Rolf-Dieter-Klein

Schnittstelle zur analogen Welt

Seit einiger Zeit schon sind preiswerte und leistungsfähige Bausteine auf dem Markt, die digitale Signale in analoge Spannungen oder Ströme umwandeln oder in der Gegenrichtung analoge Meßwerte in digitale Werte umsetzen. In diesem Artikel werden drei Schaltungen besprochen, die für den NDR-Klein-Computer und für den mc-CP/M-Computer entworfen wurden, sich aber eigentlich mit jedem Prozessor verwenden lassen.

Zunächst wird ein D/A-Wandler mit zwei Kanälen besprochen, dessen Bauteile nicht nur preiswert, sondern auch weit verbreitet sind. Als Zweites folgt ein A/D-Umsetzer mit 10-Bit-Auflösung und zum Schluß wird ein A/D-Wandler mit 16 Kanälen zu je 8 Bit besprochen.

Zu allen Schaltungen existieren Platinen für den NDR-Computer, sie sind bei der Firma Graf in Kempten erhältlich.

Der D/A-Wandler

Der hier vorgestellte Digital-/Analog-Umsetzer besitzt zwei Kanäle mit einer Auflösung von je 8 Bit. Die Schaltung ist für den Einsatz am Z80 und 68008 ausgelegt, kann aber sicher auch für andere Rechner benutzt werden. Als D/A-Bausteine werden ICs vom Typ ZN-428-8 von Ferranti verwendet, die besonders einfach anzusteuern und schnell bei der Wandlung sind.

Bild 1 zeigt die Schaltung. Mit den beiden Vergleichern 74LS85 wird die Adresse eingestellt. Der 74LS138 übernimmt die Auswahl der D/A-Bausteine. Mit $\overline{IORQ}$ und $\overline{WR}$ werden die Bausteine genau dann selektiert, wenn ein Schreibzugriff vorliegt. Der Datenbus ist direkt an die D/A-Umsetzer geführt, die einen internen Zwischenspeicher besitzen. Eine Rückmeldung, wann die Wandlung beendet ist, ist nicht nötig, da die Umsetzer eine Wandlungszeit von nur ca. 800 ns besitzen und damit viel schneller als der Prozessor sind. Die Wandlung geschieht intern mit Hilfe eines R-2R-Widerstandnetzwerkes. Die Ausgangsspannung kann zwischen 0 V und 5 V liegen. Als Referenzspannungsquelle werden intern erzeugte 2,5 V verwendet. Ein 100-pF-Kondensator am Ausgang verhindert Überschwingen bei den internen Umschaltvorgängen.

Die Ansteuerung mit Software ist denkbar einfach. *Bild 2* zeigt ein Programmbeispiel in 68000-Code. Die beiden D/A-Umsetzer liefern dann eine treppenförmige Ausgangsspannung. Mit diesem Programm kann man gleichzeitig die Schaltung testen. Als Port-Adresse wurden F8 und F9 gewählt, beim 68008, der ja „memory mapped" arbeitet, muß man daher $ffffffff8 und $ffffffff9 schreiben, da dies auf der CPU68K-Karte so kodiert wurde.

In Z80-Assembler sieht das so aus:

```
SCHLEIFE:
        INC A
        OUT (0F8H),A
        OUT (0F9H),A
        JP SCHLEIFE
```

Für den mc-CP/M-Computer ist dies das Testprogramm.

Bild 3 schließlich zeigt noch ein Beispiel in der neuen Sprache Gosi, wie sie für den NDR-Klein-Computer verfügbar ist. Dort wird eine Treppenspannung und eine Sinus-Kurve ausgegeben, allerdings viel langsamer als in Maschinensprache. Der Aufruf erfolgt mit da 1 cr (carriage return).

Da Gosi eine Logo-Verwandte ist, ist die Rekursion eine bevorzugte Technik. In dem Programmstück wird durch da :n+1 die Routine erneut aufgerufen und mit :n+1 die Zählschleife realisiert. Bei normaler Rekursionstechnik würde irgendwann einmal ein Speicherüberlauf stattfinden, das ist bei Gosi und auch bei Logo aber nicht der Fall, da vom Interpreter erkannt wird, daß in diesem Fall kein Retten von Parametern nötig ist (tail recursion).

Der A/D-Umsetzer

Es wird der Anschluß des Bausteins ZN-432-10 von Ferranti beschrieben, der eine Genauigkeit von 10 Bit besitzt und dabei eine Wandlungszeit von ca. 20 µs benötigt, also sehr schnell ist.

Der Wandler arbeitet nach dem Prinzip der sukzessiven Approximation. Die Wandlungszeit wird durch einen Takt bestimmt, den ein Schmitt-Trigger-Oszillator mit dem Baustein 7413 liefert. Der Takt sollte bei ca. 550 Hz liegen. *Bild 4* zeigt die Schaltung. Die Adreßauswahl übernimmt das IC 74LS688, das auch durch zwei 74LS85 mit 74LS04 ersetzt werden könnte. Der Baustein 74LS688 enthält acht Vergleicher. Ein Dekoder (74LS139) übernimmt die restliche Adreßdekodierung. Mit dem Lesesignal und A0=0 wird der höherwertige Teil der Bits und der Status ausgelesen, mit A1=1 wird der LSB-Teil ausgelesen. Mit A0=0 und einem Schreibsignal wird die Wandlung gestartet. Der Status zeigt an, daß eine Wandlung im Gange ist. Dabei ist Bit 7 des Ports FC (A0=0) solange auf 1 wie die Wandlung andauert. Der Eingangsspannungsbereich ist auf 0 bis 5 V eingestellt.

R4 wird dann nicht benötigt. Mit dem Wandler lassen sich aber auch andere Bereiche einstellen. Dazu muß aber der Hinweis auf das Datenblatt der Firma Ferranti genügen.

Mit R2 läßt sich der Spannungsbereich exakt abgleichen. *Bild 5* zeigt ein Beispielprogramm für den 68008. Die Routine GETAD ist die eigentliche Wandelroutine. Dadurch, daß über die niederwertige Adresse der Status mit eingelesen wird kann beim 68008 durch einen einzigen Move-Befehl sowohl der Status als auch der Datenwert eingelesen werden. Dabei liegt im Register D0 automa-

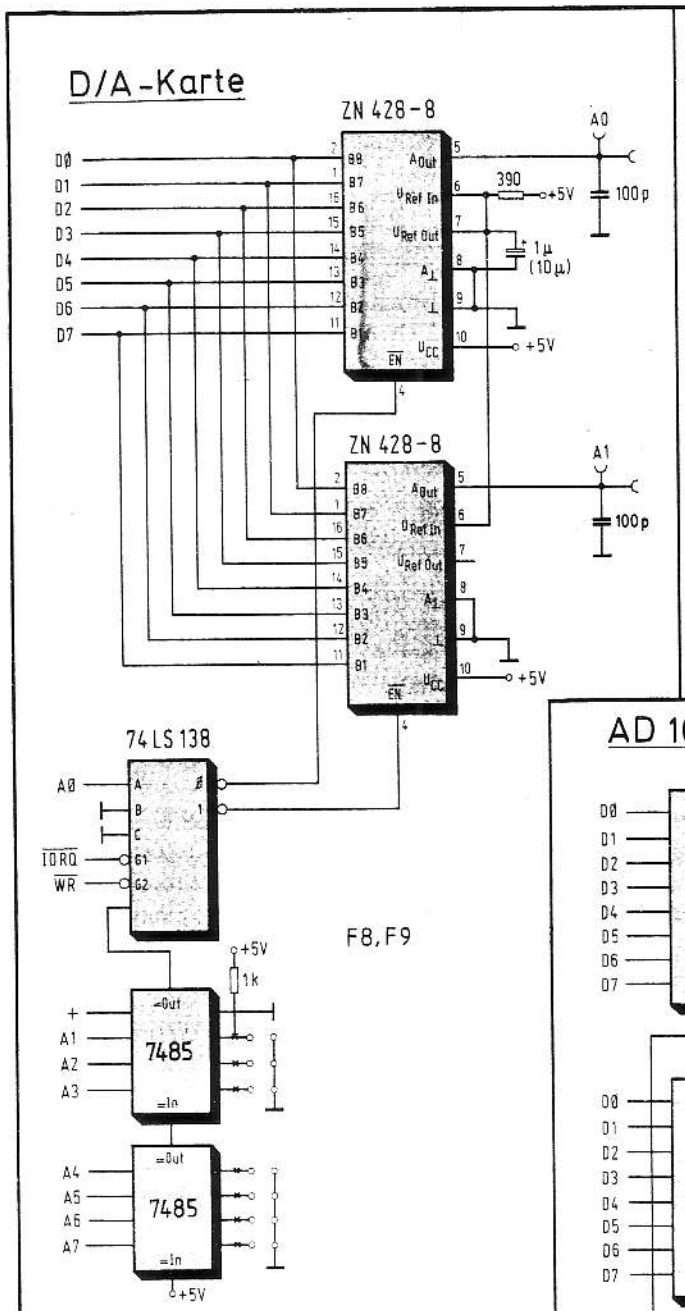

Bild 1. Die Schaltung des D/A-Umsetzers

```
lerne da :n
port 248 :n
port 249 ((:sin :n)+256)/3
da :n+1
ende

Ok gelernt, Platz zum lernen:
2501 Bytes und fuer Namen: 680 Bytes.
```

Bild 3. Ein Programm in Gosi

```
da0 equ $ffffffff8
da1 equ $ffffffff9

da:
 addq.b #1,d0
 move.b d0,da0
 move.b d0,da1
 bra.s da

Textstart=009000 Cursor=009000 Textende=009000 einf ameer CTRL-J=Hilfe
```

Bild 2. Ein 68000-Programm als Testprogramm

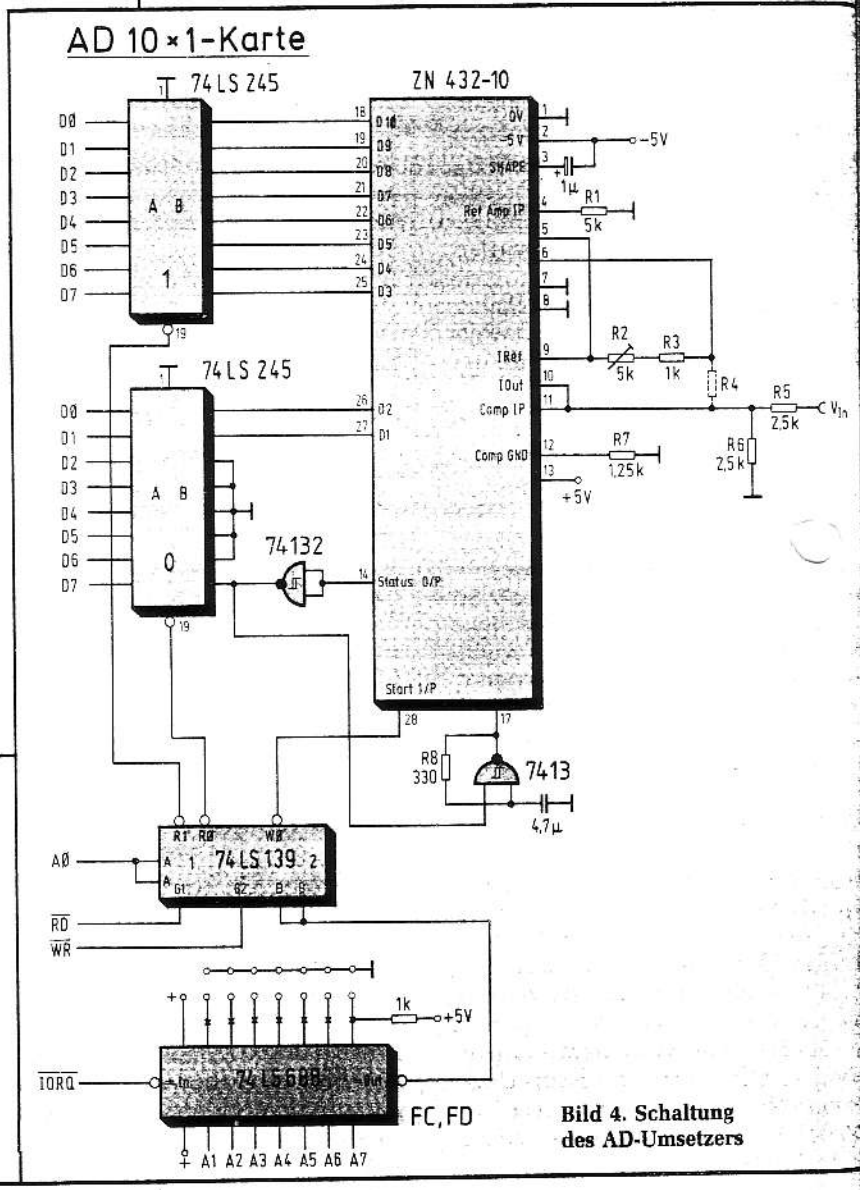

Bild 4. Schaltung des AD-Umsetzers

tisch der komplette 10-Bit-Datenwert, wenn das Status-Bit auf 0 war. Damit ist der Einlesevorgang sehr flink.

Für den Z80 würde ein Programm wie folgt aussehen:

```
GETAD:
        OUT (0FCH),A

GETLP:  IN A,(0FCH)
        OR A
        JP M,GETLP
        LD H,A
        IN A,(0FDH)
        LD L,A
        RET
```

Das 68008-Programm liest die Werte dauernd ein und gibt sie in dezimaler Form auf dem Bildschirm aus.

Der 16-Kanal-A/D-Umsetzer

Seit langer Zeit schon ist ein interessanter Baustein der Firma National Semiconductor auf dem Markt, der 16 Analog-Kanäle mit je 8-Bit-Auflösung in etwa 100 µs wandeln kann, der ADC-0816.

Bild 6 zeigt eine Schaltung mit diesen Baustein. Der Wandler belegt 16 I/O-Adressen, denn jede Adresse entspricht einem Kanal. Der Umsetzer arbeitet mit dem Sukzessiven-Approximations-Verfahren und benötigt für eine Wandlung ca. 100 µs. Für die Bearbeitung der Kanäle besitzt er einen integrierten Analog-Multiplexer.

Mit dem IC 74LS85 wird die Adreßdekodierung der Karte vorgenommen. Die Adressen E0 bis EF sollen verwendet werden. Dazu wird der Vergleichereingang bei Adresse A4 mit 0 belegt, die restlichen bleiben auf +5 V.

Wird ein Schreibzugriff von E0 bis EF vorgenommen, so startet der Wandler, wobei die Adressen von A0 bis A3 als Kanaladresse verwendet werden. Beim

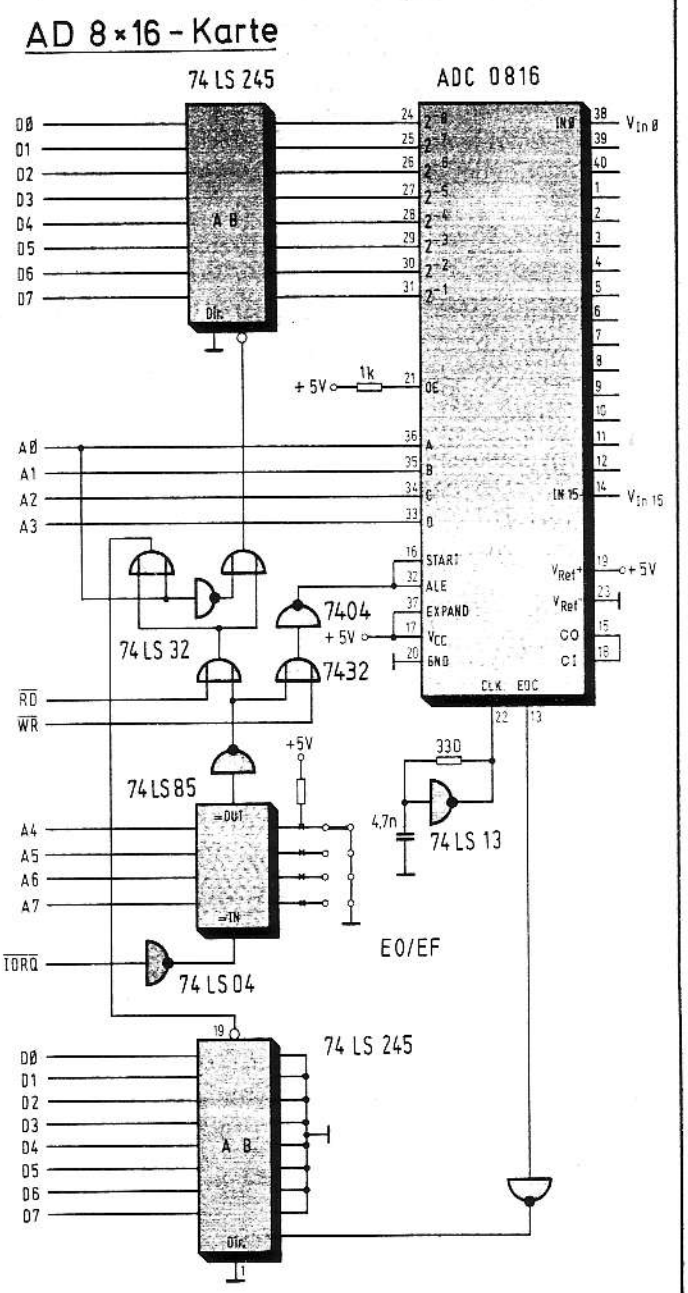

Bild 6. Schaltung des 16-Kanal-A/D-Umsetzers

```
009C00
009C00
009C00                          ********************************
009C00                          * AD 10X1 DEMOPROGRAMM RDK      *
009C00                          ********************************
009C00
009C00                           ORG $9C00
009C00
= FFFFFFFC                      AD EQU $FFFFFFFC
009C00
009C00                          GETAD:      * HOLEN EINES WERTES
009C00   4239 FFFFFFFC           CLR.B AD   * START DES WANDLERS
009C06                          GETLP:      * WARTEN AUF DEN WANDLER
009C06   3039 FFFFFFFC           MOVE AD,D0
009C0C   6BF8                    BMI.S GETLP
009C0E   4E75                    RTS        * ERGEBNIS IN D0
009C10
009C10                          START:      * TESTPROGRAMM
009C10   323C 005A               MOVE #90,D1 * RAHMEN
009C14   343C 0078               MOVE #120,D2
009C18   4EB9 0000091E           JSR $MOVETO
009C1E   0641 00C8               ADD #200,D1
009C22   4EB9 0000098A           JSR $DRAWTO
009C28   0642 0028               ADD #40,D2
009C2C   4EB9 0000098A           JSR $DRAWTO
009C32   0441 00C8               SUB #200,D1
009C36   4EB9 0000098A           JSR $DRAWTO
009C3C   0442 0028               SUB #40,D2
009C40   4EB9 0000098A           JSR $DRAWTO
009C46                          LOOP:
009C46   41F9 00009C86           LEA BUFFER,A0
009C4C   6100 FFB2               BSR GETAD * WERT HOLEN
009C50   4EB9 00000F90           JSR $PRINT4D * DEZIMAL AUSGEBEN
009C56   363C 0005               MOVE #6-1,D3
009C5A                          LP:         * REST FUELLEN
009C5A   10FC 0020               MOVE.B #$20,(A0)+
009C5E   51CB FFFA               DBRA D3,LP   * AUFUELLEN MIT BLANKS
009C62   4210                    CLR.B (A0)   * ENDEZEICHEN
009C64   41F9 00009C86           LEA BUFFER,A0
009C6A   303C 0033               MOVE #$33,D0 * SCHRIFTGROESSE
009C6E   323C 0064               MOVE #100,D1 * X-KOORDINATE
009C72   343C 0080               MOVE #128,D2 * Y-KOORDINATE
009C76   4EB9 00000D5C           JSR $WRITE   * AUSGEBEN
009C7C                          WLP:        * 20 MS WARTEN
009C7C   4EB9 00000852           JSR $SYNC
009C82   67F8                    BEQ.S WLP
009C84   60C0                    BRA.S LOOP   * ALLES WIEDERHOLEN
009C86
009C86                          BUFFER: DS.B 80
009CD6
009CD6                          END

0000     Fehler entdeckt
008A9A   Ende-Symboltabelle
```

Bild 5. Testprogramm für den 68008

Lesen enthält Kanal E0 den Status, dessen Bit 7 solange auf 1 bleibt bis die Wandlung beendet ist. E1 ist der Datenkanal. Mit dem IC 74LS13 wird der Wandlungstakt erzeugt.

Bild 7 zeigt ein Programmbeispiel für den 68008 mit graphischer Ausgabe der Kanäle. *Bild 8* zeigt dazu mögliche Kanalwerte, wie sie auf dem Bildschirm angezeigt werden.

In der Routine GETAD des Programms wird der Wandler angestoßen und im Register D0 der Datenwert nach erfolgter Wandlung abgeliefert. Es wird zunächst nach Start des Wandlers solange gewartet, bis Bit 7 des Status auf 1 liegt, der Wandler also bestätigt, daß die Wandlung begonnen hat. Dann wird gewartet bis Bit 7 des Status wieder auf 0 zurückgeht, also die Wandlung beendet ist. Danach kann der Wert eingelesen werden.

Im Hauptprogramm befindet sich der Schleifenzähler für die einzelnen Kanäle in Register D3, in Register D1 wird die X-Koordinate gespeichert, denn für jeden Kanal soll im Abstand von 20 Punkten ein Balken für einen analogen Wert dargestellt werden.

Es wird dann im Programm zuerst ein Balken gelöscht, dann der A/D-Wandler angestoßen und der neue Balken geschrieben. Wenn alle 16 Kanäle ausgegeben sind, wird mit @NEWPAGE eine neue Bildseite ausgewählt. Bei dem verwendeten Graphik-Interface zum NDR-Klein-Computer sind insgesamt vier Bildseiten vorhanden. Es wird am Ende der Schleife auch noch eine Vertauschung der Lese- und Schreibseite vorgenommen, so daß nach dem ersten Aufruf in Seite 0 geschrieben wird, und auf Seite 1 gelesen, dann auf Seite 1 geschrieben und auf Seite 0 gelesen. Dadurch ergibt sich ein sauberes stehendes Bild, bei dem die Einschreibvorgänge nicht sichtbar sind.

Spruch des Monats

„Es ist ein dem Menschen natürliches Leiden, daß er glaubt, die Wahrheit unmittelbar besitzen zu können, und daraus stammt seine Neigung, immer alles zu leugnen, was ihm unbegreifbar ist. Während er tatsächlich von Natur her nur die Lüge kennt und nur das für wahr halten darf, dessen Gegenteil ihm falsch erscheint. Deshalb muß man stets, sobald ein Satz unbegreifbar ist, das Urteil aussetzen, und man darf ihn deswegen nicht leugnen, sondern man muß das Gegenteil prüfen: Findet man, daß die gegenteilige Aussage offenbar falsch ist, so kann man kühn den ersten Satz behaupten, so unbegreifbar er auch immer sein mag."

Blaise Pascal

```
009C00
009C00                    ****************************
009C00                    * PROGRAMM FUER DEN ADC0816 *
009C00                    * RDK 840424                *
009C00                    ****************************
009C00
009C00
009C00
= FFFFFFE0                AD EQU $FFFFFFE0
009C00
009C00                    GETAD:            * D0=KANAL, DANN ERGEBNIS
009C00  41F9 FFFFFFE0      LEA AD,A0
009C06  4230 0000          CLR.B 0(A0,D0.W)   * START WANDLUNG
009C0A                    GETLP:
009C0A  1010               MOVE.B (A0),D0
009C0C  6AFC               BPL.S GETLP       * WARTEN BIS GESTARTET IST
009C0E                    GETLP1:
009C0E  1010               MOVE.B (A0),D0
009C10  6BFC               BMI.S GETLP1      * WARTEN BIS GEWANDELT
009C12  102B 0001          MOVE.B 1(A0),D0   * ERGEBNIS
009C16  4E75               RTS
009C18
009C1B                    START:            * HAUPTPROGRAMM
009C1B  363C 0000          MOVE #0,D3        * ALLE KANAELE 0..15
009C1C  323C 0064          MOVE #100,D1      * X-KOORDINATE
009C20                    LOOP:
009C20  343C 00FF          MOVE #255,D2      * LOESCHEN ALTEN WERT
009C24  4EB9 00000786      JSR §ERAPEN
009C2A  4EB9 0000091E      JSR §MOVETO
009C30  4242               CLR D2
009C32  4EB9 0000098A      JSR §DRAWTO
009C38  4EB9 00000794      JSR §SETPEN       * DANN NEUE LINIE AUSGEBEN
009C3E  3003               MOVE D3,D0        * KANAL-NR
009C40  6100 FFBE          BSR GETAD         * WERT NACH D0.B
009C44  0240 00FF          AND #$FF,D0       * ERWEITERN AUF WORT
009C48  3400               MOVE D0,D2        * Y-KOORDNATE
009C4A  4EB9 0000091E      JSR §MOVETO
009C50  4242               CLR D2
009C52  4EB9 000009BA      JSR §DRAWTO
009C58  0641 0014          ADD #20,D1        * X-KOORDINATE
009C5C  5243               ADDQ #1,D3
009C5E  0C43 0010          CMP #16,D3
009C62  66BC               BNE.S LOOP
009C64                    WARTE:
009C64  4EB9 00000852      JSR §SYNC
009C6A  67FB               BEQ.S WARTE       * NUR ALLE 20MS
009C6C  3039 00009C96      MOVE WRT,D0       * IMMER ABWECHSELND
009C72  33F9 00009C98      MOVE RDR,WRT      * SCHREIBEN,DANN KEIN
009C78  00009C96
009C7C  33C0 00009C98      MOVE D0,RDR       * FLIMMERN
009C82  3039 00009C96      MOVE WRT,D0
009C88  3239 00009C98      MOVE RDR,D1
009C8E  4EB9 00000828      JSR §NEWPAGE      * SEITENFLIP
009C94  6082               BRA.S START       * WIEDER VON VORNE
009C96
009C96  0001              WRT: DC.W 1        * START SEITE 1
009C98  0000              RDR: DC.W 0        * DANN TAUSCHEN
009C9A
009C9A                     END

0000    Fehler entdeckt
00BAAC  Ende-Symboltabelle
```

Bild 7. Programmbeispiel für den 68008

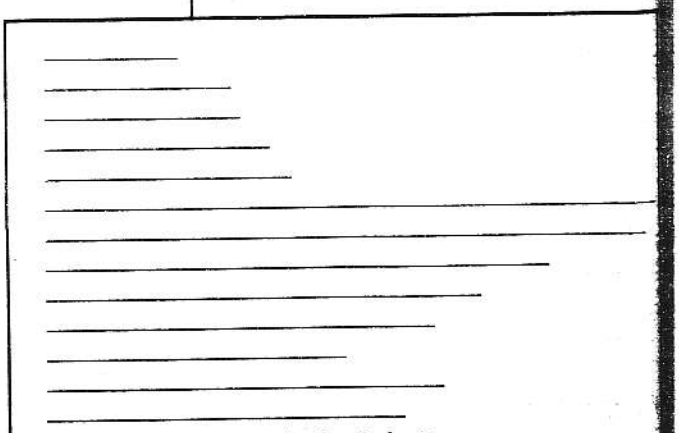

Bild 8. Bildschirmausgabe durch das Programm