

Joachim König

Z80-EMUF als universelle Fernbedienung

Fertig zu kaufende Fernsteuerungen sind zwar heute schon recht preiswert zu haben, lassen sich aber kaum an individuelle Bedürfnisse anpassen. Wir zeigen Ihnen, wie man mit modernen Bausteinen eine Fernsteuerung aufbaut, die – dank Z80-EMUF – auf Tastendruck auch komplexe Abläufe bewältigt.

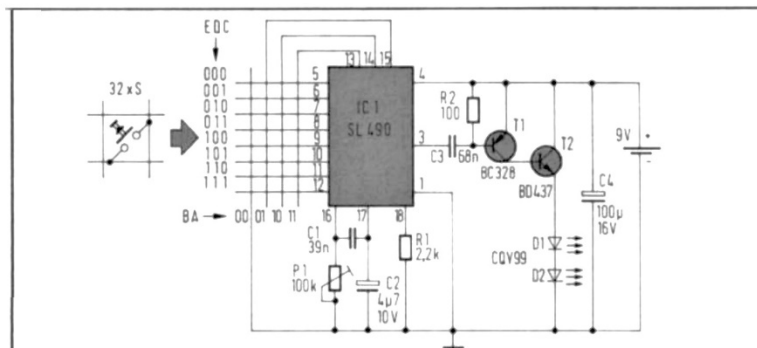


Bild 1. Der Sender wird mit einer 9-V-Batterie versorgt – ein Schalter ist nicht erforderlich



Bild 2. Das Datenwort besteht aus sechs kurzen Impulsen, deren Zwischenräume die Information darstellen. Ein kurzer Zwischenraum bedeutet „1“, ein langer „0“. Hier ist das Signal für das Datenwort 01101 (hex 0D) dargestellt

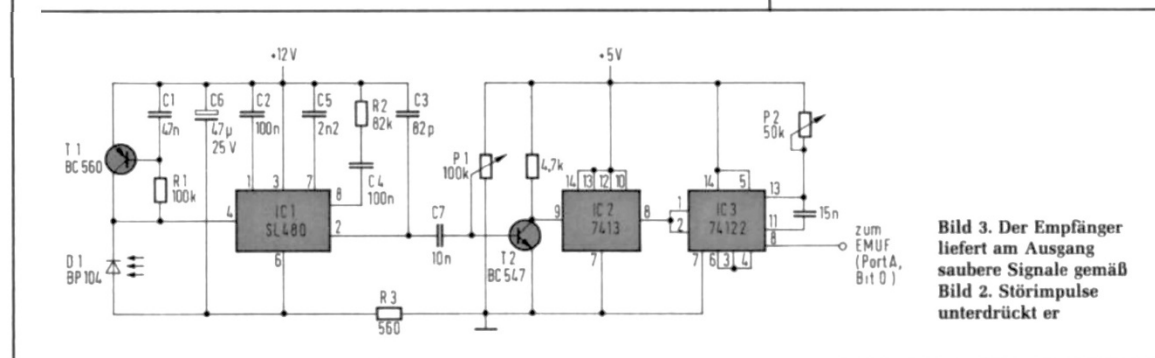


Bild 3. Der Empfänger liefert am Ausgang saubere Signale gemäß Bild 2. Störimpulse unterdrückt er

Das Kernstück des Senders ist die integrierte Schaltung SL 490 (Bild 1). In ihr ist die ganze Logik enthalten, die zum Betrieb des Senders nötig ist. Sie decodiert das Tastenfeld und erzeugt ein PPM-Signal (Puls-Pausen-Modulation).

Es können maximal 32 Tasten angeschlossen werden. Jede Taste entspricht einem fünfstelligen Binärcode. Drückt man eine Taste, so erscheinen am Ausgang des SL 490 sechs kurze Impulse.

Die Pausen zwischen diesen Impulsen enthält die Information. Eine kurze Pause bedeutet, „logisch 1“, eine lange „logisch 0“ (Bild 2). Die beiden als Darlington geschalteten Transistoren T1 und T2 dienen als Verstärker und steuern direkt die Infrarotsender D1 und D2. Der SL 490 besitzt eine Abschaltautomatik, so daß nur wenige μ A Strom fließen, wenn keine Taste gedrückt ist. Deswegen ist kein Schalter in der Stromversorgung nötig. Als Stromversorgung genügt eine einfache 9-V-Batterie. Mit P1 kann man die Zeit zwischen den Impulsen justieren. Das Verhältnis zwischen „1“ und „0“ beträgt 1:1,5.

Der Empfänger

Das mit der Empfängerdiode D1 gewonnene Signal wird mit dem Operationsverstärker SL 480 verstärkt (Bild 3).

Dann wird es über T2 dem Schmitt-Trigger IC2 zugeleitet. Der Schmitt-Trigger macht saubere Rechteckimpulse aus dem Signal, und das Monoflop IC3 verlängert diese Impulse auf 100 μ s. Da das Monoflop nicht nachgetriggert werden kann, werden Störimpulse ausgeblendet. Am Ausgang von IC3 steht eine Impulsfolge, wie in Bild 2 gezeigt, bereit. Diese Impulse werden an Port A, Bit 0 geführt. Mit dem Potentiometer P1 kann die Empfindlichkeit eingestellt werden.

0002	portid	02h	0070'	SE	ld	e, (hl)	! get low byte
0000	start:	00h	0071'	23	inc	hl	! get high byte
0003	sp, 0074h	0074'	0072'	56	ld	(table), de	! set new kontextbasis
0006	hl, inittab	0075'	0073'	ED 53 0000'	ld	endret	! adresse der konvertabelle
0008	b, 4	0076'	0074'	21 0153'	jp	hl, contab	! physische tastennummer nach e
0009	hl, (hl)	0077'	0075'	16 00	ld	d, 0	! hl = contab + physische tastennummer
0008	hl, (hl)	0078'	0076'	19	add	hl, de	! log tastennummer nach e
0009	hl, (hl)	0079'	0077'	01F4	ret	hl, 0500	! startwert fuer time-out counter
0008	hl, (hl)	0080'	0078'	50	ld	d, b	! save taste
0009	hl, (hl)	0081'	0079'	CD 0091'	erl:	call getwol	! hole naechste taste
0008	hl, (hl)	0082'	0080'	7A	ld	z	! time-out
0009	hl, (hl)	0083'	0081'	88	cp	b, a	! gleiche taste?
0008	hl, (hl)	0084'	0082'	C0	ret	nz	! andere taste gibt auch time-out
0009	hl, (hl)	0085'	0083'	E5	endret	hl	! setze return adresse fuer fehlerfall
0008	hl, (hl)	0086'	0084'	21 0091'	ld	hl, getwol	! clear number
0009	hl, (hl)	0087'	0085'	22 0002'	ld	(retadr), hl	! warte auf eine 1
0008	hl, (hl)	0088'	0086'	E1	pop	hl	! b, 0
0009	hl, (hl)	0089'	0087'	06 00	ld	b, 0	! nz, getw2
0008	hl, (hl)	0090'	0088'	D8 00	ld	g10: in	! a, (portid)
0009	hl, (hl)	0091'	0089'	E6 01	and	1	! nz, getw2
0008	hl, (hl)	0092'	0090'	C2 0088'	jp	nz, getw2	! a, h
0009	hl, (hl)	0093'	0091'	28	dec	hl	! or 1
0008	hl, (hl)	0094'	0092'	00A3'	ld	a, h	! ret z
0009	hl, (hl)	0095'	0093'	85	or	1	! g10
0008	hl, (hl)	0096'	0094'	C8	ret	z	! setze returnadresse fuer fehlerfall
0009	hl, (hl)	0097'	0095'	C3 0098'	ld	hl, getwol	! clear number
0008	hl, (hl)	0098'	0096'	21 00A9'	ld	(retadr), hl	! warte auf 1
0009	hl, (hl)	0099'	0097'	22 0002'	ld	b, 0	! check ob bit = 1
0008	hl, (hl)	0100'	0098'	06 00	ld	b, 0	! hole 1. bit
0009	hl, (hl)	0101'	0099'	08B1'	ld	a, (portid)	! schiebe bit nach b
0008	hl, (hl)	0102'	0100'	E6 01	and	1	! 2. bit
0009	hl, (hl)	0103'	0101'	CA 00B1'	ld	z, getw1	! 3. bit
0008	hl, (hl)	0104'	0102'	00B8'	ld	call getbit	! 4. bit
0009	hl, (hl)	0105'	0103'	00B9'	ld	call getbit	! 5. bit
0008	hl, (hl)	0106'	0104'	00C0'	ld	call getbit	! init timeout counter
0009	hl, (hl)	0107'	0105'	00C1'	ld	call getbit	! clear counter
0008	hl, (hl)	0108'	0106'	00C2'	ld	call getbit	! warte auf low flanke
0009	hl, (hl)	0109'	0107'	00C3'	ld	call getbit	! warte auf high flanke
0008	hl, (hl)	0110'	0108'	00C4'	ld	call getbit	! erhoehe zaehler
0009	hl, (hl)	0111'	0109'	00C5'	ld	call getbit	! decrement time out counter
0008	hl, (hl)	0112'	0110'	00C6'	ld	call getbit	! wenn timeout counter = 0
0009	hl, (hl)	0113'	0111'	00C7'	ld	call getbit	! dann springe auf fehlerroutine
0008	hl, (hl)	0114'	0112'	00C8'	ld	call getbit	
0009	hl, (hl)	0115'	0113'	00C9'	ld	call getbit	
0008	hl, (hl)	0116'	0114'	00CA'	ld	call getbit	
0009	hl, (hl)	0117'	0115'	00CB'	ld	call getbit	
0008	hl, (hl)	0118'	0116'	00CC'	ld	call getbit	
0009	hl, (hl)	0119'	0117'	00CD'	ld	call getbit	
0008	hl, (hl)	0120'	0118'	00CE'	ld	call getbit	
0009	hl, (hl)	0121'	0119'	00CF'	ld	call getbit	
0008	hl, (hl)	0122'	0120'	00D0'	ld	call getbit	
0009	hl, (hl)	0123'	0121'	00D1'	ld	call getbit	
0008	hl, (hl)	0124'	0122'	00D2'	ld	call getbit	
0009	hl, (hl)	0125'	0123'	00D3'	ld	call getbit	
0008	hl, (hl)	0126'	0124'	00D4'	ld	call getbit	
0009	hl, (hl)	0127'	0125'	00D5'	ld	call getbit	
0008	hl, (hl)	0128'	0126'	00D6'	ld	call getbit	
0009	hl, (hl)	0129'	0127'	00D7'	ld	call getbit	
0008	hl, (hl)	0130'	0128'	00D8'	ld	call getbit	
0009	hl, (hl)	0131'	0129'	00D9'	ld	call getbit	
0008	hl, (hl)	0132'	0130'	00DA'	ld	call getbit	
0009	hl, (hl)	0133'	0131'	00DB'	ld	call getbit	
0008	hl, (hl)	0134'	0132'	00DC'	ld	call getbit	
0009	hl, (hl)	0135'	0133'	00DD'	ld	call getbit	
0008	hl, (hl)	0136'	0134'	00DE'	ld	call getbit	
0009	hl, (hl)	0137'	0135'	00DF'	ld	call getbit	
0008	hl, (hl)	0138'	0136'	00E0'	ld	call getbit	
0009	hl, (hl)	0139'	0137'	00E1'	ld	call getbit	
0008	hl, (hl)	0140'	0138'	00E2'	ld	call getbit	
0009	hl, (hl)	0141'	0139'	00E3'	ld	call getbit	
0008	hl, (hl)	0142'	0140'	00E4'	ld	call getbit	
0009	hl, (hl)	0143'	0141'	00E5'	ld	call getbit	
0008	hl, (hl)	0144'	0142'	00E6'	ld	call getbit	
0009	hl, (hl)	0145'	0143'	00E7'	ld	call getbit	
0008	hl, (hl)	0146'	0144'	00E8'	ld	call getbit	
0009	hl, (hl)	0147'	0145'	00E9'	ld	call getbit	
0008	hl, (hl)	0148'	0146'	00EA'	ld	call getbit	
0009	hl, (hl)	0149'	0147'	00EB'	ld	call getbit	
0008	hl, (hl)	0150'	0148'	00EC'	ld	call getbit	
0009	hl, (hl)	0151'	0149'	00ED'	ld	call getbit	
0008	hl, (hl)	0152'	0150'	00EE'	ld	call getbit	
0009	hl, (hl)	0153'	0151'	00EF'	ld	call getbit	
0008	hl, (hl)	0154'	0152'	00F0'	ld	call getbit	
0009	hl, (hl)	0155'	0153'	00F1'	ld	call getbit	
0008	hl, (hl)	0156'	0154'	00F2'	ld	call getbit	
0009	hl, (hl)	0157'	0155'	00F3'	ld	call getbit	
0008	hl, (hl)	0158'	0156'	00F4'	ld	call getbit	
0009	hl, (hl)	0159'	0157'	00F5'	ld	call getbit	
0008	hl, (hl)	0160'	0158'	00F6'	ld	call getbit	
0009	hl, (hl)	0161'	0159'	00F7'	ld	call getbit	
0008	hl, (hl)	0162'	0160'	00F8'	ld	call getbit	
0009	hl, (hl)	0163'	0161'	00F9'	ld	call getbit	
0008	hl, (hl)	0164'	0162'	00FA'	ld	call getbit	
0009	hl, (hl)	0165'	0163'	00FB'	ld	call getbit	
0008	hl, (hl)	0166'	0164'	00FC'	ld	call getbit	
0009	hl, (hl)	0167'	0165'	00FD'	ld	call getbit	
0008	hl, (hl)	0168'	0166'	00FE'	ld	call getbit	
0009	hl, (hl)	0169'	0167'	00FF'	ld	call getbit	
0008	hl, (hl)	0170'	0168'	0100'	ld	call getbit	
0009	hl, (hl)	0171'	0169'	0101'	ld	call getbit	
0008	hl, (hl)	0172'	0170'	0102'	ld	call getbit	
0009	hl, (hl)	0173'	0171'	0103'	ld	call getbit	
0008	hl, (hl)	0174'	0172'	0104'	ld	call getbit	
0009	hl, (hl)	0175'	0173'	0105'	ld	call getbit	
0008	hl, (hl)	0176'	0174'	0106'	ld	call getbit	
0009	hl, (hl)	0177'	0175'	0107'	ld	call getbit	
0008	hl, (hl)	0178'	0176'	0108'	ld	call getbit	
0009	hl, (hl)	0179'	0177'	0109'	ld	call getbit	
0008	hl, (hl)	0180'	0178'	010A'	ld	call getbit	
0009	hl, (hl)	0181'	0179'	010B'	ld	call getbit	
0008	hl, (hl)	0182'	0180'	010C'	ld	call getbit	
0009	hl, (hl)	0183'	0181'	010D'	ld	call getbit	
0008	hl, (hl)	0184'	0182'	010E'	ld	call getbit	
0009	hl, (hl)	0185'	0183'	010F'	ld	call getbit	
0008	hl, (hl)	0186'	0184'	0110'	ld	call getbit	
0009	hl, (hl)	0187'	0185'	0111'	ld	call getbit	
0008	hl, (hl)	0188'	0186'	0112'	ld	call getbit	
0009	hl, (hl)	0189'	0187'	0113'	ld	call getbit	
0008	hl, (hl)	0190'	0188'	0114'	ld	call getbit	
0009	hl, (hl)	0191'	0189'	0115'	ld	call getbit	
0008	hl, (hl)	0192'	0190'	0116'	ld	call getbit	
0009	hl, (hl)	0193'	0191'	0117'	ld	call getbit	
0008	hl, (hl)	0194'	0192'	0118'	ld	call getbit	
0009	hl, (hl)	0195'	0193'	0119'	ld	call getbit	
0008	hl, (hl)	0196'	0194'	011A'	ld	call getbit	
0009	hl, (hl)	0197'	0195'	011B'	ld	call getbit	
0008	hl, (hl)	0198'	0196'	011C'	ld	call getbit	
0009	hl, (hl)	0199'	0197'	011D'	ld	call getbit	
0008	hl, (hl)	0200'	0198'	011E'	ld	call getbit	
0009	hl, (hl)	0201'	0199'	011F'	ld	call getbit	
0008	hl, (hl)	0202'	0200'	0120'	ld	call getbit	
0009	hl, (hl)	0203'	0201'	0121'	ld	call getbit	
0008	hl, (hl)	0204'	0202'	0122'	ld	call getbit	
0009	hl, (hl)	0205'	0203'	0123'	ld	call getbit	
0008	hl, (hl)	0206'	0204'	0124'	ld	call getbit	
0009	hl, (hl)	0207'	0205'	0125'	ld	call getbit	
0008	hl, (hl)	0208'	0206'	0126'	ld	call getbit	
0009	hl, (hl)	0209'	0207'	0127'	ld	call getbit	
0008	hl, (hl)	0210'	0208'	0128'	ld	call getbit	
0009	hl, (hl)	0211'	0209'	0129'	ld	call getbit	
0008	hl, (hl)	0212'	0210'	012A'	ld	call getbit	
0009	hl, (hl)	0213'	0211'	012B'	ld	call getbit	
0008	hl, (hl)	0214'	0212'	012C'	ld	call getbit	
0009	hl, (hl)	0215'	0213'	012D'	ld	call getbit	
0008	hl, (hl)	0216'	0214'	012E'	ld	call getbit	
0009	hl, (hl)	0217'	0215'	012F'	ld	call getbit	
0008	hl, (hl)	0218'	0216'	0130'	ld	call getbit	
0009	hl, (hl)	0219'	0217'	0131'	ld	call getbit	
0008	hl, (hl)	0220'	0218'	0132'	ld	call getbit	
0009	hl, (hl)	0221'	0219'	0133'	ld	call getbit	
0008	hl, (hl)	0222'	0220'	0134'	ld	call getbit	
0009	hl, (hl)	0223'	0221'	0135'	ld	call getbit	
0008	hl, (hl)	0224'	0222'	0136'	ld	call getbit	
0009	hl, (hl)	0225'	0223'	0137'	ld	call getbit	
0008	hl, (hl)	0226'	0224'	0138'	ld	call getbit	
0009	hl, (hl)	0227'	0225'	0139'	ld	call getbit	
0008	hl, (hl)	0228'	0226'	013A'	ld	call getbit	
0009	hl, (hl)	0229'	0227'	013B'	ld	call getbit	
0008	hl, (hl)	0230'	0228'	013C'	ld	call getbit	
0009	hl, (hl)	0231'	0229'	013D'	ld	call getbit	
0008	hl, (hl)	0232'	0230'	013E'	ld	call getbit	
0009	hl, (hl)	0233'	0231'	013F'	ld	call getbit	
0008	hl, (hl)	0234'	0232'	0140'	ld	call getbit	
0009	hl, (hl)	0235'	0233'	0141'	ld	call getbit	
0008	hl, (hl)	0236'	0234'	0142'	ld	call getbit	
0009	hl, (hl)	0237'	0235'	0143'	ld	call getbit	
0008	hl, (hl)	0238'	0236'	0144'	ld	call getbit	
0009	hl, (hl)	0239'	0237'	0145'	ld	call getbit	
0008	hl, (hl)	0240'	0238'	0146'	ld	call getbit	
0009	hl, (hl)	0241'	0239'	0147'	ld	call getbit	
0008	hl, (hl)	0242'	0240'	0148'	ld	call getbit	
0009	hl, (hl)	0243'	0241'	0149'	ld	call getbit	
0008	hl, (hl)	0244'	0242'	014A'	ld	call getbit	
0009	hl, (hl)	0245'	0243'	014B'	ld	call getbit	
0008	hl, (hl)	0246'	0244'	014C'	ld	call getbit</	

[illegible]

Mit P2 wird die Impulsbreite auf 100 μ s eingestellt. Wer kein Oszilloskop besitzt, stellt P2 in Mittenstellung. P1 muß so eingestellt werden, daß in Ruhelage (keine Impulse vom Sender) Low-Pegel am Eingang von IC2 anliegt.

Das Programm

Bei der Entwicklung des Programmes (Bild 4) wurde auf eine möglichst einfache Erweiterungsmöglichkeit der Funktion Wert gelegt. Die Software stellt nur die Auswertung der empfangenen Impulsfolgen und die davon abhängigen Verzweigungen dar. Die Routinen, die nötig sind, um die einzelnen Funktionen auszuführen, sind abhängig von der jeweiligen Funktion. Als Beispiel sind jedoch zwei Routinen abgedruckt, die eine Taste bzw. einen Schalter emulieren. Der Programmteil ENDRET wartet auf das Loslassen der gedrückten Taste, so daß im Programm eine Funktion nicht mehrmals ausgeführt wird, wenn man die Taste festhält.

Von den 32 zur Verfügung stehenden Tastencodes werden 16 benutzt, um die Aufgabe der Fernbedienung umzuschalten. So kann man 16 Aufgaben mit jeweils 16 Funktionen, also 256 Funktionen, ausführen. In der Tabelle BASIS stehen die 16 Adressen der jeweiligen Funktionsbasissen.

Ein Beispiel:
Wenn in der Variablen TABLE die Adresse der Tabelle LAMPE steht, werden beim Druck auf eine der Tasten 0...15 die jeweiligen Adressen aus der Tabelle LAMPE geholt, und das Programm wird dort ausgeführt. Eine Erweiterung der Funktionen ist also dadurch möglich, daß man die Adresse der neuen Funktion in die entsprechende Tabelle einbaut, ohne das Kernprogramm zu ändern.

Die Routine CONVERT dient zum Umwandeln des physikalischen Tastencodes in den logischen. Je nach Verdrahtung des Tastenfeldes ergeben sich die übertragenen Tastencodes. Um nun die Codes in eine einheitliche Reihenfolge zu bringen, müssen sie umgewandelt werden. Das abgedruckte Programm nimmt eine 1:1-Umwandlung vor.

Wenn die 256 Funktionen zu wenig sind, der kann das Programm auch so umschreiben, daß eine der 32 Tasten die Umschaltung einleitet und die nächste Taste die Aufgabe festlegt. Dann sind insgesamt 992 Funktionen möglich.

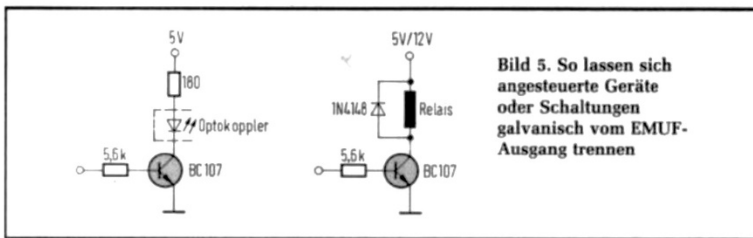


Bild 5. So lassen sich angesteuerte Geräte oder Schaltungen galvanisch vom EMUF-Ausgang trennen

In Bild 5 sind einige Möglichkeiten der Steuerung gezeigt. Man kann mit den PIOs über einen Transistor direkt ein Relais ansteuern und damit 220-V-Geräte bedienen. Wichtig: Das gesteuerte Gerät und der EMUF müssen galvanisch getrennt sein. Falls die Reichweite der Fernbedienung (ca. 6 m) unbefriedigend ist, kann man die Sendedioden in Reflektoren setzen.

Linken des Programmes

Das Listing ist so geschrieben, daß die Adressen des Programmes erst beim Linken festgelegt werden. Dazu ist ein Linker nötig, der das Programm nicht ab-

lauffähig in den Speicher legt. Der Linker von Pascal MT+ ist dazu in der Lage. Um das Objektfile damit zu linken, bekommt es den Namen FERNB.ERL; dann lautet der Aufruf LINK FERNB/P:0/D:8000. Damit wird das Programm auf Adresse 0 gelegt, die Daten auf Adresse 8000h. Wer keinen Linker besitzt, muß alle Adressen, die mit einem doppelten Anführungszeichen gekennzeichnet sind, entsprechend verändern.

Literatur

- [1] Remote Control Data, Plessey Semiconductors.
- [2] IR-Fernsteuerung. Elektor, Heft 139, Seite 97.

Wordstar liest dBase-III-Dateien

Nichts Neues, werden Sie sagen, das steht doch im Handbuch! Das stimmt, allerdings werden Sie bei Wordstar-Versionen vor 3.4 Schwierigkeiten mit den deutschen Sonderzeichen bekommen. Mit dem Befehl COPY...TO:...DELIMITED liefert dBase-III für Wordstar lesbare Daten, solange keine deutschen Sonderzeichen verwendet werden. Den von diesem Befehl erstellten Sonderzeichen müssen anschließend noch die Codes \$1B und \$1C voran- bzw. nachgestellt

werden, damit sie Wordstar/Mailmerge als Buchstaben und nicht als Steuerzeichen liest.

Dies erledigt das im Bild abgedruckte Basic-Programm, das zwar langsam aber nützlich ist. Damit können Sie Serienbriefe mit Hilfe von Mailmerge erstellen, deren Adressen aus dBase-III-Dateien stammen – auch an den Herrn Häberle aus Böblingen oder an die Frau Weißer aus München.

Roland Essinger

```
10 REM Convertierung dBase III- in WordStar/MailMerge-Daten
20 REM * 1986 * R.Essinger * Sindelfingen
30 CLS:PRINT:PRINT " Rest-Convertierung der Buchstaben ä ö ü Ä Ö Ü und ß einer"
40 PRINT:PRINT " dBase III/delimited-Datei xxx.txt in eine MailMerge-Datendatei
im Laufwerk B"
50 PRINT:PRINT " Bitte die Dateinamen eingeben (ohne Laufwerksangabe)!"
60 PRINT:LINE INPUT " Name der Quelldatei: ";N$;N$="B:";N$="OPEN" I",1,N$
70 PRINT:LINE INPUT " Name der Zieldatei: ";N$;N$="B:";N$="OPEN" O",2,N$
80 PRINT:PRINT " Bitte warten bis zur Meldung 'ENDE der Convertierung'"
90 WHILE NOT EOF(1)
100 A=ASC(INPUT$(1,1));A$=CHR$(A)
110 IF A=21 OR A=142 OR A=153 OR A=154 OR A=132 OR A=148 OR A=129 OR A=225
THEN A$=CHR$(27)+CHR$(A)+CHR$(28)
120 PRINT#2,A$
130 WEND:CLOSE 1:CLOSE 2
140 PRINT:PRINT " ENDE der Convertierung der Datei "N$
150 PRINT:END
```

dBase-III-Dateien mit deutschen Sonderzeichen konvertiert dieses Programm in Wordstar-Dateien

Wordstar preiswerter

In der Bundesrepublik haben es die CP/M-Computeristen gut: Die Firmen Digital Research, Micropro und Markt & Technik haben einen zweiten Anlauf genommen, die CP/M-2.2-/3-Welt zu beleben. In Sonderverträgen ist es zum Beispiel gelungen, den Endpreis für Wordstar unter 200 DM zu drücken. Am interessantesten für alle in der 80er-Welt Engagierten dürfte der Entschluß sein, die CP/M-Dokumentation als autorisierte Übersetzung des Originals preiswert in den Handel zu bringen. Damit werden jetzt auch Computeristen dieses Betriebssystem besser nutzen können, die bisher mit der englischen Version nicht zurecht kamen, – welche nicht nur wegen der Fremdsprachlichkeit schwer zu verstehen war, sondern auch, weil sie viele reichlich dunkle Stellen enthält, die man nur nach mehrmaligem Probieren exakt versteht.

Für mc ist diese Entwicklung besonders interessant, weil wir glauben, daß die CP/M-Welt in der Bundesrepublik durch unsere Aktivitäten in Sachen mc-CP/M-Computer (1982) entscheidende Impulse erhielt. Heute, vier Jahre danach, hält CP/M das Marktsegment der 8-Bit-Computer fest in seiner Hand, zwar nicht so spektakulär wie MS-DOS die 16-Bit-Welt besetzt, denn CP/M ist bewährt, aber nicht betagt. Die Industrie hat ebenfalls in bestimmten Marktsegmenten mit Erfolg auf CP/M gesetzt. Zum Beispiel hat die Firma Schneider von Ihren CPC-Geräten weit über 100 000 Stück abgesetzt, alles Geräte, die unter CP/M laufen. Die Anstrengungen der Firma Philips auf dem Heimcomputermarkt gelten einem MSX-System, das letztlich ebenfalls unter CP/M läuft.

In Deutschland sind

- Triumph-Adler,
- Taylorlix,
- Olympia,
- Philips,
- Amstrad/Schneider,
- Commodore,
- Vortex,
- Graf Electronic (mc-CP/M-Computer)

die wichtigsten von Digital Research genannten Lizenznehmer für CP/M. Darüber hinaus gebe es noch eine äußerst hohe Zahl von Einzelanwendern, die ein lizenziertes CP/M benutzen. Ro.