DATA1," STOP-BITS",1,2
210 DATA1,-DRAHT,3,X
220 DATA1,DUPLEX,VOLL,HALB
230 DATA7,,OHNE PAR.PRUEF.,UNGER.PARIT.,
OHNE PAR.PRUEF
240 DATAGERAD.PARIT.,OHNE PAR.PRUEF,8.BI
T AUF 1,OHNE PAR.PRUEF,8.BIT AUF 0
250 POKE650,128:REM ALLE TASTEN REPEAT
260 REM
270 REM MENUE ZUR PARAMETERAENDERUNG
280 RESTORE:FORJ=0TO6:READK%(J)
290 READK$:FORI=0TOK%(J):READK$(I,J):K$(
I,J)=K$(I,J)+K$
300 NEXTI:NEXTJ
310 J=0:K(J)=K(J)+1:IFK(J)=7THENK(J)=0
320 GOSUB680:K$="REGISTERINHALTE"
-----
330 K$=K$+" "KR=" "+STR$(KR
)+" "BR=" "+STR$(BR)+" "
340 GOSUB520:J=K(J):IFJ>0THENK$=K$(J,0):
GOSUB520:GOTO310
350 REM
360 REM TESTROUTINE ABARBEITEN
370 GOSUB710:GOSUB690:GOSUB660:CLOSE1:GO
SUB700
380 GOSUB680:OPEN1,2,0,CHR$(KR)+CHR$(BR)
:GOSUB700
390 GET#1,I$:REM PUFFER INITIALISIEREN
400 PRINT" ";:REM OPTISCHE KENNUNG FUER
SENDEN

```

```

410 FORI=32TO95:PRINTCHR$(I);:PRINT#1,CH
R$(I);:NEXT:REM ZEICHENSATZ
420 PRINT#1:PRINT:PRINT#1:PRINT
430 PRINT#1,"THE QUICK BROWN FOX JUMPS O
VER THE LAZY DOG 0123456789"
440 PRINT#1:POKE204,0:REM CURSOR BLINKEN
EIN
450 GET#1,I$:IFI$=""THEN480
460 IF(ASC(I$)AND127)<32THENWAIT207,1,1:
REM STEUERZEICHEN
470 POKE207,0:PRINTI$;:POKE204,0:REM BLI
NKEN STEuern
480 GETI$:IFI$="_"THENPOKE204,1:PRINT"
MOMENT BITTE":GOTO280
490 PRINT#1,I$;:GOTO450:REM ABFRAGESCHLE
IFE
500 REM
510 REM UNTERPROGRAMME
520 GOSUB710:PRINT" "K$""
530 FORI=0TOK%(J):IFJ<>1THENPRINT"";
540 PRINT" "K$(I,J):NEXT
550 PRINT" > CURS
OR ODER RETURN ";
560 PRINT" > :IFJ<>1THENPRINT
570 IFK(J)=0THEN600
580 FORI=1TOK(J):PRINT"";:IFJ<>1THENPRI
NT
590 NEXT
600 PRINT" "K$(K(J),J)
610 GETK$:IFK$=""THEN610
620 IFK$=""THENIFK(J)>0THENPRINT" "K$
(K(J),J):K(J)=K(J)-1:GOTO560
630 IFK$=""THENIFK(J)<K%(J)THENPRINT" "
K$(K(J),J):K(J)=K(J)+1:GOTO560
640 IFK$<>CHR$(13)THEN610
650 RETURN
660 IF(64ANDPEEK(37150))THEN660:REM PUFF
ER LEEREN
670 RETURN
680 KR=128*K(3)+32*K(2)+K(1):BR=32*K(6)+
16*K(5)+K(4):RETURN
690 FORI=0TO7:POKE828+I,K(I):NEXT:RETURN
700 FORI=0TO7:K(I)=PEEK(828+I):NEXT:RETU
RN
710 PRINT" RS-232 EIN/AUSGABE ":RE
TURN
READY.

```

280 mit der Übernahme der Textblöcke in Stringvariable bis Zeile 300.

In den Zeilen 310 bis 340 werden die Menüs zur Parameter-einstellung (Unterprogramm ab Zeile 520) angewählt und die errechneten Registerinhalte (UP 680) angezeigt. Nachdem die gewünschte Einstellung erfolgt ist, wird diese Schleife verlassen und die Testroutine ab Zeile 370 begonnen.

Darin wird zuerst der Bildschirm vorbereitet (UP 710), dann werden die aktuellen Werte von K(I) gerettet (UP 690) und gewartet, bis der Sendepuffer einer eventuell noch laufenden Übertragung ausgeschrieben ist (UP 660). Danach wird das File geschlossen. Dabei wird der RAM-Bereich der beiden Puffer für Basic wieder freigegeben. Weil hierbei ein CLR-Befehl ausgeführt wird, müssen die vorweg geretteten Parameter zurückgeholt (UP 700) und die Registerinhalte berechnet werden (UP 680). Darauf wird das File mit den neuen Parametern eröffnet, der Bereich wieder für Basic gesperrt und die dabei erneut gelöschten Parameter zurückgeholt (UP 700).

In Zeile 390 wird der Puffer initialisiert. Darauf (410 bis 420) wird ein Zeichensatz übertragen und gleichzeitig invertiert auf den Bildschirm geschrieben. In Zeile 430 wird ein weiterer Testsatz gesendet. Zur Vorbereitung des weiteren Tests mit

Zeichen von der Tastatur her wird der Cursor auf Blinken geschaltet (440).

In den Zeilen 450 bis 490 liegt die Send- und Empfangschleife. Sind alle Zeichen aus dem Empfangspuffer verarbeitet, dann erfolgt gleich der Sprung zur Tastaturabfrage (nach 480). Ein dort eingegebener Rückpfeil beendet den Test und läßt zum Einstellungsmenü verzweigen. Andere Zeichen gehen in den Sendepuffer. Wurde ein Zeichen in Zeile 450 empfangen, dann wird, falls es ein Steuerzeichen ist, gewartet, bis der blinkende Cursor wegschaltet (470), sonst wird das Blinken auch bei aktivem Cursor ausgeschaltet, das Zeichen geschrieben und das Blinken wieder eingeschaltet.

Im Unterprogramm ab Zeile 520 werden sowohl die einzelnen Gruppen als auch die veränderlichen Parameter dieser Gruppen angezeigt, durch Cursor-up- und Cursor-down-Tasten verändert und durch »RETURN« übernommen.

In diesem Testprogramm konnte auf die Statusabfrage der RS232-Schnittstelle verzichtet werden, bei einer Anwendung in einem regulären Treiberprogramm sollte sie aber benutzt werden.

(Hans-Jürgen Kellermann/ev)