

Jürgen Plate, Herwig Feichtinger

## Mikroelektronik im Fernsehen

### Teil 5

Der auf unserer Titelseite abgebildete Computer gehört, technologisch gesehen, derzeit wohl zu den fortschrittlichsten. Er arbeitet mit dem Motorola-Mikroprozessor 68008, der intern mit 32-Bit-Registern arbeitet, aber eine preisgünstige Systemkonfiguration erlaubt, weil extern nur mit 8-Bit-Datenwegen gearbeitet wird. Der folgende Beitrag stellt die Schaltung dar und geht auf einige Konstruktionsdetails ein.

Der 68008-Computer ist Bestandteil der vom NDR produzierten Fernsehserie „Mikroelektronik“, die derzeit in den dritten Programmen der Nordkette ausgestrahlt wird. In den übrigen Ländern der Bundesrepublik ist, soweit bisher zu erfahren war, eine Ausstrahlung (wenn überhaupt) frühestens im September geplant. Interessierte können die 26 Folgen von je 15 Minuten Dauer aber auch auf zwei Videokassetten beziehen (Beta, VHS oder Video-2000 angeben!). Die Bestellung erfolgt durch Übersenden eines Schecks über 250 DM (248 DM + 2 DM Porto) an den Franzis-Verlag, Abt. ZV, Postfach 37 01 20, 8000 München 37. Oder Sie überweisen die 250 DM auf unser Postscheckkonto München 813 75-809.

Zu dem 68008-System gibt es vom Franzis-Software-Service (Postfach 37 01 20, 8000 München 37) ausführliches Begleitmaterial; es lohnt sich, die kostenlose Übersicht anzufordern. Der Software-Service liefert auch die nötige Software auf EPROMs. Die Hardware-Bausätze gibt es bei den Firmen Graf, Magnusstraße 13, 8960 Kempten sowie Elektronikladen Giesler + Danne, Eggestraße 70, 4930 Detmold 18. (Nur diese Firmen sind für den Vertrieb autorisiert.)

#### Mehr Bits

So heißt die NDR-Folge, in der die 68008-CPU vorgestellt wird. Sie müssen jetzt Ihr bisheriges System mit dem SBC-II, Tastatur- und Bildschirmkarte, Kassetteninterface und IOE nicht in die Ek-

ke stellen, denn alle Peripheriekarten arbeiten auch mit der neuen CPU zusammen. Lediglich der SBC-II geht einstweilen in Rente. Weil der Prozessor 68008 zwanzig Adressenleitungen hat, kann er ein Megabyte ( $1024 \times 1024$  Byte) adressieren. Für den Nachbau bedeutet das zunächst, daß das Motherboard vollständig bestückt sein muß und nicht wie vorher nur zu zwei Dritteln. Die Karten CPU68K (Prozessor) und ROA64 (Speicher) sind daher auch länger als die anderen Platinen.

Die Speicherkarte (Bild 1) ist recht einfach aufgebaut. Es werden 8 Speicherbausteine mit einer Organisation von  $8 \text{ K} \times 8$  Bit verwendet, was eine Kapazität von 64 KByte je Karte ergibt. Als Speicher sind entweder Schreib-Lese-Speicher (RAMs) HM 6264 oder EPROMs 2764 verwendet, wobei beide Baustein Typen bunt gemischt werden können. Das Signal zur Speicherbankumschaltung wird derzeit nicht verwendet. Über eine Reihe von vier Brücken kann mit Kurzschlußsteckern der Adressbereich der Karte (also die Decodierung der vier höchsten Bits) ausgewählt werden. Bei der ersten Karte sind alle Brücken kurzgeschlossen, bei der zweiten bleibt die erste frei, bei der dritten die zweite, bei der vierten die ersten beiden und so weiter. Bei sorgfältigem Löten kann eigentlich nichts schief gehen.

#### Die 68008-Hardware...

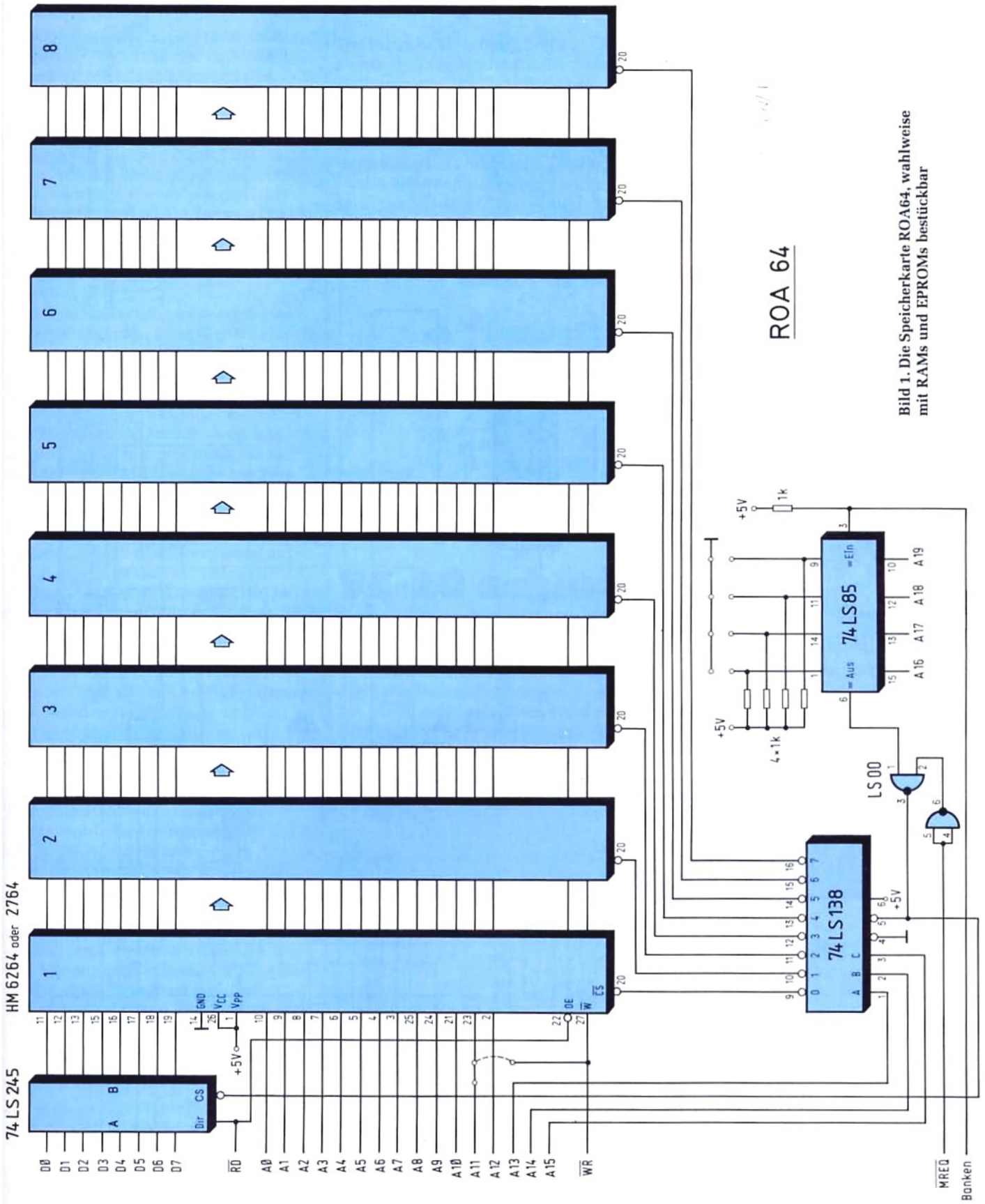
Die Prozessorkarte hat das gleiche Format wie die Speicherkarte. Zentrales

Bauteil ist der 68008 mit 48 Beinchen. Die Schaltung in Bild 2 bietet eigentlich keine Besonderheiten. Die Bustreiber 74245 und 74373 auf der rechten Seite erfüllen die gleiche Aufgabe beim 68008 wie beim Z80. Das einzig bemerkenswerte auf dieser Seite des Prozessors sind die beiden Interruptleitungen  $\overline{\text{INT}}$  und  $\overline{\text{NMI}}$ : Werden sie verbunden (Brücke bei den Pins 41/42), ist der Interrupt nicht maskierbar; bleibt sie offen, können die beiden Interruptleitungen maskiert werden. Im Augenblick werden sie ohnehin nicht benutzt. Der Taktgenerator läuft mit 8 MHz, durch Ändern einer Brücke kann der Takt auf 4 MHz heruntersetzt werden. Übrigens sind alle Brücken, die keiner Einstellung durch den „Normalbenutzer“ bedürfen, durch Brücken auf der Leiterkarte voreingestellt.

Auch die Reset-Logik mit dem Baustein 555 ist altbekannt. Aus dem Prozessor-Reset wird auch gleich ein Systemreset für den gesamten Computer abgeleitet. Legt man eine Taste über den  $10\text{-}\mu\text{F}$ -Elko, kann auch von Hand ein Reset ausgelöst werden. Wichtig für den Betrieb der CPU68K ist die Einstellung der Brücke auf der Jumperreihe am IC 74164 (etwa in Bildmitte). Da der Prozessor sehr schnell ist, muß man den Speicher- oder Peripherie-Zugriff verlangsamen. Bei offener Jumperreihe werden gar keine Wait-Zyklen beim Speicher- oder E/A-Zugriff erzeugt. Steckt ein Kurzschlußstecker ganz links, wird beim Schreiben ein Waitzyklus eingefügt, auf Position 2 ein Waitzyklus beim Lesen und Schreiben. Es wird also nach rechts immer langsamer; das Timing verläuft aber nicht ganz symmetrisch. Bei der derzeitigen Auslegung muß der Jumper ganz links stecken, weil sonst die GDP64-Karte nicht mitkommt. Eine Änderung der GDP64 zur „Beschleunigung“ ist im Schaltplanheft des Software-Service abgedruckt. Der nächste Hemmschuh sind die EPROMs; bei einer Zugriffszeit von mehr als 300 ns braucht der Prozessor schon zwei Wait-Zyklen. Wenn beim Betrieb des 68008 öfter Fehler auftreten (Illegal Instruction, Address Error), sollten Sie probeweise einen Wait-Zyklus mehr einschalten. Übrigens können alle ICs bis auf den 7405 auch vom Typ 74LS.. sein.

#### ...und die Software

Zum 68008 wurden zwei große Softwarepakete entwickelt. Einmal das Grundprogramm, das in vier EPROMs ab Adresse 0000 bis 7FFF (hex) steckt und



ROA 64

Bild 1. Die Speicherkarte ROA64, wahlweise mit RAMs und EPROMs bestückbar

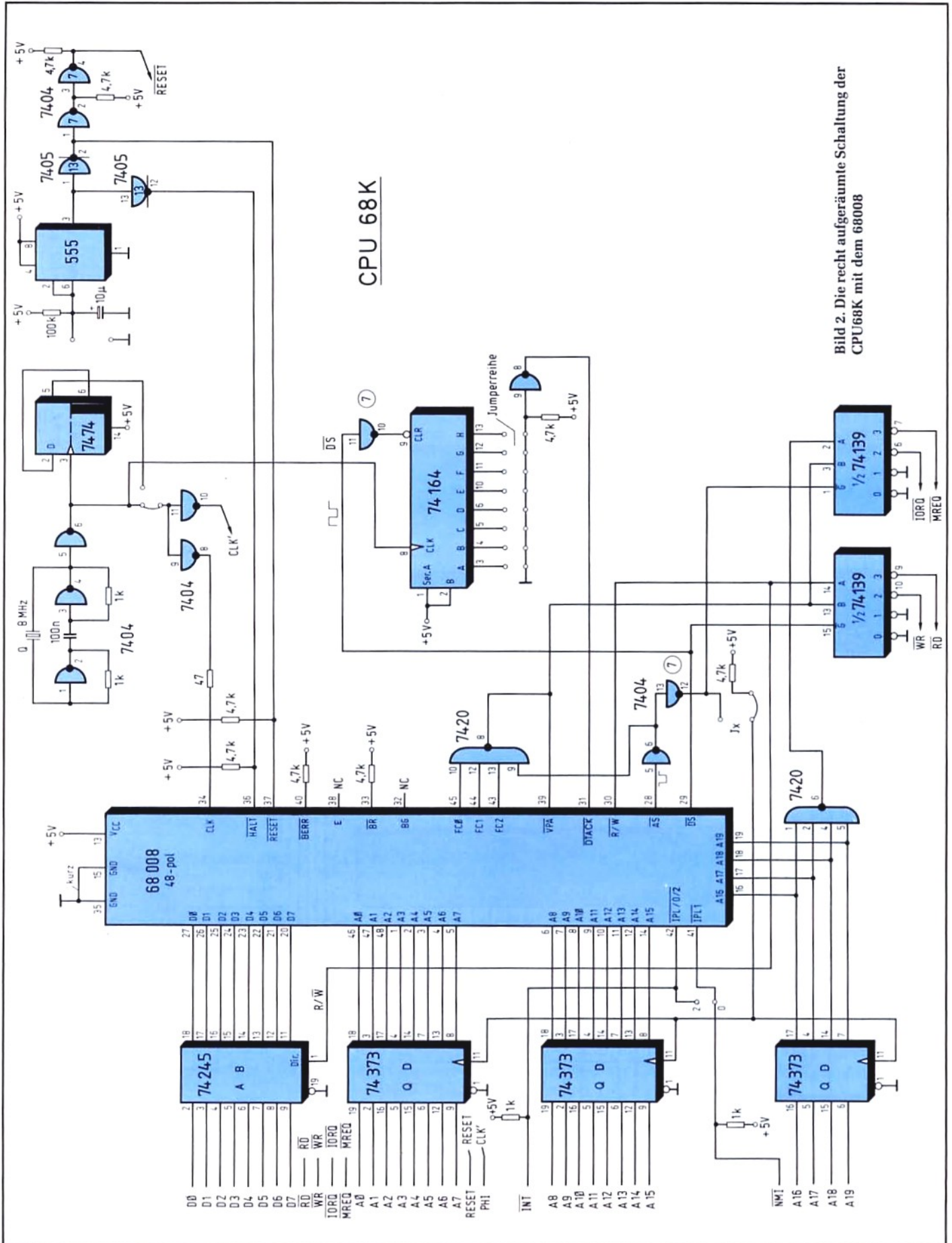


Bild 2. Die recht aufgeräumte Schaltung der CPU 68K mit dem 68008



zum anderen der Pascal-S-Interpreter, der im Bereich 10 000...17FFF „wohnt“.

Das sind zusammen schon 64 KByte EPROM-Software. Im Grundprogramm, das genau wie das des Z80 voll menügesteuert ist, befinden sich neben den schon bekannten Funktionen (Ändern, Starten, Symbole, Kassettenbetrieb, Debugger, EPROM-Programmierung) ein 68008-Assembler, ein Bildschirm-Texteditor, eine Bibliotheksfunktion und ein Menü zum Setzen von Optionen (z. B. Umleiten der Ausgabe auf einen Drucker). Eine 30seitige Beschreibung befindet sich im Schaltplanheft des Software-Service.

Zum Abschluß noch eine kurze Bemerkung zur Bibliotheksfunktion: Da ein Megabyte Speicher adressiert werden kann, bietet es sich eigentlich an, Programme, die öfter verwendet werden, in EPROMs abzulegen. Es fällt dann nicht nur das Laden von Kassette weg, sondern die EPROMs kosten auch nur ein sechstel des RAM-Preises. So kann ein Programm im Speicher (natürlich auch im RAM) mit einem Kopfeintrag versehen werden, der Namen, Länge, Startadresse und andere Informationen enthält. Solche Programme werden dann automatisch vom Bibliotheksmenü gefunden und können durch Tastendruck aufgerufen werden. Einzige Bedingung ist, daß die Konfiguration auf einer 2-KByte-Grenze beginnt. Bei kleinen Programmen sind an einer 2-KByte-Grenze auch mehrere Kopfeinträge möglich.

**Pascal-S**

Bei Pascal-S handelt es sich um einen Pascal-Compiler und -Interpreter. Verwendet wird eine Untermenge der Sprache Pascal, wie sie zum Beispiel in dem Buch „Systematisches Programmieren“ von N. Wirth verwendet wird (Teubner-Verlag). Eine kurze Sprachbeschreibung und das Quell-Listing finden sie im Pascal-Heft des Software-Service. Das mit dem Editor erstellte Pascal-C-Programm wird in einen Zwischencode übersetzt und dieser Code dann interpretiert. Erste Benchmark-Tests brachten Geschwindigkeitsresultate, die deutlich über denen anderer Mikrocomputer-Pascalversionen lagen. Die Arithmetik hat 18 Stellen, und der Aufruf von Unterprogrammen, die mit Assembler erstellt wurden, ist möglich – also auch zum Beispiel der Aufruf des Editors oder anderer Programme aus der Grundsoftware. Alles in allem ein zukunftsträchtiges Computersystem! *Fortsetzung folgt*

# Deutsche Zeichen für den Eurocom-II-Editor

Der Bildschirm-Editor „Edit 85“ für den Eurocom-II verwendet zur Zeichendarstellung nicht den Zeichensatz des Monitors, sondern enthält im RAM-Bereich einen eigenen. Dadurch ist es möglich, selten benötigte ASCII-Zeichen zu ändern. Geändert wurden die Zeichen, die nach Voreinstellung eines Druckers (z. B. des MX-80 F/T von Epson) auf den

deutschen Zeichensatz auch eine entsprechende Druckerausgabe eines mit dem Editor geschriebenen Textes ermöglichen. Die erforderlichen Änderungen und die Zuordnung der deutschen zu den amerikanischen Zeichen zeigt das Bild. Die deutschen Zeichen sind auf der Tastatur über die ASCII-Zeichen erreichbar. *Willi Bieler*

Zeichen ASCII	Deutsch	Anfangsadresse im RAM	Geänderter Character-Satz mit deutschen Zeichen
@	§	2BAB	38 40 30 48 30 08 70
[	Ä	2C68	88 70 88 88 F8 88 88
/	ö	2C6F	88 70 88 88 88 88 70
]	Ü	2C76	88 00 88 88 88 88 70
<	ä	2D48	50 00 70 08 78 88 78
!	ö	2D4F	50 00 70 88 88 88 70
~	ü	2D56	50 00 88 88 88 98 68
	ß	2D5D	70 88 88 80 88 88 80

Umcodierung nationaler Sonderzeichen beim Programm „Edit 85“

# VC-20 schluckt Hexadezimalzahlen

Will man Maschinenprogramme in Form von Hexadezimalzahlen ins RAM des VC-20 eintippen, dann leistet das im Bild abgedruckte Programm gute Dienste. Es erwartet zunächst die Anfangsadresse als vierstellige Hexadezimalzahl mit abschließendem Return. Andere Eingaben werden abgewiesen. Dann

können nacheinander die Hexadezimalwerte eingegeben werden. In jede Zeile werden vom Rechner automatisch die Adresse des ersten Wertes und dahinter die vier folgenden Werte geschrieben. Mit der Eingabe von Carriage-Return bricht das Programm ab.

*Hans-Jürgen Kellermann*

```

70 REM HEXEINGABE
80 REM 08.08.82
90 PRINT "HEXADEZIMALE EINGABE"
100 INPUT "ANFANG":A#:REM STARTADRESSE
110 IF LEN(A#)<>4 THEN 100:REM 4-STELLIG
120 P=0:F=0:FOR I=1 TO 4:I#=MID$(A#,I,1):D=ASC(I#)-48:IF D>9 THEN D=D-7
130 IF D<0 OR D>15 THEN F=1:REM FEHLER
140 P=P*16+D:NEXT I:REM ANF. DEZIMAL
150 IF F=1 THEN 100:REM FEHLERFLAG
160 PRINT " "TAB(14):P:PRINT:PRINT A#:
170 D=4*(D/4-INT(D/4))
180 PRINT TAB(6+3*D):REM 1. PLATZ
190 GOSUB 250:D=16*I:GOSUB 250:D=D+I:PRINT " ":E=E+1:POKEP,D:P=P+1:REM INS RAM
200 IF E<>4 THEN 190:REM 4 WERTE/ZEILE
210 PRINT:D=P:I#="":REM NEUE ZEILE
220 FOR I=1 TO 4:F=16*(D/16-INT(D/16)):D=INT(D/16):IF F>9 THEN F=F+7
230 I#=CHR$(48+F)+I#:NEXT I:REM HEX. ADR.
240 PRINT I# " ":E=0:GOTO 190
250 GET I#:IF I#="" THEN 250:REM EINGABE
260 IF I#<CHR$(13) THEN STOP:REM ENDE
270 I=ASC(I#)-48:IF I>9 THEN I=I-7
280 IF I<0 OR I>15 THEN 250:REM UNGÜELTIG
290 PRINT I#:RETURN
    
```

Maschinenprogramme kann man mit Hilfe dieser Basic-Routine ins RAM des VC-20 eintippen