Einplatinencomputer mit dem *U880* (Blatt 1)

Dipl.-Ing. Joachim Czepa

1. Einleitung

Der Einplatinencomputer *C-80* ist ein eigenständiger Mikrocomputer mit der CPU *U880D*, 2 KByte EPROM, 1 KByte RAM und 40 frei verfügbaren Ein-/Ausgabeleitungen. Die gesamte Schaltung benötigt lediglich 12 Schaltkreise und läßt sich auf einer Universal-Leiterplatte 95 mm × 170 mm unterbringen. Alle Signalleitungen der CPU werden über einen 58poligen Steckverbinder herausgeführt. Die Steckerbelegung entspricht der des *K-1520*-Systembus. Die Ein-/Ausgabeleitungen führen auf einen frontseitigen Steckverbinder. Der Beitrag beschreibt weiterhin den Anschluß einer Hexadezimaltastatur, einer 8stelligen 7-Segment-Anzeige und eines handelsüblichen Kassettenrecorders. Die verbleibenden 20 Ein-/Ausgabeleitungen können zum Anschluß weiterer Peripherie benutzt werden.
Das entwickelte Monitorprogramm belegt 1 KByte im Speicherbereich. Damit sind alle Voraussetzungen für das Erstellen und Testen eigener Programme auf Maschinenkode-Ebene gegeben. Die Programme können mit Hilfe eines Kassettenrecorders dauerhaft gespeichert werden.
Großer Wert wurde auf minimale Bauelementekosten und weitgehende Kompatibilität zu bestehenden Systemen gelegt. Es kommen ausschließlich Bauelemente aus der DDR-Produktion zur Anwendung. Das verwendete Aufzeichnungsverfahren für die Daten- und Programmabspeicherung auf einem Kassettenrecorder ist mit dem Verfahren des *Polycomputers PC-880* kompatibel. Die entwickelten Hard- und Softwarelösungen lassen sich auf andere *U880*-Systeme übertragen. Mit dem Anschluß weiterer Steckeinheiten kann ein Mikrorechnersystem mit erweitertem Speicherumfang und komfortableren Ein-/Ausgabemöglichkeiten aufgebaut werden.
Der beschriebene Einplatinencomputer *C-80* eignet sich zur Einarbeitung in die Hard- und Softwareproblematik des *U880*. Es lassen sich kleine leistungsarme Steuerungen auf Mikrorechnerbasis entwickeln. Der *C-80* eignet sich aber auch als CPU-Platine für ein umfangreicheres System.
Grundkenntnisse in der Funktionsweise des *U880* und der zugehörigen Peripheriebausteine sowie des praktischen Aufbaus digitaler Schaltungen werden vorausgesetzt. Alle Adressen werden hexadezimal angegeben. Allen HEX-Zahlen, die mit A...F beginnen, wird eine führende »O« vorangestellt. Der Buchstabe »H« am Schluß der Zahl weist auf die hexadezimale Schreibweise hin.

2. Schaltungsbeschreibung des *C-80*

Bild 1 zeigt den Stromlaufplan des Einplatinencomputers. Die Leiterplatte enthält die CPU *U880D*, 2 KByte EPROM *(U555C)*, 1 KByte RAM *(U214D)*, zwei PIO-Bausteine *U855D*, den Taktgenerator und einen Adreßdekoder mit CMOS-Bausteinen. Es werden keine Bustreiber eingesetzt. Die CPU-Ausgänge treiben direkt den Systembus. Alle CPU-Ausgänge sind TTL-kompatibel. Die *U880-CPU* garantiert ihre statischen und dynamischen Kennwerte bei einer Belastung der Ausgänge mit einer Standard-TTL-Last und einer Lastkapazität von 50 pF. Bei einer Vergrößerung der Lastkapazität auf 100 pF verlängern sich die Verzögerungszeiten um etwa 30 ns.
In der Schaltung nach Bild 1 wird die CPU ausschließlich durch MOS- bzw. CMOS-Eingänge belastet. Die statische Belastung kann vernachlässigt werden. Der Einfluß der Lastkapazität auf die Funktionsfähigkeit der Schaltung spielt erst im Bereich der maximalen Taktfrequenzen eine Rolle. Bei Minimalsystemen ist es in der Regel vorzuziehen, auf Bustreiber zu verzichten und eine kleinere Taktfrequenz in Kauf zu nehmen. Bustreiber bringen neben dem erhöhten Bauelementeaufwand insbesondere eine hohe Stromaufnahme mit sich.
Um die Funktionsfähigkeit der vorliegenden Schaltung zu testen, wurde der Einplatinencomputer versuchsweise an einem etwa 50 cm langen Systembus bei einer Belastung jeder Systembusleitung mit 6 Schaltkreisen *8212/8216* betrieben. Es wurde die höchste Taktfrequenz gemessen, bei der die Schaltung noch einwandfrei arbeitete. Dieser Wert ist in starkem Maße von den Daten der verwendeten Bauelemente abhängig. Mit mehreren untersuchten 2,5-MHz-CPUs und nicht ausgesuchten Bauelementen wurden in allen Fällen Taktfrequenzen weit über 3 MHz erreicht!
Diese Ergebnisse überzeugten den Autor davon, daß es zumindestens in kleinen Mikrorechnersystemen empfehlenswert ist, CMOS statt TTL einzusetzen. So können Bustreiber zum größten Teil eingespart werden. Man gelangt damit zu Systemen mit geringem Bauelementeaufwand und minimaler Verlustleistung bei vergleichbarer Leistungsfähigkeit mit industriellen Systemen.
Zwei Nachteile sollen allerdings nicht unerwähnt bleiben. Zum einen ist die *U880-CPU* nicht CMOS-kompatibel. Die CPU-Ausgänge garantieren nicht den erforderlichen »High«-Pegel von mindestens 3,5 V für die CMOS-Eingänge. Deshalb wurden die Widerstände *R*11 bis *R*19 zum Anheben des H-Pegels eingesetzt. Die Meßergebnisse kamen aber auch ohne diese Widerstände zustande.
Als zweiter Nachteil ist zu werten, daß Fremdspannungen auf den Systembusleitungen zur Beschädigung aller angeschlossenen Schaltkreise führen können. Dem kann nur durch sorgfältiges Arbeiten begegnet werden.

2.1. Taktgenerator

Der Taktgenerator wird mit einer TTL-IS aufgebaut, um eine Flankensteilheit von weniger als 30 ns für die *U880-CPU* zu garantieren. Zwei Negatoren bilden die eigentliche Oszillatorschaltung. Der Transistor VT1 ist als Emitterfolger geschaltet und bewirkt einen hohen Eingangswiderstand der Oszillatorschaltung. Die Gegenkopplung mit *R*1 stellt einen stabilen Arbeitspunkt in der Nähe des Umschaltpunktes des TTL-Negators ein. Zwischen den Punkten B und C kann ein Quarz angeschlossen werden, der im Frequenzbereich von 500 kHz bis 2,5 MHz schwingt. Die nachfolgende Triggerschaltung liefert ein etwa symmetrisches Taktsignal mit entsprechender Flankensteilheit. *R*5 gewährleistet den notwendigen »High«-Pegel des Taktsignals.
Die hochohmige Oszillatorschaltung erlaubt auch den Einsatz preiswerter 455-kHz-Piezofilter zur Schwingungserzeugung. Das Filter wird zwischen den Punkten A und B angeschlossen. Damit kann Beschaffungsproblemen für einen teuren Quarz anfangs aus dem Wege gegangen werden. Für viele Anwendungen ist diese geringe Taktfrequenz völlig ausreichend. Beispielsweise bereitete die Software für die Magnetbandspeicherung, bei der eine serielle Übertragungsgeschwindigkeit von immerhin 1 200 bit/s rein softwaremäßig realisiert wird, selbst mit dieser geringen Taktfrequenz keine Schwierigkeiten.
Bei geschlossenem Schalter S liegt die Taktfrequenz als TTL-Signal am Systembus an. Bei geöffnetem Schalter kann über den Systembus ein externer Takt eingespeist werden. Damit besteht die Möglichkeit, Einzeltaktbetrieb durchzuführen.

2.2. Adreßdekoder

Für den Adreßdekoder werden nur einfache CMOS-Gatter (IS7 bis IS10) verwendet. Der Einsatz eines 1-aus-10-Dekoders *(U4028D)* bringt keine Einsparung, dafür aber eine größere Verzögerungszeit für die $\overline{CE}$-Signale. Die $\overline{CE}$-Signale werden durch Verknüpfung der höchstwertigsten Adreßsignale A10 bis A15 und des $\overline{MREQ}$-Signals gebildet. Die EPROM werden nur bei Speicherlesezyklen ($\overline{MREQ}$ = »Low«, $\overline{RD}$ = »Low«) aktiviert. Bild 2 zeigt die sich ergebende Adreßbelegung der Speicher-IS. Alle Adressen oberhalb 1 000 H können für die externe Dekodierung weiterer Speicherschaltkreise benutzt werden. Der RAM wird im Adreßbereich von O8OOH...OBFFH ein zweites Mal

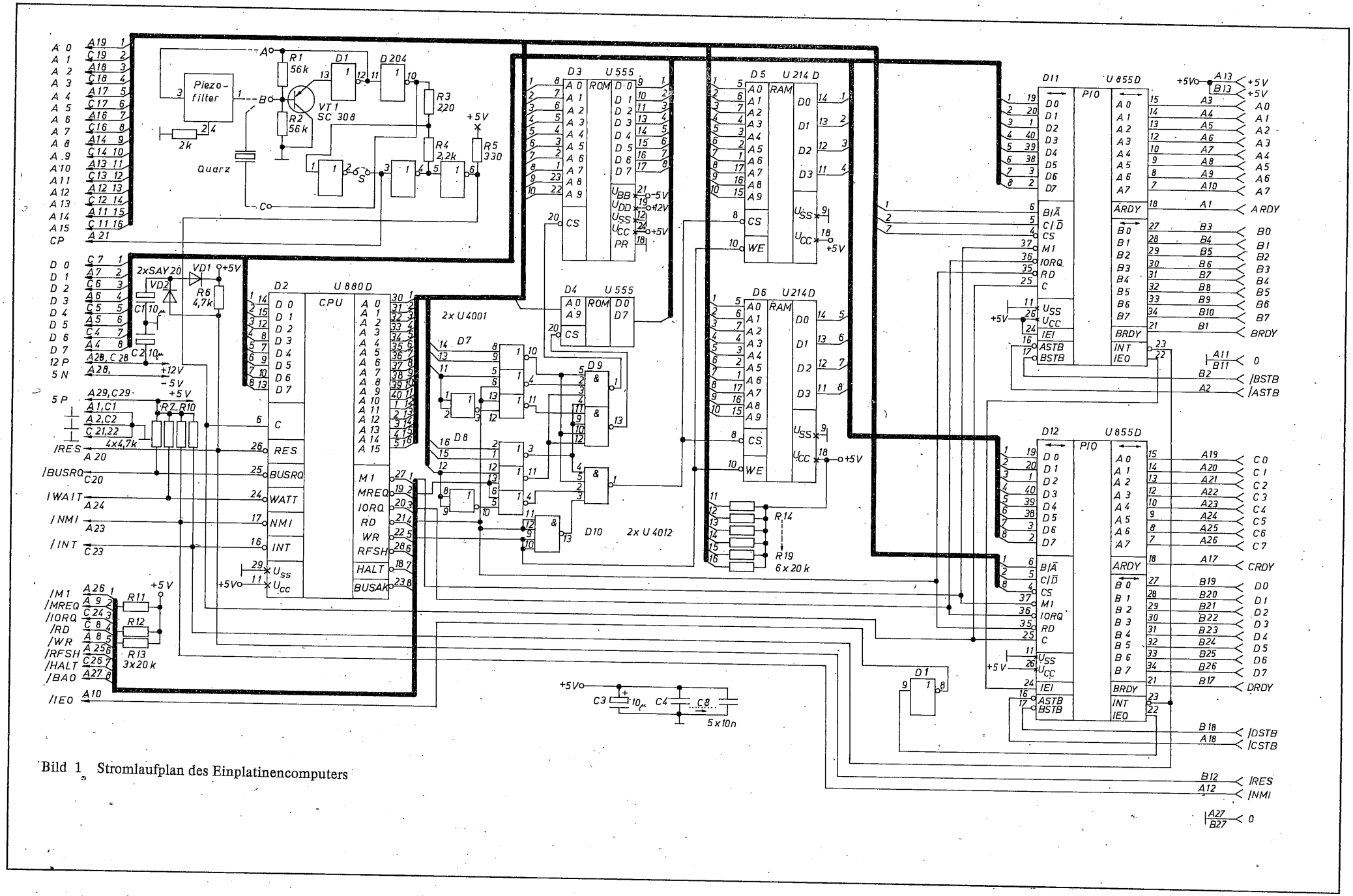


Bild 1 Stromlaufplan des Einplatinencomputers

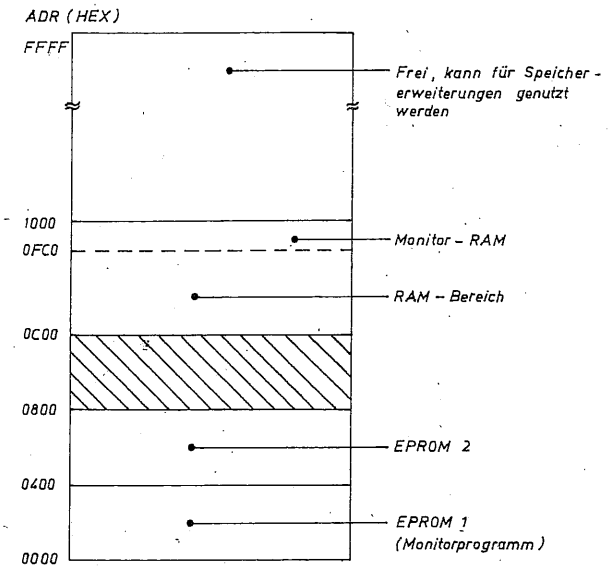
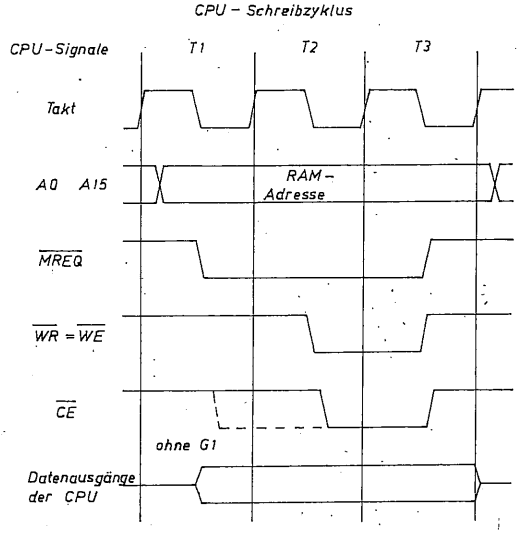


Bild 2 Speicherplan des Einplatinencomputers



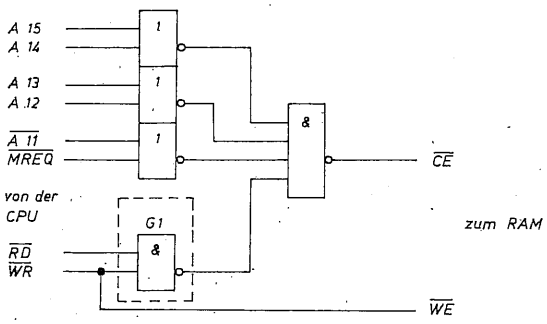


Bild 3 Anschluß des statischen RAM *U214* an die *U880-CPU*.
Ohne Gatter 1 wird das $\overline{CE}$-Signal bei Schreibzyklen zu früh aktiv. Die Datenausgänge der CPU und des RAM sind dabei zeitweise gegeneinandergeschaltet!

ausdekodiert. Diese Adressen sollten nicht angesprochen werden. Man erhält damit, abgesehen vom nicht nutzbaren Bereich, die gleiche Adreßbelegung wie die ZRE des *K-1520*-Systems. Das Monitorprogramm belegt 1 KByte im ersten EPROM (OOOOH...O3FFH) und 64 Byte im RAM (OFCOH...OFFFH). Das zweite EPROM kann für Anwenderprogramme oder Monitorerweiterungen benutzt werden.
Für den RAM-Bereich werden zwei Schaltkreise *U214D* (1 KBit × 4) eingesetzt. Diese Schaltkreise haben bidirektionale Datenleitungen, die direkt an den Datenbus der CPU geschaltet werden. Leider steht kein separater $\overline{OE}$-Eingang (»Output Enable«) zur Verfügung. Die Datenausgänge des RAM werden bei $\overline{CS}$ = »Low« und $\overline{WE}$ = »High« aktiv. Wird das $\overline{CS}$-Signal allein aus dem $\overline{MREQ}$-Signal und einer gültigen Adresse dekodiert, kommt es bei Speicherschreibzyklen durch das verzögerte $\overline{WR}$-Signal zu Konflikten auf dem Datenbus. Das $\overline{CS}$-Signal darf erst bei gültigem $\overline{WR}$- oder $\overline{RD}$-Signal der CPU aktiv werden (siehe Bild 3).

2.3. PIO-Anschluß

Für die Adressierung der Ein-/Ausgabebausteine wird kein gesonderter Dekoder verwendet. Die $\overline{CE}$-Eingänge der beiden PIO sind direkt mit den Adreßleitungen A6 bzw. A7 der CPU verbunden. Bei gleichartiger Verwendung von A2 bis A5 können extern 4 weitere E/A-Bausteine (PIO, SIO, CTC) angeschlossen werden. Durch den Einsatz eines Dekoders kann die Anzahl der zusätzlich anschließbaren Schaltkreise auf 16 erhöht werden. Die Adreßleitung AO wird zur Portauswahl (B/A) benutzt und A1 zur Unterscheidung zwischen Daten- und Steuerport (C/D). Aus Tabelle 1 sind alle Ein-/Ausgabeadressen ersichtlich.
Alle 40 Ein-/Ausgabeleitungen der PIO führen zu einer 58poligen Buchsenleiste auf der Frontseite der Leiterplatte. Zusätzlich werden die Betriebsspannungen, die $\overline{NMI}$-Leitung und die $\overline{RESET}$-Leitung herausgeführt.
Beide PIO-Bausteine können einen Interrupt auslösen. PIO 1 (D11) hat die höchste Priorität und wird vom Monitorprogramm für alle Ein-/Ausgaben verwendet. Der IEO-Ausgang von PIO 2 wird negiert auf den Systembus geführt. Bild 4 zeigt die Steckerbelegung der frontseitigen Buchsenleiste.

Tabelle 1 Belegung der Ein-/Ausgabe-Adressen

E/A-Adresse (HEX)	Bedeutung	
7C	PORT A, Datenbyte	
7D	PORT B, Datenbyte	PIO 2
7E	PORT A, Steuerbyte	(Anwender-PIO)
7F	PORT B, Steuerbyte	
BC	PORT A, Datenbyte	
BD	PORT B, Datenbyte	PIO 1
BE	PORT A, Steuerbyte	(System-PIO)
BF	PORT B, Steuerbyte	
CO...FE	frei	

2.4. Systembus

Alle CPU-Anschlüsse führen auf einen 58poligen Steckverbinder. Die Signalbelegung entspricht der des *K-1520*-Systembus; die Signale $\overline{RDY}$, $\overline{IODI}$, $\overline{MEMDI}$, $\overline{IEI}$ und $\overline{BAI}$ sind nicht beschaltet. Die maximale Belastbarkeit der Busleitungen beträgt allerdings nur eine Standard-TTL-Last. Die Taktleitung ist höher belastbar und kann als Eingang oder Ausgang benutzt werden. Alle CPU-Eingänge liegen über Widerstände an +5 V. Die Interruptmöglichkeiten der *U880-CPU* sind ohne Einschränkungen nutzbar. Das Ende der Interruptkette liegt am Systembus an ($\overline{IEO}$).

PIO – Buchsenleiste

B		A
+5 V	29	+5 V
+12 V	28	−5 V
⊥	27	⊥
D7	26	C7
D6	25	C6
D5	24	C5
D4	23	C4
D3	22	C3
D2	21	C2
D1	20	C1
D0	19	C0
$\overline{DSTB}$	18	$\overline{CSTB}$
DRDY	17	CRDY
	16	
	15	
	14	
+5 V	13	+5 V
$\overline{RES}$	12	$\overline{NMI}$
⊥	11	⊥
B7	10	A7
B6	9	A6
B5	8	A5
B4	7	A4
B3	6	A3
B2	5	A2
B1	4	A1
B0	3	A0
$\overline{BSTB}$	2	$\overline{ASTB}$
BRDY	1	ARDY

A – Reihe an der Leiterplatte anliegend

Bild 4 Steckerbelegung

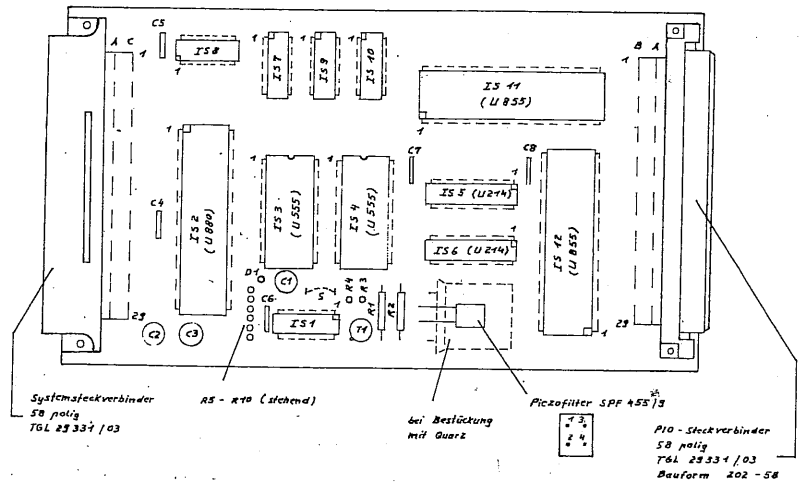


Bild 5 Anordnung der Bauelemente auf der Musterplatte

Der DMA-Betrieb mit der *U880-CPU* ist ebenfalls möglich. Dieser Anwendungsfall wird bei Minimalsystemen in der Praxis nur für Testzwecke eine Rolle spielen. Bei $\overline{BUSRQ}$ = »Low« schaltet die *U880-CPU* alle Signalausgänge in den hochohmigen Zustand, und eine andere Einheit kann die Bussteuerung übernehmen. Dabei sind die Speicher- und E/A-Schaltkreise frei zugänglich.

Die Aufladung von *C1* bewirkt beim Zuschalten der +5 V das Rücksetzen der CPU. VD2 trennt die Rücksetzschaltung nach Aufladung von *C1* von der $\overline{RESET}$-Leitung und sollte eine kleine Flußspannung aufweisen. VD1 entlädt den Kondensator beim Abschalten der Betriebsspannung. Die PIO-Schaltkreise werden hardwaremäßig nicht rückgesetzt.

Der Systembus bietet vorrangig die Möglichkeit, weitere Steckeinheiten anzuschließen. Diese sollten die gleiche Signalbelegung am Steckverbinder aufweisen. In diesem Fall kann vorteilhaft eine gleichmäßige Rückverdrahtung für alle Steckplätze eingesetzt werden. Die drei Betriebsspannungen (+5 V, −5 V, +12 V) werden über den Systembus zugeführt. Die Spannungen sollten nur gemeinsam zu- bzw. abgeschaltet werden. −5 V und +12 V werden nur für die EPROM benötigt und lassen sich unter Umständen aus einem kleinen Transverter erzeugen. Tabelle 2 gibt die berechneten und gemessenen Werte für die Stromaufnahme an.

Tabelle 2 Stromaufnahme des Einplatinencomputers

	Stromaufnahme in mA		
	+5 V	+12 V	−5 V
berechnete Maximalwerte	550	130	90
gemessen	320	60	40
gemessen (ohne IS 4 und 12)	230	30	20

2.5. Konstruktive Gestaltung

Der Einplatinencomputer *C-80* wurde auf einer Universalleiterplatte mit den Maßen 95 mm × 170 mm aufgebaut. Die Konstruktion einer Leiterplatte hätte nur bei zweiseitigem Leiterbild mit 0,3-mm-Leiterzügen Sinn. Das läßt sich aber mit amateurmäßigen Mitteln nicht mehr ausführen.

Der Musteraufbau wurde nach dem Auflöten der Fassungen und passiven Bauelemente mit einfachem 0,3-mm-CuL-Draht verdrahtet. Die Verdrahtung wurde ausschließlich von der Lötseite ausgeführt. Die Drähte werden direkt von Anschluß zu Anschluß gezogen. Dabei sind die Drähte nicht zu straff zu führen, um eine Beschädigung der Lackschicht zu vermeiden. Eine kurzzeitige Berührung mit dem heißen Lötkolben bewirkt keine Beeinträchtigung der Lackisolierung. Der Zeitaufwand für die Verdrahtung der Musterplatte lag unter 25 Stunden. Bild 5 zeigt die Anordnung der Bauelemente auf der Musterplatte.

2.6. Inbetriebnahme

Für die Inbetriebnahme sollten zumindestens ein Vielfachmesser und ein Oszillograph zur Verfügung stehen. Folgende Reihenfolge wird empfohlen:

- Messen der Übergangswiderstände zwischen Masse und den einzelnen Betriebsspannungen an der verdrahteten, aber noch unbestückten Leiterplatte.
- Anlegen der Betriebsspannungen an die unbestückte Leiterplatte und Spannungsmessung an den Schaltkreisfassungen. Damit können fehlerhafte Verdrahtungen der Stromversorgung ausgeschlossen werden.
- Bestücken des Taktgenerators und Nachweis der Taktfrequenz.
- Bestückung mit der *U880-CPU*. Die Datenleitungen werden über 10-kΩ-Widerstände an Masse gelegt. Die CPU führt in diesem Fall nach RESET nur NOP-Befehle aus, d.h. aufeinanderfolgende Befehlslesezyklen auf allen Adressen. Die Adreßausgänge verhalten sich wie ein 16-Bit-Binärzähler. AO liefert die höchste Frequenz, die ⅛ der Taktfrequenz beträgt. Fehlerhafte Verdrahtungen der Adreßleitungen können mit Hilfe des Oszillographen verfolgt werden. Nach dem Bestücken des Adreßdekoders müssen die $\overline{CE}$-Signale für die Speicher-IS erzeugt werden.
- Datenbus über die 10-kΩ-Widerstände an +5 V legen. Die CPU erkennt jetzt den Befehl RST 38H und führt einen Befehlslesezyklus und nachfolgend zwei Speicherschreibzyklen aus. Dieses Verhalten läßt sich ebenfalls mit einem einfachen Oszillographen kontrollieren.

Die weitere Funktionsprüfung nach der Bestückung der restlichen IS wird mit dem in einem EPROM gespeicherten Monitorprogramm vorgenommen.

3. Tastatur- und Anzeigeeinheit

Bild 6 zeigt den Stromlaufplan von Tastatur und Anzeige. Die Einheit wird an PIO 1 (D11) über die frontseitige Buchsenleiste angeschlossen. Es wurde ein entsprechend zugeschnittener 26poliger Steckverbinder benutzt. PIO 2 läßt sich so noch unabhängig mit einem ebenfalls zugeschnittenen Steckverbinder beschalten. Für den konstruktiven Aufbau von Tastatur und Anzeige eignet

Einplatinencomputer mit dem *U880* (Blatt 3)

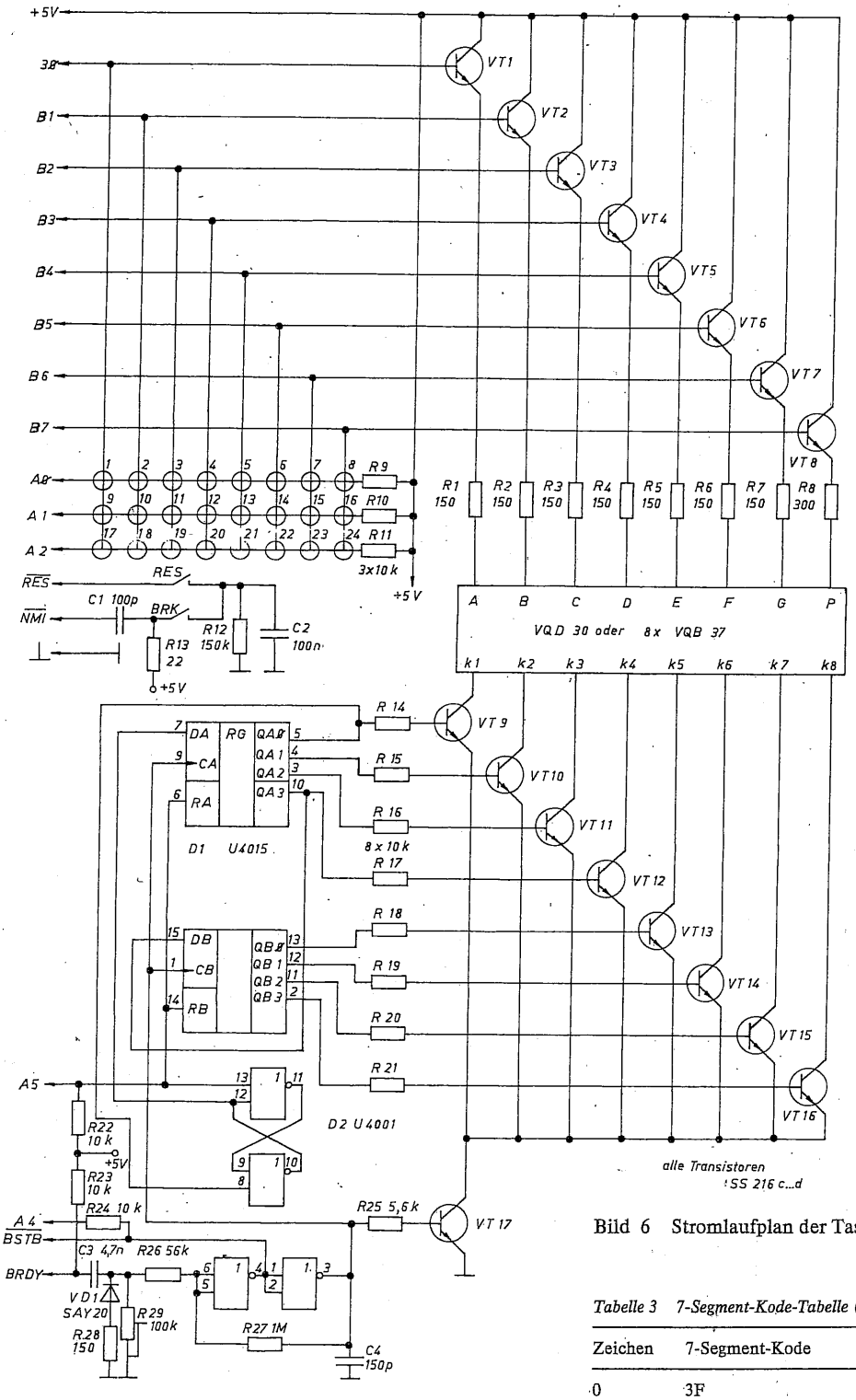


Bild 6 Stromlaufplan der Tastatur- und Anzeigeeinheit

sich ein ausgedienter Taschenrechner. Der Taschenrechnerschaltkreis wird entfernt und durch die dargestellte Schaltung ersetzt. Die Schaltung fand auf einer kleinen Universal-Leiterplatte außerhalb des Taschenrechnergehäuses Platz. Sie enthält gleichzeitig die Magnetbandansteuerung.
Port B der PIO wird in der Betriebsart Byteausgabe und Port A für Bitbetrieb programmiert. Dabei stellen A5 und A6 Ausgabeleitungen dar. Die anderen Leitungen von Port A werden für Eingaben benutzt. Über A6 und A7 wird das Magnetbandgerät angesteuert. A3 bleibt frei.

Tabelle 3 7-Segment-Kode-Tabelle (HEX)

Zeichen	7-Segment-Kode	Zeichen	7-Segment-Kode
0	3F	C	39
1	06	D	5E
2	5B	E	79
3	4F	F	71
4	66		
5	6D	S	6D
6	7D	P	73
7	07	L	38
8	7F	I	06
9	6F	X	76
A	77	Y	72
B	7C		

Tabelle 4 Kodierung der Tasten

Tasten-Nr. (siehe Bild 6)	Bezeichnung	Tastenkode (HEX)	mit FCN-Taste (HEX)
1	REG	14	34
2	GO	13	33
3	D	0D	2D
4	A	0A	2A
5	7	07	27
6	4	04	24
7	1	01	21
8	FCN	12	32
9	+	10	30
10	–	11	31
11	E	0E	2E
12	B	0B	2B
13	8	08	28
14	5	05	25
15	2	02	22
16	0	00	20
17	/	/	/
18	/	/	/
19	F	0F	2F
20	C	0C	2C
21	9	09	29
22	6	06	26
23	3	03	23
24	MEM	15	35

3.1. Anzeigesteuerung

Das Multiplexen der 8stelligen Anzeige wird im Ansteuerprogramm durch zyklische Ausgabe der einzelnen Stellen realisiert. Port B treibt über die Transistoren VT1 bis VT8 die Segmente und den Dezimalpunkt der Anzeige. Die Widerstände *R*1...*R*8 begrenzen den Segmentstrom. Ihr Wert richtet sich nach dem Strombedarf der verwendeten Anzeige und der Anzahl der Stellen.

Bei der Ausgabe einer Segmentkombination wird das Signal BRDY aktiv (BRDY = »High«). *C*3 lädt sich über *R*29 auf und bewirkt für die Zeit der Aufladung ein Umschalten der CMOS-Triggerschaltung. Das Schieberegister (D1) schaltet um eine Stelle weiter. Gleichzeitig wird über VT17 die gesamte Anzeige freigegeben. Ist die Aufladung beendet, kippt der Trigger zurück und tastet die Anzeige dunkel.

Die positive Flanke von $\overline{\text{BSTB}}$ bewirkt im PIO das Rücksetzen des BRDY-Signals und damit die schnelle Entladung von *C*3. Wurde zuvor Interrupterlaubnis für Port B programmiert, löst die positive Flanke von $\overline{\text{BSTB}}$ zugleich Interrupt aus. Die zugehörige Interrupt-Behandlungsroutine liefert die Ausgabe der nächsten Stelle. Über die Leitung A4 kann das Signal $\overline{\text{BSTB}}$ softwaremäßig abgefragt werden. Dadurch kann die Anzeigesteuerung auch ohne Interrupt im Abfrage-Betrieb (Polling) geschehen. Diese Form der Ansteuerung wird im Monitorprogramm verwendet. *R*24 verhindert ein ständiges Helltasten der Anzeige bei versehentlicher Programmierung von A4 als Ausgabeleitung.

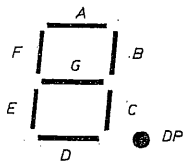
Über die Ausgabeleitung A5 werden das Schieberegister und das mit CMOS-Gattern gebildete Flip-Flop rückgesetzt. Die erste Ausgabe über Port B nach A5 = »Low« führt zur Übernahme des »High«-Pegels am seriellen Dateneingang DA. Die erste Stelle leuchtet auf. Gleichzeitig wird das CMOS-Flip-Flop gesetzt. Bei allen folgenden Ausgaben übernimmt das Schieberegister eine »0«, und die »1« wird durchgeschoben. Damit werden der Reihe nach alle Stellen aktiv. Nach Ausgabe der letzten Stelle wird A5 = »High«, und die Schaltung geht wieder in den Ausgangszustand zurück.

Bei diesem Schaltungsprinzip ist das Einbrennen von Segmenten infolge Fehlbedienung ausgeschlossen. Ein Nachleuchten kann nicht auftreten. Die Zahl der Anzeigestellen läßt sich durch Verlängern des Schieberegisters erhöhen, ohne daß mehr Leitungen der PIO benötigt werden. Mit *R*29 wird die Multiplexfrequenz beeinflußt. Im Muster betrug die Helltastzeit einer Anzeigestelle etwa 1 ms. Ein Ausgabezyklus dauert somit 8 ms. *C*4 unterdrückt kurze Störimpulse.

3.2. Tastaturabfrage

Bei A5 = »High« ist D1 rückgesetzt und die Anzeige dunkelgetastet. Port B kann jetzt zur Ansteuerung der Tastaturmatrix verwendet werden. Dazu werden die Ausgabeleitungen B0 bis B7 auf »Low« gesetzt. Ist keine Taste gedrückt, wird auf den Leitungen A0 bis A2 »High«-Pegel erkannt. Diese Abfrage wird vorteilhaft am Ende eines Ausgabezyklus durchgeführt. War keine Taste gedrückt, werden erneut alle Stellen ausgegeben. Der Zustand »Taste gedrückt« wird vom Programm erst gewertet, wenn mindestens dreimal hintereinander eine gedrückte Taste erkannt wurde. Es ergibt sich eine Entprellzeit von 24 ms (3 × 8 ms). Durch aufeinanderfolgendes Aktivieren der Leitungen B0 bis B7 (ein Ausgang »0«, alle anderen »1«) und ständiges Abfragen der Leitungen A0 bis A2 wird die gedrückte Taste lokalisiert. Das Tastaturabfrageprogramm übergibt im Register A den Tastenkode zur weiteren Auswertung. Eine zweite gedrückte Taste wird erst dann gewertet, wenn zuvor mindestens dreimal der Zustand »keine Taste gedrückt« abgetastet wurde.

Im Muster wurden nur 22 Tasten in der 3 × 8-Matrix beschaltet. Der Tastenkode wird im Abfrageprogramm aus der Nummer der gedrückten Taste in der Matrix über eine Tabelle ermittelt. Damit ist auch bei konstruktiv anderer Gestaltung in jedem Fall eine softwaremäßige Anpassung des Monitorprogramms zu erreichen.



Bit	Segment
0	A
1	B
2	C
3	D
4	E
5	F
6	G
7	DP

Bild 7 Zuordnung der Segmente zu den Ausgabebits

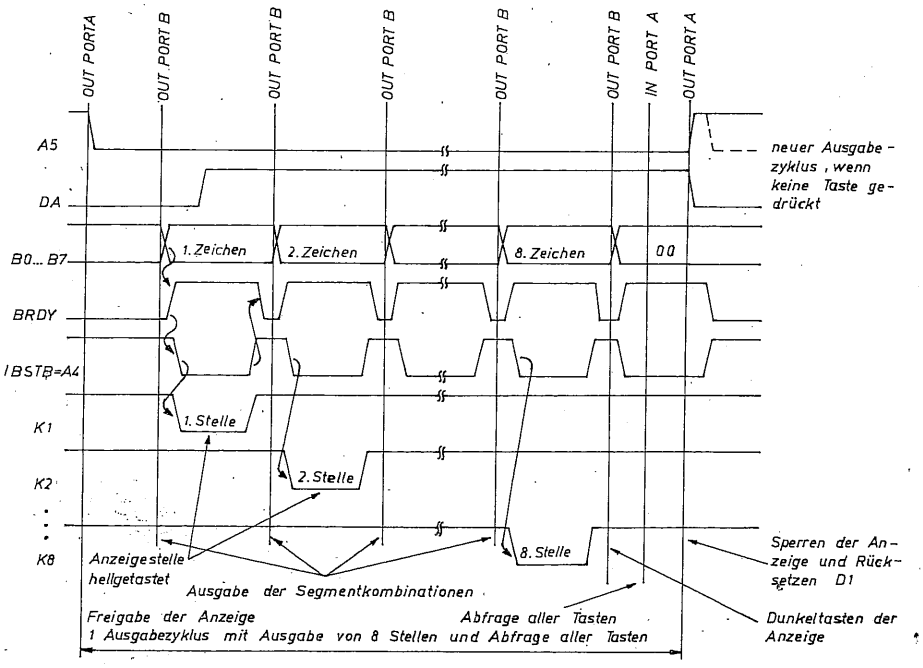


Bild 8 Zeitdiagramm für einen Ausgabezyklus (nicht maßstabsgerecht)

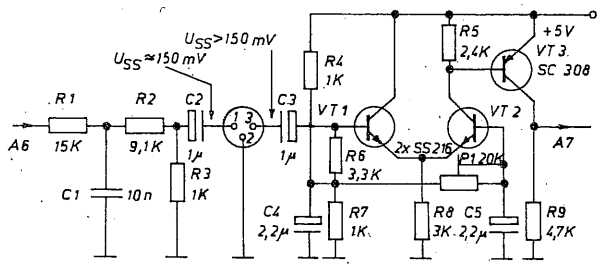
Zwei weitere Tasten sind für die Auslösung von NMI und RESET vorgesehen. Wird eine dieser Tasten gedrückt, lädt sich der Kondensator *C2* auf und gibt somit einen kurzen »Low«-Impuls auf die Leitung. Die *RC*-Kombination *C1* und *R13* verhindert ein mehrmaliges Auslösen von NMI bei Prellen der NMI-Taste.
Die Tastatur- und Anzeigeeinheit wird aus der +5-V-Spannung des Einplatinencomputers betrieben. Die Schaltung benötigt etwa 30 bis 40 mA bei Verwendung von 8 Stellen einer *VQD30*. In der beschriebenen Form lassen sich Tastatur und Anzeige an jeder PIO-Schnittstelle einsetzen. Ein entsprechendes Ansteuerprogramm ist natürlich Voraussetzung. Unter Verwendung von A3 kann auch eine 4 × 8-Matrix aufgebaut werden.

4. Magnetbandanschluß

Es wird das Aufzeichnungsverfahren des »Polycomputers« verwendet. Damit wird die Austauschbarkeit von Programmen und Daten gesichert.

4.1. Stromlaufplan des Magnetbandanschlusses

Bild 9 zeigt die Schaltung für die Magnetbandspeicherung. Die Schaltung realisiert lediglich eine pegelmäßige Anpassung der PIO an den Diodeneingang des Magnetbandgeräts. A6 wird als Ausgabeleitung und A7 als Eingabeleitung programmiert. Die Daten werden seriell mit 1 200 Bit/s aufgezeichnet. Die aufzuzeichnenden Daten werden durch Software kodiert und formatiert. Entsprechende Unterprogramme sind Bestandteil des Monitorprogramms. Diese Programme können auch vom Anwender benutzt werden.



A6 PIO – Ausgabeleitung, PORT A
A7 PIO Eingabeleitung, PORT A

Bild 9 Stromlaufplan der Magnetbandansteuerung. Mit P 1 können Symmetrie und Empfindlichkeit der Schaltung abgeglichen werden

Hinter der passiven *RC*-Kombination am PIO-Anschluß A6 steht ein gleichstromfreies Signal mit $U_{ss} \approx 150$ mV zur Ansteuerung des Magnetbandeingangs zur Verfügung. *C1* dämpft die höheren Frequenzanteile. Für die Aufzeichnung sind handelsübliche Kassetten- oder Magnetbandgeräte geeignet. Es sind keine Eingriffe in das Gerät notwendig. Der mit Transistoren aufgebaute Begrenzerverstärker hebt den Eingangspegel von $U_{ss} \approx 100$ mV auf TTL-Pegel an. Diese Schaltung kann problemlos durch gleichwertige Schaltungslösungen mit Operationsverstärkern ersetzt werden. Der Eingangswiderstand der gezeigten Schaltung liegt bei 3 kΩ.
Das Aufzeichnungsverfahren und die Eigenschaften des verwendeten Kodes sind ausführlich in [1] und [2] erläutert. Hier können nur grundlegende Hinweise gegeben werden, die für das Verständnis der Schaltung und des Programms notwendig sind.

4.2. Aufzeichnungsverfahren

Die Aufzeichnung wird im Conditioned-Diphase-Kode vorgenommen (siehe Bild 10). Jedes Bit wird durch einen Flankenwechsel eingeleitet. Die Richtung der Pegeländerung spielt bei diesem Verfahren keine Rolle. Eine logische »1« wird kodiert, indem der Pegel für die Dauer der Übertragung eines Bits konstant bleibt. Die Flanke wechselt erst mit Beginn des nächsten Bits. Eine logische »0« beinhaltet zusätzlich nach der Hälfte der Bitzeit einen Flankenwechsel. Mit der verwendeten Übertragungsgeschwindigkeit von 1 200 Bit/s ergibt sich die Dauer eines Bits (Bitzeit) zu 833 µs.

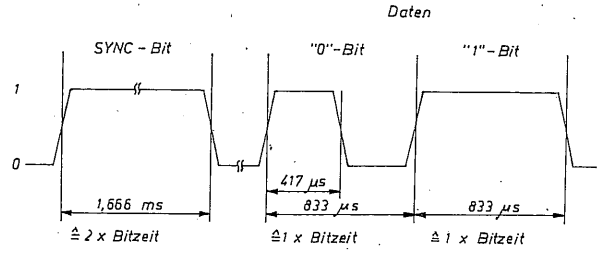


Bild 10 Aufzeichnungskode (alle Zeiten beziehen sich auf 1 200 Bit/s)

Das Ausgabeprogramm führt die Parallel-/Seriell-Wandlung der auszugebenden Bytes und die Kodierung der einzelnen Bits aus. Alle Zeiten werden durch Warteschleifen im Programm erzeugt. Damit ist die Funktionsfähigkeit des Programms unmittelbar von der Taktfrequenz der CPU abhängig. Durch Ändern der einzelnen Zeitkonstanten im Programm ist eine Anpassung an verschiedene Taktfrequenzen im Bereich von 450 kHz bis 2,5 MHz möglich. In der Beschreibung des Monitorprogramms sind die Berechnungsformeln und Speicherplätze angegeben.
Bei der Wiedergabe gespeicherter Daten bzw. Programme synchronisiert sich das Eingabeprogramm auf die Flankenwechsel an den Bitgrenzen. Der Pegel wird nach ¾ der Bitzeit abgetastet. Dabei ist theoretisch eine Zeittoleranz von ±25 % für den Abtastzeitpunkt möglich. Der praktisch auftretende Fehler muß auch im ungünstigsten Fall kleiner sein, um die Austauschbarkeit der Programme und Kassettengeräte zu gewährleisten. Dieser maximale Fehler setzt sich aus folgenden Fehlerkomponenten zusammen:

a) Geschwindigkeitsabweichung zwischen Aufnahme und Wiedergabe. Der Geschwindigkeitsunterschied kann bei Verwendung verschiedener Geräte für Aufnahme und Wiedergabe bis zu 3 % betragen.
b) Gleichlaufschwankungen des Kassettengeräts.
c) Abweichungen der CPU-Taktfrequenz. Dabei ist die zeitliche und temperaturabhängige Abweichung der Oszillatorfrequenz gemeint.
d) Ungenauigkeiten der programmierten Warteschleifen durch Verwendung ganzzahliger Zeitkonstanten. Diese Fehlerkomponente tritt sowohl bei der Aufnahme als auch bei der Wiedergabe auf. Der Fehler hängt von der Länge der Zeitschleife ab und beträgt beim verwendeten Programm maximal 13 CPU-Takte. Die Warteschleifen sind in folgender Form programmiert:

```
        LD B, ZK          ; ZK = Zeitkonstante
MARKE: DJNZ MARKE-#; Warteschleife.
```

e) Ungenauigkeit der Flankensynchronisation beim Lesen der Daten (Bild 11). Das Programm fragt ständig den Pegel der Leitung A7 ab und wartet auf einen Flankenwechsel.

```
MARKE: IN (PORTA)     ; Bit 7 lesen
       XOR C          ; Vergleich mit altem Zustand
       JPP MARKE      ; Sprung, wenn kein Wechsel
```

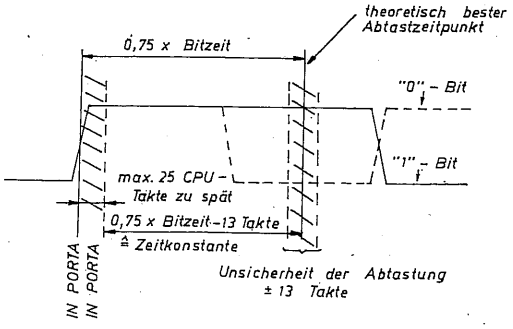


Bild 11 Abtastfehler durch die Unsicherheit der Flankensynchronisation

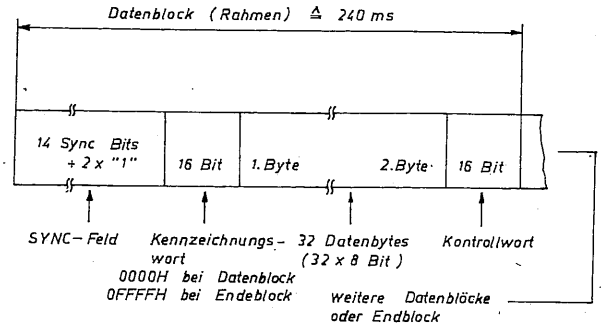


Bild 12 Aufzeichnungsformat (das jeweils niederwertigste Bit eines Bytes wird zuerst aufgezeichnet)

Tabelle 5 Fehlerabschätzung für die Ungenauigkeit der Bitabtastung bei 1200 Baud

Fehlerkomponente	Fehler bei der Taktfrequenz 455 kHz	2,5 MHz
a) Geschwindigkeitsabweichung	±3 %	±3 %
b) Gleichlaufschwankungen	±1,5 %	±1,5 %
c) Taktfrequenz	±1,5 %	±1 %
d) Zeitfehler beim Senden	±1,5 %	±0,3 %
Zeitfehler beim Lesen	±1,5 %	±0,3 %
e) Ungenauigkeit der Flankensynchronisation	±3 %	±0,6 %
Gesamt	±12 %	±6,7 %

Das Register C enthält im Bit 7 den alten Zustand vor dem Flankenwechsel. Im ungünstigsten Falle wird die Flanke 25 CPU-Takte zu spät erfaßt. Die Warteschleife für die Abtastung nach ¾ der Bitzeit wird deshalb um 13 Takte verringert, um eine maximale Abweichung von ±25/2 Takten zu erreichen.

Die Fehler d) und e) sind von der Taktfrequenz der CPU abhängig. Tabelle 5 zeigt den Einfluß aller Einzelfehler und den nur im ungünstigsten Fall auftretenden Gesamtfehler. Dieser liegt weit unter dem theoretisch möglichen Fehler.

Diese kurze Fehlerabschätzung sollte die hohe Sicherheit dieses Aufzeichnungsverfahrens unterstreichen. Die Sicherheit ist insbesondere auf die Synchronisation des Abtastzeitpunktes an jeder Bitgrenze (Bitsynchronisation) zurückzuführen. Selbst mit Kassettengeräten vom Typ Mira ist die Aufzeichnungssicherheit so groß, daß absichtlich Fehler erzeugt werden müssen, um die Fehlererkennung im Programm zu testen. Lediglich eine stark verminderte Höhenwiedergabe durch einen verschmutzten Tonkopf führt zu größeren Fehlerraten. Ein Einfluß der Taktfrequenz auf die Fehlerrate konnte beim praktischen Betrieb nicht bemerkt werden.

4.3. Aufzeichnungsformat

Bei der Aufzeichnung werden die einzelnen Bytes eines Speicherbereichs seriell, d. h. bitweise ausgegeben. Beim Lesen der gespeicherten Informationen muß neben der beschriebenen Bitsynchronisation auch die Bytesynchronität gewährleistet sein. Zu diesem Zweck werden vor den eigentlichen Daten Synchronisationsbits mit der doppelten Bitzeit aufgezeichnet. Diese Folge von SYNC-Bits entspricht einer Rechteckschwingung von 300 Hz und bildet den Kennton am Anfang der Aufzeichnung. Das Ende des SYNC-Feldes wird durch zwei normal kodierte »1« angezeigt. Danach folgen unmittelbar die Datenbits. Jeweils 8 aufeinanderfolgende Bits stellen ein Byte dar. Beim Überlesen von Bits durch Bandfehler kann die Bytesynchronisation verlorengehen. In diesem Fall werden alle nachfolgenden Bits falsch interpretiert. Damit ein solcher Fehler nicht die gesamte Aufzeichnung wertlos macht, wird die Aufzeichnung in Blöcke (Rahmen) unterteilt. Alle Blöcke beginnen mit einem SYNC-Feld, das in jedem Fall die Bytesynchronisation herstellt. Auf das SYNC-Feld folgen ein 16-Bit-Kennzeichnungswort, 32 Datenbytes und eine 16-Bit-Kontrollsumme. Die Kontrollsumme wird bei der Aufzeichnung berechnet und mit aufgezeichnet. Auch beim Lesen werden diese Berechnung und der Vergleich mit der aufgezeichneten Kontrollsumme vorgenommen. Das ermöglicht ein Erkennen fehlerhafter Blöcke. Die effektive Aufzeichnungsgeschwindigkeit liegt bei etwa 120 Byte/s. 1 KByte läßt sich mit Kennton in 10 s abspeichern.

5. Monitorprogramm

Nach Einschalten bzw. RESET beginnt die CPU auf der Adresse 0000H mit der Ausführung eines Programms. Um mit dem Rechner in Verbindung treten zu können, liegt in der Regel an dieser Stelle ein Monitorprogramm. Dieses Programm meldet sich beim Benutzer und erlaubt den Aufruf bestimmter Monitorfunktionen. Hier soll ein kleines Monitorprogramm besprochen werden, mit dem Programme in den RAM-Bereich eingegeben, getestet und geändert werden können.

5.1. Funktionen des Monitors

Das Monitorprogramm ist in einem 1-KByte-EPROM *U555C* gespeichert. Dem Anwender stehen 960 Bytes RAM auf der CPU-Platine zur Verfügung. Folgende Monitorfunktionen sind benutzbar:

- Anzeige und Modifikation von Speicherinhalten;
- Starten von Programmen und Setzen eines Haltepunktes;
- Anzeige und Modifikation von Registerinhalten;
- Speichern und Laden von Speicherbereichen von Kassette;
- Füllen von Speicherbereichen mit konstanten Daten;
- Umladen von Speicherbereichen;
- Beschreiben eines Ausgabeports;
- Lesen eines Eingabeports.

Das Monitorprogramm benutzt die vorgestellte Tastatur- und Anzeigeeinheit für alle Ein- und Ausgaben. Bild 13 zeigt die Anordnung der Tasten und der Anzeige beim Mustergerät. Die acht Stellen der 7-Segment-Anzeige sind in zwei Gruppen aufgeteilt.

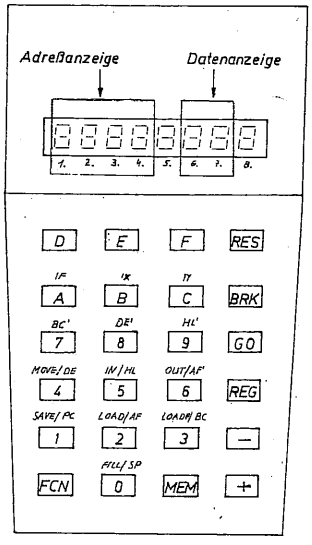


Bild 13 Anordnung der Tasten und der Anzeige beim Mustergerät

In der linken Hälfte zeigen 4 Stellen Adressen oder Registerinhalte an. Zwei Stellen in der rechten Hälfte werden zur Anzeige von Daten oder Registernamen benutzt. Die 5. und die 8. Stelle werden vom Programm nicht angesteuert.
Die Tastatur teilt sich in 16 Hexadezimaltasten, 6 Funktionstasten (+, –, MEM, REG, GO und FCN) und zwei Tasten zum Auslösen von RESET (RES) und NMI (BRK). Mit der Funktionstaste FCN werden die Hexadezimaltasten auch zur Eingabe von weniger häufig benutzten Funktionen ausgenutzt. Das Monitorprogramm fragt die Tasten in der aus Bild 6 ersichtlichen Reihenfolge ab. Tabelle 4 zeigt den Tastenkode, der den Tasten im Tastatur-Unterprogramm zugeordnet wird. Die Zuordnung ergibt sich über eine Tabelle im Monitorprogramm (Adresse O3BEH +n –1). Beim Drücken der FCN-Taste wird zum Kode der nachfolgend gedrückten Taste der Wert 2OH addiert. Die Hex-Zahlen zur Ausgabe in 7-Segment-Darstellung werden ebenfalls über eine Tabelle umkodiert (Adresse O3D6H).
Vom Monitorprogramm werden der Interruptmode und die Stellung des Interruptfreigabe-Flip-Flops nicht geändert (kein DI- oder EI-Befehl). Um Speicherplatz zu sparen, wurden die RST-Befehle mit den am häufigsten benutzten Unterprogrammen belegt.

5.2. Start des Monitors

Nach dem Einschalten der Betriebsspannungen bzw. dem Drükken der Taste RES meldet sich das Programm mit der Ausgabe des Dezimalpunktes in der 1. Stelle. Mit den Funktionstasten MEM, REG, GO oder der FCN-Taste und einer HEX-Taste wird in die zugehörige Monitorfunktion verzweigt. Beim Aufruf einer neuen Funktion wird die zuvor aufgerufene Funktion wieder verlassen. Dieser Abbruch ist auch während einer Adreß- oder Dateneingabe möglich. Bei der Eingabe von Adressen werden die Ziffern von rechts in das Adreßfeld der Anzeige geschoben. Jeweils die letzten vier eingegebenen Ziffern sind sichtbar und werden als gültige Adresse gewertet. Analoges gilt für das zweistellige Datenfeld. Führende Nullen brauchen nicht eingegeben zu werden. Mit den Tasten »+« und »–« wird eine Eingabe beendet bzw. die Ausführung der Funktion veranlaßt. Zweimaliges Drücken der FCN-Taste führt in jedem Fall zur Grundschleife zurück.
Beim Betätigen der BRK-Taste wird das gerade abgearbeitete Programm unterbrochen. Alle Register werden im RAM-Bereich des Monitors gerettet. Danach befindet sich der Monitor selbständig in der Funktion Registeranzeige (REG). Es können aber auch alle anderen Funktionen aufgerufen werden. Nachfolgend werden die einzelnen Monitorfunktionen ausführlich beschrieben.

5.3. Speicheranzeige (MEM)

Nach Drücken der Taste MEM wird der Bediener zur Eingabe einer Adresse aufgefordert. Die Adreßeingabe wird mit der Taste »+« beendet. Gleichzeitig erscheint im Datenfeld der Inhalt des angegebenen Speicherplatzes. Mit dem Betätigen der Tasten »+« oder »–« wird die Adresse erhöht oder erniedrigt und der zugehörige Speicherplatz angezeigt. Nach der Eingabe von HEX-Zahlen überschreiben diese nach Drücken der Taste »+« den angezeigten Speicherplatz. Danach wird die Adresse selbständig erhöht und angezeigt. In dieser Form können Programme eingegeben werden.

5.4. Starten von Programmen (GO)

Mit dieser Funktion werden eigene Programme gestartet und getestet. Ein Software-Haltepunkt kann gesetzt werden. Der Inhalt des durch die Haltepunktadresse angegebenen Speicherplatzes wird durch einen RST 38H-Befehl ersetzt. Erreicht die CPU beim Abarbeiten des Programms diesen Punkt, so ergibt sich ein Rücksprung in den Monitor. In der Haltepunktbehandlung werden alle Register einschließlich des Befehlszählers (PC) und des Interrupt-Flip-Flops (IFF) gerettet. Dann wird der RST 38H-Befehl im Anwenderprogramm wieder durch den ursprünglichen Befehlskode ersetzt. Der Anwender kann den RST 38H-Befehl im Programm auch direkt zum Setzen weiterer Haltepunkte verwenden. Das Monitorprogramm unterscheidet in der Haltepunktbehandlung selbständig diese Anwenderhaltepunkte vom Monitorhaltepunkt. Der Monitorhaltepunkt kann also sinnvoll nur im RAM-Bereich auf das jeweils erste Byte des Befehls gesetzt werden. Dieser Befehl wird mit Erreichen des Haltepunktes nicht mehr abgearbeitet. Die NMI-Behandlung ist mit der Haltepunktbehandlung identisch.
Nach Drücken der GO-Taste erscheint in der Anzeige der in der letzten Haltepunktbehandlung gerettete Befehlszähler. Mit Betätigen der Taste »+« wird das Anwenderprogramm an dieser Stelle fortgesetzt. Die angezeigte Adresse kann zuvor aber auch mit einer neuen Startadresse überschrieben werden. Die Taste »–« erlaubt die Eingabe einer Haltepunktadresse. Mit der BRK-Taste kann ein Anwenderprogramm jederzeit unterbrochen werden. Da hier ebenfalls alle Register gerettet werden, ist eine Fortsetzung an der unterbrochenen Stelle möglich. Die Monitorroutinen lassen sich ebenfalls mit der BRK-Taste abbrechen. Eine Fortsetzung ist aber nicht sinnvoll, da in der Regel der Stack zerstört wird.
Bei einem GO-Kommando unter Angabe einer Startadresse wird der Stackpointer (SP) vom Monitor mit der Adresse OFCOH geladen.

Tabelle 6 Mehrfachbelegung der HEX-Tasten

HEX-Taste	Funktion mit FCN-Taste	Funktion mit REG-Taste
0	FILL	Register SP
1	SAVE	Register PC
2	LOAD	Register AF
3	LOADP	Register BC
4	MOVE	Register DE
5	IN	Register HL
6	OUT	Register AF'
7		Register BC'
8		Register DE'
9		Register HL'
A		Register IF
B		Register IX
C		Register IY

5.5. Registeranzeige (REG)

Diese Monitorfunktion zeigt die in der Haltepunktbehandlung geretteten Register des Anwenderprogramms. Die Funktion wird mit der Taste REG oder nach der Haltepunktbehandlung aufgerufen. Das Monitorprogramm verwaltet einen Registerzeiger. Der Registerzeiger zeigt auf eine Speicherzelle in dem Bereich des Monitor-RAM, in dem die Anwenderregister gerettet werden. Beim Aufruf der Registeranzeige wird dieses Register angezeigt. Mit den Tasten »+« und »–« kann der Registerzeiger erhöht oder erniedrigt werden. Damit lassen sich alle Register der Reihe nach anzeigen. Bei erneutem Betätigen der REG-Taste und einer HEX-Taste kann ein Registerpaar entsprechend Tabelle 6 direkt angewählt werden.
Die Register werden grundsätzlich als Doppelregister im Adreßfeld angezeigt. Im Datenfeld erscheint der Registername. Die Register des Zweitregistersatzes werden mit dem Aufleuchten des Dezimalpunktes gekennzeichnet. »IF« bezeichnet den Inhalt des Interruptregisters der CPU und die Stellung des Interruptfreigabe-Flip-Flops.
Bei der Eingabe eines Bytes über die HEX-Tastatur erscheint die Anzeige im Datenfeld. Mit der Taste »+« wird das eingegebene

Byte in den höherwertigen Teil des Doppelregisters übernommen. Beim Drücken der Taste »–« wird der niederwertige Teil des Doppelregisters geladen. Beim Sprung zum Anwenderprogramm (GO) werden die im Monitor-RAM gespeicherten Registerinhalte in die CPU-Register übertragen. Nach dem Einschalten ist die Belegung dieser Speicherzellen unbestimmt. Bei RESET wird der Inhalt der Anwenderregister nicht verändert, während er beim Lesen vom Magnetband zerstört wird.

5.6. Füllen von Speicherbereichen, FILL (FCN 0)

Mit der Anzeige »FA« im Datenfeld wird der Anwender zur Eingabe der Anfangsadresse des zu füllenden Speicherbereichs aufgefordert. Die Taste »–« beendet die erste Adreßeingabe. Jetzt erscheint die Anzeige »FE«, und die Endadresse wird eingegeben. Die Endadresse wird mit der Taste »+« beendet. Diese Form der Eingabe von Anfangs- und Endadresse eines Speicherbereichs gilt analog auch für die folgenden Monitorfunktionen. Es wechselt nur der Kommandobuchstabe in der 6. Stelle der Anzeige.
Ist die eingegebene Endadresse kleiner als die Anfangsadresse, meldet sich der Monitor mit der Fehlermeldung »E« in der Anzeige. Nach Eingabe der Endadresse wird das Byte eingegeben, mit dem der angegebene Speicherbereich gefüllt werden soll. Nach dem Zuschalten der Betriebsspannungen haben alle RAM-Speicherzellen einen undefinierten Zustand. Es empfiehlt sich, den gesamten RAM-Bereich mit dem Byte OOH zu laden. Es fällt sonst schwer, das Ende eines eingegebenen Programms zu finden.

5.7. Magnetbandspeicherung, SAVE (FCN 1)

Zum Abspeichern von Speicherbereichen auf einem handelsüblichen Kassettengerät müssen Anfangs- und Endadresse eingegeben werden. Nach Eingabe der Endadresse beginnt sofort die Magnetbandausgabe. Das Kassettengerät muß zuvor gestartet werden. Am Beginn der Aufzeichnung wird ein etwa 3 s langer Kennton ausgegeben. Die Programme bzw. Daten werden in Blöcken zu je 32 Byte ausgegeben. Das Kennzeichnungswort der einzelnen Blöcke ist immer OOOOH. Verbleiben für den letzten Datenblock weniger als 32 Byte, so wird dieser Block mit den nachfolgenden Bytes aufgefüllt. Da beim Einlesen die Endadresse eingegeben wird, werden diese zusätzlichen Bytes nicht ausgewertet.
Abweichend vom *Polycomputer* wird nach dem letzten Datenblock ein Endeblock ausgegeben. Der Endeblock enthält das Kennzeichnungswort OFFFFH. Dieser Endeblock wird vom *Polycomputer* ebenfalls nicht ausgewertet. Er kann aber zum Beenden der Magnetbandeingabe genutzt werden.

5.8. Magnetbandeingabe, LOADP (FCN 3)

Nach Eingabe von Anfangs- und Endadresse versucht das Leseprogramm, sich mit den Impulsflanken vom Magnetband zu synchronisieren. Dazu müssen mindestens 7 unmittelbar aufeinanderfolgende SYNC-Bits gelesen werden. Ist diese Bedingung erfüllt, wird auf das Abtasten der beiden als »1« kodierten Impulse gewartet. Damit ist der Blockanfang eindeutig gekennzeichnet. Jetzt beginnt die Übernahme des Kennzeichnungswortes, der 32 Datenbytes und der Prüfsumme. Danach folgt erneut die Blocksynchronisation. Das Kassettengerät kann auch nach dem Start des Programms eingeschaltet werden. Die Aufzeichnung muß mit dem Hinweiston beginnen. Die Wahrscheinlichkeit einer Fehlsynchronisation bei Sprache oder Musik ist relativ hoch.
Die gelesenen Bytes werden, beginnend mit der Anfangsadresse, im angegebenen Speicherbereich abgelegt. Das Programm meldet sich erst bei Erreichen der Endadresse. Wurde die Endadresse zu hoch gewählt, versucht das Programm weiterzulesen. In diesem Fall kann mit RES oder BRK abgebrochen werden. Das Kennzeichnungswort wird nur für die Berechnung der Prüfsumme ausgewertet. Ist der Einlesevorgang beendet, meldet sich der Monitor bei fehlerfreier Übertragung in üblicher Weise mit dem Dezimalpunkt. Bei Lesefehlern erscheint im Adreßfeld die Anfangsadresse des letzten fehlerhaft gelesenen Blocks. Das Datenfeld zeigt die Anzahl der fehlerhaften Blöcke. Durch wiederholtes Drücken der Taste »+« können die Anfangsadressen aller fehlerhaften Blöcke rückwärts angezeigt werden. Alle 32 Bytes eines solchen fehlerhaften Blocks können falsch sein. In diesen Fällen nützt auch ein zweites Lesen nicht viel, da unter Umständen andere Blöcke fehlerhaft sind. Die Funktion LOADP verhält sich wie die entsprechende Routine des *Polycomputers*.

5.9. Magnetbandeingabe, LOAD (FCN 2)

Der Hauptnachteil des *Polycomputer*-Verfahrens besteht in der notwendigen Kenntnis der Endadresse des Programms. Aus diesem Grund ist eine zweite Funktion für die Magnetbandeingabe verfügbar. Nach Aufruf der Funktion LOAD muß ebenfalls eine Anfangs- und Endadresse eingegeben werden. Im Unterschied zur Funktion LOADP wird das Einlesen abgebrochen, wenn die Endadresse erreicht oder aber der Endeblock gelesen wird. Damit kann beispielsweise OFFFFH als Endadresse vorgegeben werden. Die Datenbytes des Endeblocks werden nur zur Prüfsummenberechnung gewertet, aber nicht abgespeichert. Beim Lesen von Programmen oder Daten, die mit dem *Polycomputer* aufgezeichnet wurden, ist die richtige Angabe der Endadresse notwendig.
Bei der LOAD-Funktion werden nur fehlerfrei gelesene Blöcke im Speicher abgelegt. Die als fehlerhaft erkannten Blöcke werden unterdrückt. Die sich ursprünglich an dieser Stelle im Speicher befindenden 32 Byte bleiben erhalten. Die Adressen werden trotzdem weiterhin fortlaufend berechnet. Daher werden die nachfolgenden fehlerfreien Blöcke auf den richtigen Adressen gespeichert. Damit wird es möglich, ein zweites Mal zu lesen, ohne richtig beschriebene Adreßbereiche zu zerstören. Insbesondere beim mehrmaligen Aufzeichnen von Programmen und dem »überlagernden« Lesen mit LOAD läßt sich die Sicherheit der Magnetbandspeicherung noch beträchtlich steigern. Fehlerhafte Blöcke werden in gleicher Weise wie bei der LOADP-Funktion angezeigt.
In der Regel ergibt sich beim Hinweiston, der aus 2 000 SYNC-Bits besteht, sofort eine Synchronisation. Eine geringfügige Bandstörung genügt aber, um das Ende des SYNC-Feldes vorzutäuschen. Damit wird das fehlerhafte Lesen eines Datenblocks ausgelöst. Bei der LOAD-Funktion beginnt die Datenübernahme erst, wenn ein Block gelesen wird, dessen Kennwort im Bit 15 eine »0« enthält. Der beschriebene Fehler läßt sich so größtenteils ausschließen, da in diesem Fall vorwiegend »1« gelesen wird. Mit dem Anschluß von Tastatur und Bildschirm wird das beschriebene Aufzeichnungsprotokoll mit Dateikennsätzen erweitert. Näheres ist in [2] zu finden. Solche zusätzlichen Dateikennsätze, die z. B. den Programmnamen enthalten, sind mit Bit 15 = »1« im Kennwort gekennzeichnet. Sie werden damit vom LOAD-Kommando am Anfang automatisch überlesen. Es besteht bei späteren Erweiterungen die Kompatibilität zu neuen Aufzeichnungsprotokollen.

5.10. Umladen von Speicherbereichen, MOVE (FCN 4)

Die MOVE-Funktion wird zum Umladen von Speicherbereichen benutzt. Wie bei der FILL-Funktion werden Anfangs- und Endadresse des umzuladenden Bereichs angegeben. Dann fordert der Monitor die Eingabe der neuen Anfangsadresse an. Nach Abschluß mit der Taste »+« wird der angegebene Speicherbereich umgeladen. Der alte und der neue Speicherbereich können sich beliebig überlappen.

5.11. Lesen eines Ports, IN (FCN 5)

Nach dem Aufruf dieser Funktion wird die Eingabe einer Adresse erwartet. Das niederwertige Byte stellt die eigentliche Portadresse dar. Der höherwertige Teil wird bei der Ausführung des Eingabebefehls über den höherwertigen Teil des Adreßbusses ausgegeben. Das kann bei einigen Anwendungen von Vorteil sein. Mit jedem Drücken der Taste »+« wird der Inhalt des adressierten Eingabeports gelesen und angezeigt. Jede andere Taste führt zum Abbruch der Funktion.

5.12. Beschreiben eines Ports, OUT (FCN6)

Die Adressen werden in gleicher Weise wie bei der IN-Funktion eingegeben. Über die HEX-Tastatur gibt man das auszugebende Byte ein. Mit Drücken der Taste »+« wird der OUT-Befehl ausgeführt. Es kann sofort ein neues Byte eingegeben werden.
Die Funktionen FCN7 bis FCNF sind nicht belegt. Beim Aufruf verzweigt der Monitor aus der Grundschleife zur Adresse O4OOH. Hier kann nach Bedarf ein zweites EPROM mit Monitorerweiterungen eingesetzt werden. Es lassen sich aber auch spezielle Anwenderprogramme aufrufen. Dabei ist die Benutzung von Unterprogrammen des Monitors möglich. Es stehen Programme für Tastatureingabe, Displayausgabe und Magnetband-Ein-/Ausgabe zur Verfügung.

5.13. Anpassen des Monitors

Tabelle 7 zeigt den Maschinenkode des Monitorprogramms. Die erste Spalte enthält die laufende Adresse. Jeweils 16 aufeinanderfolgende Bytes sind in einer Zeile abgedruckt. Aus Tabelle 8

Tabelle 7
Maschinenkode des Monitorprogramms

0000	31	D7	0F	3E	80	C3	A1	00	D9	1E	00	C3	8C	02	00	00
0010	FD	21	F3	0F	7C	C3	6B	02	97	26	00	C3	60	02	00	00
0020	21	F3	0F	06	09	C3	85	00	01	F8	0F	02	03	C3	8B	00
0030	FD	21	F8	0F	79	C3	6F	02	ED	73	EF	0F	31	ED	0F	F5
0040	C5	D5	E5	08	D9	F5	C5	D5	E5	ED	57	47	0E	00	E2	52
0050	00	0C	C5	DD	E5	FD	E5	ED	7B	EF	0F	D1	ED	73	EF	0F
0060	2A	F1	0F	7C	B5	01	18	D0	31	D7	0F	28	10	1B	E5	ED
0070	52	28	01	13	E1	7E	3C	20	04	3A	FE	0F	77	EB	22	ED
0080	0F	3E	14	18	21	36	00	23	10	FB	C9	3E	77	02	DF	FE
0090	11	20	13	E5	3E	79	02	DF	20	0C	D1	ED	52	23	D0	3E
00A0	79	E7	32	F3	0F	CF	31	D7	0F	E7	21	A6	00	E5	D6	13
00B0	28	6D	3D	CA	7A	01	3D	28	25	D6	0B	28	3A	3D	CA	E6
00C0	01	01	38	FF	3D	CA	0A	02	01	73	00	3D	28	F7	3D	CA
00D0	C6	01	3D	28	3F	3D	28	2F	FE	0A	DA	00	04	C7	DF	C0
00E0	3E	23	3E	2B	4E	F7	D7	CF	28	F7	FE	11	28	F5	D0	CD
00F0	51	02	20	F0	71	18	EA	3E	71	EF	CD	50	02	C0	EB	71
0100	1B	23	7A	B3	20	F9	C7	DF	C0	CD	50	02	C0	44	79	4D
0110	ED	79	18	F5	DF	C0	44	4D	ED	48	F7	CF	28	F8	C9	2A
0120	ED	0F	D7	CF	30	09	21	C0	0F	22	EF	0F	CD	19	00	22
0130	FC	0F	21	00	00	28	10	FE	11	C0	3E	76	32	F9	0F	DF
0140	C0	7E	32	FE	0F	36	FF	22	F1	0F	F1	21	FF	C3	22	FA
0150	0F	FD	E1	DD	E1	F1	ED	47	21	FB	D3	38	02	2E	F3	22
0160	F8	0F	E1	D1	C1	F1	08	D9	E1	D1	C1	F1	ED	7B	EF	0F
0170	C3	F8	0F	CF	D0	77	0E	34	3E	35	CD	AE	01	7E	23	66
0180	6F	D7	21	E6	03	19	11	F8	0F	ED	A0	ED	A0	21	FF	0F
0190	CF	28	E4	FE	14	28	DC	FE	11	28	DE	D0	4F	CD	AE	01
01A0	CD	52	02	23	28	05	FE	11	20	D0	2B	71	18	CC	21	FF
01B0	0F	7E	FE	0D	38	05	20	01	AF	E6	0C	77	87	21	EF	0F
01C0	5F	AF	57	ED	52	C9	3E	5C	EF	3E	77	02	0B	3E	54	02
01D0	44	4D	DF	C0	E5	ED	52	E1	38	08	0B	09	EB	09	03	ED
01E0	B8	C7	EB	ED	B0	C7	3E	6D	EF	D5	01	20	00	D9	E1	01
01F0	D0	07	11	00	00	CD	F4	02	D9	37	ED	42	23	D9	01	0E
0200	00	30	EF	11	FF	FF	CD	F4	02	C7	C5	79	EF	C1	0E	00
0210	78	D9	21	D7	0F	E5	CD	4F	03	E1	4F	06	20	7A	D9	20
0220	05	A0	FA	42	02	FC	0C	D5	CB	61	20	19	78	D9	A1	7E
0230	23	D9	20	01	12	13	2B	7C	B5	28	08	78	D9	10	EF	97
0240	18	D0	AF	B1	C8	E1	D7	F7	CF	C2	A6	00	0D	20	F6	C7
0250	97	4F	F7	CF	D0	CB	21	CB	21	CB	21	CB	21	81	18	F1
0260	6F	D7	CF	D0	29	29	29	29	85	18	F5	CD	6F	02	7D	F5
0270	07	07	07	07	CD	78	02	F1	D9	21	D6	03	E6	0F	5F	16
0280	00	19	7E	D9	FD	77	00	FD	23	C9	1E	20	16	FF	06	03
0290	21	F3	0F	CD	C9	02	AA	07	30	F4	10	F4	14	28	EF	21
02A0	BD	03	01	BD	FE	ED	41	23	DB	BC	2F	E6	07	20	06	CB
02B0	00	38	F2	18	D7	06	00	ED	41	1F	07	07	07	4F	09	7E
02C0	83	FE	12	28	C5	D9	FE	10	C9	3E	FF	D3	BE	3E	9F	D3
02D0	BE	3E	0F	D3	BF	DB	BC	CB	AF	D3	BC	C5	01	BD	09	DB
02E0	BC	CB	67	28	FA	ED	A3	20	F6	DB	BC	CB	EF	D3	BC	F6
02F0	F8	3C	C1	C9	3E	2A	3D	20	FD	CD	48	03	0B	78	B1	20
0300	F3	CD	44	03	D5	DD	E1	CD	44	03	06	07	CD	28	03	3E
0310	10	06	02	08	5E	23	56	23	DD	19	CD	28	03	08	3D	06
0320	01	20	F0	DD	E5	D1	06	04	0E	10	10	FE	CB	3A	CB	1B
0330	3F	1F	1F	47	DB	BC	A8	D3	BC	06	0B	CD	46	03	0D	C8
0340	06	08	18	E6	06	17	10	FE	DB	BC	EE	40	D3	BC	C9	08
0350	06	07	DB	BC	4F	16	19	15	28	F6	DB	BC	A9	F2	57	03
0360	7A	FE	0D	30	EB	10	EB	06	25	CD	AE	03	FA	67	03	06
0370	0D	CD	9E	03	08	A2	FA	4F	03	D5	D5	DD	E1	3E	10	08
0380	CD	9C	03	DD	19	73	23	72	23	08	3D	20	F2	7E	CD	9C
0390	03	DD	E5	E3	AF	ED	52	B4	B5	E1	D1	C9	06	08	11	00
03A0	80	CD	AE	03	17	CB	1A	CB	1B	D8	06	0E	18	F3	10	FE
03B0	DB	BC	47	A9	48	F5	DB	BC	A9	F2	B6	03	F1	C9	14	13
03C0	0D	0A	07	04	01	12	10	11	0E	0B	08	05	02	00	FF	FF
03D0	0F	0C	09	06	03	15	3F	06	5B	4F	66	6D	7D	07	7F	6F
03E0	77	7C	39	5E	79	71	6D	73	73	39	77	71	7C	39	5E	79
03F0	76	38	77	F1	7C	B9	5E	F9	76	B8	06	71	06	76	06	72

Tabelle 8 Kontrollsummen für das Monitorprogramm (HEX)

Adreßbereich	Kontrollsumme
0000–00FF	6D90
0100–01FF	7A9B
0200–02FF	7BB0
0300–03FF	672B

lassen sich die Prüfsummen für das Monitorprogramm entnehmen. Damit kann das fehlerfreie »Abtippen« kontrolliert werden. Die Prüfsumme entsteht durch einfache Addition aller Bytes in einem 16-Bit-Register.

Tabelle 9 enthält alle wichtigen Adressen des Monitorprogramms. Mit Hilfe dieser Tabelle kann es bei einiger Erfahrung an das eigene System angepaßt werden. Das wird vor allem bei anderen Ein-/Ausgabebedingungen der Fall sein. Die Zuordnung von Tasten- und 7-Segment-Kode über Tabellen im Programm dürfte diese Arbeit erleichtern. Bei unverändertem Nachbau der beschriebenen Schaltungen sind keine Änderungen notwendig.

Die RST-Adressen wurden zur Einsparung von Speicherplätzen für den Aufruf besonders häufig benutzter Unterprogramme belegt. Mit RST0 kann jederzeit in die Monitorgrundschleife gesprungen werden. Alle Monitorunterprogramme lassen sich benutzen. Dazu enthält Tabelle 9 auch die notwendigen Angaben für die Registerbelegungen. Die Anzeige-Unterprogramme schreiben die auszugebenden Daten im 7-Segment-Kode in 8 Speicherzellen (DISPLAY-RAM). Jede Speicherzelle entspricht einer Stelle in der Anzeige. Beim Aufruf von RST20H werden diese Zellen mit dem Wert OOH geladen. Fordert das Monitorprogramm eine Eingabe an, so wird das Tastaturabfrageprogramm aufgerufen. Innerhalb dieses Programms wird die eigentliche Ausgabe der 8 DISPLAY-Zellen auf die Anzeige bewirkt. Jeweils am Ende eines Ausgabezyklus wird die Tastatur abgefragt. Ist keine Taste betätigt, werden erneut alle 8 Stellen ausgegeben. Erst beim Druck auf eine Taste ermittelt das Unterprogramm den Kode dieser Taste und führt den Rücksprung aus. Die Anzeige leuchtet daher nur, wenn das Monitorprogramm auf eine Eingabe wartet. Die Monitorfunktion wird aber in der Regel so schnell ausgeführt, daß der Eindruck einer ständig leuchtenden Anzeige entsteht.

Die Unterprogramme für die Magnetband-Ein-/Ausgabe belegen zusammen etwa 200 Byte Programmspeicher. Dabei werden keine RST-Befehle benutzt. Der Wert BTA stellt die Anzahl der Rechnertakte je Übertragungszeit eines Bits dar. Er läßt sich aus dem Verhältnis von Taktfrequenz und Übertragungsgeschwindigkeit berechnen:

$BTA = f/1{,}2$, f - Taktfrequenz in kHz.

Die angegebene Formel ergibt den Wert für BTA bei einer Übertragungsgeschwindigkeit von 1200 Bit/s. Die berechneten Konstanten werden auf ganzzahlige Werte gerundet, hexadezimal kodiert und auf den entsprechenden Speicheradressen eingetragen. Damit kann das Programm an CPU-Taktfrequenzen im Bereich von 455 kHz bis 2,5 MHz angepaßt werden. Die Speicherzugriffe dürfen nicht mit WAIT verlängert werden.

Beim Magnetbandlesen werden die gelesenen Bytes eines Datenblocks im Monitor-RAM zwischengespeichert. Dabei werden die Anwenderregister zerstört. Durch Füllen des Speicherbereichs OFD7H bis OFFOH mit OOH können die Anwenderregister jederzeit in einen definierten Zustand gebracht werden.

Tabelle 9 Wichtige Adressen und UP-Aufrufe

ADR (HEX)	Bemerkungen
0000	Monitorstart mit Anzeige Dezimalpunkt (RST0)
0008	UP: Tastatureingabe mit Anzeige DISPLAY-RAM (RST8) nach RET : A = Tastenkode verändert : AF, DE', BC', HL'
0010	UP: Ausgabe ADR in DISPLAY-RAM (Stelle 1–4) (RST10) vor CALL : HL = ADRESSE verändert : AF, DE', HL', IY
0018	UP: Eingabe einer Adresse mit Anzeige (RST18) nach RET : HL = ADRESSE verändert : AF, BC', DE', HL', IY
0020	UP: Löschen DISPLAY-RAM (RST20) verändert: F, B, HL
0028	UP: Eingabe Anfangs- und Endadresse eines (RST28) Speicherbereiches mit Anzeige vor Call : A = Anzeigekode für Stelle 6 nach RET : DE = AADR, HL = Byteanzahl verändert : AF, BC, BC', DE', HL', IY
0030	UP: Ausgabe BYTE in DISPLAY-RAM (St. 6,7) (RST30) vor CALL : C = BYTE verändert : AF, DE', HL', IY
0038	Haltepunkt- und NMI-Behandlung
0085	Fortsetzung RST 20H
008D	Fortsetzung RST 28H
00A1	Fortsetzung RST 0
00DE	Monitorfunktion MEM
00F7	Monitorfunktion FILL
0107	Monitorfunktion OUT
0114	Monitorfunktion IN
011F	Monitorfunktion GO
0173	Monitorfunktion REG
01C6	Monitorfunktion MOVE
01E6	Monitorfunktion SAVE
020A	Monitorfunktion LOAD und LOADP
0250	UP: Eingabe eines Bytes mit Anzeige nach RET : C = BYTE verändert : AF, BC', DE', HL', IY
0260	Fortsetzung RST 18H
0268	Fortsetzung RST 10H
026F	Fortsetzung RST 30H
0278	UP: Umkodierung HEX in 7-Segment-Kode mit Ausgabe in DISPLAY-RAM vor CALL : A = HEX-ZIFFER (BIT 3...0) IY = Adresse DISPLAY-RAM nach RET : IY = ADR+1, (IY+0) = 7-Segment-Kode verändert : AF, DE', HL', IY
028A	Fortsetzung RST 8H
028F	Speicherzelle für Entprellzeit-Konstante (N*8ms)
02C9	UP: Einmalige Ausgabe DISPLAY-RAM über PIO 1 (UP enthält PIO – Programmierung) vor CALL : HL = Anfangsadresse DISPLAY-RAM nach RET : A = 0, wenn keine Taste gedrückt verändert : AF, HL
02F4	UP: Ausgabe eines Datenblocks auf Kassette vor CALL : BC = Anzahl der SYNC-Bits DE = Kennwort des Blocks HL = Anfangsadresse im Speicher nach RET : HL = ADR + 32 verändert : AF, AF', BC, DE, HL, IX
034F	UP: Lesen eines Datenblocks von Kassette vor CALL : A = 0FFH, Synchronisation nur bei Datenblöcken, sonst A = 0 HL = Anfangsadresse im Speicher nach RET : A = 0, wenn kein Lesefehler DE = gelesenes Kennwort HL = ADR + 32 verändert : AF, AF', BC, DE, HL, IX
03BE	Beginn Tastaturkode-Tabelle
03D6	Beginn 7-Segment-Kode-Tabelle
03E6	Beginn Tabelle der Registernamen
0C00...0FC0	Anwender-RAM, wird vom Monitor nicht benutzt
0FC1...0FD6	Monitorstack
0FD7...0FF0	Rettungszellen für Anwender-Register
0FF1...0FF2	Rettungszelle für Haltepunktadresse
0FF3...0FFB	DISPLAY-RAM
0FFC...0FFD	Hilfszellen für GO-Funktion
0FFE	Rettungszelle für Befehlskode
0FFF	Registerzeiger
1000...	1000–FFFF nicht belegt

Literatur

[1] *S. Burkhardt/U. Hübner/A. Troll*; Poly-Computer 880. Bedienhandbuch (Heft 1). VEB Polytechnik und Präzisionsgeräte Karl-Marx-Stadt.

[2] *A. Troll/U. Hübner*; Daten- und Programmabspeicherung auf Heimmagnetbandgeräten. radio fernsehen elektronik 31 (1982) H. 12, S. 796–799.

[3] *H. Kieser/M. Meder*; Mikroprozessortechnik. Berlin 1982.

[4] *H. Bartold/H. Bäurich*; Mikroprozessoren - Mikroelektronische Schaltkreise und ihre Anwendung (Teil 1...3). Band 186...188 der Amateurreihe »electronica«, Berlin 1980.