

MPS-Support jetzt auch für den MPS802

Grafikauflösungen von 640 x 400 Punkten sind ab jetzt kein Problem mehr mit dem MPS802.

Damit jene, die einen MPS802 haben, das Programm MPS-Support (Ausgabe 2/86) nutzen können, ist es entsprechend angepaßt worden. Somit lassen sich HiRes-Grafiken mit einer Auflösung von 640 x 400 Punkten mit dem MPS802 ausdrucken. Alle in Ausgabe 2/86 aufgeführten Funktionen bleiben im vol-

len Umfang erhalten. Auch die Bedienung ist die gleiche geblieben. Daher erübrigt es sich, an dieser Stelle die Anleitung zu wiederholen. Diese Programmweiterung, die an verschiedenen Stellen im Speicher steht, mußte in zwei Teile zerlegt werden, um den ohnehin schon knappen Basic-Speicher nicht noch mehr zu

verkleinern. Zum Anpassen des Programms an den MPS802 sind folgende Schritte erforderlich:

- MSE laden und starten
- MPS-Support (Ausgabe 2/86) vom MSE aus laden
- »CTRL-N« und auf die Frage nach der Startadresse »\$11c9« eingeben
- Listing 1 abtippen
- MPS-Support mit »CTRL-S« speichern
- C 64 aus- und wieder einschalten
- MSE laden
- Listing 2 abtippen und speichern

- C 64 aus- und wieder einschalten
- Zuvor geändertes Programm MPS-Support laden (nicht starten)
- Listing 2 mit LOAD "MPS-802II",8,1 laden
- Mit dem Befehl SAVE "802-SUPPORT",8 das komplette Programm speichern.

Das Programm »MPS-Support« ist in dieser geänderten Version wie ein Basic-Programm zu laden und zu starten.

(Horst Bierhals/ah)

```
name : mps-802 ii      1771 17ad
1771 : ff ff a9 6a a2 04 a0 06 00
1779 : 20 7b 12 a2 6a 20 c9 ff ff
1781 : a9 14 20 d2 ff a9 67 a2 cf
1789 : 04 a0 05 06 a2 00 8a dd 3b
1791 : 3e 03 d0 05 e8 e0 08 d0 7d
1799 : f6 60 18 a5 fc 6d 3d 03 b0
17a1 : aa f0 08 a9 20 20 d2 ff 49
17a9 : ca d0 fa 60 50 52 00 00 3e
```

Listing 2. »802-Support« Teil 2

```
name : mps-802 i      11c9 12c9
11c9 : 20 9e b7 8e 11 17 60 20 84
11d1 : 73 17 20 7b 12 a9 00 20 f6
11d9 : d2 ff a9 0d 20 d2 ff a9 a3
11e1 : 64 a2 04 a0 00 20 7b 12 bf
11e9 : a9 4f 85 ff a2 00 86 fc d9
11f1 : 86 fe 8e 3c 03 a9 00 8d ba
11f9 : 3d 03 20 86 12 20 8d 17 17
1201 : f0 29 a2 69 20 c9 ff a2 f1
1209 : 00 bd 3e 03 20 d2 ff e8 42
```

Listing 1. »802-Support« Teil 1

```
1211 : e0 08 d0 f5 a9 0d 20 d2 11
1219 : ff a2 64 20 c9 ff 20 9b db
1221 : 17 a9 fe 20 d2 ff a9 8d bf
1229 : 20 d2 ff 18 a5 fe 69 08 bd
1231 : 85 fe 90 02 e6 ff 20 e1 4c
1239 : ff f0 1e ee 3d 03 ad 3d 33
1241 : 03 c9 50 d0 b5 a2 64 20 99
1249 : c9 ff a9 0d 20 d2 ff ee 94
1251 : 3c 03 ad 3c 03 c9 32 d0 eb
1259 : 9c a9 69 20 c3 ff a9 0d 25
1261 : 20 d2 ff a9 64 20 c3 ff 76
```

```
1269 : a9 6a 20 c9 ff a9 24 20 a7
1271 : d2 ff 20 cc ff a9 6a 20 1c
1279 : c3 ff 20 ba ff a9 00 20 29
1281 : bd ff 4c c0 ff 78 a5 01 c5
1289 : 29 fd 85 01 a2 00 8a 9d c2
1291 : 3e 03 e8 e0 08 d0 f8 a9 e5
1299 : 80 85 fd a0 00 b1 fe a2 3e
12a1 : 00 0a 90 0a 48 bd 3e 03 7d
12a9 : 05 fd 9d 3e 03 68 e8 e0 b5
12b1 : 08 d0 ee 46 fd c8 c0 08 df
12b9 : d0 e3 a5 01 09 02 85 01 bd
12c1 : 58 60 ff 20 e7 ff 60 a2 92
```

Hypra-Basic wird strukturiert

Durch neue Befehle ist es möglich, mit Hypra-Basic strukturiert zu programmieren. Für Einsteiger erklären wir ausführlich, wie man Module einbaut!

Zunächst einmal ein dickes Lob an unsere Leser. Die eingesandten Module sind wirklich gut. Das soll aber nicht heißen, das die Möglichkeiten von Hypra-Basic, dem Listing des Monats aus Ausgabe 4/85 auch nur annähernd aus-

geschöpft wären. In Richtung Grafik und Sound zeigten sich noch keine Aktivitäten — obwohl gerade hier die Stärken des C 64 liegen. Unsere Aufforderung an alle Leser: Programmiert Grafik- und Sound-Module, denn hier gibt es ein weites Betätigungsfeld.

Für alle diejenigen, die mit der Anleitung zu Hypra-Basic noch nicht zurechtkommen, zeigen wir Schritt für Schritt, wie man vom Abtippen zur eigenen Erweiterung kommt (siehe Kasten).
(K. Visel/J. Haible/F. Gräf/S. Thiel/og)

Module einbauen: so geht's

Das Besondere an Hypra-Basic ist sein modularer Aufbau, das heißt, Sie bauen sich Ihre persönliche Erweiterung, mit den Befehlen, die Sie wirklich benötigen. Um dies zu erreichen, sind einige Vereinbarungen notwendig. Jedes Modul muß einem bestimmten Format entsprechen. Auf der Diskette muß das Modul einen bestimmten Namen haben, und zu jedem Modul muß ein File mit den dazugehörigen Parametern existieren.

Auf der Diskette müssen sich »HYPRABASIC« (Hauptprogramm), »BASIC.ASS« (Umrechnungsroutine und Interpreter-Erweiterung), sowie die Module mit ihren Parameter-Files befinden.

Die Namen der Module müssen dieses Format haben: "xERW.ASS"

»x« ist die jeweilige Nummer des Moduls, zum Beispiel "38ERW.ASS". Das Leerzeichen vor »ASS« ist zu beachten.

Das dazugehörige Parameter-File müssen Sie aus den Angaben des Listing-Teils selbst erstellen. Dazu wählen Sie vom Hauptprogramm den Menüpunkt »PARAMETER EINGEBEN«, der mit F5 aufgerufen wird. Dann werden Sie zuerst nach der Modulnummer gefragt, die natürlich mit der entsprechenden Erweiterung übereinstimmen muß. Gleiches gilt für die Anzahl der Befehle. Sie ist im

Listing-Teil immer angegeben. In diesem Beispiel sind es drei Befehle. Die nächste Eingabe ist der Name des ersten Befehls. Er ist beliebig, darf aber keine Basic-Schlüsselwörter enthalten. In unserem Fall ist das »FRAME«. Die Startadresse des ersten Befehls ist 49152, das Ende liegt bei 49158. Daran angehängt sind 0 Datenbyte. Die Informationen für die anderen zwei Befehle entnehmen Sie Listing 1. Antworten Sie auf die Frage »Sind Sie sicher?« mit <J>, speichert Hypra-Basic die Parameter. Bei <N> wird in das Hauptmenü zurückgegangen.

Haben Sie alle Parameter-Files erstellt, können Sie sich die erste Erweiterung generieren. Wählen Sie also »ERWEITERUNG ERSTELLEN«. Dann geben Sie, jeweils durch <RETURN> getrennt, die Nummern der gewünschten Module ein. Bei Eingabe von <E> und nach Bestätigung der Sicherheitsabfrage werden die Parameter gelesen. Dann benötigt Hypra-Basic nur noch die Anfangsadresse der Erweiterung. Dabei wird zwischen »TABELLE« und »BEFEHLE« unterschieden. Die Tabelle darf nun nicht unter dem ROM liegen, der Bereich ab 49152 ist zunächst zu empfehlen. Gleichzeitig ist der Anfang der Tabelle die Startadresse der Erweiterung. Die Befehle selbst werden dann entweder daran angehängt, oder irgendwo unter das ROM gelegt. Meistens passen die Befehle hinter die Tabelle, so daß der Vorschlag von Hypra-Basic angenommen werden kann.

Hat der Computer die Erweiterung erstellt, dann steht sie lediglich im Speicher. Es empfiehlt sich also, diese jetzt

zu speichern. Ist die Erweiterung gespeichert, brechen Sie das Programm mit RUN/STOP ab, und geben SYS 64738 ein. Laden Sie jetzt die Erweiterung mit »8,1« und starten Sie sie mit SYS (Startadresse), hier mit »SYS 49152«.

Schritt für Schritt:

- Auf der Diskette müssen sich »HYPR-BASIC« und »BASIC.ASS« befinden.
- Zu jedem Modul muß die entsprechende Parameterdatei vorhanden sein. Diese wird unter Hypra-Basic erstellt.
- Laden und starten Sie Hypra-Basic.
- Wählen Sie den Menüpunkt »ERWEITERUNG ERSTELLEN«.
- Geben Sie die Nummern der gewünschten Module, durch <Return> voneinander getrennt, ein.
- Beenden Sie die Eingabe mit <E> und bestätigen Sie die Sicherheitsabfrage.
- Wählen Sie die Startadresse der Erweiterung. Normalerweise ist das 49152.
- Den Vorschlag für die Befehlstabelle können Sie übernehmen, wenn nicht in den Bereich von 53248 bis 57343 geschrieben wird.
- Speichern Sie dann die Erweiterung. Wenn Sie die Adresse für die Module übernommen haben, genügt die Eingabe eines Namens. Die Erweiterung wird dann mit »8,1« geladen.
- Das Aktivieren der neuen Befehle wird durch SYS 49152, also der Startadresse ausgelöst.
- Die neuen Befehle stehen zur Verfügung

Listing 1. Modulnummer: 31

Anzahl der Befehle: 3

1) FRAME 48152-49158,0

Funktion:

FRAME legt die Rahmenfarbe des Bildschirms fest.

Syntax: FRAME f

f ist die Farbnummer. (0 = schwarz, 1 = weiß ...)

Erlaubt sind Werte für f von 0-255, wobei bei Werten größer als 15 nur das Low-Nibble zählt.

2) BACK 49159-49165,0

Funktion: Hintergrundfarbe festlegen

Syntax: BACK f

3) PEN 49166-49172,0

Funktion: Zeichenfarbe festlegen

Syntax: PEN f

programm : 31erw .ass c000 c017

```
c000 : 20 9e b7 8e 20 d0 60 20 79
c008 : 9e b7 8e 21 d0 60 20 9e 17
c010 : b7 8e 86 02 60 00 00 00 f6
```

Listing 2. Modulnummer: 32

Anzahl der Befehle: 2

1) FAST 49152-48165,0

Funktion: FAST schaltet beim C 128 im C 64-Modus auf 2 MHz Taktfrequenz um und den Bildschirm ab. Dadurch wird der Computer doppelt so schnell. Beim C 64 ergeben sich durch das Abschalten des Bildschirms nur etwa 5 bis 6 Prozent Geschwindigkeitsgewinn.

Syntax: FAST

2) SLOW 49166-49179,0

Funktion: SLOW schaltet wieder auf 1 MHz zurück und schaltet den Bildschirm ein.

Syntax: SLOW

programm : 32erw .ass c000 c01e

```
c000 : ad 11 d0 29 ef 8d 11 d0 e0
c008 : a9 01 8d 30 d0 60 a9 00 52
c010 : 8d 30 d0 ad 11 d0 09 10 7b
c018 : 8d 11 d0 60 00 c0 00 00 74
```

100 REM TEXT IN DREIECKFORM AUSGEBEN

110 DO

120 : PRINT "BITTE TEXT EINGEBEN" : INPUT X\$: PRINT

130 : N=1 : Q=1

140 : DO WHILE N>0

150 : PRINT LEFT\$(X\$,N)

160 : N=N+Q

170 : IF N=LEN(X\$) THEN Q=-1

180 : LOOP

190 : PRINT

200 : DO

210 : INPUT "NOCHMAL (J/N) " : A\$

220 : EXIT IF A\$="J" OR A\$="N"

230 : PRINT "FALSCH EINGABE"

240 : LOOP

250 LOOP UNTIL A\$="N"

260 END

Listing 7. Demo-Programm für Modul 34

Listing 5. Modulnummer: 35

Anzahl der Befehle: 1

HELP 49152-49247,0

Funktion: Auflisten aller Befehle der Basic-Erweiterung auf dem Bildschirm (oder Drucker) ohne Beeinflussung des im Speicher befindlichen Programmes.

Syntax: Help (Startadresse der Tabelle)

Zur Ausgabe auf den Drucker muß in diesem Beispiel vor Eingabe des Help-Befehles folgende Befehlssequenz eingegeben werden:

OPEN 4,4

CMD 4

Nach der Ausgabe wie üblich mit Print #4 und CLOSE 4 abschließen.

programm : 35erw .ass c000 c062

```
c000 : 20 8a ad 20 f7 b7 aa 98 ee
c008 : 18 69 a0 85 fb 8a 69 00 67
c010 : 85 fc a9 8d 20 d2 ff 20 08
c018 : 28 c0 20 28 c0 20 28 c0 dd
c020 : a9 20 20 d2 ff 4c 17 c0 7c
c028 : 20 59 c0 a0 00 b1 fb f0 98
c030 : 25 85 fd aa 20 59 c0 b1 20
c038 : fb 20 d2 ff ca d0 f5 a2 48
c040 : 05 20 59 c0 ca d0 fa a9 36
c048 : 0d 38 e5 fd a8 a9 20 20 43
c050 : d2 ff 88 d0 fa 60 68 68 83
c058 : 60 e6 fb d0 02 e6 fc 60 50
c060 : 20 31 8d 00 18 0a 29 0f 11
```


Listing 4. Modulnummer: 34

Anzahl der Befehle: 3

- 1) DO 49152-49207,5
- 2) LOOP 49213-49280,5
- 3) EXIT 49286-49412,25

Syntax:

- 1 a) DO
- b) DO WHILE Bedingung
- 2 a) LOOP
- b) LOOP UNTIL Bedingung
- 3 a) EXIT Bedingung
- b) EXIT IF Bedingung

Funktion: Der Programmteil zwischen den Befehlen »DO« und »LOOP« wird beliebig oft ausgeführt. Aussteigen aus dieser Endlosschleife kann man auf drei verschiedene Arten:

- 1) Zu Beginn der »DO«-Schleife kann mit »WHILE« eine Bedingung gesetzt werden. Die Schleife wird nur ausgeführt, solange die Bedingung zutrifft. Ist die Bedingung am Schleifenanfang nicht erfüllt, so wird das Programm hinter »LOOP« fortgesetzt.

Dies entspricht einer »WHILE...ENDWHILE« (-Schleife)

- 2) An beliebiger Stelle innerhalb einer »DO«-Schleife kann man mit »EXIT IF« aussteigen. Trifft die nachfolgende Bedingung zu, so wird die Schleife verlassen.

- 3) Schließlich kann die Abbruchbedingung auch noch am Schleifenende stehen. Ist die Bedingung nach »UNTIL« erfüllt, so wird die Schleife verlassen, ansonsten fortgesetzt.

Zu beachten ist noch, daß bei einer »WHILE«-Bedingung die Schleife unter Umständen gar nicht durchlaufen wird. Bei der »UNTIL«-Bedingung jedoch mindestens einmal.

Die neuen Befehle müssen unbedingt zu Beginn einer neuen Zeile stehen, dürfen jedoch mit Doppelpunkten und Leerzeichen eingerückt werden (siehe Listing 7).

programm : 34erw .ass c000 c11e

```
c000 : a9 02 20 fb a3 a5 7a 48 14
c008 : a5 7b 48 a9 bb 48 20 79 24
c010 : 00 f0 05 20 22 c0 f0 03 c0
c018 : 4c ae a7 20 b8 c0 20 59 6e
c020 : c0 00 a2 04 dd 38 c0 f0 8e
c028 : 03 4c 08 af 20 73 00 ca 7d
c030 : 10 f2 20 9e ad a5 61 60 e3
c038 : 45 4c 49 48 57 20 79 00 5b
c040 : f0 21 a2 04 dd 81 c0 f0 b9
c048 : 03 4c 08 af 20 73 00 ca 9d
c050 : 10 f2 20 9e ad a5 61 f0 25
c058 : 0a 20 9d c0 9a 68 68 68 51
c060 : 4c f8 a8 20 9d c0 bd 03 33
c068 : 01 85 7a bd 02 01 85 7b b7
c070 : 20 79 00 f0 05 20 22 c0 c6
c078 : f0 01 60 20 b8 c0 4c 59 7a
c080 : c0 4c 49 54 4e 55 20 9d 8f
c088 : c0 20 79 00 c9 8b d0 03 f9
c090 : 20 73 00 20 9e ad a5 61 1f
c098 : d0 e1 4c f8 a8 ba e8 e8 61
c0a0 : e8 e8 bd 01 01 c9 bb d0 7b
c0a8 : 04 60 ad 05 c1 ad ab c0 a2
c0b0 : 85 22 ad ac c0 4c 45 a4 14
c0b8 : a9 00 8d 1d c1 20 3b a9 c6
c0c0 : a0 02 b1 7a d0 03 4c e3 3b
c0c8 : a8 c8 c8 98 65 7a 85 7a 4f
c0d0 : 90 02 e6 7b 20 73 00 c9 bc
c0d8 : 3a f0 f9 c9 44 d0 0d 20 81
c0e0 : 73 00 c9 4f d0 d7 ee 1d 71
c0e8 : c1 4c bd c0 a2 03 dd 19 43
c0f0 : c1 d0 ca 20 73 00 ca 10 52
c0f8 : f5 ad 1d c1 f0 06 ce 1d f8
c100 : c1 4c bd c0 60 4c 4f 4f b3
c108 : 50 2f 45 58 49 54 20 57 b3
c110 : 49 54 48 4f 55 54 20 44 80
c118 : cf 50 4f 4f 4c 00 00 00 92
```

```
100 REM BEISPIEL "MITTERNACHTSFORMEL"
110 PRINT "LOESUNG EINER QUADRATISCHEN GLEICHUNG"
120 INPUT "KOEFFIZIENTEN A,B,C ";A,B,C : PRINT
130 D=B*B-4*A*C
140 CASE IF D>=0
150 : CASE IF D>0
160 : PRINT "ZWEI LOESUNGEN"
170 : PRINT "X1 =" ; (-B+SQR(D))/(2*A)
180 : PRINT "X2 =" ; (-B-SQR(D))/(2*A)
190 : CASE ELSE
200 : PRINT "EINE LOESUNG"
210 : PRINT "X =" ; -B/(2*A)
220 : CASE END
230 CASE ELSE
240 : PRINT "KEINE REELLE LOESUNG"
250 CASE END
260 END
```

READY.

Listing 6. Demo-Programm für Modul 33.

Listing 3. Modulnummer: 33

Anzahl der Befehle: 1

- 1) CASE 49152-49321,9

Syntax:

- a) CASE IF Bedingung
- b) CASE ELSE
- c) CASE END

Funktion: Trifft der Interpreter auf »CASE IF«, so wird die folgende Bedingung geprüft und entsprechend verzweigt. Ist die Bedingung erfüllt, so wird das Programm mit den darauf folgenden Anweisungen fortgesetzt. Folgt ein »CASE ELSE« so wird der Programmteil zwischen »CASE ELSE« und »CASE END« übersprungen.

Trifft die »CASE IF«-Bedingung dagegen nicht zu, so wird das Programm direkt hinter »CASE END« beziehungsweise »CASE ELSE« (sofern vorhanden) fortgesetzt.

Es ergibt sich folgende Struktur:

CASE IF Bedingung

Programmteil, der ausgeführt wird, wenn die Bedingung erfüllt ist. (Dürfte zum Beispiel eine weitere »CASE«-Struktur enthalten)

CASE ELSE

Programmteil, der ausgeführt wird, wenn die Bedingung nicht erfüllt ist. (Kann entfallen)

CASE END

Die »CASE«-Anweisungen müssen am Zeilenanfang stehen, damit sie von der Routine gefunden werden. Sie dürfen jedoch mit Doppelpunkten und Leerzeichen eingerückt werden (siehe Listing 6).

programm : 33erw .ass c000 c0b5

```
c000 : 20 79 00 c9 8b f0 2e c9 a3
c008 : 80 d0 03 4c f8 a8 20 6d 6b
c010 : c0 a9 00 8d b2 c0 20 7e 05
c018 : c0 c9 8b d0 06 ee b2 c0 de
c020 : 4c 16 c0 c9 80 d0 ef ae 8c
c028 : b2 c0 f0 06 ce b2 c0 4c 55
c030 : 16 c0 4c f8 a8 20 73 00 32
c038 : 20 9e ad a5 61 f0 03 4c 0a
c040 : f8 a8 a9 00 8d b2 c0 20 a8
c048 : 7e c0 c9 8b d0 06 ee b2 68
c050 : c0 4c 47 c0 c9 80 f0 03 8b
c058 : 20 6d c0 ae b2 c0 d0 03 af
c060 : 4c f8 a8 c9 80 d0 03 ce c4
c068 : b2 c0 4c 47 c0 a2 03 dd 5f
c070 : ae c0 f0 03 4c 08 af 20 1f
c078 : 73 00 ca 10 f2 60 20 3b c9
c080 : a9 a0 02 b1 7a d0 03 4c 03
c088 : e3 a8 c8 c8 98 65 7a 85 b4
c090 : 7a 90 02 e6 7b 20 73 00 36
c098 : c9 3a f0 f9 a2 03 dd aa 09
c0a0 : c0 d0 db 20 73 00 ca 10 46
c0a8 : f5 60 45 53 41 43 45 53 73
c0b0 : 4c 45 00 20 31 00 00 00 b6
```