

xBasic 64: eine Super-Basic-Erweiterung

Schon oft wurden Basic-Erweiterungen veröffentlicht, die aber immer nur einen bestimmten Bereich bevorzugten. xBasic 64 dagegen umfaßt das gesamte Spektrum des C 64 und stellt insgesamt 44 neue Befehle zur Verfügung.

XBasic 64 wurde für den Commodore 64 ohne besondere Peripherie geschrieben. Lediglich die Diskettenbefehle und die Hardcopy-Funktion benötigen die entsprechenden Geräte (Device 8 und 4).

xBasic 64 belegt den Speicher von \$8000—\$9310. (Näheres hierzu siehe Speicherbelegung im Innenteil, Seite 59ff).

xBasic 64 stellt Befehle für die Grafikprogrammierung, die Tonerzeugung, die Programmerstellung und Fehlersuche, das strukturierte Programmieren und die labelabhängige Verzweigung zur Verfügung.

In Tabelle 1 eine Übersicht über die von xBasic 64 bereitgestellten Befehle.

HRG	GCOLOR	TEXT	SET	RESET
REVERS	PRINT AT	FILL	PAINT	RESCUE
RENUM-	REPEAT	UNTIL	JUMP	CALL
BORDER				INVERS
SUBEND	PROCE-	EXIT	BRANCH	VOL
	DURE			
ENVE-	WAVE	DUMP		HCOPY
LOPE				
MEMORY	CHRCOPY	RAM	NEGATE	CREATE
CHANGE	MOVE	OVER	ERROR	SPRITE
MOBEX	MULTI	CLEAR	AUTO	DIR
BORDER	PAPER	INK	DOKE	KILL
				DELETE

Tabelle 1. Befehlsliste

Insgesamt besitzt xBasic 64 mehr als 40 neue Befehle.

Befehlserklärung:

1. Grafikbefehle

1.1. Hochauflösende Grafik (320 x 200)

HRG Syntax: HRG x,y

Effekt: Der Computer wird in den hochauflösenden Modus versetzt. Es werden die Punktfarben x und die Hintergrundfarbe y gesetzt.

TEXT Syntax: TEXT

Effekt: Der Computer wird wieder in den Textmodus (40 x 25 Positionen) zurückgeschaltet. TEXT wird bei jeder Programmunterbrechung und nach Beendigung eines Programms ausgeführt.

GCOLOR Syntax: GCOLOR (pf,hf,rf)

Effekt: Punktfarbe pf (0 bis 15), Hintergrundfarbe hf (0 bis 15) und Rahmenfarbe rf (0 bis 255) werden gesetzt.

SET Syntax: SET (x,y)

Effekt: Es wird ein Bildpunkt an die durch x (0 bis 319) und y (0 bis 199) festgelegte Koordinate gesetzt. Bei einer Bereichsüberschreitung wird ?INCORRECT XY VALUE gemeldet.

INVERS Syntax: INVERS x,y

Effekt: Es gilt das unter SET Gesagte, jedoch wird der Punkt mit den Koordinaten x/y.

REVERS Syntax: REVERS

Effekt: Der gesamte Grafikschrift wird invertiert.

1.2 Blockgrafik (40 x 25 Positionen) FILL

Syntax: FILL s, z, lx, ly, char

Effekt: Füllt einen rechteckigen Bildschirmbereich mit dem Zeichen char (Bildschirmcode) aus. Dabei ist s die Spalte und z die Zeile der linken oberen Ecke des Rechtecks. lx ist die Länge in der Horizontalen und ly in der Vertikalen. Liegt das Rechteck nicht vollständig im zugelassenen Bereich, wird ?RANGE TOO LARGE angezeigt.

PAINT Syntax: PAINT s, z, lx, ly, col

Effekt: Dieser Befehl wirkt fast wie der zuvor genannte, jedoch wird hier der Bereich mit der Farbe col angefüllt.

BORDER Syntax: BORDER col

Effekt: Der Bildschirmrahmen erhält die Farbe col.

PAPER Syntax: PAPER col

Effekt: Der Bildschirmhintergrund nimmt die Farbe col an.

INK Syntax: INK col

Effekt: Alle folgenden Bildschirmausgaben erfolgen in der Farbe col.

AT Syntax: AT PRINT (x,y) Ausdruck
Ein in Anführungszeichen stehender Text oder der Wert einer Variablen wird auf dem Bildschirm ab Spalte x und Zeile y ausgegeben.

1.2.1. User defined graphics (UDG)

xBasic 64 gestattet auch die Erstellung eines eigenen Zeichensatzes. Dieser liegt ab \$C800. Wird in diesem Modus gearbeitet, so verschiebt sich das Video-RAM nach \$C400 (dez. 50176).

CHRCOPY Syntax: CHRCOPY

Effekt: Der Commodore-Zeichensatz wird an die oben erwähnte Stelle kopiert.

RAM Syntax: RAM

Effekt: Der Commodore 64 wird auf den kopierten Zeichensatz umgestellt.

CREATE Syntax: CREATE char, b1, b2, b3..., b8

Effekt: Mit diesem Befehl wird dem Zeichen char im neuen Zeichensatz ein anderes Bitmuster zugewiesen, welches durch die Bytes b1 bis b8 festgelegt ist.

Alle folgenden Befehle sind auch auf Sprites anwendbar. Darüber entscheidet der erste Parameter. Ist er 0, so bezieht sich der Befehl auf die UDG, ist er 1, wirkt er auf die Spritegrafik.

CHANGE Syntax: CHANGE 0, char, bnr, b oder CHANGE 1, sbl, x, y, b

Effekt: UDG — Das Zeichen mit dem Code char wird verändert. Das Byte mit der Nummer bnr (0 bis 7) wird durch das Byte b ersetzt.

Sprites — Der Sprite-Datenblock mit der Nummer sbl wird verändert. Das Byte, das im Block durch x und y maskiert ist, wird durch b ersetzt.

MOVE Syntax: MOVE 0, char 1, char 2 oder MOVE 1, sbl 1, sbl 2

Effekt: UDG — Das Zeichen mit dem Code char 2 erhält das Bitmuster von char 1.

Sprites — Der Spritedatenblock sbl 1 wird nach sbl 2 kopiert.

OVER Syntax: OVER 0, char 1, char 2 oder OVER 1, sbl 1, sbl 2

Effekt: UDG — Das Zeichen mit dem Code char 1 wird mit dem Bitmuster von Char 2 verflochten.

Sprites — Die Bitmuster von sbl 1 und sbl 2 werden verknüpft, wobei das Ergebnis in sbl 2 steht.

NEGATE Syntax: NEGATE mod, nr

Effekt: Das Zeichen im neuen Zeichensatz mit der Nummer nr (mod=0) oder das Sprite mit der Nummer nr (mod=1) wird invertiert.

1.3. Spritegrafik SPRITE

Syntax: SPRITE snr, x, y, sbl, col, mod

Effekt: Das Sprite mit der Nummer snr (0 bis 7) wird an der durch x (0-511) und y (0 bis 255) festgelegten Position auf den Bildschirm gebracht. Es hat die Farbe col und bezieht sein Bitmuster aus sbl. Ist mod 0, so wird das Sprite im HRG-Modus, ist mod 1, wird es im Multicolormodus dargestellt.

Fortsetzung auf Seite 59


```

1118 DATA 45,53,53,45,20,20,20,3A,20, 490 <172>
1119 DATA 24,00,92,05,20,50,52,4F,47, 531 <180>
1120 DATA 52,41,4D,4D,20,3A,20,00,12, 441 <195>
1121 DATA 20,20,2A,2A,2A,20,46,41,4C, 433 <211>
1122 DATA 53,43,48,45,20,45,49,4E,47, 614 <208>
1123 DATA 41,42,45,20,2A,2A,2A,20,20, 422 <193>
1124 DATA 92,00,0D,0D,2A,2A,2A,20,45, 399 <243>
1125 DATA 4E,44,45,20,2A,2A,2A,00,13, 392 <223>
1126 DATA 05,20,20,12,44,92,49,53,48, 532 <189>
1127 DATA 20,4F,44,45,52,20,12,54,92, 610 <190>
1128 DATA 41,50,45,0D,00,13,20,20,49, 383 <181>
1129 DATA 2F,4F,20,20,20,46,45,48,4C, 522 <247>
1130 DATA 45,52,00,20,D1,B1,20,48,B2, 851 <215>
1131 DATA A0,B3,A9,CF,20,FF,B1,20,8E, 1353 <115>
1132 DATA B4,85,FC,20,8E,B4,85,FB,C5, 1500 <115>
1133 DATA 61,A5,FC,E5,62,90,23,A5,FB, 1436 <098>
1134 DATA C5,5F,A5,FC,E5,60,B0,19,20, 1267 <097>
1135 DATA A7,B4,D0,14,60,20,A7,B4,F0, 1290 <065>
1136 DATA 0C,85,F9,20,A7,B4,F0,05,85, 1151 <066>
1137 DATA FB,4C,EF,B0,68,68,20,43,B3, 1225 <090>
1138 DATA 4C,5F,B4,20,CF,FF,C9,4C,D0, 1330 <146>
1139 DATA 09,20,D1,B1,20,48,B2,4C,0B, 796 <010>
1140 DATA B6,C9,0D,60,A9,00,85,5E,20, 920 <019>
1141 DATA 5F,B4,20,EA,B1,20,0D,B5,24, 980 <040>
1142 DATA 5E,30,05,20,E4,FF,F0,FB,20, 1185 <097>
1143 DATA E1,FF,F0,26,20,9F,B2,24,5E, 1257 <110>
1144 DATA 10,09,20,4E,B5,20,0D,B5,20, 574 <246>
1145 DATA 60,B5,20,33,B2,20,3F,B2,90, 955 <000>
1146 DATA D7,A0,B4,A9,28,20,FF,B1,20, 1260 <095>
1147 DATA E4,FF,C9,0D,D0,F9,A9,00,85, 1456 <141>
1148 DATA 5E,A5,61,85,FB,A5,62,85,FC, 1388 <131>
1149 DATA 20,E0,B2,4C,64,B1,A5,FC,20, 1236 <092>
1150 DATA 4E,B1,A5,FB,85,FF,20,4E,B1, 1346 <144>
1151 DATA A9,20,A0,3A,20,F2,B1,A0,00, 1030 <053>
1152 DATA 20,ED,B1,B1,FB,20,4E,B1,C8, 1361 <128>
1153 DATA C0,08,90,F3,20,ED,B1,24,5E, 1163 <090>
1154 DATA 30,03,A9,12,2C,A9,20,20,D2, 725 <255>
1155 DATA FF,20,10,B2,A5,FF,20,4E,B1, 1188 <123>
1156 DATA A9,92,20,D2,FF,4C,EA,B1,A9, 1468 <157>
1157 DATA FF,85,B8,85,B9,A9,04,85,BA, 1382 <143>
1158 DATA 20,C0,FF,A2,FF,4C,C9,FF,20, 1460 <165>
1159 DATA CC,FF,A9,FF,4C,C3,FF,20,5F, 1536 <215>
1160 DATA B4,A9,80,85,5E,20,4E,B5,20, 1027 <088>
1161 DATA 48,B2,A2,24,A9,20,20,D2,FF, 1159 <121>
1162 DATA CA,D0,FA,20,EA,B1,20,EA,B1, 1546 <162>
1163 DATA 20,60,B5,4C,C1,B4,20,BB,B5, 1155 <093>
1164 DATA A6,5F,A4,60,A9,61,20,DB,FF, 1290 <131>
1165 DATA B0,0A,20,B7,FF,29,BF,D0,03, 1099 <135>
1166 DATA 4C,FB,B4,A9,01,20,C3,FF,20, 1191 <131>
1167 DATA 68,B6,A0,B4,A9,4F,20,FF,B1, 1338 <147>
1168 DATA 20,F9,B1,4C,FB,B4,20,68,B6, 1283 <130>
1169 DATA A9,37,A0,B4,20,FF,B1,20,F9, 1309 <126>
1170 DATA B1,A2,08,C9,44,F0,06,A2,01, 1025 <079>
1171 DATA C9,54,D0,F1,A9,01,AB,20,BA, 1290 <123>
1172 DATA FF,A0,00,E0,01,F0,1A,A9,40, 1139 <110>
1173 DATA 8D,20,02,A9,3A,8D,21,02,B9, 763 <050>
1174 DATA 01,02,99,22,02,C8,CC,00,02, 598 <013>
1175 DATA 90,F4,C8,C8,D0,0C,B9,01,02, 1196 <121>
1176 DATA 99,20,02,C8,CC,00,02,D0,F4, 1045 <092>
1177 DATA 98,A2,20,A0,02,4C,8D,FF,20, 1060 <119>
1178 DATA B8,B5,A5,BA,C9,08,90,33,A6, 1286 <146>
1179 DATA B9,86,57,A9,01,20,C3,FF,A9, 1227 <136>
1180 DATA 60,85,B9,20,C0,FF,B0,28,A5, 1274 <126>
1181 DATA BA,20,B4,FF,A5,B9,20,96,FF, 1440 <174>
1182 DATA 20,A5,FF,85,61,A5,90,4A,4A, 1139 <128>
1183 DATA B0,13,20,A5,FF,85,62,20,AB, 1081 <112>
1184 DATA FF,A5,57,85,B9,A9,00,20,D5, 1239 <141>
1185 DATA FF,90,03,4C,A3,85,86,5F,84, 1183 <144>
1186 DATA 60,A5,BA,C9,01,D0,0A,AD,3D, 1101 <149>
1187 DATA 03,85,61,AD,3E,03,85,62,4C, 778 <063>
1188 DATA FB,B4,A9,13,20,D2,FF,A2,1C, 1306 <168>
1189 DATA 20,ED,B1,CA,D0,FA,60, 1202 <104>

```

Listing von MSE (Schluß)

xBasic 64: Eine Super-Basic-Erweiterung

xBasic 64 gibt Ihnen viele Hilfen, die Sie im Standard-Commodore-Basic nicht finden. Es unterstützt nicht nur Grafik und Musik, sondern besitzt auch eine Reihe von Programmierhilfen und Floppy-Befehlen.

Fortsetzung der Beschreibung von Seite 52.

MOBEX Syntax: MOBEX snr,ex,ey

Effekt: Das Sprite mit der Nummer snr wird expandiert. Bei ex gleich 1 wird es in X-Richtung expandiert, bei ey gleich 1 in Y-Richtung.

MULTI Syntax: MULTI col1,col2

Effekt: Es werden die beiden zusätzlichen Farben col1 und col2 für Multicolorsprites festgelegt.

CLEAR Syntax: CLEAR oder CLEAR snr

Effekt: Im ersten Fall werden alle Sprites vom Bildschirm gelöscht, im zweiten nur das Sprite mit der Nummer snr.

1.4. Sonstige Grafikbefehle

HCOPY Syntax: HCOPY

Effekt: Es wird eine Hardcopy vom Textschirm im Groß-/Grafikmodus erzeugt (Gerätenummer 4).

2. Befehle für die Tonerzeugung

VOL Syntax: VOL lt

Effekt: Setzt Lautstärke lt (0-15) für alle 3 Stimmen.

WAVE Syntax: WAVE st,wf,fl

Effekt: Für die Stimme st (1-3) wird die Wellenform wf (16,32,64 oder 128 gesetzt. fl schaltet die Stimme ein (fl = 1) oder aus (fl = 0).

ENVELOPE Syntax: ENVELOPE st,an,ab,ha,au

Effekt: Setzt die Hüllkurve für Stimme st. Die Parameter an (Anschlag), ab (Abschwellen), ha (Halten) und au (Ausklängen) dürfen im Bereich von 0 bis 15 liegen.

Anmerkung: Wird bei WAVE eine ungültige Wellenform angegeben, so meldet xBasic 64 »?NOT EXISTING WAVE«

3. Programmierhilfen

HELP Syntax: HELP

Effekt: Es werden alle von Exbasic 64 verstandenen Befehle aufgelistet.

MEMORY Syntax: MEMORY

Effekt: Die momentane Speicheraufteilung wird angezeigt (Programm, Variable etc.).

PROGRAMM : xBasic 64 8000 8DFB

8000 :	4E 80 5E FE C3 C2 CD 38 00	8250 :	8A 29 07 18 65 BB 85 BB F5	8480 :	A0 00 B1 FB 85 FD C8 B1 0B
8008 :	30 93 11 20 20 20 20 20 0E	8258 :	A5 BC 69 00 85 BC 18 A5 A0	8488 :	FB 85 FE C8 C8 C8 B1 FB 0A
8010 :	2A 2A 2A 2A 20 43 4F 4D 13	8260 :	14 29 F8 65 BB 85 BB A5 16	84C0 :	C9 DB F0 14 A0 01 B1 FD 0A
8018 :	4D 4F 44 4F 52 45 20 36 44	8268 :	15 65 BC 85 BC 18 A9 E0 05	84C8 :	D0 04 A9 06 D0 C5 A5 FD 93
8020 :	34 20 45 58 42 41 53 49 CF	8270 :	65 BC 85 BC A5 14 29 07 DA	84D0 :	85 FB A5 FE 85 FC D0 D8 D1
8028 :	43 20 2A 2A 2A 2A 0D 0D 8D	8278 :	49 07 AA A9 01 CA 30 03 52	84D8 :	C8 B1 FB F0 E7 18 A5 FB 63
8030 :	20 36 34 4B 20 52 41 4D 16	8280 :	0A D0 FA A0 00 A2 35 28 FF	84E0 :	69 06 85 BB A9 00 65 FC 4F
8038 :	20 53 59 53 54 45 4D 20 A7	8288 :	78 86 01 90 04 11 BB E0 AF	84E8 :	85 BC A4 02 B1 BB D1 B2 DA
8040 :	20 00 8B E3 8D 80 7C A5 DD	8290 :	04 49 FF 31 BB 91 BB A2 DB	84F0 :	D0 DC 88 10 F7 A0 02 B1 42
8048 :	14 81 49 81 86 AE 20 50 9E	8298 :	36 86 01 58 60 20 03 82 75	84F8 :	FB 85 39 C8 B1 FB 85 3A A3
8050 :	FD 20 15 FD 20 A3 FD 20 8A	82A0 :	4C 4C 23 82 20 03 82 18 6C	8500 :	18 A5 FB 69 03 85 7A A5 A8
8058 :	58 FF 58 20 BF E3 A0 00 6B	82A8 :	40 23 82 A2 35 78 86 01 AE	8508 :	FC 69 00 85 7B 68 68 4C 9F
8060 :	8C 83 02 84 37 84 FB A9 9A	82B0 :	A0 00 A2 E0 84 FB 86 FC 51	8510 :	E1 A7 AD EA 07 C9 80 D0 F0
8068 :	A0 85 FC A2 20 B1 FB 91 01	82B8 :	A2 20 B1 FB 49 FF 91 FB 29	8518 :	05 A9 07 4C 84 81 A8 A5 7F
8070 :	FB 88 D0 F9 E6 FC CA D0 46	82C0 :	88 D0 F7 E6 FC CA D0 F2 DA	8520 :	7A 79 80 8E C8 A5 7B 99 34
8078 :	F4 A9 36 85 01 A9 09 A0 42	82C8 :	A2 36 86 01 58 60 20 9E 8D	8528 :	80 8E C8 A5 39 99 80 8E 56
8080 :	80 94 38 8C 84 02 8C A2 82	82D0 :	B7 E0 28 30 03 4C 0D 82 D3	8530 :	C8 A5 3A 99 80 8E C8 8C 45
8088 :	A8 8C 25 A0 8C 06 A6 20 A8	82D8 :	8A 48 20 FD AE 20 9E B7 24	8538 :	EA 07 20 79 00 4C 8F 84 87
8090 :	2D E4 A9 20 8D A0 A8 A9 72	82E0 :	E0 19 B0 F1 68 A8 60 20 45	8540 :	AD EA 07 D0 04 A9 08 D0 8E
8098 :	4C 8D 04 A6 A9 C7 8D A1 D3	82E8 :	CE 82 18 20 F0 FF 20 FD 8D	8548 :	D2 A8 88 B9 80 8E 85 3A CF
80A0 :	A8 A9 CC 8D 24 A0 A9 E2 B5	82F0 :	AE 4C A0 AA 20 CE 82 A9 18	8550 :	88 B9 80 8E 85 39 88 B9 5E
80A8 :	8D 05 A6 20 B1 80 4C 9D F1	82F8 :	04 85 FC A9 00 85 FB 86 5C	8558 :	88 8E 85 7B 88 B9 80 8E 66
80B0 :	E3 A2 08 BD 42 80 9D 00 FD	8300 :	FD 84 FE 18 98 65 FB 85 B2	8560 :	85 7A 8C EA 07 4C F8 A8 AB
80B8 :	03 CA 10 F7 60 A9 36 85 5B	8308 :	FB A5 FC 69 00 85 FC CA F8	8568 :	D0 04 A9 09 D0 AD A9 FF E7
80C0 :	01 20 F3 81 4C 17 84 20 CE	8310 :	30 10 A9 28 18 65 FB 85 5F	8570 :	85 4A 20 8A A3 9A C9 8D C5
80C8 :	8A AD 4C F7 B7 D0 03 4C E1	8318 :	FB A5 FC 69 00 85 FC CA 08	8578 :	F0 04 A9 0A D0 90 68 68 82
80D0 :	1D A8 20 C7 80 20 13 A6 E5	8320 :	10 F0 20 FD AE 20 9E B7 46	8580 :	68 68 20 79 00 4C A0 45
80D8 :	38 A5 5F E9 01 A4 60 4C 47	8328 :	8A 18 65 FE C9 28 30 05 A0	8588 :	A8 D0 04 A9 09 D0 8C AC 71
80E0 :	24 A8 A0 00 84 08 88 CA D9	8330 :	A9 01 4C 84 81 86 FE 20 86	8590 :	EA 07 D0 04 A9 08 D0 83 F0
80E8 :	C8 E8 BD 00 02 38 F9 A0 9F	8338 :	FD AE 20 9E B7 8A 18 65 63	8598 :	88 88 88 88 8C EA 07 20 14
80F0 :	8B F0 F5 C9 80 D0 08 18 89	8340 :	FD C9 19 B0 EB 86 FD 20 A9	85A0 :	79 00 4C A0 A8 20 9E B7 B6
80F8 :	A5 0B 69 CC 4C C7 A5 A6 FE	8348 :	FD AE 20 9E B7 8A 48 A6 B6	85A8 :	E0 10 30 03 4C 48 B2 8E EC
8100 :	7A E6 08 C8 B9 9F 88 10 B0	8350 :	FD A4 FE 91 FB 88 10 FB CD	85B0 :	18 D4 60 A9 05 A0 D4 85 33
8108 :	FA E9 A0 8B D0 DC BD 00 63	8358 :	18 A5 F9 69 28 85 FB A5 59	85B8 :	FB 84 FC 20 E8 85 A2 02 82
8110 :	02 4C 07 A6 30 03 4C F3 03	8360 :	FC 69 00 85 FC 68 48 CA 8B	85C0 :	86 BB 20 DC 85 0A 0A 0A AC
8118 :	A6 C9 FF F0 F9 24 0F 30 1E	8368 :	10 E7 68 60 20 CE 82 A9 68	85C8 :	0A 85 BC 20 DC 85 18 65 ED
8120 :	F5 C9 CC B0 03 4C 24 A7 85	8370 :	D8 85 FC 4C FB 82 A5 2B 94	85D0 :	BC A0 00 91 FB E6 FB C6 83
8128 :	38 E9 CB AA 84 A9 A0 FF B2	8378 :	A4 2C 85 22 84 23 A0 03 C2	85D8 :	5B D0 E7 60 20 FD AE 20 E3
8130 :	CA F0 08 C8 B9 A0 88 10 7C	8380 :	C8 B1 22 D0 FB C8 98 18 5C	85E0 :	9E B7 E0 10 B0 C6 8A 60 C0
8138 :	FA 30 F5 C8 B9 A0 88 10 D0	8388 :	65 22 A0 00 91 2B A5 23 76	85E8 :	20 9E B7 8A F0 BE E0 04 27
8140 :	03 4C EF A6 20 47 AB D0 C7	8390 :	69 00 C8 91 2B 88 A2 03 E5	85F0 :	B0 BA CA F0 09 18 A9 07 D4
8148 :	F2 20 73 00 20 52 81 4C 5A	8398 :	E6 22 D0 02 E6 23 B1 22 96	85F8 :	65 FB 85 FB D0 F4 60 A9 C5
8150 :	AE A7 D0 01 60 E9 80 90 9F	83A0 :	D0 F4 CA D0 F3 A5 22 69 7F	8600 :	04 A0 D4 85 FB 84 FC 20 52
8158 :	15 C9 4C B0 17 C9 23 B0 29	83A8 :	02 85 2D A5 23 A9 00 85 F5	8608 :	8E 85 20 FD AE 20 9E B7 50
8160 :	10 0A A8 B9 0D A0 48 B9 41	83B0 :	2E 4C 63 A6 20 8A AD 20 FF	8610 :	8A A2 03 D0 35 86 F0 08 C3
8168 :	0C A0 48 4C 73 00 4C A5 14	83B8 :	F7 B7 A5 14 85 FB A5 15 70	8618 :	CA 10 FB A9 0C 4C 84 81 96
8170 :	A9 4C 0E A8 38 E9 4C 0A F0	83C0 :	85 FC 20 FD AE 20 8A AD FD	8620 :	85 02 20 FD AE 20 9E B7 44
8178 :	A8 B9 81 8C 48 A9 80 8C 5C	83C8 :	20 F7 B7 A5 2B A4 2C 85 1A	8628 :	E0 02 B0 C0 18 8A 65 02 BD
8180 :	48 4C 6B 81 0A AA BD E0 A8	83D0 :	22 84 23 A0 03 A5 FC 91 86	8630 :	A0 00 91 FB 60 10 20 40 3C
8188 :	8D 85 22 8D E1 8D 4C 45 5E	83D8 :	22 88 A5 FB 91 22 88 B1 D7	8638 :	80 A5 2D A4 2E 85 14 84 D3
8190 :	A4 A9 38 8D 11 D0 A9 08 D8	83E0 :	22 F0 25 85 25 88 B1 22 16	8640 :	15 C4 30 D0 02 C5 2F B0 4A
8198 :	8D 18 D0 A9 00 BD 00 DD C3	83E8 :	85 24 A0 02 18 B1 22 65 4A	8648 :	18 69 02 90 01 C8 85 22 58
81A0 :	A0 00 A2 E0 84 FB 86 FC 41	83F0 :	14 91 24 C8 B1 22 65 15 DB	8650 :	84 23 20 7F 86 20 B3 86 A3
81A8 :	A2 20 98 91 FB 88 D0 FB F2	83F8 :	91 24 B0 0D A5 24 85 22 3F	8658 :	8A 10 07 20 BC 86 4C 6E BE
81B0 :	E6 FC CA D0 F6 60 20 FA CA	8400 :	A5 25 85 23 A0 01 D0 D7 03	8660 :	86 60 98 30 06 20 CC 86 E4
81B8 :	AE 20 EB B7 A5 14 C9 10 AA	8408 :	60 A0 02 A9 00 91 24 C8 1D	8668 :	4C 6E 86 20 D5 86 A9 0D E3
81C0 :	30 03 4C 48 B2 A0 0A 0A 46	8410 :	91 24 A9 02 4C 84 81 A9 A0	8670 :	20 D2 FF A5 14 A4 15 18 99
81C8 :	0A 85 14 E0 10 B0 F3 8A 21	8418 :	00 8D E9 07 8D EA 07 4C 1F	8678 :	69 07 90 C1 C8 B0 BE A0 0F
81D0 :	18 65 14 A0 00 A2 C0 84 D5	8420 :	83 A4 AC E9 07 C0 80 D0 78	8680 :	0F B1 14 AA 29 7F 20 D2 68
81D8 :	FB 86 FC A2 04 91 FB 88 78	8428 :	05 A9 03 4C 84 81 20 06 2D	8688 :	FF C8 B1 14 AB 29 7F 08 BE
81E0 :	D0 FB E6 FC CA D0 F6 20 56	8430 :	A9 18 98 65 7A AC E9 07 7B	8690 :	03 20 D2 FF 8A 10 11 98 F6
81E8 :	FD AE 20 9E B7 8E 20 D0 2A	8438 :	99 00 8E C8 A5 7B 69 00 6A	8698 :	30 0A A9 2A 20 D2 FF 68 E6
81F0 :	4C F7 AE A9 1B 8D 11 D0 1D	8440 :	99 00 8E C8 A5 39 99 00 20	86A0 :	68 4C 6E 86 A9 25 D0 4E 3E
81F8 :	A9 15 8D 18 D0 A9 03 8D 14	8448 :	8E C8 A5 3A 99 00 8E C8 50	86A8 :	98 10 04 A9 24 D0 47 60 25
8200 :	00 D0 60 20 FA AE 20 EB 88	8450 :	8C E9 07 60 20 9E AD A5 98	86B0 :	20 D2 FF A9 20 20 D2 FF BD
8208 :	B7 E0 C8 90 05 A9 00 4C AA	8458 :	61 D0 24 AC E9 07 D0 04 E2	86B8 :	A9 3D D0 3A A0 00 B1 22 90
8210 :	84 81 A5 15 C9 01 90 08 58	8460 :	A9 04 D0 C7 88 99 00 8E AC	86C0 :	AA C8 B1 22 A8 8A 20 95 09
8218 :	D0 F3 A5 14 C9 40 B0 ED 0B	8468 :	85 3A 88 B9 00 8E 85 39 61	86C8 :	B3 4C CF 86 20 A6 BB 20 CC
8220 :	4C F7 AE 08 8A 29 FB 85 F5	8470 :	88 B9 00 8E 85 7B 88 B9 70	86D0 :	DD 8D 4C 1E AB 20 F4 86 FF
8228 :	FE 85 BB A9 00 85 BC 06 38	8478 :	00 8E 85 7A 60 AE A7 AC E3	86D8 :	A0 02 B1 22 85 25 88 B1 31
8230 :	BB 26 BC 06 BB 26 BC 18 FE	8480 :	E9 07 88 88 88 8C E9 F3	86E0 :	22 85 24 88 B1 22 85 26 6E
8238 :	A5 BB 65 FE 85 BB A5 BC 3A	8488 :	07 20 79 00 4C ED A7 D0 72	86E8 :	F0 0A B1 24 20 D2 FF C8 F8
8240 :	69 00 85 BC 06 BB 26 BC F3	8490 :	05 A9 05 4C 84 81 8D 3C 38	86F0 :	C4 26 D0 F6 A9 22 4C D2 5D
8248 :	06 BB 26 BC 06 BB 26 BC 9D	8498 :	03 A2 00 20 73 00 F0 06 F7	86F8 :	FF A9 0D 20 D2 FF A0 A0 04
		84A0 :	9D 3D 03 E8 D0 F5 86 02 95	8700 :	A9 8B 84 FB 85 FC A0 00 D2
		84A8 :	A5 2B 85 FB A5 2C 85 FC 8F	8708 :	84 07 B1 FB F0 44 10 3C E5

DUMP Syntax: DUMP

Effekt: Alle Variable und deren Inhalt werden ausgegeben (Real,Integer=%,Strings=\$,Funktionen=*)

START Syntax: START

Effekt: xBasic 64 wird neu initialisiert. Das derzeit im Speicher befindliche Programm kann mit RESCUE gerettet werden.

RESCUE Syntax: RESCUE

Effekt: Zurückholen eines Programms nach NEW oder START.

RENUMBER Syntax: RENUMBER stz,sw

Effekt: Das im Speicher befindliche Basic-Programm wird neu durchnummeriert. Dabei ist stz die erste Zeile des »neuen« Programms und sw die Schrittweite. Wird der für Zeilenummern erlaubte Bereich überschritten, so wird die erste nicht mehr unterzubringende Zeile mit der Nummer 0 belegt und »ILLEGAL LINENUMBER« ausgegeben.

8710 : 38 E9 80 20 D2 FF E6 07 38
8718 : A5 07 C9 04 D0 0E A9 0D 72
8720 : 20 D2 FF C8 B4 02 A9 00 C1
8728 : 85 07 F0 11 C8 B4 02 38 B8
8730 : A9 0A E5 02 AA A9 20 20 51
8738 : D2 FF CA D0 FA 18 A5 02 E1
8740 : 65 FB 85 FB 90 02 E6 FC 32
8748 : A0 00 F0 BE 20 D2 FF C8 26
8750 : D0 B8 A9 0D 4C D2 FF A9 37
8758 : 04 85 BA 85 72 A9 7E 85 F8
8760 : B8 A9 00 85 71 85 87 85 C8
8768 : B9 20 C0 FF A6 B8 20 C9 A6
8770 : FF A2 19 A9 0D 20 D2 FF 59
8778 : 20 E1 FF F0 2E A0 00 B1 F2
8780 : 71 85 67 29 3F 06 67 24 B0
8788 : 67 10 02 09 80 70 02 09 3F
8790 : 40 20 D2 FF C8 C0 28 D0 6A
8798 : E6 98 18 65 71 85 71 90 A7
87A0 : 02 E6 72 CA D0 CD A9 0D 47
87A8 : 20 D2 FF 20 CC FF A9 7E A6
87B0 : 4C C3 FF A9 26 A0 88 20 DD
87B8 : 1E AB 38 A5 2D E5 2B AA 73
87C0 : A5 2E E5 2C 20 1C 88 A9 D4
87C8 : 33 A0 88 20 1E AB 38 A5 DD
87D0 : 2F E5 2D AA A5 30 E5 2E 62
87D8 : 20 1C 88 A9 40 A0 88 20 C9
87E0 : 1E AB 38 A5 31 E5 2F AA EB
87E8 : A5 32 E5 30 20 1C 88 A9 7E
87F0 : 4D A0 88 20 1E AB 38 A5 1F
87F8 : 37 E5 33 AA A5 38 E5 34 60
8800 : 20 1C 88 A9 5A A0 88 20 93
8808 : 1E AB 38 A5 33 E5 31 AA 3B
8810 : A5 34 E5 32 20 1C 88 A9 E7
8818 : 0D 4C D2 FF 20 CD BD A9 BB
8820 : 67 A0 88 4C 1E AB 0D 50 97
8828 : 52 4F 47 52 41 4D 4D 20 32
8830 : 3A 20 00 0D 56 41 52 49 67
8838 : 41 42 4C 45 20 3A 20 00 AA
8840 : 0D 41 52 52 41 59 53 20 39
8848 : 20 20 3A 20 00 0D 53 54 69
8850 : 52 49 4E 47 53 20 3A EF
8858 : 20 00 0D 46 52 45 45 20 29
8860 : 20 20 20 20 3A 20 00 20 81
8868 : 42 59 54 45 00 A0 00 84 23
8870 : FB 84 FD A9 C8 B5 FE A9 6A
8878 : D0 85 FC 78 A2 33 86 01 39
8880 : A2 08 B1 FB 91 FD 88 D0 DF
8888 : F9 E6 FC E6 FE CA D0 F2 80
8890 : A2 36 86 01 58 60 A9 00 3E
8898 : 8D 00 DD 8D FE B1 A9 12 15
88A0 : 8D 18 D0 8D F9 81 A9 C4 FB
88A8 : 8D 88 02 8D FB 82 60 A9 24
88B0 : 03 8D 00 DD 8D FE B1 A9 60
88B8 : 15 8D 18 D0 8D F9 81 A9 86
88C0 : 04 8D 88 02 8D FB 82 60 59
88C8 : 20 9E B7 86 FB A9 00 85 0E
88D0 : FC A2 03 06 FB 26 FC CA 19
88D8 : D0 F9 18 A9 C8 65 FC 85 97
88E0 : FC 60 20 C8 88 B6 02 20 32
88E8 : FD AE 20 9E B7 8A A4 02 7F
88F0 : 91 FB C8 C0 08 F0 04 84 EA
88F8 : 02 D0 EC 60 20 AB 89 B0 78
8900 : 21 20 FD AE 20 C8 88 20 31
8908 : FD AE 20 9E B7 E0 08 30 3B
8910 : 03 4C 48 B2 8A 48 20 FD 09
8918 : AE 20 9E B7 68 AB 8A 91 8E
8920 : FB 60 20 FD AE 20 4F 89 4F
8928 : 20 FD AE 20 9E B7 E0 03 28
8930 : B0 DF 86 FD 20 FD AE 20 1E
8938 : 9E B7 E0 15 B0 D3 8A F0 42
8940 : 08 18 A9 00 69 03 CA D0 3A
8948 : FB 65 FD 48 4C 16 89 20 5A
8950 : 8A AD 20 F7 B7 A5 15 C9 48
8958 : 02 B0 B6 A5 14 85 FB A5 BD
8960 : 15 85 FC A2 06 06 FB 26 98
8968 : FC CA D0 F9 60 20 AB 89 F9

8970 : B0 1E 20 FD AE 20 C8 88 17
8978 : A5 FB 85 FD A5 FC 85 FE 92
8980 : 20 FD AE 20 C8 88 A0 07 B0
8988 : B1 FD 91 FB 88 10 F9 60 CD
8990 : 20 FD AE 20 4F 89 A5 FB 2E
8998 : 85 FD A5 FC 85 FE 20 FD F1
89A0 : AE 20 4F 89 A0 3E D0 E0 64
89A8 : 20 9E B7 E0 02 30 03 4C 68
89B0 : 48 B2 38 CA 10 01 18 60 E3
89B8 : 20 AB 89 B0 20 20 FD AE FD
89C0 : 20 C8 88 A5 FB 85 FD A5 4A
89C8 : FC 85 FE 20 FD AE 20 C8 B2
89D0 : 88 A0 07 B1 FD 11 FB 91 1C
89D8 : FB 88 10 F7 60 20 FD AE 77
89E0 : 20 4F 89 A5 FB 85 FD A5 EE
89E8 : FC 85 FE 20 FD AE 20 4F DF
89F0 : 89 A0 3E D0 DE 20 AB 89 18
89F8 : B0 12 20 FD AE 20 C8 88 99
8A00 : A0 07 B1 FB 49 FF 91 FB E2
8A08 : 88 10 F7 60 20 FD AE 20 8F
8A10 : 4F 89 A0 3E D0 EC 20 9E 46
8A18 : B7 E0 08 B0 92 86 FD A9 00
8A20 : 01 CA 30 04 0A CA 10 FC 44
8A28 : 85 FE 60 20 16 8A 20 FD 7B
8A30 : AE 20 EB B7 A5 15 C9 02 0E
8A38 : B0 E1 8A 48 A5 FD 0A AA 4C
8A40 : A5 14 9D 00 D0 E8 68 9D 88
8A48 : 00 D0 A5 15 D0 0A A5 FE AE
8A50 : 49 FF 2D 19 D0 4C 5D 8A 01
8A58 : A5 FE 0D 10 D0 8D 10 D0 1D
8A60 : 20 FD AE 20 9E B7 8A A6 4E
8A68 : FD 9D FB 07 20 FD AE 20 40
8A70 : 9E B7 8A A6 FD 9D 27 D0 6C
8A78 : 20 FD AE 20 2B 89 B0 0A F4
8A80 : A5 FE 49 FF 2D 1C 0D 4C 86
8A88 : 8F 8A A5 FE 0D 1C D0 8D B6
8A90 : 1C D0 A5 FE 0D 15 D0 8D 35
8A98 : 15 D0 60 20 16 8A 20 FD 64
8AA0 : AE 20 AB 89 B0 0A A5 FE AA
8AA8 : 49 FF 2D 1D D0 4C B5 8A 3B
8AB0 : A5 FE 0D 1D D0 8D 1D D0 4B
8AB8 : 20 FD AE 20 AB 89 B0 0A 34
8AC0 : A5 FE 49 FF 2D 17 D0 4C 9E
8AC8 : CF 8A A5 FE 0D 17 D0 8D 0D
8AD0 : 17 D0 60 20 9E B7 8A 8D 58
8AD8 : 25 D0 20 FD AE 20 9E B7 03
8AE0 : 8A 8D 26 D0 60 D0 06 A9 CC
8AE8 : 00 8D 15 D0 60 20 16 8A 83
8AF0 : A5 FE 49 FF 2D 15 D0 8D 40
8AF8 : 15 D0 60 A9 08 85 BA 20 9A
8B00 : B4 FF A9 6F 85 B9 20 96 E0
8B08 : FF 20 A5 FF 20 D2 FF C9 AD
8B10 : 0D D0 F6 20 AB FF 60 A9 D6
8B18 : 08 85 BA A9 60 85 B9 A9 33
8B20 : 01 85 B7 A9 24 85 FD A9 C1
8B28 : FD 85 B8 A9 00 85 BC 20 6B
8B30 : D5 F3 A5 BA 20 B4 FF A5 B2
8B38 : B9 20 96 FF A9 00 85 90 79
8B40 : A0 03 84 FD 20 A5 FF 85 7D
8B48 : FE A4 90 D0 2F 20 A5 FF 61
8B50 : A4 90 D0 28 A4 FD 88 D0 73
8B58 : E9 A6 FE 20 CD BD A9 20 0A
8B60 : 20 D2 FF 20 A5 FF A6 90 03
8B68 : D0 12 AA F0 06 20 D2 FF B7
8B70 : 4C 63 88 A9 0D 20 D2 FF A3
8B78 : A0 02 D0 C6 4C 42 F6 20 19
8B80 : 9E B7 8E 20 D0 60 20 9E 6F
8B88 : B7 8E 21 D0 60 20 9E B7 DA
8B90 : 8E 86 02 60 4C E2 FC FF BE
8B98 : FF FF FF FF FF FF FF FF 97
8BA0 : 48 52 C7 47 43 4F 4C 4F 6B
8BA8 : D2 4E 52 CD 53 45 D4 52 47
8BB0 : 45 53 45 D4 52 45 56 45 BE
8BB8 : 52 D3 40 50 52 49 4E D4 60
8BC0 : 46 49 4C CC 50 41 49 4E 2B
8BC8 : D4 52 45 53 43 55 C5 52 1C

8BD0 : 45 4E 55 4D 42 45 D2 52 7A
8BD8 : 45 50 45 41 D4 55 4E 54 99
8BE0 : 49 CC 4A 55 4D D0 43 41 B7
8BE8 : 4C CC 50 52 4F 43 45 44 A5
8BF0 : 55 52 C5 53 55 42 45 4E 63
8BF8 : C4 45 58 49 D4 42 52 41 C9
8C00 : 4E 43 C8 56 4F CC 45 4E FA
8C08 : 56 45 4C 4F 50 C5 57 41 11
8C10 : 56 C5 44 55 4D D0 48 45 0C
8C18 : 4C D0 48 43 4F 50 D9 4D C0
8C20 : 45 4D 4F 52 D9 43 48 52 A7
8C28 : 43 4F 50 D9 52 41 CD 52 6D
8C30 : 4F CD 43 52 45 41 54 05 BC
8C38 : 43 48 41 4E 47 C5 4D 4F 30
8C40 : 56 C5 4F 56 45 D2 49 4E C4
8C48 : 56 45 52 D3 53 50 52 49 E3
8C50 : 54 C5 4D 4F 42 45 D8 4D 10
8C58 : 55 4C 54 C9 43 4C 45 41 50
8C60 : D2 45 52 52 4F D2 44 49 E3
8C68 : D2 42 4F 52 44 45 D2 50 D4
8C70 : 41 50 45 D2 49 4E CB 53 62
8C78 : 54 41 52 D4 00 00 00 00 9C
8C80 : 90 81 B5 81 F2 81 9C 82 21
8C88 : A3 82 AA 82 E6 82 F3 82 BF
8C90 : 68 83 75 83 B3 83 21 84 6F
8C98 : 53 84 8E 84 11 85 3A A9 DB
8CA0 : 3F 85 67 85 88 85 A4 85 7F
8CA8 : B2 85 FE 85 38 86 FB 86 36
8CB0 : 56 87 B2 87 6C 88 95 88 DA
8CB8 : AE 88 E1 88 FB 88 6C 89 FC
8CC0 : B7 89 F4 89 2A 8A 9A 8A 21
8CC8 : D2 8A E4 8A FA 8A 16 8B DD
8CD0 : 7E 88 85 88 CC 88 93 88 71
8CD8 : 00 00 00 00 00 00 00 00 D9
8CE0 : 49 4E 43 4F 52 52 45 43 5E
8CE8 : 54 20 58 59 20 56 41 4C E0
8CF0 : 55 C5 52 41 4E 47 45 20 59
8CF8 : 54 4F 4F 20 4C 41 52 47 72
8D00 : C5 49 4C 4C 45 47 41 4C 33
8D08 : 20 4C 49 4E 45 4E 55 4D 21
8D10 : 42 45 D2 54 4F 4F 20 4D BF
8D18 : 41 4E 59 20 52 45 50 45 F6
8D20 : 41 54 2E 2E 2E 20 55 4E B3
8D28 : 54 49 CC 55 4E 54 49 4C 44
8D30 : 20 57 49 54 48 4F 55 54 D6
8D38 : 20 52 45 50 45 41 20 4D 29
8D40 : 49 53 53 49 4E 47 D0 4C 69
8D48 : 41 42 45 CC 50 52 4F 43 F1
8D50 : 45 44 55 52 45 50 4E 4F 84
8D58 : 54 20 46 4F 55 4E C4 54 BB
8D60 : 4F 4F 20 4D 41 4E 59 20 35
8D68 : 43 41 4C 4C 2E 2E 2E 20 36
8D70 : 53 55 42 45 4E C4 53 55 AA
8D78 : 42 45 4E 44 20 57 49 54 04
8D80 : 48 4F 55 54 20 43 41 4C 09
8D88 : CC 4D 49 53 53 49 4E 47 FF
8D90 : 20 4C 49 4E 45 4E 55 4D A9
8D98 : 42 45 D2 45 58 49 54 20 3B
8DA0 : 57 49 54 48 4F 55 54 20 EB
8DA8 : 47 4F 53 55 C2 42 52 41 20
8DB0 : 4E 43 48 20 57 49 54 48 58
8DB8 : 4F 55 54 20 43 41 4C CC D4
8DC0 : 4E 4F 54 20 45 58 49 53 B2
8DC8 : 54 49 4E 47 20 57 41 56 AC
8DD0 : C5 00 00 00 00 00 00 00 96
8DD8 : 00 00 00 00 00 00 00 00 D9
8DE0 : E0 8C F2 8C 01 8D 13 8D 38
8DE8 : 2B 8D 3F 8D 4C 8D 5F 8D 25
8DF0 : 76 8D 89 8D 9B 8D AD 8D 39
8DF8 : C0 8D 2F 00 FF 00 FF 00 4B

Listing des Monats: xBasic 64. Benutzen Sie zur Eingabe den MSE, den Sie ebenfalls in dieser Ausgabe finden.

4. Diskettenbefehle

DIR Syntax: DIR

Effekt: Das Directory der momentan eingelegten Diskette wird angezeigt, ohne das im Speicher befindliche Programm zu löschen.

ERROR Syntax: ERROR

Effekt: Tritt ein Diskettenfehler auf, so wird mit diesem Befehl die Fehlernummer, der Fehlertext, der Sektor und die Spur ge-

holt, in der der Fehler auftrat. Das Ganze wird durch Kommata getrennt ausgegeben.

5. Befehle für strukturiertes Programmieren

REPEAT...UNTIL

Syntax: REPEAT (beliebige Anweisungen) UNTIL (Bedingung)

Effekt: Die Anweisung(en) zwischen REPEAT und UNTIL werden solange ausgeführt, bis die Bedingung hinter UNTIL erfüllt ist. Die Befehle dürfen in zwei Zeilen stehen.

Es dürfen maximal 32 REPEAT...UNTIL-Schleifen geschachtelt werden. Ansonsten erfolgt die Meldung »?TOO MANY REPEAT...UNTIL«. Ein UNTIL ohne vorhergehendes REPEAT führt zu dem Fehler »?UNTIL WITHOUT REPEAT«.

JUMP Syntax: JUMP Label

Effekt: Es erfolgt ein Sprung zum Label (Label). Dieses muß in einer PROCEDURE-Zeile stehen. Ein JUMP ohne nachfolgendes Label verursacht den Fehler »?MISSING LABEL«. Wird versucht, zu einem nicht existierenden Label zu verzweigen, wird »?PROCEDURE NOT FOUND« gemeldet.

PROCEDURE Syntax: PROCEDURE Label

Effekt: Markiert eine Zeile als Labelzeile. Der Befehl selbst wird wie ein REM behandelt. Deshalb dürfen vor und hinter PROCEDURE keine anderen Statements stehen.

CALL...SUBEND Syntax: CALL Label...SUBEND

Effekt: Hierbei handelt es sich um eine Art GOSUB...RETURN. Jedoch wird hier nicht zu einer Zeilennummer, sondern zu der durch Label gekennzeichneten Labelzeile gesprungen. Bei SUBEND springt das Programm dann auf das Statement, das unmittelbar hinter dem entsprechenden CALL steht. Auch hier gilt: maximal 32 CALL...SUBEND-Aufrufe schachteln, sonst gibt's einen ?TOO MANY CALL...SUBEND-Fehler. Ein SUBEND ohne CALL gibt »?SUBEND WITHOUT CALL«.

EXIT Syntax: EXIT Zeilennummer

Effekt: Mit diesem Befehl kann beliebig oft aus einem Unterprogramm ohne RETURN herausgesprungen werden. Das Programm verzweigt zu der hinter EXIT angegebenen Zeilennummer. Fehlt sie, so wird ein ?MISSING LINENUMBER-Fehler ausgegeben.

BRANCH Syntax: BRANCH Zeilennummer

Effekt: Ermöglicht ein Verlassen eines Unterprogrammes ohne SUBEND. Ansonsten gilt das bei EXIT Gesagte.

xBasic 64 ermöglicht außerdem das Setzen des DATA-Zeigers auf eine bestimmte Zeilennummer mit RESTORE. Auch Variable sind als Ausdruck zugelassen. Das gleiche gilt auch für GOTO und GOSUB.

Ein Nachteil von xBasic 64 ist, daß ein xBasic 64-Befehl bei einer IF...THEN-Anweisung durch einen Doppelpunkt vom THEN getrennt werden muß.

Beispiel:

```
100 IF A=SIN(45) THEN:EXIT 500
```

Zur Beachtung:

Verwechseln Sie bei den Spritebefehlen nicht Spritenummer und Spritedatenblocknummer. Die Spritenummer bezieht sich auf die auf dem Schirm darstellbaren Sprites, während sich die Spritedatenblocknummer auf den Datenblock bezieht, aus welchem die Sprites ihre Bitmuster beziehen können. Die Spritenummer darf im Bereich von 0 bis 7 liegen, während die Datenblocknummer Werte von 0 bis 511 annehmen kann. Achten Sie jedoch darauf, daß die im freien RAM liegenden Blöcke vor dem Überschreiben geschützt werden. Die Blöcke 0 bis 10 bitte nicht benutzen!

Der Spritepointer kann natürlich nur die Blöcke 0 bis 255 adressieren. Um auf die anderen Blöcke zugreifen zu können, müssen diese erst mit MOVE in den entsprechenden Bereich geschoben werden.

xBasic 64 arbeitet mit drei verschiedenen Video-RAMs. Das »normale« Video-RAM ab \$0400, das HRG-Video-RAM, das als Farb-RAM dient, ab \$C000 und das Video-RAM bei Gebrauch der UDG ab \$C400.

Weiter belegt xBasic 64 die Bereiche \$8E00-8E7F als REPEAT...UNTIL-Stack und \$8E80-8EFF als CALL...SUBEND-Stack.

Programmtechnische Details

Zur Initialisierung bedient sich xBasic 64 eines weitverbreiteten Verfahrens: Damit der Commodore 64 ein ROM-Modul beim Einschalten erkennt, muß dieses ab Adresse \$8004 das Kennwort »CBM80« enthalten. Die beim Einschalten angesprungene RESET-Routine ruft ihrerseits eine Routine auf, die den erwähnten Speicherbereich nach dem Wort durchsucht. Wird es gefunden, so springt der Computer in der Programmausführung an eine Adresse, die in den Speicherstellen \$8000 und \$8001 abgelegt ist. Durch Einstecken einer ROM-Kassette am Expansionsport wird der Eingang EXROM auf 0 gesetzt. Dadurch wird der 64er auf ROM umgestellt. Ist EXROM gleich 1, so wird bei einem RESET einfach der RAM-Bereich ab \$8004 auf das Kennwort untersucht. Und genau hier setzt xBasic 64 ein. Es enthält nämlich an dieser Stelle das geforderte Kennwort. Der Computer springt also nicht auf den im ROM stehenden Vektor (\$A000), sondern auf den von xBasic 64 bereitgestellten (\$8000). Dieser zeigt auf eine eigene RESET-Routine. Diese kopiert zum Beispiel das Basic-ROM in das darunterliegende RAM. Nun werden die Veränderungen an den Routinen GOTO, GOSUB und RESTORE vorgenommen. Also Maschinenprogrammierer Vorsicht! In Verbindung mit xBasic 64 ist der Bereich von \$A000 bis \$BFFF tabu!

Oberhalb von xBasic 64 ist der für Maschinenprogramme freie Teil nur der Bereich von \$9000 bis \$9FFF.

xBasic 64 besitzt einige neue Fehlermeldungen. Die entsprechende Routine liegt ab \$8184. Sie wird mit der Nummer der Fehlermeldung im Akku angesprungen. Sie holt sich dann die Adresse der Fehlermeldung aus einer Tabelle (\$8DE0).

Geben Sie einmal folgendes ein:

```
POKE 780,2:SYS 33156
```

Sie müßten die Meldung »?ILLEGAL LINENUMBER ERROR« erhalten. Das ist die Fehlermeldung mit der Nummer 2. Die Nummern 0 bis 12 werden von xBasic 64 belegt, die restlichen sind jedoch für eigene Fehlermeldungen frei.

Für die Befehle REPEAT...UNTIL und CALL...SUBEND benutzt xBasic 64 zwei Stacks. Auf diesen Stacks wird beim Aufruf von REPEAT oder CALL die momentane Zeilennummer und der Programmzeiger abgelegt.

Die Befehle UNTIL und SUBEND holen diese Werte wieder zurück. Jeder Stack hat seinen Pointer, der auf die Elemente im Stack zeigt. Die beiden Pointer liegen bei \$07E8 (RU-Stack) und \$07E9 (CS-Stack). Die beiden Stacks liegen ab \$8E00 (RU) und 8E80 (CS).

Mit den von xBasic 64 zur Verfügung gestellten Routinen sollte es keine Schwierigkeit sein, eigene Befehle einzubauen. Mit dem entsprechenden Algorithmus und den Routinen Koordinaten holen und Punkt setzen, ist es bestimmt nicht sehr kompliziert, eine LINE-Routine oder andere Grafikbefehle zu implementieren (siehe auch Tabelle 1).

(Stephan Blietz/gk)

Die Eingabe von xBasic 64

xBasic 64 muß mit dem MSE eingegeben werden, den Sie ebenfalls in diesem Heft finden oder bereits aus den vorhergehenden 64'er-Ausgaben kennen. Nach dem Abtippen speichern Sie das Programm auf jeden Fall vor dem ersten Test.

xBasic 64 läßt sich mit LOAD "Name",8,1 von der Diskette (oder LOAD von der Kassette) laden und mit SYS 64738 (!) starten.

Tabelle 2. Die Speicherbelegung von xBasic 64.

\$8000	Startvektor	\$8635	Tabelle der Wellenformen
\$8002	NMI-Vektor	\$8639	Befehl DUMP
\$8004	>CBM80<	\$86F9	Befehl HELP
\$8009	Aktivierungsmeldung	\$8757	Befehl HCOPY
\$8042	Tabelle der Basic-Vektoren	\$87B3	Befehl MEMORY
\$804E	RESET-Routine	\$8826	Ausgabemaske für MEMORY
\$80B1	Basic-Vektoren laden	\$886D	Befehl CHRCOPY
\$80BD	Basic-Warmstart	\$8896	Befehl RAM
\$80C7	Erweiterung für GOTO, GOSUB und RESTORE	\$88AF	Befehl ROM
\$80E2	xBasic 64-Befehle codieren	\$88C8	Holt Zeichennummer und setzt Zeiger
\$8114	xBasic 64-Befehle decodieren	\$88E2	Befehl CREATE
\$8149	Befehlsadresse holen	\$88FC	Befehl CHANGE
\$8184	Neue Fehlermeldung ausgeben	\$894F	Holt Spriteblock und setzt Zeiger
\$8191	Befehl HRG	\$896D	Befehl MOVE
\$81B6	Befehl GCOLOR	\$89A8	Holt Flag und verändert Carry-Bit
\$81F3	Befehl NRM	\$89B8	Befehl OVER
\$8203	Hires-Koordinatenpaar holen	\$89F5	Befehl INVERS
\$8223	Hires-Punkt setzen/löschen	\$8A16	Holt Spritenummer und berechnet Byte
\$829D	Befehl SET	\$8A2B	Befehl SPRITE
\$82A4	Befehl RESET	\$8A9B	Befehl MOBEX
\$82AB	Befehl REVERS	\$8AD3	Befehl MULTI
\$82CE	Lores-Koordinatenpaar holen	\$8AE5	Befehl CLEAR
\$82E7	Befehl @PRINT	\$8AFB	Befehl ERROR
\$82F4	Befehl FILL	\$8B17	Befehl DIR
\$836C	Befehl PAINT	\$8B7F	Befehl BORDER
\$8376	Befehl RESCUE	\$8B86	Befehl PAPER
\$83B4	Befehl RENUMBER	\$8B8D	Befehl INK
\$8417	RU- und CS-Stackpointer auf 0 setzen	\$8B94	Befehl START
\$8422	Befehl REPEAT	\$8BA0	Tabelle der xBasic-Befehlswörter
\$8454	Befehl UNTIL	\$8C80	Adressen der xBasic-Befehle-1
\$848F	Befehl JUMP	\$8CE0	Tabelle der xBasic-Fehlermeldungen
\$8512	Befehl CALL	\$8DE0	Adressen der Fehlermeldungen
\$8540	Befehl SUBEND	\$8E00	RU-Stack
\$8568	Befehl EXIT	\$8E80	CS-Stack
\$8589	Befehl BRANCH	\$C000	Farb-RAM bei hochauflösender Grafik
\$85A5	Befehl VOL	\$C400	Video-RAM bei UDG
\$85B3	Befehl ENVELOPE	\$C800	Neuer Zeichensatz
\$85DC	Holt Nibble nach X-Register	\$E000	Hires-Bitmap
\$85E8	Holt Stimme und setzt Zeiger		
\$85FF	Befehl WAVE		

64 — Tastaturänderung

Wenn Sie die Tastaturbelegung auf Ihrem C 64 ändern möchten, gibt es eine einfache Methode, um jeder beliebigen Taste eine andere Bedeutung zu geben.

Wie Sie sicher schon wissen, ist der Checksummer, der in jedem 64'er abgedruckt ist, eine wichtige Hilfe bei der Eingabe der Listings. Da bei diesem Programm das Basic-ROM und das Betriebssystem ins RAM kopiert werden, kann man die Tastaturbelegung ganz einfach ändern.

Wollen Sie zum Beispiel die Y-Taste und die Z-Taste vertauschen, so sind dazu lediglich zwei POKE-Befehle notwendig: **POKE 60289+25,ASC("Z");POKE 60289+12,ASC("Y")**

Wie Sie jede andere Taste umbelegen, sehen Sie aus der folgenden Tabelle:

POKE 60289+ X ,ASC(" gewünschtes Zeichen "); ergibt Tasten ohne Shift

POKE 60354+ X ,ASC(" gewünschtes Zeichen "); ergibt Tasten mit Shift

POKE 60419+ X ,ASC(" gewünschtes Zeichen "); ergibt Tasten mit C=

POKE 60536+ X ,ASC(" gewünschtes Zeichen "); ergibt Tasten mit CTRL (Helmut Pilz/gk)

Werteliste für X:										
	0	1	2	3	4	5	6	7	8	9
00	DEL	RET.	CRSR RIGHT	F7	F1	F3	F5	CRSR DOWN	3	W
10	A	4	Z	S	E	SHIFT LEFT	5	R	D	6
20	C	F	T	X	7	Y	G	8	B	H
30	U	V	9	I	J	0	M	K	O	N
40	+	P	L	-	.	:	@	,	£	*
50	;	HOME	SHIFT RIGHT	=	1	/	1	-	CTRL	2
60	SPACE	C=	Q	STOP						

Tabelle. Tastaturbelegung des C 64