

Stephan Thienel, Thomas Sauer, Wolfgang Kostal

# Entwicklungshilfe

Ein Monitorprogramm für den 6502-EMUF

Der 6502-EMUF [1] erhält zum Entwickeln und Testen von Programmen einen Terminal-Anschluß, ein Monitorprogramm mit 65C02-Disassembler, eine Speichererweiterung bis 24 KByte RAM und eine Single-Step-Einrichtung.

Kaum ein gängiger 6502-Mikrocomputer verfügt über so viele freie Ports wie der 6502-EMUF. Deswegen lassen sich Programme für ihn oft nur mit einem erheblichen Verschleiß an EPROMs und auch an Löschlampen-Lebensdauer entwickeln. Bei 8 KByte RAM, die sogar, wie anschließend beschrieben, mühelos bis 24 KByte erweitert werden können, liegt es auf der Hand, den EMUF selbst als Entwicklungssystem zu verwenden. Alles was fehlt, ist ein Terminal und ein Monitorprogramm.

## Das Terminal: Beispiel C-64

Das Monitorprogramm (Bild 1) enthält die Routinen für eine V.24-Schnittstelle, wie sie auch für den Anschluß eines Akustikkopplers verwendet wird. Als Terminal eignet sich also im Prinzip jeder Rechner, der auch einen Anschluß für ein Modem hat. Bild 2 zeigt zum Beispiel, wie der EMUF mit der RS-232-Schnittstelle des C-64 von Commodore verbunden wird. Hier sind nicht einmal Pegelwandler notwendig, weil der User-Port des C-64 ebenfalls TTL-Signale liefert bzw. verarbeitet. Als Eingang am EMUF dient PC7, Ausgang ist PD0. In Bild 3 ist ein kleines Kommunikationsprogramm für den C-64 aufgelistet. Es erklärt sich durch seine Kommentare selbst.

## Das Monitorprogramm

Die Ein-/Ausgabe-Einheit ist ein seriell angeschlossener Terminal-Rechner. Das Problem der Übertragungsgeschwindigkeit wurde ähnlich wie einst beim legendären KIM1 von Commodore gelöst: das Terminal schickt als erstes Zeichen \$FF, damit der EMUF durch Messung der

Länge eines Startbits die Baudrate erkennen kann. Vor dem Übertragen des Bytes \$FF sollte am EMUF unbedingt ein Reset ausgelöst werden. Es sind Übertragungs-

geschwindigkeiten bis 2400 Baud und darüber hinaus möglich. Die Tabelle 1 zeigt eine Zusammenstellung der möglichen Befehle, die durchaus noch individuell erweitert werden können. Denkbar sind zum Beispiel Befehle zum Verschieben eines Speicherbereichs in einen anderen, Vergleichen und Füllen von Speicherbereichen usw. Die vorgestellte Befehlsliste wurde für den Betrieb an einem Terminalrechner ausgelegt, der über einen Assembler verfügt, so daß solche Routinen eigentlich nicht gebraucht werden und nur unnötig Speicherplatz belegen. Wichtiger sind in diesem Falle Routinen zum Übertragen von Hex-Code zwischen Terminalrechner und EMUF, damit ein assembliertes Programm zum Testen ins RAM des EMUF geschrieben werden kann. Dabei wird jedes Byte gemäß seiner Darstellung im Hex-Code in zwei ASCII-Zeichen zerlegt und dann erst übertragen. Der Befehl „L“ dient zum Laden eines Programms vom Terminalrechner in den

|      |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |      |      |
|------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------|------|
| 0000 | a2 | ff | 9a | 86 | f4 | 20 | 34 | fa | a9 | ff | 8d | ef | 00 | 2c | 1f | 60   | 08d2 |
| 0010 | 30 | fb | a9 | fc | 18 | 69 | 01 | 90 | 03 | ee | ef | 00 | ac | 1f | 60 | 10   | 06fd |
| 0020 | f3 | 8d | ee | 00 | a2 | 08 | 20 | b8 | fa | 20 | 26 | fa | 20 | 21 | fa | 20   | 0785 |
| 0030 | ac | f9 | 20 | d6 | f9 | a0 | 00 | b1 | fb | 20 | b7 | f9 | 20 | d6 | f9 | a9   | 0a48 |
| 0040 | 00 | 85 | f9 | 85 | fa | 20 | aa | fa | 20 | d8 | f9 | 48 | 20 | 8a | fa | 68   | 0906 |
| 0050 | a2 | 11 | dd | 6c | f8 | f0 | 06 | ca | 10 | f8 | 4c | 45 | f8 | 8a | 0a | aa   | 0883 |
| 0060 | bd | 7e | f8 | a8 | e8 | bd | 7e | f8 | 48 | 98 | 48 | 60 | 4e | 2b | 2d | 2e   | 0852 |
| 0070 | 52 | 47 | 4d | 4c | 53 | 2a | 7f | 00 | 6e | 67 | 6d | 6c | 73 | 72 | a1 | f8   | 065a |
| 0080 | a7 | f8 | ad | f8 | bb | f8 | c4 | f8 | 05 | f9 | 18 | f9 | 6c | f9 | 8d | f9   | 0bad |
| 0090 | d9 | fd | 2b | f8 | 2b | f8 | a1 | f8 | 05 | f9 | 18 | f9 | 6c | f9 | 8d | f9   | 0aaf |
| 00a0 | c4 | f8 | 20 | e2 | fa | 4c | 2c | f8 | 20 | db | fa | 4c | 2c | f8 | 38 | a5   | 096a |
| 00b0 | fb | e9 | 01 | 85 | fb | b0 | 02 | c6 | fc | 4c | 2c | f8 | a0 | 00 | a5 | f9   | 0987 |
| 00c0 | 91 | fb | 4c | a8 | f8 | 20 | 13 | fa | a2 | 03 | 20 | d6 | f9 | b5 | f4 | 20   | 0902 |
| 00d0 | b7 | f9 | ca | 10 | f5 | 20 | d6 | f9 | a5 | f2 | 20 | b7 | f9 | a5 | f1 | 20   | 0a8b |
| 00e0 | b7 | f9 | 20 | d6 | f9 | a2 | 08   | 0a16 |
| 00f0 | a5 | f3 | 85 | e3 | 06 | e3 | a9 | 30 | 69 | 00 | 20 | d8 | f9 | ca | d0 | f4   | 09aa |
| 0100 | 20 | 21 | fa | 4c | a8 | f8 | a6 | f4 | 9a | a5 | fc | 48 | a5 | fb | 48 | a5   | 09d1 |
| 0110 | f3 | 48 | a6 | f6 | a4 | f5 | a5 | f7 | 40 | a9 | 02 | 85 | e3 | 20 | 21 | fa   | 099a |
| 0120 | 20 | eb | fa | 20 | ac | f9 | 20 | d6 | f9 | a2 | 08 | 86 | e9 | a0 | 00 | b1   | 0923 |
| 0130 | fb | 20 | b7 | f9 | 20 | d6 | f9 | 20 | db | fa | c6 | e9 | d0 | ef | 20 | f4   | 0b31 |
| 0140 | fa | 20 | d6 | f9 | a2 | 08 | 86 | e9 | a0 | 00 | b1 | fb | c9 | 20 | 30 | 0a   | 0871 |
| 0150 | c9 | 7b | 10 | 06 | 20 | d8 | f9 | 4c | 5f | f9 | a9 | 2e | 20 | d8 | f9 | 20   | 07d7 |
| 0160 | db | fa | c6 | e9 | d0 | e2 | c6 | e3 | d0 | b3 | 4c | 2c | f8 | 20 | a3 | f9   | 0b8e |
| 0170 | b0 | 19 | a5 | f9 | 48 | a5 | fa | 48 | 20 | 7b | fa | a0 | 00 | 91 | fb | 68   | 08bf |
| 0180 | 85 | fa | 68 | 85 | f9 | 20 | db | fa | 4c | 6d | f9 | 4c | 2c | f8 | 20 | a3   | 093f |
| 0190 | f9 | b0 | 0d | a0 | 00 | b1 | fb | 20 | b7 | f9 | 20 | db | fa | 4c | 8e | f9   | 099a |
| 01a0 | 4c | 2c | f8 | a5 | fb | c5 | f9 | a5 | fc | e5 | fa | 60 | a5 | fc | 20 | b7   | 0b26 |
| 01b0 | f9 | a5 | fb | 20 | b7 | f9 | 60 | 85 | f8 | 4a | 4a | 4a | 4a | 20 | c8 | f9   | 094f |
| 01c0 | a5 | f8 | 20 | c8 | f9 | a5 | f8 | 60 | 29 | 0f | c9 | 0a | 18 | 30 | 02 | 69   | 0739 |
| 01d0 | 07 | 69 | 30 | 4c | d8 | f9 | a9 | 20 | 85 | ff | 48 | 86 | fd | 84 | fe | 20   | 0877 |
| 01e0 | 51 | fa | ad | 10 | 60 | 29 | fe | 8d | 10 | 60 | 20 | 51 | fa | a2 | 08 | ad   | 074e |
| 01f0 | 10 | 60 | 29 | fe | 4e | ff | 00 | 69 | 00 | 8d | 10 | 60 | 20 | 51 | fa | ca   | 067f |
| 0200 | d0 | ed | ad | 10 | 60 | 09 | 01 | 8d | 10 | 60 | 20 | 51 | fa | a6 | fd | a4   | 0793 |
| 0210 | fe | 68 | 60 | a2 | 00 | bd | 46 | ff | 20 | d8 | f9 | e8 | 20 | d0 | f5 | 0a08 |      |
| 0220 | 60 | a9 | 0d | 4c | d8 | f9 | a2 | 00 | bd | 30 | ff | 20 | d8 | f9 | e8 | 00   | 097a |
| 0230 | 16 | d0 | f5 | 60 | d8 | 78 | a2 | 00 | 8e | 13 | 60 | a2 | 01 | 8e | 12 | 60   | 06d1 |
| 0240 | 8e | 10 | 60 | a9 | 0e | 85 | ea | 85 | ec | a9 | ff | 85 | eb | 85 | ed | 58   | 0977 |
| 0250 | 60 | ad | ef | 00 | 8d | f0 | 00 | ad | ee | 00 | 38 | e9 | 01 | b0 | 03 | ce   | 07b7 |
| 0260 | f0 | 00 | ac | f0 | 00 | 10 | f3 | 60 | ad | ef | 00 | 8d | f0 | 00 | ad | ee   | 08a3 |
| 0270 | 00 | 4a | 4e | f0 | 00 | 90 | e3 | 09 | 80 | b0 | e0 | 20 | aa | fa | 20 | 8a   | 0782 |
| 0280 | fa | 20 | aa | fa | 20 | 8a | fa | a5 | f9 | 60 | c9 | 30 | 30 | 1b | c9 | 47   | 08b4 |
| 0290 | 10 | 17 | c9 | 40 | 30 | 03 | 18 | 69 | 09 | 2a | 2a | 2a | 2a | a0 | 04 | 2a   | 0363 |
| 02a0 | 26 | f9 | 26 | fa | 88 | d0 | f8 | a9 | 00 | 60 | 86 | fd | 84 | fe | a2 | 08   | 0947 |
| 02b0 | 2c | 1f | 60 | 30 | fb | 20 | 51 | fa | 20 | 68 | fa | ad | 1f | 60 | 29 | 80   | 0698 |

**Bild 1.** Das Monitorprogramm fürs EPROM, die Adresse \$0000 im EPROM entspricht \$F800 im EMUF. Das Programm ist also im EPROM, falls dieses größer als 2 KByte ist, immer in den obersten 2 KByte unterzubringen

|      |   |      |
|------|---|------|
| 02c0 | 4e ff 00 0d ff 00 8d ff 00 20 51 fa ca d0 ec 20   | 07f6 |
| 02d0 | 68 fa a5 ff a6 fd a4 fe 4a 2a 60 e6 fb d0 02 e6   | 0ab8 |
| 02e0 | fc 60 a5 f9 85 fb a5 fa 85 fc 60 a5 fb 85 e7 a5   | 0bab |
| 02f0 | fc 85 e8 60 a5 e7 85 fb a5 e8 85 fc 60 00 08 0a   | 0955 |
| 0300 | 18 1a 28 2a 38 3a 40 48 4a 58 5a 60 68 6a 78 7a   | 049e |
| 0310 | 88 8a 98 9a a8 aa b8 ba c8 ca d8 da e8 ea f8 fa   | 0c10 |
| 0320 | 10 30 50 70 80 90 b0 d0 f0 09 29 49 69 89 a0 a2   | 072f |
| 0330 | a9 c0 c9 e0 e9 04 05 06 14 24 25 26 45 46 64 65   | 05e1 |
| 0340 | 66 84 85 86 a4 a5 a6 c4 c5 c6 e4 e5 e6 01 21 41   | 0945 |
| 0350 | 61 81 a1 c1 e1 11 31 51 71 91 b1 d1 f1 15 16 34   | 078c |
| 0360 | 35 36 55 56 74 75 76 94 95 b4 b5 d5 d6 f5 f6 96   | 0933 |
| 0370 | b6 12 32 52 72 92 b2 d2 f2 0c 0d 0e 1c 20 2c 2d   | 0582 |
| 0380 | 2e 4c 4d 4e 6d 6e 8c 8d 8e 9c ac ad ae cc cd ce   | 08a1 |
| 0390 | ec ed ee 1d 1e 3c 3d 3e 5d 5e 7d 7e 9d 9e bc bd   | 0823 |
| 03a0 | dd de fd fe 19 39 59 79 99 b9 be d9 f9 6c 7c 3f   | 09e3 |
| 03b0 | 3f 3f 42 52 4b 50 48 50 41 53 4c 43 4c 43 49 4e   | 048e |
| 03c0 | 43 50 4c 50 52 4f 4c 53 45 43 44 45 43 52 54 49   | 04b2 |
| 03d0 | 50 48 41 4c 53 52 43 4c 49 50 48 59 52 54 53 50   | 04dc |
| 03e0 | 4c 41 52 4f 52 53 45 49 50 4c 59 44 45 59 54 58   | 04e4 |
| 03f0 | 41 54 59 41 54 58 53 54 41 59 54 41 58 43 4c 56   | 04ee |
| 0400 | 54 53 58 49 4e 59 44 45 58 43 4c 44 50 48 58 49   | 04dc |
| 0410 | 4e 58 4e 4f 50 53 45 44 50 4c 58 42 50 4c 42 4d   | 04d0 |
| 0420 | 49 42 56 43 42 56 53 42 52 41 42 43 43 42 43 53   | 0484 |
| 0430 | 42 4e 45 42 45 51 4f 52 41 41 4e 44 45 4f 52 41   | 0489 |
| 0440 | 44 43 42 49 54 4c 44 59 4c 44 58 4c 44 41 43 50   | 049b |
| 0450 | 59 43 4d 50 43 50 58 53 42 43 54 53 42 4f 52 41   | 04c7 |
| 0460 | 41 53 4c 54 52 42 42 49 54 41 4e 44 52 4f 4c 45   | 04ac |
| 0470 | 4f 52 4c 53 52 53 54 5a 41 44 43 52 4f 52 53 54   | 04f5 |
| 0480 | 59 53 54 41 53 54 58 4c 44 59 4c 44 41 4c 44 58   | 04e2 |
| 0490 | 43 50 59 43 4d 50 44 45 43 43 50 58 53 42 43 49   | 04a4 |
| 04a0 | 4e 43 4f 52 41 41 4e 44 45 4f 52 41 44 43 53 54   | 049b |
| 04b0 | 41 4c 44 41 43 4d 50 53 42 43 4f 52 41 41 4e 44   | 047f |
| 04c0 | 45 4f 52 41 44 43 53 54 41 4c 44 41 43 4d 50 53   | 049a |
| 04d0 | 42 43 4f 52 41 41 53 4c 42 49 54 41 4e 44 52 4f   | 049a |
| 04e0 | 4c 45 4f 52 4c 53 52 53 54 5a 41 44 43 52 4f 52   | 04df |
| 04f0 | 53 54 59 53 54 41 4c 44 59 4c 44 41 43 4d 50 44   | 04c6 |
| 0500 | 45 43 53 42 43 49 4e 43 53 54 58 4c 44 58 4f 52   | 04c2 |
| 0510 | 41 41 4e 44 45 4f 52 41 44 43 53 54 41 4c 44 41   | 047b |
| 0520 | 43 4d 50 53 42 43 54 53 42 4f 52 41 41 53 4c 54   | 04b7 |
| 0530 | 52 42 4a 53 52 42 49 54 41 4e 44 52 4f 4c 4a 4d   | 04b9 |
| 0540 | 50 45 4f 52 4c 53 52 41 44 43 52 4f 52 53 54 59   | 04e2 |
| 0550 | 53 54 41 53 54 58 53 54 5a 4c 44 59 4c 44 41 4c   | 04ee |
| 0560 | 44 58 43 50 59 43 50 44 45 43 43 50 58 53 42 04b4 | 04b4 |
| 0570 | 43 49 4e 43 4f 52 41 41 53 4c 42 49 54 41 4e 44   | 0491 |
| 0580 | 52 4f 4c 45 4f 52 4c 53 52 41 44 43 52 4f 52 53   | 04d2 |
| 0590 | 54 41 53 54 5a 4c 44 59 4c 44 41 43 4d 50 44 45   | 04b9 |
| 05a0 | 43 53 42 43 49 4e 43 4f 52 41 41 4e 44 45 4f 52   | 0490 |
| 05b0 | 41 44 43 53 54 41 4c 44 41 4c 44 58 43 4d 50 53   | 049c |
| 05c0 | 42 43 4a 4d 50 4a 4d 50 23 7c b2 24 2d 39 51 59   | 04d8 |
| 05d0 | 61 73 75 7d 97 a8 b1 b2 b3 b4 20 d6 f9 a0 00 b1   | 090f |
| 05e0 | fb a2 b2 dd fc fa f0 03 ca d0 f8 86 e3 b9 c8 fd   | 0c8e |
| 05f0 | cb c5 e3 90 f8 84 e4 a0 03 a9 af a2 fb 18 65 e3   | 0a58 |
| 0600 | 90 01 e8 88 d0 f7 85 e5 86 e6 b1 e5 20 d8 f9 c8   | 0aed |
| 0610 | c0 03 d0 f6 20 d6 f9 a2 00 a5 e3 dd cb fd 90 03   | 09da |
| 0620 | 20 35 fe a2 00 18 a5 fb 65 e4 85 fb a5 fc 69 00   | 0880 |
| 0630 | 85 fc 4c 2c f8 e8 dd cb fd b0 fa 8a 0a aa bd f2   | 0b15 |
| 0640 | fe a8 e8 bd f2 fe 48 98 48 60 a0 01 18 b1 fb a4   | 09cc |
| 0650 | fc aa 10 01 88 65 fb 90 01 c8 18 69 02 aa 90 01   | 06b6 |
| 0660 | c8 4c 72 fe a9 24 4c d8 f9 a0 01 b1 fb aa c8 b1   | 09de |
| 0670 | fb a8 20 64 fe 98 20 b7 f9 8a 4c b7 f9 a9 23 20   | 08ff |
| 0680 | d8 f9 4c 8a fe a9 28 20 d8 f9 20 64 fe a0 01 b1   | 093b |
| 0690 | fb 4c b7 f9 20 85 fe 20 a2 fe 4c 9d fe a9 29 4c   | 095f |
| 06a0 | d8 f9 a9 2c 20 d8 f9 a9 58 4c d8 f9 a9 2c 20 d8   | 0982 |
| 06b0 | f9 a9 59 4c d8 f9 20 85 fe 20 9d fe 4c ac fe 20   | 098c |
| 06c0 | 8a fe 4c a2 fe 20 8a fe 4c ac fe 20 85 fe 4c 9d   | 099e |
| 06d0 | fe 20 69 fe 4c a2 fe 20 69 fe 4c ac fe 20 ac fe   | 09f8 |
| 06e0 | 4c 9d fe 20 ec fe 20 a2 fe 4c 9d fe a9 28 20 d8   | 0961 |
| 06f0 | f9 4c 69 fe 49 fe 7c fe 89 fe 93 fe b5 fe be fe   | 0bf4 |
| 0700 | c4 fe ca fe 68 fe d0 fe d6 fe dc fe e2 fe 85 f7   | 0dc8 |
| 0710 | 68 85 f3 68 85 f1 85 fb 68 85 f2 85 fc 84 f5 86   | 0a9d |
| 0720 | f6 ba 86 f4 20 34 fa 4c c5 f8 6c ea 00 6c ec 00   | 092f |
| 0730 | 0d 0a 45 4d 55 46 20 49 49 49 20 2d 20 4d 4f 4e   | 0396 |
| 0740 | 49 54 4f 52 0d 0a 0d 0a 20 41 43 20 58 52 20 59   | 0353 |
| 0750 | 52 20 53 50 20 50 43 20 20 20 53 3a 4e 56 20      | 03c9 |
| 0760 | 42 44 49 5a 43 0d 0a 4c 6a c4 f0 07 4c 7d ac 68   | 05d1 |
| 0770 | 4c 7d ac a5 49 ee 11 c7 4c be ad 20 2e af ad 1e   | 07a8 |
| 0780 | c7 d0 e9 a9 0f 4c 59 ae a2 0c 20 fb ae 4c 7d ac   | 0877 |
| 0790 | a2 f6 20 ab a6 aa b9 35 c7 c9 2a d0 0a c8 20 a7   | 08c4 |
| 07a0 | b1 f0 3a a2 3f d0 1a c9 28 d0 0f c8 20 a7 b1 a2   | 0858 |
| 07b0 | fb 20 ab a6 a2 34 d0 09 d0 07 20 a7 b1 f0 0c a2   | 0808 |
| 07c0 | 4a 20 0a bd aa 4c 44 ab e8 e8 e8 e8 20 fb ae a5   | 0924 |
| 07d0 | 49 20 2e af a5 4a 4c 7b ae e8 e8 e8 e8 e8 e8 e8   | 0a02 |
| 07e0 | ed e8 20 fb ae a5 49 20 2e af a5 4a f0 0a a2 00   | 080f |
| 07f0 | ad 13 c7 f0 03 20 60 a4 4c 7e 2a ff 00 fb 2d ff   | 07b5 |

EMUF. Bei „S“ dagegen (S ≙ save) wird ein Speicherbereich des EMUF im gleichen Verfahren an den Terminalrechner gesendet. Weil Hex-Code übertragen wird, kann man während der Übertragung direkt mitlesen.

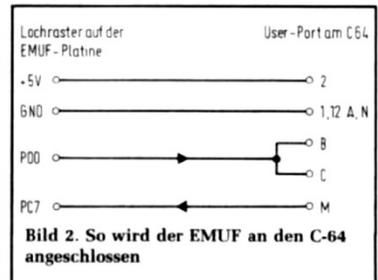
Die wichtigsten Monitor-Einsprünge sind in *Tabelle 2* zusammengestellt. In der Zeropage werden lediglich die Zellen von \$E3 bis \$FF benutzt. Die Zellen \$EA/\$EB dienen als NMI-Zeiger (für Single-Step auf \$FF0E voreingestellt), \$EC/\$ED enthalten den IRQ-Vektor (für Breaks auch auf \$FF0E gesetzt).

### Eine RAM-Speichererweiterung

Der 6502-EMUF kann bis zu 8 KByte RAM und 16 KByte EPROM ansprechen. Soll die Entwicklung auch der größtmöglichen Programme unterstützt werden, so sind zusätzlich 16 KByte RAM notwendig. Bei der Konzeption des EMUF wurde dies bereits berücksichtigt und die Ports auf die Adressen ab \$6000 gelegt. So sind insgesamt 24 KByte für RAM frei. Es gibt auch keine Platz- oder Dekodierschwierigkeiten, da die drei RAM-Bausteine 6264 einfach übereinander gelötet werden können. Das erste RAM steckt dabei ganz normal im entsprechenden Sockel. Das nächste wird einfach mit allen Anschlüssen (bis auf Pin 20) auf das erste gelötet. Lediglich Pin 20 muß am Lochraster mit SEL 2/3 verbunden werden. Genauso verfährt man mit dem dritten RAM-IC. Sein Anschluß 20 kommt an SEL 4/5.

### Single-Step-Betrieb

Der Anschluß SYNC am 6502 liefert nach dem Abarbeiten eines Maschinenbefehls ein Signal. Mit ihm kann ein NMI ausgelöst werden, der z. B. die Registerinhalte anzeigen kann. Der NMI-Zeiger bei \$EA/\$EB wird beim Kaltstart bereits auf die entsprechende Routine bei \$FF0E voreingestellt. Leider würde ein solcher Interrupt auch seine eigene Interrupt-Service-Routine anhalten. Des-



wegen wird in der kleinen Zusatzschaltung in Bild 4 der Interrupt immer dann hardwaremäßig (mit SEL C/D/E/F) verhindert, wenn der Befehl im EPROM liegt (die Schaltung macht übrigens von den freien Gattern der EMUF-Schaltung Gebrauch). Es ist kein großer Nachteil, daß damit nur noch Programme im RAM im Single-Step-Verfahren abgearbeitet werden können.

## Die Entwicklung eines EMUF-Programms

Nach der Entwicklung eines ausführlichen Flußdiagramms wird man mit dem Schreiben des Programms in Assembler beginnen. Dies geschieht auf dem Terminalrechner. Nach der Übersetzung liegt das assemblierte Maschinenprogramm dann irgendwo im RAM. Man wird es aus Sicherheitsgründen erst einmal abspeichern und dann ins RAM des EMUF übertragen. Dies kann zum Beispiel auch mit einem kleinen Basic-Programm geschehen. Jetzt kann der Test beginnen. Verläuft der erste Start nicht gleich erfolgreich und dafür sprechen alle Statistiken, so können spezielle Programmteile mit Breaks und Single-Step genau untersucht werden. Hier kann auch der eingebaute 65C02-Disassembler von großem Nutzen sein. Ist das Programm entzinkt und das Source-Programm entsprechend auf den neuesten Stand gebracht, dann ändert man die Startadresse des Programms so, daß es im EPROM-Bereich des EMUF liegt. Nach der Übersetzung muß es nur noch in ein EPROM gebrannt und dieses in den EMUF gesteckt werden.

## Literatur

- [1] Thienel, S., Sauer, T.: Mehr Speicher – mehr Anwendungen. mc 1985, Heft 5, Seite 96.

```

100 open 2,2,0,chr$(128+6)+chr$(0):get#2,a$:rem baudrate c-64 festlegen
110 :
120 print#2,chr$(255):rem          $ff senden zum messen der baudrate
130 :
140 get b$:if b$="" then 170:rem   taste gedruickt?
150 print#2,b$:rem                wenn ja --> senden
160 :
170 get#2,c$:if c$="" then 140:rem zeichen empfangen?
180 print c$:rem                  wenn ja --> ausgeben
190 goto 140:rem                  usw.
    
```

Bild 3. Ein kleines Terminalprogramm für den C-64

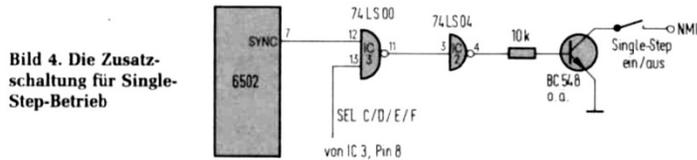


Bild 4. Die Zusatzschaltung für Single-Step-Betrieb

Tabelle 1. Die Liste der Monitorbefehle

|       |  |
|-------|--|
| XXXXN | aktuelle Adresse auf XXXX setzen   |
| +     | aktuelle Adresse um eins erhöhen   |
| -     | aktuelle Adresse um eins erniedrigen                                     |
| XX.   | Byte XX in aktuelle Adresse übernehmen                                   |
| *     | Befehl in aktueller Adresse disassemblieren                              |
| R     | Registerinhalte anzeigen   |
| M     | Inhalte der nächsten 16 Speicherstellen anzeigen (Hex- und ASCII-Format) |
| G     | Programm ab aktueller Adresse starten                                    |
| XXXXL | Hex-Code von aktueller Adresse bis zur Zelle XXXX einlesen               |
| XXXXS | Hex-Code von aktueller Adresse bis XXXX senden                           |

Tabelle 2. Die Systemeinsprünge

|        |        |  |
|--------|--------|--|
| \$F9D8 | OUTCH  | Akku als 8 Bits ausgeben   |
| \$F9D6 | OUTSP  | Blank ausgeben   |
| \$FA21 | CRLF   | CR und LF ausgeben   |
| \$F9B7 | PRTBYT | Akku als zwei ASCII-Zeichen ausgeben   |
| \$F9AC | PRTPNT | Inhalt von \$00FB/FC als Adresse (vier ASCII-Zeichen) ausgeben                   |
| \$FAAA | GETCH  | 8 Bits in Akku einlesen  |
| \$FA7B | GETBYT | zwei ASCII-Zeichen als Byte in den Akku einlesen                                 |
| \$F800 | COLD   | Kaltstart  |
| \$F82C | WARM   | Warmstart  |
| \$FF0E | NMIV   | NMI-Vektor für Register-Retten (wird auch in \$00EC/ED als Break-Vektor benutzt) |

## Der Kuchen ist aufgeschnitten

Die Tortengrafik zeigt den US-amerikanischen Softwaremarkt für „gehobene“ Mikrocomputer (Apple-II ist dabei). Die Aufteilung ist deshalb interessant, weil hier ganz klar die Machtverhältnisse zutage treten. In Europa dürften der Atari ST und der Amiga von Commodore zwar besser im Rennen liegen als in den USA, aber ob wesentlich besser, das ist die Frage. IBM-PCs aller Klassen und ihre Imitate bilden die Mehrheit. Ro.

