

MICROCOMPUTER MC 80



Nutzerhandbuch

VEB Elektronik Gera

Hersteller:
VBB Elektronik Gera
Parkstraße 3
Postschließfach III/334
G e r a
DDR 6500

Das vorliegende Handbuch gibt keinerlei Auskunft über Liefermöglichkeiten und beinhaltet keine Verbindlichkeiten für die Produktion.

Die technischen Angaben dieser Beschreibung dienen der Information zum Gerätesystem MC 80 und zu dessen Handhabung. Änderungen im Sinne des wissenschaftlich-technischen Fortschrittes behalten wir uns vor.

Für Informationen, die über den vorliegenden Umfang der Geräte- und Systembeschreibung hinausgehen, sowie auch für Beratungen zu spezifischen Anwenderproblemen, steht Ihnen unsere

Abteilung Applikation (KBA)
Telefon Gera 622-2212

zur Verfügung.

Ausgabe 12/86

Inhaltsverzeichnis

1. Allgemeine Gerätebeschreibung des MC 80 I
- 1.1. Kurzcharakteristik 1...14
- 1.2. Variantenübersicht
- 1.3. Technische Daten
- 1.4. Garantiebedingungen
2. Hinweise zur Aufstellung und zum Inbetriebsetzen II 1..10
- 2.1. Transport und Lagerung
- 2.2. Aufstellung und Inbetriebnahme
- 2.3. Wartung und Pflege des MC 80
- 2.4. Benutzung der geräteinternen EPROM-Löscheinrichtung
- 2.5. Tastatur
- 2.6. Display
- 2.7. Kassettenmagnetbandeinheit
3. Technische Beschreibung der Gerätekonfiguration III 1..16
- 3.1. Struktur des MC 80
- 3.1.1. Blockschaltbild und Systembus
- 3.1.2. Zentrale Recheneinheit
- 3.1.3. Festwertspeicher
- 3.1.4. Schreib-Lese-Speicher
- 3.1.5. Displaylogik (DPL)
- 3.1.6. Bildschirmansteuerung (BSA)
- 3.1.7. Tastatur
- 3.1.8. Magnetbandinterface (MBI)
- 3.1.9. EPROM-Programmiereinheit
- 3.2. Aufteilung des Adreßraumes
- 3.2.1. Speicheradressen
- 3.2.2. Geräteadressen
- 3.3. Hinweise zur Erweiterung und Modifikation des Gerätes
- 3.3.1. Gewährleistung der Gerätefunktion
- 3.3.2. Stromversorgung für zusätzliche Steckeinheiten
- 3.3.3. Rückverdrahtung
- 3.3.4. Prioritätsbestimmende Ketten

4.	Zugriff zu den Funktionsprogrammen des MC 80	IV
4.1.	"TASTENTEST"	1...32
4.2.	"INIT"	
4.3.	"BAND"	
4.4.	"EDIT"	
4.4.1.	Beschreibung des Objektcodeeditors	
4.4.2.	Aufruf des Editors	
4.4.3.	Verlassen des Editors	
4.4.4.	Der Anzeigemodus	
4.4.5.	Der Quellcode-Eingabemodus	
4.4.6.	Binden von Programmen	
4.4.7.	Der Testmodus	
4.5.	"RAM"	
4.6.	"EPROG"	
5.	Softwareschnittstellen im Betriebssystem	V
5.1.	Displayprogramme	1...14
5.2.	Tastaturprogramm	
5.3.	Ändern der Tabulator-Positionen	
5.4.	Benutzung der Graphik des Display	
5.5.	Vereinbaren einer Programmtabelle	VII
6.	Hinweise zur Anwenderprogrammierung	1...14
6.1.	Aufruf von Anwenderprogrammen	
6.2.	Verwendung von NMI und RST	
6.3.	Bildschirm- und Tastaturinterrupt	
6.4.	Beispiel	VII
Anlage 1	: Befehlsliste des Objektcodeeditors des MC 80	1...13
Anlage 2	: GAB- Nachweis	VIII 1...4
Anlage 3	: TKO-Test "TKO 2"	IX 1...6

1. Allgemeine Gerätebeschreibung des MC 80

1.1. Kurzcharakteristik

Ein Gerät der Gerätefamilie MC 80 ist ein mikroprozessor-gesteuertes, speicherprogrammierbares Basisgerät für all-gemeine technische Anwendungen.

Durch entsprechende Modifizierung ist es einsetzbar als

- Steuergerät für Labor- und Prüffeldautomatisierung, für technologische Spezialausrüstungen und industrielle Prozesse
- Test- und Inbetriebnahmeggerät für die Entwicklung, Prüfung und Fehlersuche mikroprozessorgesteuerter Geräte und Anlagen.
- Programmiergerät für die Erstellung der Software und Programmierung auf EPROM.

Das Grundgerät MC 80 enthält einen Rechnerkern auf der Basis des Prozessorschaltkreises U 880 mit residentem (ROM-geladenen) Betriebssystem.

Ein Bildschirm mit alphanumerischer und analoger Anzeige sowie einer alphanumerischen Tastatur dienen der inter-aktiven Kommunikation Mensch-Maschine.

In einer Kassettenmagnetbandeinheit können Anwenderprogramme und- daten gespeichert werden.

Weitere Funktionsgruppen des Gerätes sind die EPROM-Program-miereinheit mit Aufnahmevorrichtung und eine EPROM-Lösch-kammer.

Für die vielfältigen Anwendungsmöglichkeiten sind Erweite-rungen über die im Basisgerät vorhandenen freien Steckplätze möglich. So können z. B. Anschlußsteuerungen für externe Geräte und Zusatzspeicherkarten für Programmerweiterung auf den Systembussteckplätzen sowie weitere Steckeinheiten auf frei verdrahtbaren Steckplätzen untergebracht werden. Diese Modifizierungen erfolgen durch den Anwender nach seiner eigenen Konzeption.

1.2. Variantenübersicht

Die angebotenen Gerätevarianten unterscheiden sich hinsichtlich

- des Vorhandenseins eines Zwischenbodens in der Steckeinheitenaufnahme (Schaffung von 2 x 5 Steckplätzen für Steckeinheiten, 95 x 170 mm²)
- der Speicherausstattung RAM und (EP)ROM
- des Vorhandenseins der EPROM-Programmier- und Löscheinrichtung
- Softwaremodule

Die Bestellbezeichnungen der Varianten sind:

Prozeßsteuergerät	MC 80 . X X / 1 S ...
Gerätetyp _____	
a) Steckeinheitenaufnahme _____	
b) Speicherausstattung _____	
c) Zusatz für EPROM-Programmier- und Löscheinrichtung _____	
d) Programmodule _____	

a) Steckeinheitenaufnahme

1 entfällt

2 ohne Zwischenboden nur für STE 215 x 170 mm²

b) Speicherausstattung (zusätzlich zu ZRE K 2521

oder K 2523 mit 3 bzw. 4 KByte ROM und 1 KByte RAM für Grundbetriebssystem).

Typ	Speicher	freie Steckplätze	
		mit Bus- anschluß	frei ver- drahtbar
MC 80.21	1x16 KByte ROM/EPROM (PFS K 3820) 1x16 KByte dyn.RAM (OPS K 3525)	5(215x170mm ²)	-
MC 80.22*	wie MC 80.21	5(215x170mm ²)	-

* elektronisch gesteuertes Laufwerk

Unter den freien Steckplätzen stehen in Klammern die Formate der Leiterplatten.

c) Für die EPROM-Programmier- und Löscheinrichtung und dem Handler für die EPROM-Programmierung folgt der Zusatz:/1

d) Programmodule

- S 1 Objektcodeeditor U 880
- S 2 Handler für EPROM-Programmierung
- S 3 EPAS 80
- S 4 BASIC 80

Die Auslieferung der Programmmodule erfolgt auf (EP)ROM-Steckeinheiten.

Bei der Bestellung MC 80... ist das Betriebssystem Softwaremodul S0 enthalten und braucht nicht gesondert angegeben zu werden.

Bei einer Bestellung der Gerätevariante MC 80.../1 ist nicht die Angabe des Softwaremoduls S2 notwendig, da es Bestandteil des Zusatzes .../1 ist.

Die Module S1 bis S4 können auch einzeln käuflich erworben werden.

e) Zusatzsteckeinheiten (gehören nicht zum Lieferumfang)

8 KByte EPROM	(95 x 170 mm ²)
4 KByte DRAM mit Schreibsperre	(95 x 170 mm ²)
UNI-Leiterplatte unbestückt	(95 x 170 mm ²)
Adapter-Leiterplatte	(95 x 170 mm ²)

f) Servicedokumentation

Die Servicedokumentation kann vom Anwender käuflich erworben werden.

1.3. Technische Daten

Mikrorechner

Systembus:	nach TGL 37271/01 (Linieninter- face K 1520)
Zentrale Recheneinheit:	ZRE K 2521 (bzw. K 2523)
Prozessortyp:	U 880 D (TGL 26176)
Verarbeitungsbreite:	8 bit parallel
Befehlssatz:	siehe Anlage 1
Adressierungsraum:	64 KByte
E/A-Adreßbereich:	256 Ein-/Ausgabeadressen
Unterbrechungsarten:	- maskierbarer Interrupt - nichtmaskierbarer Interrupt
Refreshsteuerung:	vorhanden
Waitsteuerung:	mit zeitlichen Einschränkungen vorhanden
Systemtaktzyklus:	407 ns \pm 0,1 %
Speicher auf ZRE:	- 3 KByte EPROM/ROM für Betriebs- system mit Adressen 0000H...0FFFH - 1 KByte stat. RAM für Daten des Betriebssystems mit Ad- ressen 0C00H... 0BFFH
Zusatzspeicher:	nach Ausstattungsvariante (siehe 1.2.)

Datensichtanzeige

Bildröhre:	B 31 G 21
Zeichenanordnung:	8 Zeilen zu 32 Zeichen
Zeichenvorrat:	alle Großbuchstaben, Ziffern und Zeichen nach ISO 646
Zeichendarstellung:	nach Zeichengenerator U 402 D BM 510

Kursordarstellung: blinkend
Kurvendarstellung: 4 eindeutige Kurven durch Speicherabruf darstellbar
Helligkeit: einstellbar
Einlaufzeit: max. 5 min

Tastatur

Tastenbauelement: TSH 19F, TGL 38855
Anordnung: 59 Tasten nach TGL 6990
 Interpretationsebenen: - Buchstaben, Ziffern
- Sonderzeichen
- Kommandos
Kursorsteuerung: - 4 Richtungstasten
- 1 Tabulatortaste
Sondertasten: Rücksetzen (rot)
Testlaufunterbrechung/Interrupt (BK)
Kommandoausführung (ENTER)
Rückkehr in vorigen Kommandozustand (OFF)
Anzeige: Lumineszenzdiode (gelb)
für Umschaltung in die Sonderzeichenebene

Kassettenmagnetbandspeicher

empfohlener Datenträger:	Digitalkassette Typ 490 TGL 24522/07
Kapazität:	128 KByte pro Bandseite
Spurenanzahl:	2
Kanalanzahl:	1
Bandgeschwindigkeit:	4,76 cm/s
Transport:	- Bandtransport vorwärts - schneller Bandtransport vorwärts/rückwärts
Umspulzeit:	< 90 s
Bandendabschaltung:	automatisch
Laufwerksteuerung:	Tastenvorauswahl am Laufwerk, programmierbare Ein-/Aus-Schaltung bzw. softwaregesteuert
Aufzeichnungsverfahren:	Phasenmodulation (Richtungstakt- schrift) nach ISO 3407
Bitdichte:	32 bit/mm nach ISO 3407
Bandaufbau:	
- Datenblock:	Präambel, 2 Byte Blocknummer, 128 Byte Daten, 2 Byte CRC Postambel
- Bandmarke:	Präambel, 2 Byte mit Inhalt 00. Postambel
- Namensblock:	wie Datenblock mit der Block- nummer FFFF
- Blockzwischenraum:	nominell 20,3 mm; min 17,8 mm, max. 250 mm

EPR0M-Programmierungseinrichtung

Schaltkreisfassung:	24-pol. Fassung nach TGL 36665 mit Auswurf
Lebensdauer:	200 Steckungen (Goldlegierung)
programmierbarer Schalt- kreistyp:	U 555 C TGL 37787 und kompatible Typen
Betriebsarten:	- Testen - Lesen - Programmieren - Vergleichen
Programmierdauer:	ca. 2 min

EPR0M-Löschereinrichtung

Strahler:	HNS 8
Kapazität der Löschkammer:	2 Schaltkreise
Löschdauer:	ca. 20 min

Software

Betriebssystem:	- 4 KByte ROM und 1 KByte RAM für Tastatur, Bildschirm und Magnetband auf ZRE
Objektcodeeditor:	- 5 KByte ROM - Grundlage Befehlssatz und Mnemonic des U 880 D - umfaßt Assembler, Reassembler Syntaxprüfung und Testpro- gramm
EPR0M-Programmierung:	- 1 KByte ROM

Hilfsenergie

Netzspannung:	220V $\begin{matrix} +10\% \\ -15\% \end{matrix}$
Frequenzbereich:	47...63 Hz
Energiebedarf:	
- Grundausstattung	ca. 100 W ~
- bei max. sekundärseitiger Belastung	ca. 200 W ~

Konstruktive Gestaltung

Gehäuseform:	Auftischgehäuse pultförmig, Tastatur abnehmbar
äußere Abmessungen: (b x h _{max} x t)	577 x 310 x 320 mm ³
Netzanschluß:	Gerätestecker Schutzleiteranschluß TGL 7783

Steckeinheitengefäß

- STE-Formate:	Karteneinschub ungeschützt A1 nach TGL 25068/01 215 mm x 170 mm 95 mm x 170 mm (siehe 1.2.)
- Steckverbinder:	indirekt 58-polig, 13,5 mm breit nach TGL 29331/03
- Steckraaster:	20 mm
Anwendersteckplätze:	siehe 1.2.
Masse:	ca. 25 kg

Einsatz-, Lager- und Transportbedingungen

Einsatzklasse:	ELK 3 nach TGL 26465 (5/40/30/80//10-1 _B) 84-107 kPa
Transportklasse:	TKL 3 nach TGL 26465 (-40/+50/+35/85//12-1 _{LT}) min 84 kPa
Lagerklasse:	LKL 2 nach TGL 26465 (-30/+40/+30/90//12-1 _{LT}) min 84 kPa

Zuverlässigkeit

Betriebszuverlässigkeit

(MTBF): 3000 h

Sicherheitsparameter

Schutzgrad:	IP 20 RGW-St 778
Schutzklasse:	I TGL 21366
Gruppe der Kriech- und Luftstrecken:	3 TGL 16559
Funktentstörgrad:	F1/12 TGL 20885
GAB-Nachweis:	Anlage 2

1.4. Garantiebedingungen

Gemäß § 46 VG wird für das Gerät MC 80 eine gesetzliche Garantiezeit von 6 Monaten gewährt, wobei sich die Garantiebedingungen nach § 45 VG und dem Nutzerhandbuch richten. Gemäß § 94 (4) VG gilt eine Frist zur Mängelbeseitigung, Nachbesserung oder Ersatzleistung innerhalb von 3 Wochen.

2. Hinweise zur Aufstellung und zum Inbetriebsetzen des Gerätes

2.1. Transport und Lagerung

Bei Transport, Umschlag und Lagerung des Gerätes MC 80 sind folgende Grundsätze zu beachten:

- bis zum Einsatzort nur in der Versandpackung transportieren und lagern
- grobe Stöße und Erschütterungen vermeiden
- vor Staub und Feuchtigkeit schützen
- starke Temperaturschwankungen vermeiden
- Lagertemperatur einhalten
- Lagerung 6 Monate nicht überschreiten

Bei Transport des Gerätes sind stets Netzkabel und Tastatur abzunehmen. Das Abnehmen und Anstecken der Tastatur erfolgen auf einer ebenen Tischplatte.

2.2. Aufstellung und Inbetriebnahme

Vor Inbetriebnahme des Gerätes ist die Tastatur anzustecken.

Das Gerät ist über das mitgelieferte Netzkabel mit Schutzkontaktstecker und Gerätekupplung an 220 V Netzspannung mit Schutzmaßnahmen nach TGL 200-602 mit Schutzleiteranschluß anzuschließen.

Die im Gehäuse eingearbeiteten Lüftungsschlitze dürfen nicht verdeckt werden.

Das Gerät ist erst einzuschalten, wenn es die Umgebungstemperatur des entsprechenden Raumes, in dem es betrieben werden soll, angenommen hat.

Um ein ruhigstehendes Schirmbild zu gewährleisten, sind magnetische Störfelder vom Gerät fernzuhalten.

Gegebenenfalls ist die Gerätelage zur Störquelle zu ändern bzw. die Störquelle in geeigneter Weise abzuschirmen. Über den Netzschalter (1) wird das Gerät in Betrieb genommen. Die Bereitschaft des Gerätes wird über die grüne Leuchtdiode (2) angezeigt (siehe Bild 2.1). Ist dies nicht der Fall, so ist zu prüfen, ob das Netz Spannung führt und ob die Gerätesicherung (17) in Ordnung ist (siehe Bild 2.2). Bestätigt die Leuchtdiode (2) die Funktion des Gerätes, so muß nach ca. 20 s eine Display-Ausschrift erscheinen. Geschieht dies nicht, ist die RES-Taste (10) zu bedienen. Die Helligkeit der Bildschirmausschrift läßt sich mit dem Helligkeitsregler (6) R 001 für das Display einstellen.

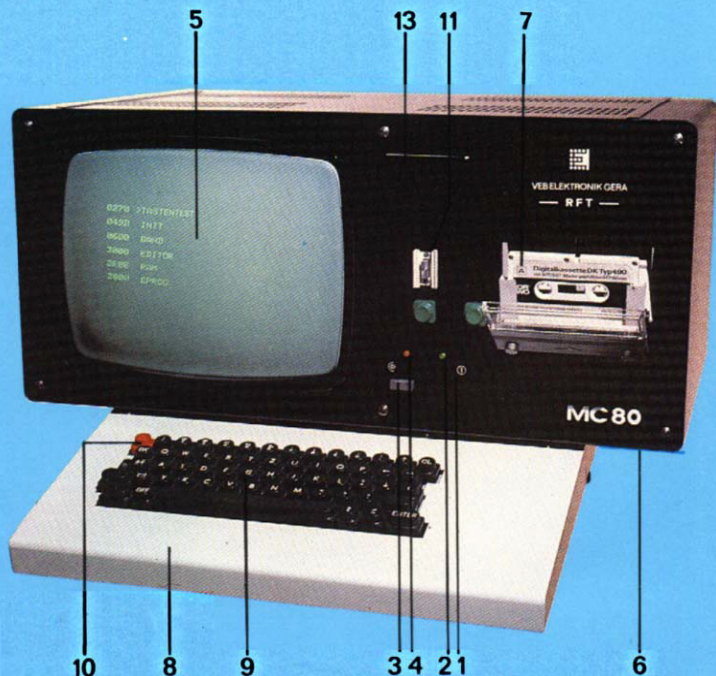
Sicherheitsmaßnahmen

Das Gerät darf vom Anwender nur zum Zwecke des Anschlusses von Zusatzperipherie an den Mikrorechner geöffnet werden. Dazu ist lediglich die Demontage des linken Teiles der Rückwand (von hinten gesehen) gestattet. Um den Schutzgrad IP 20 wieder herzustellen wird empfohlen, in die Rückwand eine Aussparung für den Anschlußsteckverbinder einzuarbeiten. Zum Anschluß der Zusatzperipherie sind Steckverbinder mit Griffschalen nach TGL 29331/08 und Fernmeldeplastschlauchleitungen zu verwenden.

Achtung!

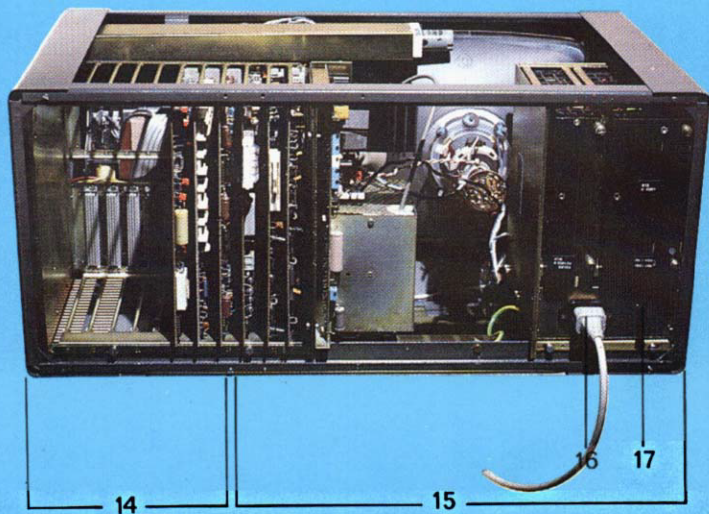
Trotz des Implosionsschutzes der Bildröhre ist diese vor Schlag und Stoß zu schützen. Bei Abnahme der Rückwand und beim Auswechseln der Gerätesicherung muß unbedingt der Netzstecker gezogen sein! Ein Öffnen des Gerätes über die Rückwand hinaus zum Zwecke der Reparatur und Prüfung ist nur geschultem Fachpersonal gestattet. (Es treten Hochspannungen bis zu 12 kV im Gerät auf). Bei Sicherungstausch ist eine Geräteschmelzsicherung 2,0 A träge einzusetzen.

**Bild 2.1 Darstellung des MC 80
mit Bedienelementen**



- 1 Netzschalter Gerät
- 2 Anzeige Betriebsbereitschaft Gerät (grün)
- 3 Netzschalter Löschkammer
- 4 Anzeige Betriebsbereitschaft Löschkammer (gelb)
- 5 Bildschirm für Datensichtanzeige
- 6 Helligkeitsregler
- 7 Kassettenmagnetbandlaufwerk
- 8 Tastatur
- 9 Tastensatz
- 10 Rücksetztaste (rot)
- 11 EPROM-Schaltkreisfassung
- 12 Löschkammeranschluss

Bild 2.2 Rückansicht MC 80



- 14 Rückwand Teil 1
- 15 Rückwand Teil 2
- 16 Gerätestecker
- 17 Sicherungshalter (Feinsicherung T 2,0 A)

2.3. Wartung und Pflege des MC 80

Der MC 80 ist ein wartungsarmes Gerät.

Das Gehäuse ist nur mit nichtaggressiven Spülmitteln zu pflegen.

Die vom Hersteller im MC 80 eingesetzten EPROM's werden mit einem lichtundurchlässigen Klebeband abgedeckt und gekennzeichnet. Dem Anwender wird empfohlen durch diese oder andere geeignete Maßnahmen die EPROM's, die eigene Programme speichern, vor Informationsverlust zu schützen.

Die Reinigung des Bildschirms erfolgt mit Glasreinigungsmittel und einem weichen Tuch im ausgeschalteten Zustand des MC 80. Aus Sicherheitsgründen ist das Bearbeiten und Reinigen der Bildwiedergaberöhre mit spitzen oder scharfen Gegenständen nicht zulässig.

Treten beim Betrieb Störungen auf, ist der MC 80 sofort auszuschalten, um Folgefehler zu vermeiden.

Aus diesem Grund ist bei größerer Staubbeanspruchung jährlich einmal die Bildwiedergaberöhre im Gerät staubfrei zu machen, wobei die einschlägigen ASAO-ABAO-Vorschriften und Punkt 2.2. dieser Dokumentation zu beachten sind.

Im Rahmen des Wartungszyklus ist zu empfehlen, die Kassettenmagnetbaineinheit (MBE), speziell den A/W-Kopf, die Tonwelle und die Gummiandruckrolle bei Verschmutzungen durch Bandabrieb oder nach Transport des Gerätes, jedoch spätestens nach jeweils 100 Betriebsstunden zu reinigen. Die Reinigung erfolgt mit einem Magnetbandservicestab, aber nicht mit harten oder metallischen Gegenständen, bei geöffnetem Kassettenfach und gedrückter Wiedergabetaste. Es empfiehlt sich den Filz des MB-Servicestabes mit Spiritus zu befeuchten.

Der Verwendung von Digitalreinigungskassetten zur Kopfreinigung wird abgeraten. Die Antriebsmechanik des Laufwerkes darf nicht geölt und gefettet werden.

Das Kassettenfach ist aus Gründen des Staubschutzes geschlossen zu halten und die Digitalkassette Typ 490 ist in dem zugehörigen Plastbehälter aufzubewahren, um eine größere Staubeinwirkung auf elektrische und magnetische Funktionsteile (A/W-Kopf, Magnetband) zu verhindern.

2.4. Benutzung der geräteinternen EPROM-Löscheinrichtung
Die Löschkammer wird über Netzschalter (3) eingeschaltet (siehe Bild 2.1). Die Leuchtdiode (4) signalisiert den Zustand "EIN".

Beachte:

Die Löschkammer ist nur bei eingeschaltetem Gerät betreibbar.

Die EPROM-Schaltkreise sind mit ihren Quarzglasfenstern nach oben in den Löschkammereinschub zu legen. Zur Vermeidung des Ozongeruches ist die Löschkammer geschlossen zu betreiben und bei längerer Benutzung für gute Belüftung zu sorgen.

2.5. Tastatur

Die Tastatur besteht aus 59 Tasten, deren Anordnung im Bild 2.3 im Überblick dargestellt ist. Dabei erfolgt die Interpretation der Tasten in drei Ebenen.

1. Ebene: Buchstaben A-Z
 Ziffern 0-9
 Leerzeichen
 Funktionstasten
 Zeichen
2. Ebene: Sonderzeichen (bei Doppelbesetzung der Tasten
 obenstehende Zeichen)
3. Ebene: Steuerzeichen

Das Umschalten in die 2. Ebene geschieht mittels der Umschalttasten (18) links und rechts der Tastatur. Solange eine dieser Tasten gedrückt wird, befindet man sich in der 2. Ebene (analog der Schreibmaschine). Nach Betätigung der Feststelltaste ist die 2. Ebene ständig eingeschaltet. Sie wird nach Betätigung einer der Umschalttasten wieder ausgeschaltet. Einige Tasten haben Sonderfunktionen:

RES (RESET); setzt das Gerät in den Ausgangszustand zurück, die RAM-Inhalte werden nicht verändert, das Betriebssystem wird abgearbeitet.

- BK (BREAK); unterbricht ein Programm, das während der Testung im Echtzeitlauf gestartet wurde oder löst einen vom Anwender speziell initialisierten hochpriorisierten Interrupt aus
- CL (CLEAR); Eingabelöschtaaste
löscht das Zeichen, das vor der aktuellen Schreibposition steht und setzt den Cursor auf die gelöschte Position
- ENTER; Kommandoausführungstaste,
löst die mit der Eingabe verbundene Programmfunktion aus
- OFF; vorheriger Bedienzustand wird wieder eingenommen
- Kursorsteuerung:
- ↑
↓
←
→
┌───┐
└───┘
- Kursor nach oben
- Kursor nach unten
- Kursor nach links
- Kursor nach rechts
- Tabulator; Einnahme der Schreibposition beim nächsten Tabulator. Die Tabulatorpositionen können im Betriebssystem geändert werden (Abschnitt 5.3.).
- CTR; Diese Taste wird in Verbindung mit einer Zeichentaste benutzt.
Das gleichzeitige Betätigen der CTR-Taste und einer Zeichentaste laut Tab. 5.3, Abschnitt 5, erzeugt ein sogenanntes Steuerzeichen (Control-Code). In speziellen Programmen des Betriebssystems (Abschnitt 5.) und in dem Funktionsprogramm "TASTENTEST" (Abschnitt 4.1.) wird diese Ebene wirksam.

In der Grundvariante wird nach "RES" (sowohl nach Tastenbetätigung als auch nach Einschalten) die Betriebssystem-Kommandoabfrage angesprungen. Ist ein spezielles Prozeßsteuerprogramm auf dem Speicher enthalten (siehe Abschnitt 6), so wird dieses abgearbeitet und der zugehörige spezielle Bedienzustand angelaufen. Das ist aber erst nach Anwenderspezifizierung des Gerätes der Fall.

Auch in diesem Fall kann man den Bedienzustand Betriebssystem-Kommandoabfrage erreichen, indem man während des gleichzeitigen Drückens der Umschalttaste "UC" und "OFF" die RES-Taste betätigt. In diesem Fall wird die Abarbeitung des Prozeßsteuerprogrammes verhindert.

Beispiel: Betätigen Sie nach der Taste "RESET" zuerst die "ENTER"-Taste und dann nacheinander alle Zeichentasten in beiden Ebenen und betrachten Sie die entstehende Ausschrift auf dem Display.
Nach Betätigen der "RES"-Taste wird der Bedien- grundzustand wieder eingenommen.

2.6. Display

Das Display ist dialogorientiert und dient sowohl der Programmerstellung als auch der Anzeige und Auswertung von Prozeßdaten.

Auf dem Display können 63 verschiedene Zeichen dargestellt werden. Ein Cursor bestimmt die Displayposition, auf welcher das nächste Zeichen geschrieben werden soll.

Das Display kann direkt von der Tastatur beschrieben werden, es kann auch programmgesteuert vom Prozessor beschrieben und gelesen werden.

Tastatur und Display dienen der Kommunikation Prozessor-Bediener.

Der Bildschirm dient auch zur Darstellung von Informationen in Form von analogen Kurvenzügen. Die Umschaltung geschieht prozessorgesteuert. Für je eine Analogkurve entfällt eine alphanumerische Zeile in der Ausschrift (2., 4., 6., und/ oder 8. Zeile für max. 4 Analogkurven).

Näheres dazu ist in Abschnitt 5.4. aufgeführt.

Bild 2.3 Tastenübersicht



- 10 „RESET“-Taste (rot)
- 18 Umschalttasten für die 2. Ebene (UC)
- 19 Umschalttaste in die Kommandoebene (CTR) –
Taste mit Gefühlspunkt
- 20 Feststeller für 18
- 21 Umschaltanzeige für 18 und 20

Anmerkung

Für folgende auf d. Tastatur angegebene Zeichen erzeugt der Zeichengenerator ein anderes Bild:

Tastensymbol	Display
Ⓐ	Ⓐ
&	%
^	∞
∪	∅

2.7. Kassettenmagnetbandeinheit

Das Kassettenmagnetbandgerät SK 1200 stellt in Zusammenwirken mit der Ansteuertechnik den Massenspeicher des Gerätes MC 80.22 dar. Die Benutzung des Laufwerkes erfolgt programmgesteuert, so daß keine Funktionstasten am Laufwerk erforderlich sind. Die für das Laufwerk nötigen Betriebsspannungen und Steuersignale werden durch das jeweils gewählte Programm aktiviert.

Links neben dem Kassettenfach befindet sich die Drucktaste für das Öffnen des Kassettenauswurfes. Die Betätigung dieser Taste ist jederzeit, also auch bei laufender Kassettenfunktion möglich, da die Taste beim Erreichen des Druckpunktes eventuell laufende Magnetbandprogramme unterbricht und die Kassette zum Auswurf freigibt. Ein Überschreiten des Druckpunktes führt zum Öffnen des Kassettenfaches.

Wird ein laufendes Magnetbandprogramm durch Betätigung des Auswurfes unterbrochen, setzt der Prozessor die Abarbeitung des Programms an der Adresse 0600H fort (Magnetband-Menü). Das Stoppen einer Magnetbandfunktion mittels Auswurf Taste sollte nur in begründeten Ausnahmefällen durchgeführt werden (z. B. fehlerhafter Kassettenlauf). Die verschiedenen Magnetbandfunktionen sind im Pkt. 4.3. erläutert.

3. Technische Beschreibung der Gerätekonfiguration

3.1. Struktur des MC 80

3.1.1. Blockschaltbild und Systembus

Das Blockschaltbild des MC 80 ist im Bild 3.1 dargestellt. Der Systembus (Tab. 3.1) ist nach TGL 37271/01 ausgeführt und kompatibel zum Mikrorechnersystem K 1520.

C	Kontakt	A
5P	29	5P
12P	28	12P
/BAI	27	/BAO
/HALT	26	/M1
/RDY	25	/RFSH
/IORQ	24	/WAIT
/INT	23	/NMI
∅∅	22	/IODI
∅∅	21	TAKT
/BUSRQ	20	/RESET
AB 1	19	AB ∅
AB 3	18	AB 2
AB 5	17	AB 4
AB 7	16	AB 6
5N	15	5N
AB 9	14	AB 8
AB11	13	AB10
AB13	12	AB12
AB15	11	AB14
/IEI	10	/IEO
/MEMDI	9	/NREQ
/RD	8	/WR
DB ∅	7	DB 1
DB 2	6	DB 3
DB 4	5	DB 5
DB 6	4	DB 7
5PG	3	5PG
∅∅	2	∅∅
∅∅	1	∅∅

Tab. 3.1: Systembus des MC 80

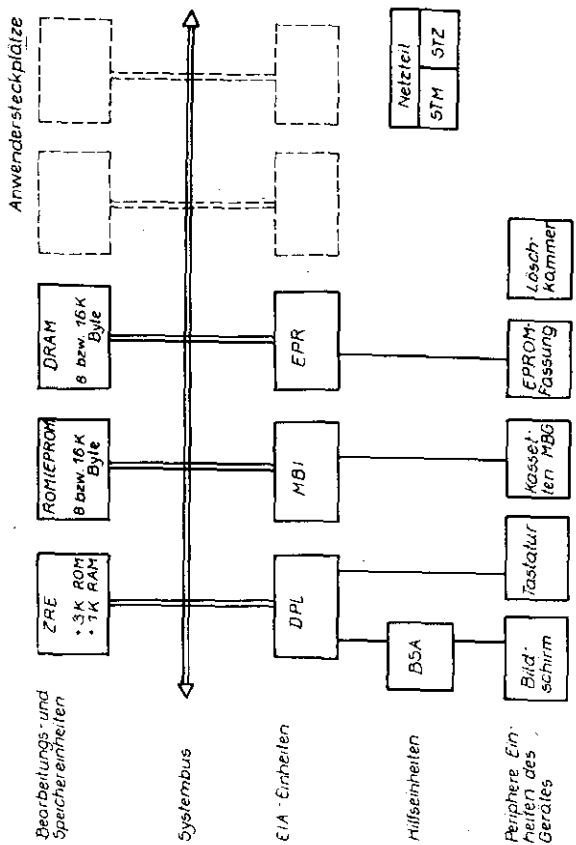


Bild 3.1 : Blockschaftbild

3.1.2. Zentrale Recheneinheit

Die Zentrale Recheneinheit ZRE ist als Steckereinheit K 2521 oder K 2523 im Mikrorechner bestückt und enthält die Schaltungsteile

Mikroprozessor
Takterzeugung
Rücksetzschaltung

Speicher: 3 KByte ROM/EPROM für Betriebssystem
1 KByte stat. RAM u.a. für Bildwiederholungspeicher

Parallel-Ein-/Ausgabe (PIO) zur Verwendung durch den Anwender
Zähler/Zeitgeber (CTC nur auf K 2521) zur Verwendung durch den Anwender

und die erforderlichen Bustreiber.

Die Signalbelegung auf dem Steckverbinder X2 für den Koppelbus ist der Tab. 3.2 zu entnehmen.

C		Kontakt	A
00		29,28	00
-		27,26	-
CTC-CLK/TRG 0		25	CTC-ZC/TO 0
-CLK/TRG 1		24	-ZC/TO 1
-CLK/TRG 2		23	-ZC/TO 2
-		22	-CLK/TRG 3
/MEMDI 2		21	/MEMDI 1
-		20,19,18	-
PIO-/ASTB		17	PIO-/BSTB
PIO-A 1		16	PIO-A 0
-A 3		15	-A 2
-A 5		14	-A 4
-A 7		13	-A 6
-ARDY		12	-BRDY
-B 1		11	-B 0
-B 3		10	-B 2
-B 5		9	-B 4
-B 7		8	-B 6
/IEP	2)	7	UM
/PRDY	1)	6	/PSTB
/RESET	1)	5	00
TAKT 0	1)	4	00
-		3,2	-
5P		1	5P

- 1) Verwendung bei Mehrrechnerkopplung
- 2) Signal für externe Beschleunigungsschaltung der Interruptprioritätenkette

Tab. 3.2: Belegung des Koppelbussteckverbinders X2 der ZRE

Für den MC 80 sind auf der ZRE folgende Wickelverbindungen durchgeführt:

X 6 - X 7	Takt auf Koppelbus
X8:1-X9:1	MEMDI
X14:2-X15:2	IBI in Prioritätskette eingeordnet.

Die Speicherschaltkreise (U 555 oder U 505) des Betriebssystems S0 sind nach Bild 3.2 angeordnet.

Adressen: S0.1	0000...03FF
S0.2	0400...07FF
S0.3	0800...0BFF

Nähere Angaben können der Betriebsdokumentation des K 1520 entnommen werden.

3.1.3. Festwertspeicher

Als Festwertspeicher werden entweder die Steckereinheit PFS K 3820 mit 16 KByte Kapazität oder die Steckereinheit 8 KByte ROM bestückt.

Die Speicherschaltkreise (U 555 bzw. U 505) sind auswechselbar auf 24-polige Steckfassungen gesetzt. Auf der Steckereinheit PFS sind Brücken gewickelt für

X8:2-X9:2	Adreßbereich 2000H...5FFH
X6:1-X7:1	MEMDI und
X10 -X11	WAIT-Unterdrückung (darf nicht gelöst werden)

Die Anordnung der Schaltkreise im gewickelten Adreßbereich zeigt Bild 3.3.

Weitere Angaben zur Steckereinheit PFS K 3820 sind der Betriebsdokumentation des K 1520 zu entnehmen.

Die Steckereinheit 8 KByte ROM ist durch entsprechende Brücken (X1:2-X2:2; X3:1-X4:1; X5:1-X6:1) für den Speicherbereich 2000H bis 3FFFH adressiert. Für diesen Adreßbereich sind die Anfangsadressen der Speicherschaltkreise in Bild 3.4 dargestellt.

2000-3FFF	X1:2-X2:2	X3:1-X4:1	X5:1-X6:1
4000-5FFF	X1:1-X2:1	X3:2-X4:2	X5:1-X6:1
6000-7FFF	X1:2-X2:2	X3:2-X4:2	X5:1-X6:1
8000-9FFF	X1:1-X2:1	X3:1-X4:1	X5:2-X6:2
A000-BFFF	X1:2-X2:2	X3:1-X4:1	X5:2-X6:2
C000-DFFF	X1:1-X2:1	X3:2-X4:2	X5:2-X6:2
E000-FFFF	X1:2-X2:2	X3:2-X4:2	X5:2-X6:2

Tab. 3.3: Brücken für die hexadezimalen Adreßbereiche der Steckeinheit 8 KByte ROM

Soll die Speicherkapazität für einen anderen Adreßbereich eingesetzt werden, so sind die Brücken gemäß Tabelle 3.3 einzusetzen. Die Lage der Brücken ist ebenfalls aus Bild 3.4 zu ersehen. Dabei ändern sich die Anfangsadressen der Speicherschaltkreise in Bild 3.4 entsprechend.

3.1.4. Schreib-Lese-Speicher

Als Schreib-Lese-Speicher werden entsprechend der Variante entweder eine Steckeinheit OPS K 3525 oder zwei Steckeinheiten 4 KByte dyn. RAM eingesetzt. Die Steckeinheit OPS K 3525 ist ein dynamischer RAM mit 16 KByte Kapazität, realisiert mit den Schaltkreisen K 565, RU 1. Auf dieser Steckeinheit sind Wickelbrücken durchgeführt für

X8:3-X9:3) Adreßbereich C000...FFFFH
X8:4-X9:4	
X6:1-X7:1	MEMDI
X11 -X12	WAIT Unterdrückung (darf nicht gelöst werden)

Weitere Angaben sind der Betriebsdokumentation des K 1520 zu entnehmen.

Die Steckeinheiten 4 KByte dyn. RAM enthalten ebenfalls dynamische RAM-Schaltkreise K 565 RU 1 und sind darüber hinaus mit einer Schreibsperre ausgestattet.

	X1-X2	X3-X4	X5-X6	X7-X8
1000-1FFF	:2-:2	:2-:2	:2-:2	:1-:1
2000-2FFF	:2-:2	:2-:2	:1-:1	:2-:2
3000-3FFF	:2-:2	:2-:2	:1-:1	:1-:1
4000-4FFF	:2-:2	:1-:1	:2-:2	:2-:2
5000-5FFF	:2-:2	:1-:1	:2-:2	:1-:1
6000-6FFF	:2-:2	:1-:1	:1-:1	:2-:2
...				
F000-FFFF	:1-:1	:1-:1	:1-:1	:1-:1

Tab. 3.4: Brücken für die Adreßbereiche

Im ausgelieferten Gerät sind die beiden Steckeinheiten mit Brücken für die Adreßbereiche E000H...EFFFH bzw. F000H...FFFFH versehen. Die Lage der Lötbrücken ist aus Bild 3.5 zu entnehmen.

Die Schreibsperre wird durch einen U 202 realisiert. Sie kann für 1 KByte Bereiche das Speicherschreibsignal WR freigeben oder sperren. Die Programmierung der Schreibsperre erfolgt mit OUT-Befehlen an die Peripherieadressen

01 für Freigabe

02 für Sperren,

wobei gleichzeitig auf den höherwertigen Adreßbus die Belegung des ausgewählten 1 KByte-Bereiches auszugeben ist.

Die Belegung des Datenbusses ist beliebig.

Mit dem Start des Betriebssystems werden zunächst alle Speicherbereiche zum Schreiben freigegeben.

Programmierbeispiel:

Der Bereich 8400H...87FFH ist zu sperren und der Bereich 8000H...83FFH ist freizugeben.

LD A, 84H

OUT 002H

LD A, 80H

OUT 001H

Der Inhalt des A-Registers wird während des OUT-Befehles auf A₈...A₁₅ gegeben.

Die Abfrage der Schreibsperre auf den aktuellen Programmierstand ist nicht zugelassen. Bei Verwendung eines IN-Befehles erfolgt eine fehlerhafte Programmierung der Schreibsperre.

3.1.5. Displaylogik (DPL)

Die Steckeinheit Displaylogik (DPL) hat einen festen Platz im Mikrorechner und darf nicht gezogen werden. Sie ist über einen 26-poligen Steckverbinder mit der Tastatur und über einen 15-poligen Steckverbinder mit den Zusatzspannungen $\pm 15V$ des Netzteiles und mit der Steckeinheit Bildschirmsteuerung (BSA) verbunden.

Von Steckeinheit DPL werden die Signale für X-Auslenkung, Y-Auslenkung und Helligkeitssteuerung und für die Ansteuerung der Tastenmatrix erzeugt. Über die DPL können nach einem oszillographischen Prinzip auf dem Bildschirm 8 Zeilen mit je 32 Zeichen oder bis zu 4 eindeutige Analogkurven dargestellt werden. Die Strahlablenkung entspricht nicht den Normen der Fernsehtechnik. Während einer X-Auslenkung wird eine Zeichenzeile oder eine Analogkurve vollständig geschrieben. Eine Analogkurve wird im Austausch gegen eine geradzahlige Zeichenzeile dargestellt.

Die Zeicheninformation wird aus dem Bildwiederholpeicher (Teil des RAM auf der ZRE) geholt, während die y-Werte der Analogkurven über OTIR-Befehle aus einem beliebigen, zusammenhängenden 1/4 KByte-Bereich des Speichers ausgegeben und fortlaufend dargestellt werden.

Die Ausgabe der Zeichen erfolgt per Interrupt in Abständen von 64,8 μs . Die Interruptserviceroutine benötigt zur Ausgabe eines Zeichens 27,6 μs . In den dazwischenliegenden interruptfreien Zeiten, die bei Zeilen- bzw. Bildende größer sind, werden andere Programme, wie z. B. Anwender- oder Magnetbandprogramme bearbeitet. Der Interrupt hat die höchste Priorität. Nach jedem Zeilenende wird die Tastatur abgefragt, indem eine Zeilenbelegung an die Tastenmatrix ausgegeben und die Spaltenbelegung eingelesen wird.

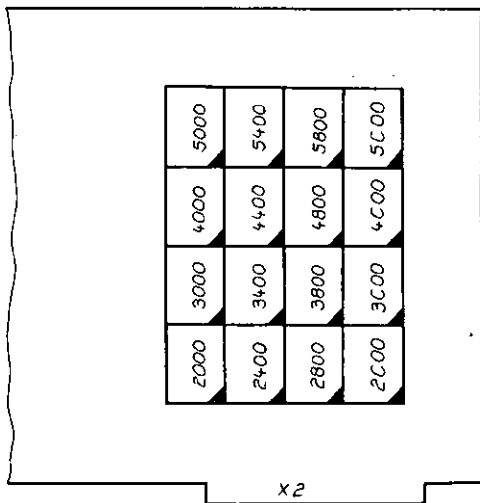


Bild 3 3: Anfangsadressen der Speicher-
schaltkreise der STE PF5

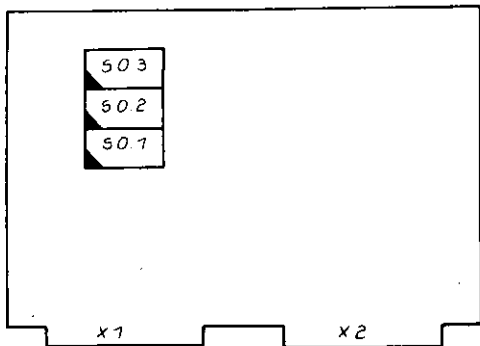


Bild 3 2: ZRE - Speicherbestückung

Auf der STE DPL wird außerdem ein weiterer Interrupt für BREAK (Unterbrechung im Testmodus) erzeugt. Siehe hierzu auch Abschnitt 6.1.

3.1.6. Bildschirmsteuerung (BSA)

Die Steckeinheit BSA ist neben dem Steckeinheitengefäß angebracht. Sie stellt alle für das Betreiben der Bildwiedergaberöhre erforderlichen Betriebsspannungen (Hochspannung, Heizspannung, Gitterspannung) zur Verfügung und verstärkt die Helligkeitssignale für Zeichen- und Kurvendarstellung. Außerdem wird durch die BSA die Ablenkeinheit für die horizontale und vertikale Richtung angesteuert.

3.1.7. Tastatur

Als Tasten finden die kontaktlosen Tasten TSH 19F Anwendung, die auf Grund einer induzierten Hallspannung schalten. Die Tasten sind in einer 8 x 7 Matrix angeordnet. Von der roten Taste (10) wird ein Rücksetzen des Mikrorechners ausgelöst. Der Rücksetzimpuls wird zeitlich durch ein Monoflop begrenzt, damit das Auffrischen des dynamischen RAM-Speichers gewährleistet bleibt. Über die Taste "BK" wird der BREAK-Interrupt im Rahmen des Testmodus auf der Steckeinheit DPL ausgelöst. Die Ebenenumschaltung wirkt auf ein RS-Flip-Flop, dessen Zustand über eine gelbe Lumineszenzdiode angezeigt wird (Leuchtzustand: Obere Ebene). Mit der Controll-Taste (19) wird in eine 3. Ebene umgeschaltet, die für Steueroperationen (Kommandos) nutzbar ist.

3.1.8. Magnetbandinterface (MBI)

Die Steckeinheit Magnetbandinterface (MBI) hat einen festen Platz im Mikrorechner und darf nicht gezogen werden. Sie ist über einen 26-poligen Steckverbinder mit dem Kassettenlaufwerk und über einen 15-poligen Steckverbinder mit den Zusatzspannungen ± 15 V des Netzteiles verbunden. Die Steckeinheit MBI realisiert die Kommunikation des Rechners mit der Kassettenmagnetbandeinheit.

Sie übernimmt die Wandlung der vom Rechner auf Magnetband zu speichernden Daten in das entsprechende Aufzeichnungsformat (pulsdauermodulierte Richtungstaktschrift mit einer Informationsdichte von 32 Bit/mm nach Standard ISO 3407) bzw. die Rückgewinnung der auf Magnetband gespeicherten Informationen.

Die Steckeinheit MBI ist nur mit der dazugehörigen Software funktionsfähig, entsprechende Bedienhinweise sind im Punkt 4.3. gegeben.

Die MBI besteht aus folgenden Funktionsgruppen:

- programmierbare E/A-Schnittstelle
- programmierbare Zähler/Zeitbasisgeber
- programmierbare Laufwerksteuerung
- Adressierungseinrichtung
- Aufnahme/Wiedergabesteuerung

Der Datenaustausch zwischen CPU und MBI erfolgt über die E/A-Schnittstelle interruptgesteuert und in Abhängigkeit von der auf der MBI gebildeten Zeitbasis.

Als Kassettenlaufwerk wird ein modifiziertes Zweimotoren-Laufwerk (SK 1200) genutzt. Die Bandgeschwindigkeit beträgt 4.76 cm/sec.

3.1.9. EPROM-Programmiereinheit

Diese Steckeinheit hat ebenfalls, wie die STE DPL und MBI, einen festen Platz und darf nicht gezogen werden.

Über 3-Ausgabe- und 1 Eingaberegister werden die Signale und Spannungen für den zu behandelnden EPROM bereitgestellt bzw. Daten eingelesen. Die erforderlichen Spannungen werden über Relais zugeschaltet.

Der gesamte Ablauf der EPROM-Programmierung erfolgt programmgesteuert.

3.2. Aufteilung des Adreßraumes

3.2.1. Speicheradressen

Adreßbereich (hex.)	Speicher- typ	Verwendung
0000-03FF	(EP)ROM	Betriebssystem S0.1
0400-07FF	(EP)ROM	Betriebssystem S0.2
0800-0BFF	(EP)ROM	Betriebssystem S0.3
0C00-0CFF	SRAM	Bildwiederholtspeicher
0D00-0D0F	SRAM	Arbeitszellen des Betriebssystems
0D10-0D17	SRAM	Startadressen-Graphik
0D18-0D1B	SRAM	Tabulatorpositionen
0D1C-0D2B	SRAM	Arbeitszellen des Betriebssystems
0D2C-0DEF	SRAM	Interruptstartadressen Anwender
0DF0-0E9F	SRAM	Arbeitszellen des Magnetbandprogrammes
0F00-0FFF	SRAM	Stack-Pointer-Ebenen
2000-23FF	(EP)ROM	Betriebssystem S0.4
2400-27FF	(EP)ROM	Freibereich für Copy-Modus und Editorerweiterungen
2800-2BFF	(EP)ROM	Handler für EPROM-Programmierung S2
2C00-2FFF	(EP)ROM	Testmodus des Editors S1.5
3000-33FF	(EP)ROM	Editor S1.1
3400-37FF	(EP)ROM	Editor S1.2
3800-3BFF	(EP)ROM	Editor S1.3
3C00-3FFF	(EP)ROM	Editor S1.4
4000-5FFF	(EP)ROM	BPAS/BASEX
6000-BFFF	beliebig	Freibereich für Anwender
C000-EFFF	DRAM	Arbeitsspeicher für Anwender
F000-F01F	DRAM	Editor-Arbeitszellen (vorzugsweise)
F020-FXXX	DRAM	Markentabelle (vorzugsweise)
FXXX-FFFF	DRAM	Menü-Tabelle

3.2.2. Geräteadressen

Adresse (hex.)	Verwendung
00	intern belegt
01	Schreib-Freigabe für RAM
02	Schreibsperre für RAM
03-0F	intern belegt
10-7F	Freibereich für Anwender
80	ZRE: CTC-Kanal 0
81	ZRE: CTC-Kanal 1
82	ZRE: CTC-Kanal 2
83	ZRE: CTC-Kanal 3
84	ZRE: PIO Port A, Daten
85	ZRE: PIO Port B, Daten
86	ZRE: PIO Port A, Steuerung
87	ZRE: PIO Port B, Steuerung
88-8F	intern belegt
90-DF	Freibereich für Anwender
E0-FF	intern belegt

3.3. Hinweise zur Erweiterung und Modifizierung des Gerätes

3.3.1. Gewährleistung der Gerätefunktion

Da die CPU U 880 auf der Steckereinheit ZRE das laufende Beschreiben des Bildschirms mit Informationen aus dem Bildwiederholungspeicher, als Teil des Arbeitsspeichers, vornimmt, muß die Arbeitsweise der CPU ständig aufrechterhalten werden. Aus diesem Grund müssen vermieden werden:

- Änderung des Interruptmodus und des I-Registers
- Programmierung des Interruptverbotes (Befehl DI) oder Interruptserviceroutine ohne BI an deren Anfang)
- lange WAIT-Zyklen
(Es sollten höchstens für E/A-Zyklen WAIT-Signale mit einer maximalen Dauer von 10 µs erzeugt werden).

In den genannten Fällen ist mit dem Ausfall der Bildschirmanzeige zu rechnen.

Um Einbrennstrukturen in den Bildschirm zu vermeiden, sind Anzeigen nicht länger als nötig stehen zu lassen.

So wird empfohlen, in der Anwendung als Prozeßsteuergerät den Bildschirm dunkel zu steuern, wenn keine Bedienung durch den Menschen erfolgt.

Die Dunkeltastung erfolgt durch Eintragung von Leerinformation in den Bildwiederholungspeicher (siehe 5.1., Unterprogramm BSA). Durch diese Maßnahme kann die Lebensdauer der Bildröhre verlängert werden.

3.3.2. Stromversorgung für zusätzliche Steckeinheiten

Von der Stromversorgungseinheit werden bei primärseitiger Netzeinspeisung von 220V $\begin{matrix} +10\% \\ -15\% \end{matrix}$ sekundärseitig die Spannungen 5P, 5N, 12P, 15N, 15P bereitgestellt. Diese Ausgangsspannungen werden von den Schaltnetzteilen, STM K0361.03 und STZ K0367-5/+12/-15+15, erzeugt.

Durch das Grundgerät wird im begrenzten Umfang die durch die Stromversorgung bereitgestellte Leistung (P_V max) durch Eigenbedarf verbraucht.

Folgende Übersicht soll die Reserven der Stromversorgung des MC 80 für den Anwender darstellen.

Nennspannung	max. Ausgangs-nennstrom	min. Stromreserve	Bemerkung
5P	9A	3,7A	im Koppelbetrieb
5N	1A	0	
12P	2A	0	
15N	1,5A	0	
15P	1,5A	0	

Tabelle 3.5: Reserven der Stromversorgung des MC 80

Bei der Angabe der min. Stromreserve der Stromversorgungseinheit wurde als Grundlage der maximale Stromverbrauch geräteeigener Steckeinheiten verwendet. Es besteht somit in den meisten Fällen die Möglichkeit, eine weitere Speichersteckeinheit OPS/K 3525 oder PFS K 3820 einzusetzen.

Wird der maximal zulässige Ausgangsnennstrom einer Ausgangsspannung des STM oder STZ überschritten, sinkt die Ausgangsspannung ab.

Etwa 5s nach Erreichen dieser Unterspannung erfolgt eine irreversible Abschaltung aller Ausgangsspannungen. Ein erneutes Einschalten des Gerätes ist ca. 5s nach Ausschalten durch Betätigung des Netzschalters möglich.

3.3.3. Rückverdrahtung

Die Rückverdrahtungsleiterplatte entspricht in ihrer Konfiguration und Anschlüssen dem Systembus nach TGL 37271/01. Dieser Systembus ist vom Anwender nicht zu ändern. Die Zuführung der Spannungen von der Stromversorgungseinheit zu den Steckeinheiten erfolgt bei 5P, 5N, 12P, 5PG über den Systembus und bei 15N und 15P für die systemeigenen Steckeinheiten über 15-polige Steckverbinder.

Den geräteeigenen Anschlußsteuerungen DPL, MBI und EPR(PPE) sind 3 feste Plätze (9,10,11) zugeordnet. Die Belegung der Steckplätze 1 bis 8 ist für den Anwender variabel.

Es besteht die Möglichkeit, auf den variablen Steckplätzen einen beliebigen Koppelbus zu realisieren, wobei die freien Steckverbinderplätze mit Buchsenleisten 58-polig, Form 424 (Lötanschlüsse) oder 434 (Wickelanschlüsse) zu bestücken und zu verdrahten sind, um die gewünschten Koppelbedingungen zwischen den Steckeinheiten zu realisieren.

Um diese Arbeiten ausführen zu können, ist der Ausbau der Magnetbandeinheit notwendig.

Bei den Varianten MC 80.1X besteht außerdem die Möglichkeit, durch den eingebauten Zwischenboden in der oberen Ebene einen beliebigen Systembus zu realisieren (Mehrprozessorsystem oder Busverlängerung).

Bei Busverlängerung ist die Einhaltung der Signallastbedingungen nach TGL 37271/01 zu beachten. Dies gilt auch für alle vom Anwender entworfenen und realisierten Zusatzsteckeinheiten.

3.3.4. Prioritätsbestimmende Ketten

Auf dem Systembus sind zwei prioritätsbestimmende Ketten für

Interrupt: IBI-IEO und
Busanforderung: BAI-BAO

derart realisiert, daß die höchste Priorität von hinten gesehen auf dem rechten Steckplatz vorliegt und nach links abnimmt (11...1). Von den Steckeinheiten DPL und MBI müssen die beiden höchstpriorisierten Steckplätze (11und10) eingenommen werden.

Vom Anwender können 3 Interruptquellen zu den geräteinternen problemlos bezüglich der Durchschaltzeit vom ersten IBI bis zum letzten IEO ergänzt werden.

Falls mehr Interruptteilnehmer benötigt werden, sind zusätzliche Schaltungsmaßnahmen zur Signalbeschleunigung erforderlich (siehe Technische Beschreibung U 855 und 857 vom VEB Funkwerk Erfurt). Es ist stets darauf zu achten, daß die prioritätsbestimmenden Ketten durchgängig sind.

Ist für ein Anwenderprogramm ein Interrupt des CTC bzw. der PIO auf der Steckeinheit ZRE erforderlich, so ist die ZRE entsprechend in die Interruptkette einzuordnen. Sie darf nicht für höchste Interruptpriorität gewickelt werden (siehe 3.1.2.).

4. Zugriff zu den Funktionsprogrammen des MC 80

Das Betriebssystem des MC 80 ist hauptspeicherorientiert. Das bedeutet, daß der MC 80 zur Ausführung der Betriebssystemfunktionen keinen Magnetbandzugriff benötigt. Sind die Anwenderprogramme im Hauptspeicher (auf ROM oder CMOS-RAM) enthalten, so wird auch zum Start und Lauf der Anwenderprogramme das Magnetband nicht benötigt. Die Arbeit mit dem Magnetband könnte sich darauf beschränken, nur Anwenderdaten (z.B. Prozeßüberwachung und -protokollierung) aufzuzeichnen. Jedoch können auch Anwenderprogramme auf dem Magnetband gespeichert, eingelesen und gestartet werden. Die Hauptspeicherorientierung ermöglicht die sofortige Betriebsbereitschaft des Gerätes nach Einschalten, "RESET" oder kurzzeitigem Netzausfall.

Das Betriebssystem des MC 80 unterstützt die Arbeit der Anwenderprogramme mit Display, Tastatur und Magnetband durch Bereitstellung geeigneter Softwareschnittstellen, ermöglicht das Laden von Anwenderprogrammen vom Magnetband auf beliebige RAM-Bereiche, eine RAM-Initialisierung sowie den Start aller Programme, die sich im Hauptspeicher befinden (resident auf ROM-Karten oder im CMOS-RAM oder geladen vom Magnetband in den RAM) und in Programmtabellen vereinbart sind.

Nach "RESET" meldet sich der MC 80 bei nicht vorhandenem Prozeßsteuerprogramm (Abschnitt 6) mit der Darstellung der Programmtabelle (MENU) auf dem Display, z.B.

```
0270> TASTENTEST
049D INIT
0600 BAND
3000 EDIT
2EBE RAM
2800 EPROG
```

Bild 4.1: Programmtabellenausschrift

Links in jeder Displayzeile stehen die hexadezimalen Startadressen der Programme. Es folgt der Programmname. Das Zeichen ">" vor dem Programmnamen markiert das angewählte Programm. Dieses Zeichen kann mittels der Tasten "↑", "↓" nach oben bzw. unten bewegt werden, somit ist eine Programmanwahl möglich. Nach Betätigung der Taste "ENTER" wird dieses Programm aufgerufen und läuft eigenständig ab. Umfaßt die Programmtabelle mehr als 8 Programme, so kann die nächste Seite (nächste 8 Programme) durch Betätigung der Taste "N" angezeigt werden. Das gilt auch für weitere Seiten. Die Zahl der anwählbaren Programme ist also nicht begrenzt. Nach Betätigung der Taste "OFF" springt die Anzeige zum Anfang der Programmtabelle.

Wird das Markierungszeichen ">" an das Ende der Programmtabelle bewegt, an dem kein Name mit Startadresse mehr angezeigt wird, und "ENTER" betätigt, so kann nach Angabe einer beliebigen Startadresse (hexadezimal mit 4 Ziffern) und nochmaliger "ENTER"-Betätigung diese Startadresse im Hauptspeicher aufgerufen werden. Das ermöglicht dem Anwender auch den Start von Programmen, die sich im Hauptspeicher befinden, jedoch in einer Programmtabelle nicht vereinbart sind. Diese Möglichkeit sollte man jedoch nur bei Notwendigkeit gebrauchen, da Fehleingaben zum Löschen wichtiger RAM-Bereiche führen können. Wie man eigene Programme mit Namen in der Programmtabelle vereinbaren kann, ist im Abschnitt 5.5. beschrieben.

Der Inhalt der Programmtabelle hängt von der Konfiguration der gesteckten ROM-Karten und den in den RAM geladenen Programmen ab. Wird z.B. die ROM-Karte "Objektcodeeditor" nicht gesteckt und statt dessen eine Anwender-EPROM-Karte mit vereinbarten Anwenderprogrammen eingefügt, so erscheinen statt dessen diese Anwenderprogramme in der Programmausschrift. Eine freie Ausstattung des Gerätes MC 80 mit Anwenderprogrammen wird damit unterstützt.

ROM- und RAM-Bereiche befinden sich auf folgenden Adressen

(hexadezimal):

0000...0BFF	} Betriebssystem	} ROM-Bereich 1 (8 kByte)
2000...23FF		
2400...27FF	} frei	
2800...2BFF	} EPROM-Programmierung	
2C00...3FFF	} Objektcodeeditor	
4000...5FFF	} ROM-Bereich 2 (8 KByte)	
6000...7FFF	Prozeßsteuerprogramm (ROM/RAM)	
C000...FFFF	RAM (16 KByte) oder	
8000...FFFF	RAM (32 KByte)	

Tab. 4.1: Speicherbereichaufteilung

Der ROM-Bereich 2400H...27FFH ist für die Anschlußsteuerung Drucker und Lochbandperipherie und der ROM-Bereich 2 für die problemorientierten Sprachen EPAS/BASEX vorgesehen.

<u>Taste</u>	<u>Funktion</u>
↓	Anwahl folgendes Programm
↑	Anwahl vorheriges Programm
ENTER	Bestätigung der Auswahl, Programmstart
N	Anwahl nächste Seite
OFF	Herstellen des Grundzustandes

Tab. 4.2: Bedienkommandos im Betriebssystem

Beispiel: Wählen Sie nacheinander durch das Zeichen "↓" alle Programme an und überprüfen Sie, ob das jeweilige Programm angesprochen wurde. Nach jeder Programmauswahl muß "ENTER" betätigt werden. Durch "OFF" erfolgt der Rücksprung ins "MENÜ".

4.1. "TASTENTEST"

Nach "RESET" oder Einschalten des Gerätes ist dieses Programm angewählt. Es muß nur die Taste "ENTER" betätigt werden und das Programm "TASTENTEST" wird aus dem Betriebssystem aufgerufen.

Bei dieser Programmanwahl kann das Display direkt mit der Tastatur beschrieben werden. Der Bediener kann sich von der Funktionstüchtigkeit der einzelnen Tasten (mit Ausnahme der Taste "EK") überzeugen.

4.2. "INIT"

Nach dem Einschalten des Gerätes ist der RAM mit zufälligen Daten belegt. Da das Betriebssystem (Programmtabelle) sowie auch andere Programme von einer definierten RAM-Belegung ausgehen, muß der RAM-Bereich nach dem Einschalten mit den Daten FFH belegt werden. Der Speicherbereich ab F020 wird für den Aufbau der Markentabelle genutzt. Das erste Byte der Markentabelle ist mit 00 zu belegen. Diese beiden Funktionen realisiert das Programm "INIT".

Wenn der Prozessor beim Testen unbekannte oder unzulässige Programmteile abarbeitet und die fehlerfreie RAM-Belegung in Frage gestellt ist, muß "INIT" aufgerufen werden und sämtliche im RAM befindlich gewesenen Programme müssen z.B. vom Magnetband neu geladen werden.

4.3. "BAND"

Mit Hilfe der im MC 80 enthaltenen Kassettenmagnetbandeinheit kann der Nutzer auf RAM bzw. ROM laufende Systemprogramme bzw. Daten von der Magnetbandkassette einlesen oder eigene erstellte Programme bzw. Daten auslagern, um sie später erneut einzulesen.

Die Bedienung der Magnetbandeinheit ist sowohl dialogorientiert aus dem Betriebssystem heraus (4.3.1.) als auch direkt durch den Prozessor initiiert (4.3.2. Unterprogrammtechnik) möglich. Die Aufzeichnung der Information erfolgt durch phasenmodulierte Richtungstaktschrift mit einer Informationsdichte von 32 Bit/mm. Vom Betriebssystem des MC 80 wird folgende Datenformatierung benutzt:

Es werden 80H Informationsbytes zu einem Block zusammengefaßt. Jedem Block werden eine Präambel (AAH) und eine Blocknummer (vom Bandanfang beginnend, fortlaufend 2 Byte) vorangestellt sowie eine Prüfsumme (CRC16, 2 Byte) und eine Postambel (AAH) nachgesetzt, so daß ein Datenblock aus insgesamt 86H Bytes besteht. Die Datenblöcke sind untereinander durch Blocklücken getrennt (konstante Magnetisierungsrichtung). Die in einem Aufzeichnungsvorgang ausgegebene Menge von Datenblöcken (Datei) wird durch einen vorangestellten Namensblock (ohne Blocknummer, Kennung PFFPH) gekennzeichnet. Die Dateien sind untereinander durch Bandmarken getrennt (Bytefolge: AA 00 00 AA).

Da der Informationsaufbau des Bandes auf einer fortlaufenden Numerierung der Datenblöcke basiert und kein Inhaltsverzeichnis des Bandes im Band selbst angelegt wird, ist es für den Nutzer unbedingt erforderlich, eine Liste der benutzten Blocknummern zu führen (für weitere Aufzeichnungen muß die letzte benutzte Blocknummer bekannt sein). Eine Änderung von Datenblöcken in einer Datei ist nicht vorgesehen und wird vom Betriebssystem verhindert.

Bei der ab Blocknummer 0001 aufzuzeichnenden Datei wird die Aufzeichnung mit einer Vorlücke (ca. 30 sec.) begonnen, um das Vorspannband und die BOT-Marke der Kassetten zu überstreichen.

4.3.1. Magnetbandbedienung

Nach Anwahl des Unterprogrammes "BAND" meldet sich das Betriebssystem mit folgender Bildschirmauschrift:

```
MAGNETBAND :
  SCHREIBEN
  LESEN
  LOESCHEN
```

Durch Rücken des Zeichens ">" mittels Kursortasten kann eines der drei Programme angewählt werden.

MB-SCHREIBEN

Vom Bediener werden folgende Angaben bzw. Kommandos gefordert:

1.) AB BLOCK-NR.: _...

Hier ist die erste Blocknummer der zu schreibenden Datei vierstellig hexadezimal einzugeben und mit der Taste "ENTER" abzuschließen.

Durch Betätigen der Taste "OFF" anstelle von "ENTER" kann das Schreibprogramm verlassen werden (Rücksprung ins Magnetband-Menü).

Fehlerhafte Eingaben, die keiner Hexadezimalzahl entsprechen, werden vom Rechner nicht akzeptiert.

Die erfolgte Übernahme wird durch den Rechner durch Hochrücken der eingegebenen Blocknummer in die erste Bildschirmzeile angezeigt.

2.) ADRESSE VON BIS

Es ist der Adressbereich des Hauptspeichers einzugeben, der als Datei aufgezeichnet werden soll. Dabei wird zunächst die Anfangsadresse des Speicherbereiches als vierstellige Hexadezimalzahl gefordert. (Abschluß mit "ENTER"). Wird die Taste "OFF" genutzt, erfolgt der Rücksprung ins Magnetband-Menü. Danach ist die letzte Speicheradresse anzugeben, die bei der Aufzeichnung gespeichert werden soll. Die eingegebene Speicherendadresse wird vom Programm nach Betätigung von "ENTER" auf eine ganzzahlige Anzahl von Datenblöcken aufgerundet. (Taste "OFF" führt wieder ins MB-Menü). Das Programm akzeptiert nur einen aufsteigenden Adressbereich im Bereich von 0000 bis FFFF. Wird diese Folge nicht eingehalten, fordert das Programm die erneute Adresseingabe.

Die erfolgte Adressübergabe wird durch das Erscheinen des gerundeten Adressbereichs in der ersten Bildschirmzeile angezeigt. Gleichzeitig wird die benötigte Blockanzahl berechnet und ebenfalls in der ersten Bildschirmzeile angezeigt.

Nach der Bildschirmausschrift "NAME" kann der Inhalt des Kennzeichnungsblockes auf den Bildschirmzeilen 4...7 eingegeben werden (Zeile 7 nur mit 12 Zeichen). Bei Betätigung der Taste "OFF" ist es nochmals möglich, die in der ersten Bildschirmzeile genannten Dateikennzeichnungen entsprechend Punkt 1 und 2 zu ändern. Die Eingabe des Kennzeichnungsblockes wird mit Taste "ENTER" abgeschlossen. Vom Programm werden dann die die Datei kennzeichnenden Informationen (erste Blocknummer, Blockanzahl, Anfangsadresse, Endadresse) am Schluß der Bildschirmzeile 7 abgelegt und so zum Inhalt des Kennzeichnungsblockes.

Die Dateiaufzeichnung wird nun vom Prozessor ohne weitere Eingriffsmöglichkeit durch den Bediener vorgenommen (dem Bediener bleibt die im Abschnitt 2.7. genannte Möglichkeit über die Auswurf Taste). Ist keine Kassette im Kassettenfach vorhanden, meldet sich das Programm mit der Ausschrift "KAS.EINLEGEN" und wartet auf Ausführung. Ist die Kassette gegen Überschreiben durch Ausbruch der entsprechenden Riegel gesichert, erfolgt die Ausschrift "KAS.GESPERRT."

Durch Wechseln der gesperrten Kassette wird das Aufzeichnungsprogramm fortgesetzt. Die Suche der entsprechenden Bandstelle zur Aufzeichnung übernimmt der Prozessor. Ist die erste Blocknummer mit 0001 angegeben, wird die Kassette zunächst zum Bandanfang zurückgespult und anschließend über etwa 30 Sekunden konstant magnetisiert. Danach erfolgt die Aufzeichnung entsprechend der gewählten Parameter. Die Aufzeichnung wird in der 2. Bildschirmzeile von folgender Protokollauschrift begleitet:

```
WR xxxx yyyy zzzz
WR - Aufzeichnung läuft
xxxx - aktuelle Blocknummer
yyyy - Anzahl der noch zu schreibenden
      Blöcke
zzzz - Adresse des Hauptspeichers,
      bis zu dem geschrieben wurde.
```

Erfolgt die Dateiaufzeichnung von einer Blocknummer ungleich 0001 wird zunächst ein automatischer Suchlauf für die der Aufzeichnung vorhergehende Blocknummer gewählt. Dieser Suchlauf erfolgt in iterativer Weise. Bei beendeter Aufzeichnung erfolgt ein Rücksprung ins MB-Menü.

Fehlerauschriften:

"FDAT": Es erfolgte ein Schreibversuch innerhalb einer vorher angelegten Datei. Die Aufzeichnung wurde verhindert. Durch Betätigen einer beliebigen Taste ist das MB-Menü erreichbar. Das Überschreiben einer Datei als Ganzes ist dagegen möglich.

"Block xxxx fehlt":

Beim automat. Suchlauf kann der der angegebenen Blocknummer vorangehende Block nicht gefunden werden. Durch Betätigen einer beliebigen Taste ist das Magnetband-Menü erreichbar. Die Aufzeichnung wurde nicht begonnen.

MB-LESEN

Ähnlich dem Schreibprogramm werden vom Bediener folgende Daten gefordert:

1.) AB BLOCK-NR.:

- Es ist die erste zu lesende Blocknummer vierstellig hexadezimal einzugeben. Die Eingabe ist mit Bestätigung von "ENTER" gültig, durch "OFF" wird in das Magnetband-Menü gesprungen.

2.) ADRESSE VON ... BIS

- Es ist der RAM-Bereich anzugeben, auf den die Information gelesen werden soll.

Weitere Erläuterung siehe MB-Schreiben.

Die Anzahl der zu ladenden Datenblöcke wird vom Programm selbst ermittelt und die eingegebene RAM-Endadresse gegebenenfalls aufgerundet. Die endgültigen Parameter des Lesevorgangs erscheinen in der ersten Bildschirmzeile. Der weitere Leseablauf erfolgt prozessorgesteuert. Ist im Fach keine Kassette eingelegt, erfolgt in der zweiten Zeile die Bildschirmausschrift "KAS. EINLEGEN". Das Programm wird mit Einlegen einer Kassette fortgesetzt. Vom Prozessor wird ein autom. Suchlauf für die zu lesenden Datenblöcke gestartet. Wird nach 30 Sekunden kein Block gefunden, wird das Band bis zum Anfang rückgesetzt und erneut eine Synchronisation versucht.

Ist eine Blocknummer gefunden, wird durch Einschalten des schnellen Vor- und Rücklaufes eine iterative Annäherung an die zu lesende Bandstelle eingeleitet.

Beim Suchvorgang gefundene Kennzeichnungsblöcke von Dateien werden auf den Bildschirmzeilen 4 bis 7 wiedergegeben. Wurde die geforderte Datei gefunden, erscheint in der 2. Bildschirmzeile das folgende Leseprotokoll:

RD xxxx yyyy zzzz

RD - Lesevorgang läuft

xxxx - aktuelle Blocknummer

yyyy - Anzahl der noch zu lesenden Blöcke

zzzz - Hauptspeicheradresse, bis zu der schon gelesen wurde.

Nach erfolgtem Lesen wird ins MB-Menü zurückgesprungen. Die Fehlerausschrift "BLOCK xxxx FEHLT" erscheint, wenn die angegebene Blocknummer nicht gefunden wurde, bzw. wenn ein Datenblock eine fehlerhafte Prüfsumme aufweist.

Der Lesevorgang wurde vor dem fehlenden Block beendet, durch Betätigen einer beliebigen Taste ist ein Rücksprung bis MB-Menü möglich.

MB-LOESCHEN

Nach Anwahl dieses Programmes erscheint zunächst ein Fragezeichen als zusätzliche Sicherung. Soll die Kassette gelöscht werden, ist gleichzeitig die Taste "CTR" (Taste mit Gefühlspunkt) und "L" zu betätigen. Bei "OFF" wird in das MB-Menü zurückgesprungen.

Die Kassette wird zunächst zum Bandanfang gespult und anschließend mit 4,76 cm Laufgeschwindigkeit gelöscht. Der Löschvorgang kann mit Betätigen der Taste "OFF" beendet werden.

4.3.2. Unterprogramme für Magnetband

Programmversion
12.01.84

4.3.2.1. Arbeitszellen Magnetbandprogramm

E/A-Adressen:

<u>Adresse</u>	<u>Bezeichnung</u>	<u>Bemerkung</u>
00CFH	REG	Register für Laufwerksteuerung
00F8H	CT0	Kanal 0 CTC
00F9H	CT0+1	Kanal 1 CTC
00FAH	CT0+2	Kanal 2 CTC
00FBH	CT0+3	Kanal 3 CTC
00FCH	PIA-1	PIO-Kanal A, Daten
00FDH	PIA	PIO-Kanal A, Steuerwort
00FEH	PIB-1	PIO-Kanal B, Daten
00FFH	PIB	PIO-Kanal B, Steuerwort

Arbeitszellen

<u>Bezeich.</u>	<u>Adr.</u>	<u>Bytezahl</u>	<u>Bemerkung</u>
NAZ	00E6H	2	Namenszielgebiet; Beginn des Pufferbereiches, auf dem bei Magnetband-Lesen der Inhalt der Kennzeichnungsblöcke umgelagert wird bzw. von dem bei Magnetband-Schreiben der Kennzeichnungsblock geladen wird. Pufferlänge = 80H. Bei Menübedienung ist (NAZ)=C60H (entspr. Zeile 4...7 des BWS)
PAZ	00E8H	2	Pausenzähler; wird zur Festlegung der Blocklückenlänge vom Programm benutzt; kein Zugriff durch Anwender!
FEZ	00E7H	1	Fehlversuchszähler; Zähler, der die Anzahl der Iterationsschnitte bei automatischer Blocksuche zählt bzw. die Wiederholungszahl bei fehlerhaftem Lesen speichert; kein Zugriff durch Anwender!
CRC	00E5H	2	Zwischenspeicher Prüfsumme; wird bei Prüfsummenbildung und Prüfsummentest benutzt; kein Zugriff durch Anwender!
BYZ	00F0H	2	Bytezähler; Zähler für die beim Lesen erkannte Bytezahl pro Block bzw. für die pro Block zu schreibende Bytezahl. (Informations- und Zusatzbytes).
SPH	00F2H	1	Semaphore; Übergabezelle für Informationen aus Unterprogrammen

<u>Bezeich.</u>	<u>Adr.</u>	<u>Bytezahl</u>	<u>Bemerkung</u>
PUZ	0DF3H	2	Pufferzeiger, zeigt die aktuelle Adresse innerhalb des Magnetbandpuffers
BLZ	0DF5H	2	Blockanzahl; Anzahl der zu lesenden/schreibenden Datenblöcke in einer Datei
BYE:	0DF7H	2	Bytezahl; Anzahl der im Block enthaltenen Datenbytes; Menübedienung legt Zelle mit 80H fest
END:	0DF9H	2	Endadresse; letzte Adresse des Hauptspeichers, die beim Datenaustausch benutzt wird
RAM:	0DFBH	2	Anfangsadresse der Datei im Hauptspeicher
BLN:	0DFDH	2	Blocknummer; vor Operation erste zu lesende/schreibende Blocknummer; während Magnetbandoperation aktuelle Blocknummer
MBB:	0DFFH	1	Magnetbandbetriebsart; 00H-Schreiben 01H-Lesen 02H-Löschen BIT7 = 0: Bildschirmbedienung BIT7 = 1: Bildschirm wird nicht benutzt
PUA:	0E00	256	Puffer für Magnetband bis 0EFFH

4.3.2.2. Magnetband-Unterprogramme (erste Ebene) des MC 80.22

Das Magnetband-Programm wurde aus Unterprogrammen aufgebaut, die es gestatten, das Magnetband auch vom Prozessor aus zu starten. Es sind mehrere Programmebenen verfügbar. Die erste (höchste) Ebene gestattet durch einfache UP-Aufrufe das Lesen bzw. Schreiben von Dateien. Dazu sind vorher die benötigten Arbeitszellen entsprechend zu laden.

<u>Name</u>	<u>Adresse</u>	<u>Fehler- meldung</u>	<u>benutzte Arbeits- zellen...</u>	<u>Bemerkung</u>
MRD	0630H	Cy-Flag	RAM, BLZ, BLN, MBB, NAZ	Magnetband lesen
MWR	0633H	Cy, Z- Flags	RAM, BLZ, BLN, MBB, NAZ	Magnetband schreiben
MCL	0636H	-	-	Magnetband löschen


```

RAM: EXT 00FBH
BLZ: EXT 00F5H
BLN: EXT 00FDH
MBB: EXT 00FFH
NAZ: EXT 00EAH
MRD: EXT 0030H
BYE: EXT 00F7H

```

Beispiel 2:

Wie Beispiel 1, jedoch soll der Bildschirm beim Lesen nicht benutzt werden, z. B. um das Meßprotokoll nicht zu zerstören. Kennzeichnungsblöcke sollen auf den Speicherbereich 0E800H bis 0E87F geladen werden.

```

AN2: LD HL, 0E400H
      LD(RAM), HL ; Anfangsadresse Hauptspeicher
      LD HL, 00008H
      LD(BLZ), HL ; Blockanzahl
      LD HL, 00401H
      LD(BLN), HL ; erste zu lesende Blocknummer
      LDA, 081H
      LD(MBB), A ; MB-Lesen ohne Bildschirmbedienung
      LDA, 080H
      LD(BYE), A ; MC-80-Datenformat
      LD HL, 0E800H
      LD(NAZ), HL ; Kennzeichnungsblöcke auf E800
      CALL MRD ; UP Lesen
      RET ; hier ggf. Fehlerauswertung (Cy=1)

```

Alle benutzten Marken sind entsprechend zu vereinbaren.

Magnetband schreiben (MWR):

Bei Aufruf des Programmes wird das eingelegte Magnetband entsprechend der eingestellten Parameter beschrieben. Durch das übergeordnete Programm ist die Dateiverwaltung des behandelten Magnetbandes zu sichern. Dem Schreibprogramm sind folgende Parameter zu übergeben:

```

BYE(00F7) ; Bytezahl (Daten) pro Block (4-stellig)
BLN(00FD) ; 1. zu lesende Blocknummer (4-stellig)
MBB(00FF)=0 ; Magnetband schreiben
RAM(00FB) ; Anfangsadresse des auszulagernden Hauptspeicher-
           ; bereiches
BLZ(00F5) ; Anzahl der zu schreibenden Blöcke
NAZ(00EA) ; Quellgebiet Kennzeichnungsblock

```

Danach kann das Programm MWR aufgerufen werden.

Von diesem Programm werden folgende Unterprogramme genutzt:

```

GRZ: Grundzustand Arbeitsspeicher
LIN: Initialisierung Lesen
RCK: Rückspulen bis Bandanfang
WIN: Initialisierung Schreiben
WVO: Blockvorbereitung Schreiben
WPR: Protokoll der Schreibparameter auf Bildschirm

```

LES: Einleseprogramm
 AUS: Endabschaltung Laufwerk
 BLE: ↑ Blockstruktur Lesen

Durch MWR werden nacheinander folgende Funktionen ausgeführt:

a) Schreiben ab Block-Nr. 1

- Rückspulen bis Bandanfang
- 30 sec. Initiallücke Schreiben
- Schreibprogramm entsprechend eingegebener Parameter ausführen
- Endabschaltung Laufwerk

b) Schreiben ab Block-Nr. >1

- automat. Suche der entsprechenden Bandstelle
- Prüfung auf Vorhandensein einer Bandmarke als Trennzeichen zwischen den Dateien
- Schreiben der Daten entsprechend der eingegebenen Parameter als Datei
- Abschluß der Datei durch Schreiben einer Bandmarke
- Abschaltung Laufwerk

Das Programm meldet sich zurück mit Carry-Flag=0 bei ordnungsgemäßer Aufzeichnung. Bei fehlerhafter Aufzeichnung wird Carry gesetzt. Das Zero-Flag kennzeichnet die Art des aufgetretenen Fehlers:

- Cy=1; Z=1: Vorhergehender Block nicht gefunden oder nicht lesbar
- Cy=1; Z=0: Schreibversuch innerhalb einer vorher aufgezeichneten Datei

Beispiel 3:

Schreiben einer Datei von 8 Blöcken ab Block Nr.: ~~0601H~~, Quellgebiet sei der Hauptspeicher ab ~~0C000H~~; Bildschirmaus-schrift der Parameter soll erfolgen, der Kennzeichnungsblock befinde sich im RAM auf den Adressen ~~0E000H~~ bis einschließlich ~~0E07FH~~.

```

AN3: LD HL, 0C000H
      LD(RAM), HL      ; Anfangsadresse Quelldaten
      LD HL, 00003H
      LD(BLZ), HL     ;Blockanzahl
      LD HL, 00601H
      LD(BLN), HL     ; erste Block-Nummer
      LD A, 080H
      LD(BYE), A      ; Bytezahl pro Block = 80H
      LD A, 000H
      LD(MBB), A      ; MB-Schreiben
      LD HL, 0E000H
      LD(NAZ), HL     ; Kennzeichnungsblock von 0E000H
      CALL MWR        ; UP: Schreiben
      RET             ; hier ggf. Fehler Fehlerauswertung
  
```

Die benutzten Marken sind entsprechend zu vereinbaren.

Magnetband Löschen (MCL)

Dieses Programm dient zum einseitigen Löschen einer eingelegten Kassette. Dieses Programm benutzt für Aufforderungen zur Kommandoeingabe den Bildschirm. Nach Programmauswahl wird der Löschvorgang vom Bandanfang durch Betätigen der Control-Taste und der Taste "L" gestartet. Eine Rückkehr ist durch Betätigen der Taste "OFF" vor Löschbeginn möglich. Das gestartete Programm wird verlassen durch Betätigen der Taste "OFF".

4.3.2.3. Magnetband-Unterprogramme (zweite Ebene) des MC 80.22

Neben den in Punkt 4.3.2.2. erläuterten Unterprogrammen existieren weitere Unterprogramme, die es ermöglichen, verschiedene Funktionen des Laufwerkes auszulösen. Dabei ist jedoch zu beachten, daß vor Aufruf der Programme entsprechende Arbeitszellen geladen werden bzw. bestimmte Programme vorher aufgerufen wurden. Die folgende Tabelle gibt einen Überblick über nutzbare Unterprogramme.

<u>Name</u>	<u>Adresse</u>	<u>benutzte UP</u>	<u>benutze Arb.zellen</u>	<u>Bemerkg.</u>
LWK	0651H	PIN,KAS,SSP	MBB,Register E	Laufwerk einschalten
PIN	0654H	-	-	Initialisierung Steuerport Laufw.
RCK	0639H	-	-	Rückspulen bis Bandanfang danach "Lauf"
AUS	063CH	-	-	Endabschaltung Laufwerk
LIN	063FH	LWK	MBB,Register E	Initialisierung lesen
GRZ	0648H	-	-	Grundzustand Pufferspeicher
R1S	064BH	ZEI	-	Rücksetzen
BLE	062DH	GRZ,Zei	BYE	Lesen eintr. Blockstruktur
LES	064EH	BLE,CRR,VGL	BYE,BLN,MBB,RAM BLZ,NAZ	Leseprogramm
VGL	0657H	ZEI	BLN,Register BC	automat. Suchprogramm
WIN	0642H	-	-	Initialisierung Schreiben
WVO	0645H	CRR	SPH,BYZ,RAM	Vorbereitung schreiben
ZEI	065AH	-	Register A,B,C	Zeitschleife

Erläuterungen:

PIN: Initialisierung des Steuerportes (PIO, Kanal B) für die Laufwerkssteuerung. Das Steuerport B wird mit den entsprechenden Steuerwörtern programmiert, die Interruptstartadresse für die interruptfähigen Bits wird in die entsprechende Tabelle eingetragen (D2C), die Interruptquellen sind aber noch gesperrt.

LWK: Laufwerk einschalten
Es werden nacheinander folgende Funktionen ausgeführt:

- Initialisierung Steuerport (PIN)
- Abfrage: Kasette eingelegt? (KAS)
- bei MP-Schreiben (MBB=0) oder Löschen (MBB=2)
Abfrage: Schreibsperre gesetzt? (SSP)
- Ausgabe Steuerdaten (Register E) an Steuerport B
E0 = 1 Lesen * 0 Schreiben
E1 = 1 Capstan ein * 0 Capstan aus
E2 = beliebig
- E3 = 1 Daten schreiben * 0 Pausenpegel schreiben
- Magneten einschalten
- Warten auf Gleichlauf (ZEI)

RCK: Rückspulen der eingelegten Kasette bis Bandanfang; vor Aufruf von RCK ist LWK auszulösen. Von RCK wird bei eingelegter Kasette ein Schnellrücklauf ausgelöst; bei Endabschaltung am Bandende wird auf Vorlauf mit 4,76 cm/s umgeschaltet und das Programm beendet.

AUS: Endabschaltung des Laufwerkes
Durch das Programm werden die aktiven Peripherieschaltkreise CTC und PIO rückgesetzt, am Laufwerk werden alle evtl. laufenden Motoren und aktive Magnete abgeschaltet.

LIN: Initialisierung Lesen
Mit diesem Unterprogramm wird die gesamte Ansteuerelektronik für MB-Lesen initialisiert. Es werden folgende Funktionen ausgeführt:

- Laufwerk einschalten (LWK) mit Register E = 03H
- Freigabe Synchronisationsinterrupt (Blockendeerkennung)
- Initialisierung Datenport (PIOA) auf Byte-Eingabe
- Initialisierung CTC für MB-Lesen

GRZ: Grundzustand Pufferspeicher
Bei Aufruf dieses Programmes werden die Zwischenpufferzellen initialisiert. Im Einzelnen werden folgende Funktionen ausgelöst:

- Pufferzeiger (PUZ) wird auf Anfang gestellt (0E00H)
- Puffer (0E00H ... 0EFFH) wird mit 00H geladen
- Semaphore (SPH) wird gelöscht (00H)
- CRC-Akkumulator wird gelöscht (0000H)
- Bytezähler (BYZ) wird gelöscht (0000H)

R1S: Rücksetzen

Mit diesem Programm wird das Band ca. 1 sec. rückgesetzt, damit kann die unmittelbare Synchronisation auf den vorhergehenden Block realisiert werden. Vor Aufruf des Programms ist das Programm L'WK (Register E=3) oder ein anderes MB-Laufprogramm durchzuführen.

BLE: Einlesen einer Blockstruktur (Datenblock, Kennzeichnungsblock, Bandmarke)

Vor Aufruf des Programms muß Lesen initialisiert sein (LIN). Das Programm BLE wartet auf das Erkennen eines Blockendes (MRS-Interrupt). Wird nach ca. 35 Sekunden kein Blockende gemeldet, wird das eingelegte Band bis zum Bandanfang rückgespult. Im Bit 7 der Zelle FEZ wird gespeichert, daß in Bandmitte ein Synchronisationsversuch durchgeführt wurde. Ist Bit 7 bereits bei vorherigem Aufruf von BLE gesetzt worden, wird nach der 35-Sekundensuche nur um ca. 10 sec. rückgespult und der Fehlerversuchszähler (FEZ) dekrementiert (FEZ ist also vorher mit einer entsprechenden Zahl zu laden).

- Bei erkanntem Blockende wird die Art der gelesenen Bandstruktur geprüft. Bit 6=1 von SPH meldet eine gelesene Bandmarke, Bit 5=1 von SPH einen gelesenen Block. Die Blockstrukturen werden dabei auf Einhaltung der entsprechenden Strukturmerkmale geprüft (Präambel, Postambel, Bytezahl). Eine CRC-Prüfung erfolgt noch nicht. Die gelesene Blockstruktur ist im Zwischenpufferbereich (0E00H-0EFFH) verfügbar.

LES: Lesen der eingelegten Kassette entsprechend übergebener Parameter

Vor Aufruf des Programms muß Lesen initialisiert sein (LIN). Das Programm LES realisiert dann das Einlesen der Information vom Band nach den übergebenen Parametern (BLN, BLZ, RAM, BYE, NAZ). Es werden folgende Funktionen ausgelöst:

- Blockstrukturen einlesen über BLE
- Fehlerversuchszähler mit 10H laden
- CRC-Summe des eingelesenen Blockes überprüfen
- Kennzeichnungsblöcke in entsprechendes Gebiet umlagern
- Gelesenen Block auf Bildschirm schreiben (falls Bit 7 von MBB=0).
- Suchen der entsprechenden Bandstelle (VGL)
- bei gefundener Bandstelle: Einlesen der Daten entsprechend der eingegebenen Parameter
- bei Aufruf aus Schreibprogrammen heraus (MBB=0) Ende von LES bei gefundener Bandstelle
- nach mehrmaligen Versuchen (FEZ=10H) nicht gefundene oder nicht lesbare Blöcke werden mit Cy=1 gemeldet, die fehlende Blocknummer ist in BLN gespeichert
- bei ordnungsgemäßem Abschluß des Programms ist Cy=0

VGL: Programm Blocknummernvergleich und Suchlauf

Vor Aufruf des Programms müssen die Programme LIN und BLE aufgerufen werden, im Register BC ist die zu findende Blocknummer zu übergeben.

Von VGL wird dann die gelesene Blocknummer (Inhalt der Adressen $\overline{OE01}$, $\overline{OE02}$) mit der zu lesenden (Register BC) verglichen und im Ergebnis ein entsprechender Suchlauf ausgelöst. Ist die gelesene Blocknummer größer als die zu lesende, wird rückgespult, ist sie kleiner (mindestens um 20H) wird vorgespult. Ist der Fangbereich (1FH vorhergehende Blöcke) erreicht, wird das Programm ohne Spulvorgang mit laufender Wiedergabegeschwindigkeit verlassen.

Die Spulvorgänge erfolgen iterativ, d. h., gegebenenfalls ist VGL nochmals aufzurufen, bis die entsprechende Bandstelle erreicht ist. Dabei muß vorher erneut BLE aufgerufen werden, da der Zwischenpufferbereich nach VGL nicht mehr nutzbar ist.

WIN: Initialisierung Schreiben

Vor Aufruf des Programms müssen die Unterprogramme PIN und LWK ausgelöst worden sein.

Vom Programm WIN wird das Steuerport der Magnetbandlogik auf Schreiben programmiert, der Interruptvektor für den Zeitgeberbaustein (CTC) wird entsprechend geladen sowie der CTC für die entsprechenden Funktionen vorbereitet.

WVO: Vorbereitung Schreiben

Allgemeines Vorbereitungsprogramm für das bevorstehende Schreiben einer Bandmarke, Kennzeichnungs- oder Datenblockes. Voraussetzung ist der vorherige Aufruf von GRZ zur Herstellung des Grundzustandes der Arbeitszellen. In der Übergabezelle SPH wird dem Programm die Art des vorzubereitenden Blockes übergeben.

SPH, Bit 0 = 1: Bandmarke

SPH, Bit 1 = 1: Kennzeichnungsblock

SPH, Bit 2 = 1: Datenblock

Vom WVO wird dann der Pufferbereich mit den entsprechenden Daten geladen. Außerdem wird bei Kennzeichnungs- und Datenblöcken die CRC-Prüfsumme berechnet und in den Pufferbereich eingefügt. Die Datenaufzeichnung erfolgt Interrupt-gesteuert nach Freigabe mit WIN. Aus den Interrupt-Routinen erfolgt die Fertigmeldung über einen geschriebenen Block mit (BYZ) = 0. Danach kann über WVO der nächste Block vorbereitet werden. Die Pausenzeiten werden ebenfalls durch Interrupt festgelegt.

ZEI: Zeitschleife

Das Programm ZEI stellt eine Warteschleife dar, die vom Magnetbandprogramm benutzt wird, um auf die Reaktion von Motoren und Magneten zu warten. Die Schleife besteht aus 3 Zyklen B, C und A, wobei C und A dem Programm zu übergeben sind. Register B stellt die innere, Register C die mittlere und Register A die äußere Zählschleife dar.

Beispiel 4: Rückspulen bis Bandanfang, danach Halt

```
AN4: CALL LIN ; Initialisieren Lesen
      CALL RCK ; Rückspulen bis Bandanfang
      CALL AUS ; Abschalten Laufwerk
```

Die benutzten Marken sind entsprechend zu vereinbaren.

Beispiel 5: Blockweise Lesen in den Zwischenpuffer, danach Stoppen des Laufwerkes

```
AN5: CALL LIN ; Initialisieren Lesen
      CALL BLE ; Blockstruktur lesen
      CALL AUS ; Abschalten Laufwerk
```

Die benutzten Marken sind entsprechend zu vereinbaren. Der gelesene Block befindet sich im Zwischenpufferbereich ~~OE00H~~ ... ~~OEFFH~~.

4.4. "EDIT"

4.4.1. Beschreibung des Objektcodeeditors

Der Editor beinhaltet den Assembler zur Übersetzung von Befehlen vom Mnemonikcode in Maschinencode und den Reassembler zur Rückübersetzung des Maschinencodes in die mnemonische Darstellung. Außer dem Maschinencode wird noch eine Markentabelle abgespeichert. Die Notwendigkeit eines schnellen Massenspeichers entfällt damit.

Durch den line-per-line-Assembler (Assemblierung sofort nach jeder Zeile) werden Syntaxfehler sofort bei der Eingabe erkannt und angezeigt.

Ein Testprogramm ermöglicht die schrittweise Testung von Anwenderprogrammen. Um eine übersichtliche Arbeit zu gewährleisten und alle Funktionen einer interaktiven Programmerstellung zu ermöglichen, werden unterschiedliche Funktionen in mehreren Modi (Betriebsarten) wahrgenommen, die sich jeweils durch eine typische Bildschirmausschrift und spezifische Kommandos unterscheiden.

Diese Modi sind:

- Anzeigemodus
- Quellcode-Eingabemodus
- RAM-Modus
- Testmodus

4.4.2. Aufruf des Editors

Soll ein schon teilweise erstelltes Programm weiter bearbeitet werden, so ist dieses mit seinen zwei Komponenten Markentabelle und Maschinencode vom externen Datenträger zuvor auf den RAM zu laden. Danach erfolgt der Aufruf des Editors aus der Programmtabelle des MC 80. Der Editor meldet sich mit der Bildschirmausschrift:

```
EDIT MTB#nnnn
```

Nach MTB# steht die Anfangsadresse der Markentabelle. Die angegebene Vorzugsbelegung kann überschrieben werden, wenn die Markentabelle auf einem anderen RAM-Bereich angelegt ist. Dabei ist zu beachten, daß vor der Markentabelle noch 20 Byte für den Editor als Zwischenspeicher zur Verfügung stehen müssen. Danach ist "ENTER" zu betätigen und auf dem Display erscheint die für den Anzeigemodus typische Bildschirmausschrift. Von hier aus sind alle anderen Modi erreichbar. Im Anzeigemodus kann man mit dem Kommando "G#adr" (siehe Tabelle 4.3., Erläuterung 7) die gewünschte Programmstelle (Adresse) anwählen. Bild 4.5 zeigt die Übergänge zwischen den einzelnen Modi. Dabei stellt eine Besonderheit dar, daß der RAM-Modus sowohl über den Anzeigemodus als auch direkt vom MENÜ aus erreichbar ist.

Die Adresse, die im Quellcode-Eingabemodus oder Testmodus aktuell ist, wird bei Rückkehr in den Anzeigemodus gespeichert. Dadurch wird ein fortlaufendes Zusammenwirken dieser Modi möglich.

Das Ende der Markentabelle wird immer mit ØØ gekennzeichnet. Werden während der Quellcode-Eingabe Marken vereinbart, so wird die Markentabelle automatisch gefüllt. Jede Markenvereinbarung und jeder Zugriff auf eine Marke benötigen 5 Byte. Es können maximal 127 Marken vereinbart werden. Die Anzahl der Zugriffe wird nur durch den für die Markentabelle vorgesehenen RAM-Bereich beschränkt. Ein Überlaufen der Markentabelle in einen nicht existierenden RAM-Bereich oder in ein Programm wird nicht verhindert.

Der Bediener kann sich jederzeit im RAM-Modus über die Ausdehnung der Markentabelle (den belegten RAM-Bereich) informieren.

4.4.3. Verlassen des Editors

Das Verlassen des Editors und der Wiedereintritt in das Betriebssystem geschieht mittels Betätigen der "OFF"-Taste. Vor Beendigung der Arbeit, auch in Zwischenphasen, sollte das erstellte Programm wieder auf den externen Datenträger (Magnetband) ausgelagert werden. Über den Umfang der Markentabelle und den insgesamt von Anwenderprogrammen belegten RAM-Bereich informiert man sich zuvor im RAM-Modus. Es sind der von der Markentabelle belegte RAM-Bereich und der vom Anwenderprogramm (vom Maschinencode) belegte RAM-Bereich auszulagern.

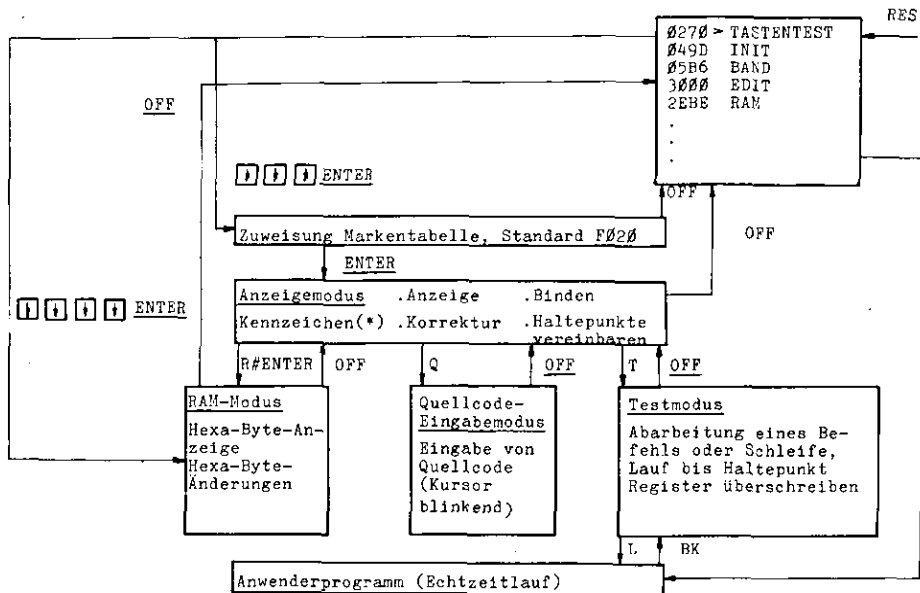


Bild 4.5: Übergänge zwischen den Modi

4.4.4. Der Anzeigemodus

In diesem Modus ist es möglich, das bereits erstellte Programm zu kontrollieren, (Notierungskontrolle), interessierenden Stellen zu suchen und anzuwählen sowie zu korrigieren. Der Anzeigemodus ist durch folgende Bildschirmausschrift gekennzeichnet (Bsp.):

```
EDIT 8234                                MUL 822F
* LD HL,00000H
MU1: ADD HL,HL
      ADC A
      JRNC MU2
      LD A,C
      SUB 004H
      LD C,A
```

Bild 4.6: Bildschirmausschrift im Anzeigemodus (Bsp.)

Das "*" steht vor der Befehlsausschrift des zur Bearbeitung anstehenden Befehls. Die oberste Zeile enthält nach der Kennzeichnung "EDIT" die Adresse (hexadezimal) des zur Bearbeitung anstehenden Befehls (Anzeigeadresse, Programmzähler für den Testmodus). Danach folgt die Angabe der davor stehenden Globalmarke, das ist eine Marke, die als drittes Zeichen einen Buchstaben enthält, und deren Adresse. Bezeichnet man Programmanfänge oder markante Stellen im Programm mit Globalmarken, so erhält man dadurch einen Überblick, in welchem Programmstück man sich befindet. Danach folgt die Auflistung der Befehle.

Im Anzeigemodus gibt es Anzeige- und Korrekturkommandos. Die Anzeigekommandos dienen der Kontrolle des Programmes und der Positionierung zwecks Korrektur und Eingabe. Angezeigt wird das, was der Reassembler an Maschinencode aus dem RAM angeboten bekommt. Demzufolge werden Zahlen in hexadezimaler Form in einem Standardformat ausgeschrieben. Pseudobefehle (DB, DM, DW) werden nicht richtig reassembliert. Außerdem werden die Informationen aus der Markentabelle genutzt.

Liegt zu einer Befehlsadresse eine Markenvereinbarung vor, so wird die Marke mit angegeben. Gibt es bei einer Befehlsadresse einen Markenzugriff, so wird statt der im RAM stehenden Hexazahl die Marke ausgeschrieben.

Kommandos im Anzeigemodus:

Kommando-Taste	weitere Angaben	Funktion
T	-	Eintritt in den Testmodus an dem angezeigten Befehlszählerstand (Anzeigeadresse)
H	-	Vereinbarung der Anzeigeadresse als Haltepunkt 8)
R	#adr ENTER	RAM-Anzeige ab adr, Eintritt in den RAM-Modus 7)
R	ENTER	RAM-Anzeige ab Anzeigeadresse, Eintritt in den RAM-Modus
Q	-	Eintritt in den Quellcodeeingabemodus
↓	-	Anzeigeadresse = Adresse des nächsten Befehls
↑	-	davorstehender Befehl wird aktueller Befehl 1)
N	-	Ausführung Kommando "↓" siebenmal
Z	-	Ausführung Kommando "↑" siebenmal 1)
+	-	Anzeigeadresse incrementieren 2)
A	-	Anzeige ab davorstehender Globalmarke 1)
G	#adr ENTER	Anzeige ab adr 7)
G	#Ø ENTER	Anzeige der mit EXT zugewiesenen Marken 9)
G	ENTER	Anzeige ab Programmende 3)
B	ENTER	Binden des Programms auf Speicheradresse
B	#adr ENTER	Binden des Programms so, daß aktueller Befehl auf adr läuft 7)
P	-	Annahme des im Testmodus verlassenen Programmzählerstandes
C	-	Sprung auf Adresse 2403 (reserviert für Ansteuerung Drucker, Stanzer)

Kommando-Taste	weitere Angaben	Funktion
X	-	Sprung auf Adresse 2400 (reserviert für Ansteuerung Lochbandleser)
M	#adr1 ENTER #adr2 ENTER	Verschieben des Programms mit Marken von adr1 bis ausschließlich adr2 auf den Bereich ab Anzeigeadresse 4), 5), 7)
M	#adr ENTER # ENTER	Verschieben von adr bis Programmende 3) auf Bereich ab Anzeigeadresse 4), 7)
V	#adr1 ENTER #adr2 ENTER	Verschieben nur der Marken von adr1 bis ausschließlich adr2 auf Bereich ab der Anzeigeadresse 6), 5), 7)
V	#adr ENTER # ENTER	Verschieben nur der Marken von adr bis Programmende 3) auf den Bereich ab Anzeigeadresse 6), 7)
E	#adr ENTER	Schaffen eines Eingabebereiches ab der Anzeigeadresse bis ausschließlich adr. 4), 7) Das folgende Programm mit den Marken wird auf adr verschoben
E	ENTER	Schaffen eines Bytes Eingabebereich vor der Anzeigeadresse 4)
S	#adr ENTER	Streichen der Befehle von der Anzeigeadresse bis adr 5), 7)
S	ENTER	Streichen des anstehenden Befehls
OFF		Verlassen des Editors

Tab. 4.3: Kommandos im Anzeigemodus

1) Als Globalmarke wird eine Marke bezeichnet, die als drittes Zeichen einen Buchstaben enthält, Bsp. MAX im Gegensatz zu MA1.

Die Rückassemblierung eines Befehls und die Erkennung des nachfolgenden Befehls sind daran gebunden, daß die Befehlsadresse auf das erste Byte des Befehlscodes im RAM zeigt, und daß nacheinander lückenlose Befehle folgen. Ein voranstehender Befehl kann nicht direkt richtig adressiert werden.

Die Kommandos "Z", "↑" und "A" bedingen deshalb, daß zuerst die voranstehende Globalmarke gesucht wird (die einen Befehl im ersten Byte adressiert), danach wird aufeinanderfolgend der nächste Befehl ermittelt, bis die Bedingungen für die Kommandos "Z" und "↑" erfüllt sind (Anzeige des direkt oder einem Bild voranstehenden Befehls). Ein Befehlsblock benötigt deshalb an seiner Anfangsadresse immer eine Globalmarke. Existiert im Bereich 256 Bytes vor der aktuellen Befehlsadresse bis zu dieser keine Globalmarke, so werden die Kommandos "Z" und "↑" nicht ausgeführt. In diesem Fall muß man sich mit "G"-Kommando an den Programmanfang oder eine markante Programmstelle zurückbewegen und eine Globalmarke eingeben.

2) Dieses Kommando ist nötig, um zwischen Datenblöcken einen Befehlsanfang wiederfinden zu können. Dieser soll aber mit einer Globalmarke gekennzeichnet werden.

3) Ein Programmende ist eine Adresse, ab der mindestens viermal der Speicherinhalt "FF" steht.

4) Ein Verschieben von Programmteilen kann nur in einem Speicherbereich erfolgen, der zuvor mit 00 oder FF geladen wurde. Damit wird ein versehentliches Zerstören eines Programms vermieden. Für das Kommando "M" bedeutet das, daß der Bereich ab der Eingabeadresse mit der Länge adr2 - adr1 frei sein muß. Der Quellbereich wird gelöscht.

Bei Anwendung des Kommandos "E" muß der Bereich ab Programmende genügend viele freie Speicherplätze enthalten. Im anderen Fall wird das Kommando nicht ausgeführt. Weiterhin ist darauf zu achten, daß die Anzahl der zu schaffenden Bytes Eingabebereich (NOP-Befehle) mit der Byteanzahl des im Quellcode-Eingabemodus einzufügenden Befehls übereinstimmen muß bzw. nur größer sein darf.

5) Bei der Angabe von zwei Bereichsadressen muß die Endadresse > Anfangsadresse sein. Beim Kommando "S" muß die angegebene Adresse > Anzeigeadresse sein. Im anderen Fall wird das Kommando nicht ausgeführt. Die Angabe der Adresse Ø beim Kommando "S" verhindert also die Kommandoausführung.

6) Die Verschiebung von Marken allein kann angewendet werden, wenn das zugehörige Maschinencodeprogramm sich schon auf dem Zielbereich befindet. Z.B. kann es sein, daß der ursprüngliche RAM-Bereich nicht mehr im Gerät enthalten ist und das Programm auf den Zielbereich geladen wurde bzw. sich das Programm in gebundener Form auf einer anderen Adresse befindet.

7) Unter adr kann eine hexadezimale Zahl mit führender Ziffer oder eine vereinbarte Marke angegeben werden. Der Aufruf z.B. der Adresse C000 erfordert die Befehlseingabe G#0C000 "ENTER". Die Eingabe einer nicht vereinbarten Marke verhindert die Kommandoausführung.

8) Nach Betätigung dieser Taste ist die entsprechende Anzeigeadresse als Haltepunkt für den Schrittbetrieb im Testmodus markiert.

Bei der schrittweisen Abarbeitung des Programmes erfolgt der Testlauf bis zu diesem Haltepunkt. Bei Rückkehr in den Anzeigemodus wird dieser Haltepunkt, wenn er angelaufen wurde, wieder gelöscht. Ebenfalls wird er bei Verlassen des Editors gelöscht. Bei der Vereinbarung mehrerer Haltepunkte wird nur der zuletzt eingegebene als Haltepunkt bewertet.

9) Es wird jeweils nur eine mit "EXT" zugewiesene Marke angezeigt. Durch Betätigen der Taste "ENTER" erfolgt die fortlaufende Markenanzeige.

4.4.5. Der Quellcode-Eingabemodus

Nach Eintritt in den Quellcode-Eingabemodus (Betätigen der Taste "Q") kann der anstehende Befehl (Anzeigeadresse) korrigiert werden. Ein Überschreiben von Befehlen ist unbedingt zu vermeiden, da bei teilweisem Überschreiben von Befehlen mit Markenzugriff die Anweisungen in der Markentabelle nicht gelöscht werden. Beim Binden des Programmes (siehe Punkt 4.4.6.) kommt es dann zur Zerstörung von Programmteilen. Das Vorgehen muß in diesen Fällen immer aus den Schritten, "Streichen des Befehls", "Schaffen eines Eingabebereiches" und "Quellcodeeingabe" bestehen (siehe Tabelle 4.3).

Nach Betätigung der "ENTER"-Taste wird der anstehende Befehl übersetzt. Liegt ein Syntaxfehler vor, so wird auf der letzten Position der Eingabezeile "F" ausgeschrieben. Wird eine Marke zum zweiten Mal vereinbart, so erfolgt dort die Ausschrift "M". Ein "O" wird angezeigt, wenn bei relativen Sprüngen die Sprungdistanz überschritten wurde. In diesen Fällen muß die Eingabe korrigiert und nochmals "ENTER" betätigt werden.

Der richtig übersetzte Befehl wird im Maschinencode ab der Anzeigeadresse abgelegt. Bei Markenangaben werden diese in die Markentabelle eingetragen. Zuvor auf die Anzeigeadresse bezogene Eintragungen in der Markentabelle werden gestrichen. Der übersetzte Befehl und die nachfolgenden Befehle werden reassembliert zur Kontrolle angezeigt. Vor dem aktuellen Befehl steht das Zeichen "+".

Nach Betätigen der Taste "ENTER" wird das Programm ab der nächsten Befehlsadresse angezeigt und die Quellcode-Eingabe für den nächsten Befehl kann folgen. Der Wiedereintritt in den Anzeigemodus geschieht durch das Betätigen der Taste "OFF". Die Eingabe der Pseudooperation "END" bewirkt ebenfalls das Verlassen des Quellcode-Eingabemodus.

Folgende Pseudooperationen werden verarbeitet:

DB nn	... definiere Byte mit dem Wert nn
DW nnnn	... definiere Doppelbyte mit dem Wert nnnn
DM 'aabc'	... definiere Zeichenkette mit den Zeichen aabc (ASCII-Code)
END	... Verlassen der Quellcode-Eingabe
MAR: EXT nnnn	... definiere Marke mit der externen Adresse nnnn

Bei mehrmaliger Vereinbarung einer Marke mit einer externen Adresse wird jeweils die vorherige Eingabe gelöscht. Dadurch besteht die Möglichkeit der Adreßkorrektur.


Pseudoanweisungen werden nicht in Maschinenbefehle des MC 80 übersetzt.

Der Befehlssatz des Objektcodeeditors des MC 80 ist in Anlage 1 dargestellt.

Zahlen können hexadezimal oder dezimal eingegeben werden. Sie müssen mit einer Ziffer beginnen (z.B. 33FFH, 0BFFFH), Hexadezimalzahlen sind mit dem Zeichen "H" abzuschließen, bei Dezimalzahlen erfolgt keine weitere Kennzeichnung. Für alle Zahlen können Marken angegeben werden, die geeignet vereinbart werden. Nach Angabe einer Marke kann beginnend mit der Kennzeichnung "+" oder "-" ein Offset als Zahl im Bereich + 127...- 128 angegeben werden (z.B. LD DE, MAR+0AH). Diese Informationen werden in der Markentabelle gespeichert. Die Zuweisung des Zahlenwertes erfolgt beim Binden (Punkt 4.4.6.).

Das Zeichen "#", mit der folgenden Angabe eines Offsets kann für die Angabe des Zuweisungszählers geschrieben werden. Nach der Assemblierung wird sofort der bekannte Wert des Zuweisungszählers in den Maschinencode eingetragen, eine Reassemblierung dieser Angabe erfolgt nicht!

Eingabeformat:

Das Markenfeld beginnt ab der zweiten Position der Eingabezeile, auf welcher der Cursor zu Anfang steht. Eine Marke muß aus 3 Zeichen bestehen und mit einem Buchstaben beginnen! Die Trennung Markenfeld - Operationscodefeld erfolgt durch mindestens ein Leerzeichen. Wird keine Marke angegeben, so genügt die Angabe eines Leerzeichens bis zum Operationscode oder das Betätigen des Tabulators "  ". Operationscode und Operanden dürfen nur durch ein Leerzeichen getrennt werden. Kommentar kann nicht mit angegeben werden.

4.4.6. Binden von Programmen

Das Binden des erstellten Anwenderprogramms erfolgt nach dem Kommando "B" im Anzeigemodus auf eine wählbare Adresse, oder nach Eintritt in den Testmodus auf die Speicheradresse. Es kann nur das erstellte Programm als Gesamtmodul mit einem Schritt gebunden werden. Die Bindeoperation bezieht sich auf den gesamten 64 KByte-Speicherbereich. Alle Zahlen, die symbolisch mit einer Marke angegeben wurden (diese Information liegt in der Markentabelle vor), werden entsprechend dem Markenwert in den Maschinencode des erstellten Programms eingetragen. Damit wird der Maschinencode lauffähig.

Beim Binden auf die Speicheradresse entsprechen den Marken die Adressen, auf der sie vereinbart wurden. Beim Binden auf eine angegebene Adresse werden die Markenwerte so verschoben, daß das Programm auf der angegebenen Adresse lauffähig ist, unabhängig von der Speicheradresse. Marken, die mittels der Pseudooperation "EXT" vereinbart wurden, werden dabei nicht mit verschoben, sondern behalten ihren Wert (externe Eintrittsadressen, die sich nicht auf das zu bindende Programm beziehen, sondern auf andere fertige oder noch zu bindende Programm-Module).

Externe Marken (mit "EXT" vereinbart) lassen sich im Anzeigemodus auf einen beliebigen Bereich verschieben und können somit entsprechend der Bindeadresse eines anderen Programmmoduls positioniert werden.

Eine nicht zugewiesene Marke wird mit ihrem Namen und der Adresse, an der auf sie zugegriffen wird, ausgeschrieben. Wird der Bereich der Distanz bei relativen Sprüngen überschritten, so erfolgt die Ausschrift der Adresse, auf welcher der relative Sprung steht. Nach dieser Protokollauschrift auf dem Display kann die Taste "ENTER" betätigt werden, um weitere Fehlerstellen zu finden. Die Bindeoperation wird aber nicht weiterbearbeitet.

Der Inhalt der Markentabelle wird beim Binden nicht verändert.

4.4.7. Der Testmodus

In diesem Modus ist es möglich, ein Anwenderprogramm schrittweise oder abschnittsweise (Halt bei jeweils vereinbarten Haltepunkten) zu testen oder von einem bestimmten Programmzählerstand mit angebbaren Registerbelegungen im Echtzeitlauf zu starten.

Bei Eintritt in den Testmodus mit dem Kommando "T" wird auf die Speicheradresse gebunden. Bei einer Fehlerausschrift sofort den Modus mit "OFF" verlassen, den Fehler korrigieren und neu in den Testmodus eintreten.

```
TEST  _  SCHRI      MAR:822F
      8230      LD HL,00000H
      AF:7740 BC:0000 DE:1357
      HL:2136 M:86   SP:1CD7
      IX:0000 IY:1D00
                CSPZ:0001
```

Bild 4.7: Displayausschrift im Testmodus (Bsp.)

Nach der Ausschrift "TEST" blinkt der Cursor. Abhängig von der gedrückten Kommandotaste wird auf dieser Position kurzzeitig T, N oder L geschrieben, gefolgt von einem entsprechenden Text. (siehe Tabelle 4.4)

Danach wird die Globalmarke mit ihrer Adresse angezeigt. Die nächste Zeile beinhaltet die Adresse des nächsten auszuführenden Befehls und den zur Bearbeitung anstehenden Befehl mit Markenangaben. Die Displayausschrift setzt sich mit der Angabe der Registerinhalte in Doppelregisterdarstellung fort. Auf der Speicherzelle M wird der Inhalt der vom HL-Register gezeigten Zelle angegeben. Das Flagregister F gibt die Belegung der vier Flags Carry, Sign, Parity und Zero an.

Nach Betätigen der Taste R können durch Rücken des Cursor und Überschreiben die Inhalte der Register, außer dem Stackpointer und Programmzähler, geändert werden.

Die Flags werden durch Überschreiben der Byteausschrift geändert. Jede Änderung ist mit "ENTER" abzuschließen.

Kommandotaste	Funktion	Text auf dem Display	Bedeutung
OFF	Eintritt in den Anzeigemodus		
T	Testlauf bis Haltepunkt oder Ausführung eines Befehls 1)	SCHRI	Schritt
N	Testlauf bis zum nächsten notierten Befehl, Abarbeitung von Schleifen und Unterprogrammen 2)	SCHLE	Schleife
L	Start des Echtzeitlaufs 3)	LAUF	
R...ENTER	Register überschreiben	REG.K	Registerkorrektur

Tab. 4.4: Kommandos im Testmodus

1) Beim Testlauf wird das Anwenderprogramm nach jedem Befehl von einem im Testmodus initialisierten Interrupt (Test-interrupt) unterbrochen. Soll ein Testlauf bis zu einem Haltepunkt ausgeführt werden, so wird in diesem Interrupt die Haltebedingung überprüft. Ist sie nicht erfüllt, so erfolgt der Rücksprung aus dem Interrupt in das Anwenderprogramm, Abarbeitung eines Befehls, wieder Unterbrechung usw. Ist die Haltepunktbedingung erfüllt oder kein Haltepunkt angegeben worden (Kommando "T" ohne vorherige Vereinbarung eines Haltepunktes), so werden die Registerbelegungen und der anstehende Befehl ausgeschrieben und der Editor befindet sich wieder im Testmodus.

Während des Testlaufes wird nach jedem Anwenderbefehl der Programmzählerstand auf den Bildschirm geschrieben. Damit ist er visuell verfolgbar. Die Unterbrechung des Anwenderprogramms nach jedem Befehl dauert etwa eine Millisekunde. Damit wird das Anwenderprogramm etwa 200...500 mal langsamer abgearbeitet als im Echtzeitlauf. Der Testlauf kann mittels Betätigen der Taste "OFF" unterbrochen werden.

Ein Haltepunkt kann im Anzeigemodus mit dem Kommando "H" vereinbart werden. Die Anzeigeadresse gilt dann als Haltepunkt (siehe Tabelle 4.3, Erläuterung 8).

2) Beim Kommando "N" wird die Adresse des nächsten notierten Befehls als Haltepunkt vereinbart und der Testlauf gestartet. War der anstehende Befehl ein CALL-Befehl, so wird erst nach dem Unterprogrammrücksprung angehalten. War der anstehende Befehl ein bedingter Sprung, so wird angehalten, wenn die Sprungbedingung nicht erfüllt ist. Somit können auch Schleifen im Testverlauf abgearbeitet werden, deren Wiederholung bei einem erfüllten bedingten Sprung erfolgt (z.B. DJNZ als Schleifenbefehl).

Der Testlauf kann mit "OFF" unterbrochen werden.

3) Der Echtzeitlauf wird nach Abarbeitung des Befehls RST 7 (Befehlscode 0FFH) unterbrochen. Mit der Taste "BREAK" kann der Testlauf abgebrochen werden.

Es werden folgende Bedingungen gestellt, welche das Anwenderprogramm zu erfüllen hat, wenn es getestet werden soll:

- Der Interruptmodus 2 darf nicht verlassen werden.
- Das I-Register (I= 0DH) in der CPU muß seinen Inhalt behalten.
- Es darf nicht auf die geräteinternen Adressen außer Schreibsperre zugegriffen werden (siehe Abschnitt 3). Der für den Testinterrupt benötigte CTC-Kanal (Adresse 0F3H) darf nicht anderweitig verwendet werden. Die Startadresse des Testinterrupts auf dem durch den Interruptvektor adressierten Speicherplatz darf nicht überschrieben werden.
- Initialzeilen und Arbeitsspeicher für Tastatur- und Bildschirm dürfen nicht unzulässig geändert werden!
- Die Strichregister dürfen nicht benutzt werden.

Weiterhin gelten folgende Bedingungen für die Unterbrechung des Anwenderprogramms im Testlauf:

- Nach einem Befehl DI bis nach dem Befehl, der auf die Interruptfreigabe mit EI folgt, läuft das Anwenderprogramm im Echtzeitlauf und wird nicht unterbrochen.
- Nach einer Interruptannahme bis nach dem Befehl, der auf EI folgt, läuft das Anwenderprogramm im Echtzeitlauf und wird nicht unterbrochen.

4.5. "RAM"

Das RAM-Programm ist einmal über den Anzeigemodus im Editor als auch direkt über die Programmtabelle erreichbar. Nach Abruf aus dem Betriebssystem werden in hexadezimaler Aufschrift 64 Datenbytes ausgeschrieben.

```
aaaa          00 01 02 03 04 05 06 07
bbbb>_       08 09 0A 0B 0C 0D 0E 0F
              10 11>12 13 14 15 16 17
              18 19 1A 1B 1C 1D 1E 1F
              20 21 22 23 24 25 26 27
              28 29 2A 2B 2C 2D 2E 2F
              30 31 32 33 34 35 36 37
              38 39 3A 3B 3C 3D 3E 3F
```

Bild 4.8: RAM-Programmausschrift (Bsp.)

Die Anzeige erfolgt dynamisch, d.h. der aktuelle Inhalt des Speichers wird beständig angezeigt.

Die Anfangsadresse wird im Bild links oben ausgeschrieben, im Bild 4.8 ist sie mit aaaa gekennzeichnet. Diese Anfangsadresse wird beim Verlassen des RAM-Programms im Speicher festgehalten und steht bei erneutem Aufruf wieder zur Verfügung. Die Anfangsadresse kann durch das Kommando "G aaaa ENTER" sowie die Kommandos "↑", "↓", "←", "→" und "ENTER" geändert werden (vgl. Kommandotabelle 4.5).

Die vor dem Zeichen ">" stehende Adresse bezeichnet die RAM-Zelle, auf welche sich die nächste Eingabe bezieht. Das Zeichen ">" ist auf der Datenbyteauschrift wiederzufinden und kennzeichnet nochmals das betreffende Byte. Mit den Kommandos "←" und "→" kann diese Eingabeadresse decrementiert bzw. incrementiert werden, damit kann man voranstehende bzw. nachfolgende Bytes ansprechen. Wenn das erste bzw. letzte Byte mit ">" bezeichnet wird und die Eingabeadresse wird weiter decrementiert bzw. incrementiert, so würde das bezeichnete Byte aus dem Anzeigefeld verschwinden.

Um das zu verhindern, wird die Anzeigeadresse um 8 decremen-
tiert bzw. incrementiert, so daß die Anzeige einer Zeile
nach unten bzw. oben bewegt wird. Das entspricht auch der
Wirkung der Kommandos "↑" und "↓" auf die Anzeigeadresse.
Dadurch befindet sich das bezeichnete Byte immer im Anzeige-
feld. Selbiges gilt auch, wenn mittels "ENTER"-Kommando auf
das letzte angezeigte Byte Daten eingegeben werden und die
Eingabeadresse dementsprechend danach incrementiert wird.

Die Tastenkommandos "←", "→", "↑", "↓" wirken repetierend,
d.h., wenn die Tasten länger betätigt werden, (> 600 ms) so
erfolgt eine automatische Neuauslösung der Tastenfunktion.
Im RAM-Programm-Grundzustand (nach Programmaufruf) blinkt
der Cursor nach dem Zeichen ">". Auf dieser Position können
Datenbytes in Form von Hexaziffern eingetragen werden.
Falsche Eingaben können mit der Taste "CL" gelöscht werden.
Der Cursor kann nicht aus dem zulässigen Bereich herausbe-
wegt werden. Die Eingabe eines Bytes wird mit der Taste
"ENTER" vollzogen. Sind die Daten vollständig angegeben,
werden sie auf die Eingabeadresse geschrieben, diese wird
incrementiert und eine nächste Eingabe kann folgen. Die
gleichen Daten können durch weiteres Betätigen von "ENTER"
auf die nächsten Eingabeadressen gegeben werden.

Das Tastenkommando "I" dient zum Eingeben von Daten im Block
bis zu einer Endadresse. Nach dem Betätigen des "I" blinkt
der Cursor in der dritten Zeile. Dort wird die hexadezimal
mit 4 Ziffern anzugebende Endadresse verlangt. Dieser Bedien-
zustand kann mittels "OFF" wieder verlassen werden (z.B. bei
Fehleingaben). Ansonsten wird die Eingabe der Endadresse mit
"ENTER" abgeschlossen. Ist diese vollständig angegeben und
größer als die Eingabeadresse, so wird der gesamte Speicher-
bereich bis zur Endadresse mit den vollständig angegebenen
Daten (nach >) belegt. Ist die Endadresse nicht vollständig,
so wird sie weiter verlangt. Sind die Daten nicht vollstän-
dig angegeben oder ist die Endadresse kleiner als die Ein-
gabeadresse, so wird ohne Aktion der Grundzustand einge-
nommen.

Das Tastenkommando "M" dient dem Verschieben von RAM-Bytes. Nach dem Betätigen von "M" werden jeweils durch Kursorrücken gekennzeichnet, zwei Adressen verlangt. Dies sind nacheinander die Quellanfangs- und die Quellendadresse. Die Adressen sind jeweils mit 4 Hexaziffern anzugeben und die Eingabe ist mit "ENTER" abzuschließen.

Sind beide Adressen angegeben und die Quellendadresse ist größer als die Quellanfangsadresse, so wird der Speicherinhalt ab der Quellanfangsadresse bis ausschließlich der Quellendadresse auf den Bereich ab der Eingabeadresse verschoben (umgeladen). Dabei bleibt der Quellbereich dem Anwender erhalten. Dieser kann im RAM-Modus mit dem Kommando "I" gelöscht werden, indem der Quellbereich mit den Daten FF belegt wird.

Es ist auch zulässig, daß sich Quell- und Eingabebereich teilweise überdecken. Das RAM-Programm realisiert intern das Umladen der Daten vom Anfang oder vom Ende beginnend. Auch bei dem Kommando "M" kann mittels Betätigen von "OFF" ohne Aktion der Grundzustand wieder eingenommen werden, womit Fehlbedienungen wieder rückgängig gemacht werden.

Das RAM-Programm kann im Grundzustand durch Betätigen der Taste "OFF" verlassen werden. Es wird das aufrufende Programm wieder eingenommen.

Kommando- taste	weitere Angaben	verbale Beschreibung der Wirkung
↑	-	Anzeige- und Eingabeadresse um 8 decrementieren
↓	-	Anzeige- und Eingabeadresse um 8 incrementieren
←	-	Eingabeadresse decrementieren
→	-	Eingabeadresse incrementieren
0...9 A...F CL	-	Eingabedaten angeben
ENTER	-	vollst. Eingabedaten auf Ein- gabeadresse schreiben, Eingabe- adresse incrementieren
I	eeee ENTER	vollst. Eingabedaten auf Bereich ab Eingabeadresse bis ausschließl. eeee eintragen 1)
M	qqqq ENTER eeee ENTER	Daten ab Adr. qqqq bis ausschl. eeee auf Bereich ab Eingabeadr. verschieben 1)
G	aaaa ENTER	Anzeige der Zieladresse aaaa 1)
OFF	-	Verlassen des RAM-Programms
I, M, G	OFF	Einnahme des Grundzustandes ohne Aktion

Tab. 4.5: Kommandos in RAM-Programm

1) Als Adreßeingabe wird eine vierstellige hexadezimale Zahl verlangt. Es werden keine führenden Ziffern gefordert, da in Verbindung mit diesem Kommando nur Adreßeingaben folgen.

Das RAM-Programm dient zum Anzeigen und zur Eingabe von Daten entsprechend einem Anwenderproblem, wenn keine spezielle Eingaberoutine vorliegt. Es ermöglicht den Zugriff auf alle RAM-Bereiche, auch auf interne Betriebssystembereiche ohne Verhinderung von programmzerstörenden Fehlbedienungen. Die Benutzung sollte also bedacht erfolgen!

Durch die dynamische Anzeige und die Möglichkeit der Vorbereitung der Dateneingabe (Ausführung gezielt mit "ENTER") kann auch in laufende Programme (Interruptebenen) eingegriffen werden. Das ist ein besonderes Merkmal der Echtzeitfähigkeit dieses Programms.

4.6. "EPROG"

Der Aufruf der EPROM-PROGRAMMIERUNG erfolgt über die Programmtable. Das Display meldet sich mit der Ausschrift "EPROM-PROGRAMMIERUNG".

Als erstes muß die Anfangs- und Endadresse des Speicherbereiches, der auf den EPROM übertragen werden soll, eingegeben werden. Danach wird die Anfangsadresse des EPROM gefordert. Die EPROM-Anfangsadresse muß im Adreßbereich des EPROM zwischen 0000H und 03FFH liegen. Die Endadresse wird nach Ausführung intern errechnet und auf dem Bildschirm angezeigt. Wird eine Endadresse berechnet, die den EPROM-Bereich überschreitet (größer 03FFH), so erfolgt eine Ausschrift "FEHLER". Die eingegebenen Adressen können nochmals korrigiert werden, wobei jede neue Eingabe mit Tastendruck "ENTER" abgeschlossen wird.

Nach Aufforderung durch das Betriebssystem schließt sich das Stecken des EPROM in die Programmierfassung an. Der Anschluß 1 an der EPROM-Programmierfassung ist besonders markiert.

Auf dem Display erscheinen danach die Namen der Programme, die mit dem Zeichen "*" ausgewählt werden können.

Folgendes Schema soll den Ablauf im Programm "EPROM-Programmierung" demonstrieren:

- Ablauf:
- Anfangsadresse des zu übertragenden Speicherbereiches eingeben
 - Betätigen der Taste "ENTER"
 - Endadresse des Speicherbereiches eingeben
 - Betätigen der Taste "ENTER"
 - Anfangsadresse des EPROM eingeben
 - Tastendruck "ENTER"
 - EPROM stecken
 - Tastendruck "ENTER"
 - Funktion auswählen, (Anfangsbuchstabe der Funktion eingeben)
- T - TEST
V - VERGL
P - PROG
L - LESSEN

Die Adressen sind vierstellig, hexadezimal einzugeben.

Programmfunktionen:

- TEST: prüft, ob der EPROM beschrieben oder leer ist
VERGL: Inhalt von EPROM und vom RAM werden verglichen
PROGR: Übertragen der Information vom angegebenen Speicherbereich auf den EPROM
LESEN: Übertragen der Information vom EPROM in den RAM

Auswahl TEST: :

Ist der EPROM leer (Datenbelegung FF), so erscheint wieder die Bildschirmausschrift "EPROM STECKEN". Nach "ENTER" kann eine neue Funktion ausgewählt werden. Wird eine Abweichung festgestellt, so werden die Adresse und der zugehörige Inhalt für RAM und EPROM protokolliert. Nach Betätigen der Taste "ENTER" wird der nächste Fehler protokolliert usw. Die Anzahl der Fehler wird ebenfalls angezeigt. Durch Drücken der Taste "L" ("LAUF") wird der gesamte EPROM getestet und die Fehleranzahl angezeigt. Ist "TEST" beendet, meldet sich das Steuerprogramm mit "EPROM-STECKEN". Rechts erscheint die Fehlerzahl.

Auswahl VERGL:

Tritt kein Fehler auf, meldet sich das Steuerprogramm mit "EPROM STECKEN". Wird eine Abweichung festgestellt, so werden die fehlerhafte Adresse und der zugehörige Inhalt für RAM und EPROM angezeigt. Die weitere Verfahrensweise ist analog der Betriebsart "TEST".

Auswahl PROGR:

Der gesteckte EPROM wird zuerst auf Programmierfähigkeit überprüft. Das heißt, es darf kein Bit mit 0 belegt auftreten, das mit 1 programmiert werden soll. Die Fehlerprotokollierung erfolgt adäquat zur Betriebsart "TEST". Danach wird auf jeden Fall (auch bei Fehler) ein "?" ausgegeben. Nach nochmaligem Betätigen der "ENTER"-Taste erfolgt dann die Programmierung. Wird eine andere Taste gedrückt, wird nicht programmiert. Auf dem Display werden während des Programmierprozesses die Zyklenzahl (62H rückwärts bis 01H) und die Fehleranzahl angegeben. Nach Beendigung meldet sich das Steuerprogramm mit der Aufschrift "EPROM STECKEN".

Auswahl LESEN:

Nach beendetem Lesen erfolgt der Rücksprung ins Steuerprogramm.

Durch Betätigen der Taste "OFF" erfolgt der Rücksprung in die MENÜTABELLE.

5. Softwareschnittstellen im Betriebssystem

5.1. Displayprogramme

Folgende Programme dienen dem Beschreiben des Displays

Bezeichnung	Adr. (hex.)	Eingaberegister	Ausgaberegister	Funktion
BSN	000B	A	-	Zeichen auf Cursorposition schreiben bzw. Steuerzeichen ausführen
BSS	001B	A, DE	F	Zeichen schreiben
BSL	0023	DE	A	Zeichen lesen
BKS	002B	DE	-	Kursor setzen
BKL	0033	-	DE	Kursorposition lesen
HBS	0300	A, DE	-	Byte schreiben
HDS	03DD	HL, DE	-	Doppelbyte ausschreiben
HBA	0429	DE	A, F	Byte lesen (mit Anfordg.)
HDA	0432	DE	HL, F	Doppelbyte lesen (mit Anforderung)
ZLE	03E8	A, DE	HL, AF	Zahl lesen (bit 7, A=1 Hexadezimal bit 7, A=0 Dezimal)
BSZ	0375	A, DE, C	-	Zeichen in Zeile schreiben
BBS	0270	-	F	Taste auf Bildschirm bis ENTER/OFF
BBZ	0364	DE, C	F	Taste auf Zeile bis ENTER/OFF
TXS	0465	HL	HL, A="?"	Text ausschreiben ab KPS
TNS	0466	HL, A	HL, A="?"	numerierten Text ausschreiben ab KPS
BSA	047E			Display abschalten
BSK	04AC			Menüausschrift auf Display schreiben

"?" = Unbestimmt

Tab. 5.1: Displayprogramme

KPS = Cursorposition

Bei allen Unterprogrammen gilt: Die nach außen nicht benutzten Register werden nicht verändert. Es werden einheitliche Übergabeschnittstellen verwendet:

DE gibt eine Displayposition an

D = Zeilennummer

E = Spaltennummer

Die Zählung erfolgt von 0...7 (D) bzw. 0...31 (E).

Ist $E \geq 32$, so wird die nächste Zeile angewählt.

Ist $32 \cdot D + E \geq 256$, so wird die letzte Displayposition angesprochen.

A enthält bei den zeichenweise arbeitenden Programmen einen Zeichencode nach der ASCII-Norm.

Das Programm BSN kann auch Steuerzeichen verarbeiten

(Tab. 5.3). Es werden nur die Zeichencodes 20H...5FH

(ASCII) dargestellt. Die Zeichen mit den Codes 60H...7FH

werden in solche der Zeichencodes 40H...5FH umgewandelt.

Das bedeutet, Kleinbuchstaben werden durch Großbuchstaben dargestellt.

A enthält in den Programmen HBS und HBA eine 8 Bit Hexazahl

A enthält in TNS die Textnummer

HL beinhaltet in den Programmen HDS und HDA eine 16 bit Hexazahl

HL enthält in den Programmen TNS und TXS die Texttabellenanfangsadresse vor Aufruf des Programms und die Adresse des Bytes nach dem Text bei Verlassen des Programmes.

C enthält bei den Programmen BSZ und BBZ die Zahl der schreibbaren Zeichen (Begrenzung der Schreibzeile nach rechts).

Zehner Bit 7 = 1 = 0	Einer																	
	0	1	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
A0	20	␣	!	"	#	␣	%	%	'	()	*	+	,	-	.	/	
B0	30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?	
C0	40	◇	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	
D0	50	P	Q	R	S	T	U	V	W	X	Y	Z	[0]	␣	-	

Tab. 5.2: Zeichencode im MC 80;
entspricht teilweise dem ASCII-Code

Code (hex.)	CTR Zeichen	Reaktion
02	cB	Kursor an Bildanfang
03	cC	Tabulator
08	cH	Kursor nach links
09	cI	Kursor nach rechts
0A	cJ	Kursor nach unten
0B	cK	Kursor nach oben
0C	cL	Bild löschen und Kursor an Bildanfang
0D	cM	Kursor an Zeilenanfang
1A	cZ	Kursor nach links, Zeichen streichen (clear)
18	cX	Zeile löschen, Kursor an Zeilenanfang

c-Betätigen der CTR-Taste

Tab. 5.3: Steuercode im MC 80

Den Code der nicht von Tabelle 5.3. umfaßten Funktionstasten zeigt folgende Übersicht:

Code (hex.)	CTR Zeichen	Funktionstaste
06	cF	ENTER
16	cV	OFF

Tab. 5.4: Code der Funktionstasten ENTER und OFF

Einige Unterprogramme können auch Steuerzeichen verarbeiten, die jeweils eine in Tabelle 5.3 beschriebene Reaktion auslösen. Die Programme BSN, TXS und TNS verstehen alle Steuerzeichencodes nach Tabelle 5.3 und das UP BSZ akzeptiert die Codes: Ø3, Ø8, Ø9, 1A.

Bei einigen Unterprogrammen, welche die Tastatur aktivieren, können mit Hilfe der Controllebene Steuerzeichen erzeugt werden. Dies geschieht durch das gleichzeitige Betätigen der CTR-Taste und der entsprechenden Buchstabentaste. Bei der Arbeit mit dem Programm BBS ist die Erzeugung aller Steuerzeichen möglich und vom Programm BBZ werden die Steuerzeichen "TAB", "CL", "—" , "—" innerhalb des eingegrenzten Bereiches zugelassen.

Beispiel: Wählen Sie aus der Menütabelle die Funktion "TASTENTEST", welche ebenfalls alle Steuerzeichen versteht, aus und beschreiben Sie das Display mit einigen Zeichen. Danach erzeugen Sie mit Hilfe der Tastatur alle Steuerzeichen nach Tabelle 5.3 und überprüfen die ausgelöste Reaktion.

Nähere Beschreibung der Programme:

BSN: schreibt das im A-Register enthaltene Zeichen nach Tab. 5.2 ab Cursorposition auf den Bildschirm und rückt den Cursor, Steuerzeichen werden ihrer Bedeutung nach ausgeführt. Den Steuercode des MC 80 zeigt Tabelle 5.3. Das Programm verändert keine Registerinhalte und keine Flags.

BSS: schreibt das im A-Register enthaltene Zeichen auf die Displayposition, die durch das D-Register (Zeile) und das E-Register (Spalte) angegeben wird, unabhängig von der Cursorposition. Liegt der Zeichencode zwischen ØØ...1FH oder 8ØH...FFH, wird das Programm mit Cy-Flag = 1 ohne Aktion verlassen. Im anderen Fall ist Cy = Ø. Die anderen Flags werden undefiniert gesetzt.

BSL: liest das Zeichen, das auf der durch DE adressierten Displayposition steht und übergibt es im A-Register. Das A-Register bekommt einen Inhalt zwischen 2ØH...5FH.

BKS: setzt den Cursor auf die durch das DE-Register angegebene Position, der alte Cursor wird gelöscht.

- BKL: ermittelt die aktuelle eingenommene Kursorposition und übergibt den Wert im DE-Register (Zeile/Spalte)
- HBS: schreibt einen Zahlenwert aus dem A-Register in zwei Hexaziffern auf die Position DE des Display, E wird um 2 incremented, d.h. DE adressiert nach Verlassen des Unterprogrammes die erste Zeichenposition nach der Hexazahl, D wird nicht verändert.
- HDS: schreibt den Zahlenwert, der im HL-Register enthalten ist, im hexadezimalen rechtsbündigen (Komma rechts) Format auf das Display, die Position der ersten Ziffer wird von DE angegeben, der Wert von E wird um 4 incremented.
- HDA: fordert einen hexadezimalen Zahlenwert in Form von 4 Hexaziffern auf dem Display an, DE gibt die Position der ersten Ziffer an. Steht auf dem Display ab DE keine Folge von 4 Hexaziffern, wird die Tastatur aktiviert. Ist die Hexazahl vollständig angegeben und erfolgt der Abschluß der Eingabe mit "ENTER", wird das Programm mit dem Zahlenwert im HL-Register und Cy-Flag = 0 verlassen. Bei Betätigung der "OFF"-Taste wird das Cy-Flag = 1 und der Inhalt des HL-Registers Null gesetzt.
- HBA: fordert in gleicher Weise wie HDA eine Hexazahl mit zwei Ziffern an, die an A (statt HL) übergeben wird.
- ZLE: liest eine Zahl dezimal oder hexadezimal vom Display. Die maximal zu lesende Zifferanzahl wird dem Programm im A-Register in den Bit 0...6 übergeben (A=1...5). Bit 7 spezifiziert die Zahl.
 Bit 7 = 0: Lesen als Dezimalzahl
 Bit 7 = 1: Lesen als Hexazahl
- Der gelesene Zahlenwert wird im HL-Register übergeben (max. 65535 = FFFFH).
 Die Flags und das A-Register enthalten eine Rückmeldung über Stellenzahl und Art der gelesenen Zahl nach folgender Tabelle:

Cy	Z	A	HL	Kommentar
0	0	1)	Zahl	Zahl mit allen Ziffern gelesen
0	1	1)	00	Zahl ist Null, mit allen Ziffern
1	0	2)	Zahl	Zahl mit weniger als angegebenen Ziffern gelesen
1	1	0	00	auf 1. abgefragter Displayposition steht keine Ziffer

1) A enthält Zeichencode (ASCII) des letzten Zeichens
 2) A enthält Zahl der gelesenen Ziffern (binär)

Tab. 5.5: Registerbelegung nach ZLE

BSZ: schreibt das in A enthaltene Zeichen auf die Cursorposition und rückt den Cursor weiter, wenn sich der Cursor auf einer Position der Zeile D und der Spalten E bis E+C befindet. Ansonsten wird bei Aufruf der Cursor auf den Schreibbereichsanfang bzw. das Schreibbereichsende gesetzt. Dabei gibt C die Zahl der innerhalb der Zeile D maximal zu schreibenden Zeichen an. Als Steuerzeichen werden nur akzeptiert: "TAB", "CL", "←", "→"

BBS: ermöglicht das Beschreiben des Display mit der Tastatur. Die Rückkehr aus dem Programm erfolgt bei Betätigung der Tasten "ENTER" oder "OFF". Die Markierung, durch welche Taste die Rückkehr ausgelöst wurde, erfolgt im Cy-Flag:

Cy	Kommentar
0	ENTER betätigt
1	OFF betätigt

Tab. 5.6

Von der Tastatur werden alle Zeichen (auch Steuerzeichen) akzeptiert, die das Programm BSN versteht, lt. Tab. 5.2 und Tab. 5.3.

BBZ: ermöglicht das Beschreiben einer Displayzeile mit der Tastatur. Die Eingrenzung der Zeile erfolgt wie bei dem Programm BSZ, ebenso die Auswertung der Zeichen, die von der Tastatur erzeugt werden. Das Programm wird wie BBS nach Betätigung von "ENTER" oder "OFF" verlassen.

Beide Programme, BBS und BBZ, blockieren den Prozessor in der zugeordneten Interruptebene. Ein Echtzeitbetriebssystem würde in der Wartezeit ein Abarbeiten niedriger Interruptebenen ermöglichen. Höhere Interruptebenen werden nicht blockiert. Die Programme sind deshalb nicht uneingeschränkt für Echtzeitaufgaben einsetzbar. In einem Echtzeitbetriebssystem sind adäquate Programme mit den gleichen Softwareschnittstellen erstellbar, die einen Ersatz ermöglichen.

TXS: Das Programm schreibt einen Text auf das Display. Der Text steht im Speicher, der Anfang des Textpuffers wird mit HL gezeigt. Der Text muß aus einer Folge von Zeichen lt. Tab. 5.2 und Tab. 5.3 bestehen. Das erste Byte eines jeden Textes wird mit bit 7=1 gekennzeichnet. Der für das erste Byte anzugebende Code ist ebenfalls aus Tabelle 5.2 ersichtlich. Nach Abschluß des Programmes zeigt HL das auf den Text folgende Byte. Das A-Register ist unbestimmt. Das Textende wird durch ein Datenbyte mit Bit 7=1 (z.B. 80H, 0FFH) gekennzeichnet.

- TNS: Das Programm schreibt einen Text aus einer umfassenden Texttabelle auf das Display. Die Nummer des Textes wird im A-Register übergeben (A = 0...n). HL zeigt den Anfang der Texttabelle. Die Einzeltexte beginnen jeweils mit Bit 7=1. Es wird der Text ausgeschrieben, der mit dem Bit 7=1 gekennzeichneten Byte beginnt, nachdem in der Texttabelle ab den durch HL gezeigten Anfang A Bit 7=1 aufgetreten ist. Nach Abschluß des Programmes zeigt HL das auf den Text folgende Byte. Das A-Register ist unbestimmt. Die Texttabelle muß durch ein Datenbyte mit Bit 7=1 abgeschlossen sein.
- BSA: tastet den Bildschirm dunkel, auch der Cursor blinkt nicht. Die Wirkung des Programmes BSA ist mit programmgesteuerten Beschreiben des Bildschirms aufgehoben.
- BSK: Dieses Programm stellt die Betriebssystem-Menüverwaltung dar. Es organisiert die Menüauschrift und die Menübedienung. Ein Sprung auf BSK ermöglicht dem Anwender jederzeit die Rückkehr zur Menüdarstellung.

Einige Beispiele zu Displayprogrammen:

Die folgenden Beispiele sollen mittels Objektcodeeditor (Abschnitt 4.4.) eingegeben werden.

- Hinweise:
- Eingabe eines Programmes im Quellcode-Eingabemodus auf einen freien RAM-Bereich
 - symbolische Adressierung der verwendeten Unterprogramme oder direkte Adreßeingabe der UP
z.B. BSN: EXT 000BH
 - Binden des Programmes
 - Programmstart
z.B. Anwahl über Menütabelle (siehe Punkt 4.) oder Start Echtzeitlauf "L" im Testmodus

```

AAA:LD A,00CH          ; Steuerzeichen Display löschen
    CALL BSN           ; Steuerzeichen ausführen
    LD A,41H
    CALL BSN           ; ASCII:A
    LD A,0AH          ; neue Zeile
    CALL BSN
    LD A,20H          ; ASCII:Leerzeichen
    CALL BSN
    LD A,42H          ; ASCII:B
    CALL BSN
    LD A,0AH          ; neue Zeile
    CALL BSN
    LD A,43H          ; ASCII:C
    CALL BSN
    JR #

```

BSN:EXT 000BH

Es entsteht die Displayausschrift A B C _

```

BBB:LD HL,TBU
    CALL TXS
    JR #

```

```

TBU:DB 8CH
    DB 41H
    DB 0AH
    DB 20H
    DB 42H
    DB 0AH
    DB 43H
    DB 0FFH

```

TXS:EXT 0465H

Dieses Programm bewirkt die gleiche Ausschrift

```

CCC:LD A,ØCH
      CALL BSN           ; Display löschen
      LD DE,Ø31ØH       ; Bildschirmposition laden
      LD A,5AH          ; ASCII:Z
      CALL BSS          ; Zeichen schreiben
      JR #
BSN:EXT ØØØBH           ; nur wenn noch nicht vereinbart
BSS:EXT ØØ1BH

```

Auf Zeile 4 in Spalte 16 wird der Buchstabe "Z" geschrieben.
Der Cursor befindet sich am Bildanfang.

```

DDD:LD A,ØCH
      CALL BSN           ; Display löschen
DD4:LD DE,1Ø2H         ; Position laden
      LD C,5            ; Zahl der schreibbaren Zeichen
      CALL BEZ          ; Eintragen von max. 5 Zeichen ab DE
      JRC DDD          ; bei OFF wieder Anfangsbed. schaffen
      LD A,5            ; bei ENTER:
      CALL ZLE          ; Dezimalzahl lesen, Bit 7 in A=0
      JRNC DD2         ;
      JRZ DDD          ; keine führende Ziffer
DD2:LD DE,2Ø2H         ; Zahl gelesen, neue Position laden
      CALL HDS          ; hexadezimal darunter ausschreiben
      LD DE,1ØØH
      CALL BKS          ; Cursor vor Eingabeposition setzen
      JR #
BSN:EXT ØØØBH
BEZ:EXT Ø364H
ZLE:EXT Ø3E8H
HDS:EXT Ø3DDH
BKS:EXT ØØ2BH

```

Bei einer Eingabe von max. 5 Ziffern und "ENTER"-Betätigung wird diese Zahl hexadezimal darunter ausgeschrieben. Der Cursor wird vor den Eingabebereich gesetzt. Bei "OFF"-Betätigung kann die Eingabe wiederholt werden.

Ebenso werden bei Angabe keiner führenden Ziffer (z.B. Hexazahl) wieder die Anfangsbedingungen hergestellt. Wird der letzte Befehl in "JR DD1" geändert, blinkt der Cursor durch die Wirkung von BBZ wieder auf der Position DE=102H und ein fortlaufendes Überschreiben kann erfolgen.

5.2. Tastaturprogramm

Bezeichnung	Adresse	Ausgaberegister	Funktion
TST	0013H	AF	Tastaturabfrage

Tab. 5.7: Tastaturprogramm

Den momentanen Zustand der Tastatur kann man nach Aufruf des Unterprogramms TST erfahren. Ist zur Abfrage eine Taste gedrückt, steht der Zeichencode im A-Register. Dem Zeichencode ist die ASCII-Norm zugrunde gelegt. Wurde keine Taste betätigt, enthält das A-Register den Wert 0.

Die Belegung von Zero- und Carry-Flag dient der dynamischen Kennzeichnung.

A	Cy	Z	Kommentar
00	0	1	keine Taste betätigt
ASCII	1	0	Taste neu gedrückt, d.h. nach Betätigung das <u>erste Mal</u> abgefragt
ASCII	0	1	Taste betätigt, <u>wurde</u> aber <u>bereits abgefragt</u>
ASCII	1	1	Taste meldet sich repetierend, d.h. sie wurde bereits abgefragt, wird aber länger als ca. 600 ms betätigt.

Tab. 5.8: Registerbelegung nach TST

Mit dieser Flagauswertung ist es also möglich:

- das einmalige Niederdrücken der Taste zu erkennen:
Zero auswerten
- den ständigen Tastendruck zu erkennen:
A-Register auswerten
- die Taste repetierend wirken zu lassen:
Carry auswerten.

Um diese Wirkung der Tastatur selbst zu erkennen, dienen folgende Programmbeispiele, die mittels Objektcodeeditor eingegeben werden sollen. (Abschn. 4.4.)

1. AAA:CALL TST
CANZ BSN ; Programm BSN vgl. 5.1.
JR AAA ; Ergebnis: Der Bildschirm wird mit dem jeweiligen Tastenzeichen nur beim Niederdrücken der Taste beschrieben

2. BBB:CALL TST
CAC BSN
JR BBB ; Der Bildschirm wird mit dem gleichen Zeichen repetierend beschrieben, wenn die Taste länger als 600 ms gedrückt bleibt, die Repetierfrequenz ist etwa 10 Zeichen/Sekunde

3. CCC:CALL TST
CALL BSN
JR CCC ; Der Bildschirm wird ständig wiederholt bei Tastendruck mit dem entsprechenden Zeichen beschrieben, so schnell wie es die Rechenzeit der Programme TST und BSN erlaubt.

4. DDD:CALL TST
CANZ BSN
LD DE,71FH
CALL BSS
JR DDD

Das Zeichen wird bei Niederdruck geschrieben; außerdem wird das Zeichen der gedrückten Taste auf der letzten Position des Bildschirms protokolliert. Mit diesem Programm kann man auch überprüfen, wie die Tastatur auf Betätigen mehrerer Tasten gleichzeitig und abwechselnd reagiert: Das Zero-Flag wird nur rückgesetzt, wenn während einer Tastaturabfrage der Zustand "keine Taste betätigt" vorliegt. Wenn die Tastaturabfrage beständig läuft, d.h. öfter als die Bedienung der Taste, wird damit die zuerst gedrückte Taste ausgeschrieben. Das gleichzeitige Betätigen mehrerer Tasten danach hat also keinen Einfluß bei Anwendung nach Beispiel 1.

Werden zwei Tasten abwechselnd gedrückt, so enthält das A-Register auch den Code der nachfolgend betätigten Taste. Dieser erscheint auf dem Bildschirm bei Anwendungen nach Beispielen 2 und 3. Werden gleichzeitig zwei oder mehrere Tasten betätigt, dann beinhaltet das A-Register den Code einer der betätigten Tasten nach einer bestimmten internen Priorität. Ein vollkommenes Fehlverhalten wird also nicht ausgelöst. Im Beispiel 4 wird die zuerst betätigte Taste nach Niederdruck ausgeschrieben, auf der letzten Bildschirmposition erscheint aber der aktuelle Inhalt des A-Registers. Da die Tastaturabfrage nicht zu einer Warteschleife auf Tastenbetätigung (Eigenschleife) führt, kann sie in entsprechende Echtzeitprogramme eingebunden werden. Der Entscheidung, was bei Nichtbetätigen der Taste zu machen ist, bleibt dem Anwender überlassen.

5.3. Ändern der Tabulator-Positionen

Eine Änderung der Tabulator-Positionen kann der Anwender durch Eingriff in den Betriebssystem-RAM vornehmen. Auf folgenden Zeilen stehen Bildschirmpositionen (Spalte); auf denen sich in jeder Zeile gleich der Tabulator befinden kann:

TAB 1	ØD 18
TAB 2	ØD 19
TAB 3	ØD 1A
TAB 4	ØD 1B
TAB 5	= nächste Zeile (Inhalt 2ØH !); ØD 1C (nicht ändern)

Tab. 5.9: Zellen für Tabulatorpositionen

In diese Zellen können in aufsteigender Reihenfolge hexadezimale Zahlen zwischen Ø1 und 1F eingetragen werden, die die neuen Tabulatorpositionen bestimmen.

Nach RESET werden in diese Zellen wieder die Tabulatorpositionen der Initialisierung eingetragen.

5.4. Benutzung der Grafik des Display

Bildwiederholtspeicher für die maximal 4 analogen Kurvenzüge, die auf dem Display darstellbar sind, können sich in jedem beliebigem RAM-Bereich befinden. Die Abbildung erfolgt so, daß die Adressen die Abszissenwerte darstellen und der Inhalt der adressierten Bytes den jeweiligen Ordinatenwert bestimmen.

Es werden 253 Abszissenwerte dargestellt, ebenso groß ist der Umfang eines Analog-Bildwiederholtspeichers. Für die Ordinatenwerte wird der Wertebereich des Bytes, also 0...255 (00...0FFH) ausgeschöpft.

Die Anfangsadressen der Analog-Bildwiederholtspeicher werden auf folgenden Adressen niedergelegt:

0D10 ABWS 1	ersetzt Zeile 2
0D12 ABWS 2	ersetzt Zeile 4
0D14 ABWS 3	ersetzt Zeile 6
0D16 ABWS 4	ersetzt Zeile 8

Tab. 5.10: Zellen für Analog-Bildwiederholtspeicher

Steht auf der jeweiligen Zelle und nachfolgenden Zelle die Adresse 0000, so wird keine analoge Kurve, sondern die zugehörige alphanumerische Zeile angezeigt; wird auf dieser Adresse die Adresse eines Analog-Bildwiederholtspeichers geschrieben, so erscheint statt der alphanumerischen Zeile die zugehörige analoge Kurve.

Beispiel:

Folgendes Programm wird mit Hilfe des Objektcodeeditors eingegeben, gebunden und vom Anfang gestartet.

```
AAA:LD HL,0E000H ; Analog-BWS 1
LD (0D10H),HL ; Einschalten der Analogdarstellung
LD DE,0E1FFH ; Ende des Analog-BWS 2
LD B,0F0H ; Schleifenzähler
LD A,5 ; erster Wert
AA1:LD (HL),A ; Analogwert eingetragen
LD (DE),A ; Analogwert in ABWS 2
INC HL ; nächster Abszissenwert
DEC DE
INC A ; nächster Ordinatenwert
DJNZ AA1 ; Schleife
LD (0D12H),DE ; Einschalten zweite Analogdarstellung
RET
```

Arbeiten Sie dieses Programm zuerst schrittweise, dann im Lauf ab!

5.5. Vereinbaren einer Programmtabelle

Wie im Abschnitt 4 erläutert, dient die Menütabelle zum Aufruf eines namentlich genannten Programmes aus dem Betriebssystem. Programmtabellen können sich rückwärts ab folgenden Speicheradressen befinden:

intern im Betriebssystem
3FFF ROM-Bereich 1
5FFF ROM-Bereich 2
7FFF Prozeßsteuerprogramm
FFFF RAM

Ab der Adresse FFFF wird die Referenztafel von BASEX usw. angelegt. Innerhalb dieser Tafel können mit entsprechenden Kommandos, die in den dazugehörigen Abschnitten erläutert sind, Programmnamen vereinbart werden. Möchte sich der Anwender eigene Programme im Bereich ROM1, ROM2 oder Prozeßsteuerprogramme generieren, so hat er folgende Struktur zu befolgen:

Adresse	Inhalt	Bedeutung
xFFF	00	Kennzeichen Programmtabelle vorhanden
xFFE	AH	erstes Byte der Zeile: Adresse high
xFFD	AL	Adresse low (Startadresse)
xFFC	80	Kennbyte, ist dieses ≠ 80, so ist dies kein Programmname
xFFB	ASCII	erster Buchstabe des Programmnamens im ASCII-Code, Bit 7=0
.		
.		
xFFx	ASCII	letzter Buchstabe des Programmnamens im ASCII-Code, Bit 7=1 zur Kennzeichnung des Zeilenendes
xFFx-1	AH	erstes Byte der nächsten Zeile: Adresse
-2	AL	usw.
-3	80	Kennbyte
.		
.		

Das Tabellenende wird durch ein Kennbyte FF (hexadezimal) gekennzeichnet.

Beispiel:

Ein Programm mit dem Namen ABC soll auf der Startadresse 7C00 und ein weiteres mit dem Namen XYZ auf der Startadresse 7C45 vereinbart werden. Der ROM ab 7FF0 muß dazu folgenden Inhalt besitzen:

7FF0: FF FF FF DA 59 58 80 45 7C C3 42 41 80 00 7C 00

6. Hinweise zur Anwenderprogrammierung

Wird der MC 80 als Prozeßsteuergerät (Mikroprozeßrechner) eingesetzt, so müssen sich in seinem Hauptspeicher Prozeßsteuerprogramme befinden. Diese können nach Einschalten des Gerätes vom Magnetband auf den RAM-Bereich geladen werden oder sie werden für ständige Anwendung auf EPROM implementiert. Nur variable Programmteile bzw. Steuerdaten werden dann noch von der Kassette in den RAM geladen. Der Speicherbereich ab der Adresse 6000H bis 7FFFH ist für die Implementierung eines Prozeßsteuerprogramms vorgesehen. Empfohlen wird die Verwendung der Baugruppe K 3620 aus dem Sortiment des Mikrorechners K 1520 mit 6 KByte ROM und 2 KByte RAM. Ist dieser RAM-Bereich ausreichend und wird für den Einsatzfall der Objektcodeeditor nicht benötigt, so kann die Grundvariante MC 80.20 (ohne Speichererweiterungen) benutzt werden.

6.1. Aufruf von Anwenderprogrammen

Kommandoprogramm

Wird ein Anwenderprogramm durch Bedienung aus dem Betriebssystem aufgerufen, so ist dieser Aufruf als Kommando zu verstehen.

Die Menütabelle des MC 80-Betriebssystems kann um diese Anwenderkommandos erweitert werden. Dies ist im Abschnitt 5.5. beschrieben.

Das durch Kommando aufgerufene Programm läuft in der Hintergrundebene des Prozessors. Am Ende dieses Programms muß ein RET-Befehl stehen. Dieser bewirkt eine Rückkehr des Prozessors in die Betriebssystem-Kommandoverwaltung.

Anwenderprogramme dieser Art werden also als Unterprogramme formuliert.

RESET-Initialisierungsprogramm

Soll ein Anwenderprogramm ohne Bedienung sofort nach RESET (Einschalten, Betätigen der RESET-Taste, nach kurzzeitigem Netzausfall) starten, so muß auf der Adresse

6000H : NOP und auf der Adresse

6001H : JMP PRO notiert werden.

PRO ist dabei die Startadresse dieses Anwenderprogramms.

Das Programm kann in einer Endlos-Schleife laufen. In diesem Fall wird die Betriebssystem-Kommandoverwaltung nicht ange-
laufen. Endet dieses Programm mit einem RET-Befehl (Formu-
lierung als Unterprogramm), so erfolgt nach dessen Abarbeitung
der Aufruf der Betriebssystem-Kommandoverwaltung. Damit wird
dieses Anwenderprogramm nach RESET neben der normalen Geräte-
funktion zusätzlich abgearbeitet.

Das Durchlaufen dieses Programms nach "RESET" kann verhindert
werden, indem gleichzeitig mit RESET die Tasten "UC" und "OFF"
betätigt werden, (entspricht dann normalen RESET).

vgl. dazu auch Abschnitt 2.4.

Zyklisches Programm

Ein Anwenderprogramm mit zyklischem Aufruf ist als Interrupt-
programm zu formulieren. In einem dazugehörigen Initiali-
sierungsprogramm ist ein Zeitgeber (CTC-Schaltkreis, z. B.
auf der ZRE frei verfügbar) mit den entsprechenden Daten zu
programmieren.

Folgende Vorschrift gilt für die Struktur eines Interrupt-
programms:

```
ABC:EI                            ; Startadresse des Interruptprogramms,  
                                 ; sofortige Freigabe für höher priori-  
                                 ; sierte Interrupts (Bildschirmbedienung)
```



```
  PUSH AF                        ; Retten aller im Interruptprogramm ver-  
  PUSH HL                        ; wendeten CPU-Register  
  PUSH BC  
  PUSH IX  
  .  
  .  
  .
```

```

POP IX      ; Rückholen der Registerbelegungen
POP BC      ; vor dem Interruptprogrammaufruf
POP HL      ; LIPO (last in, first out)!
POP AF
RETI        ; Rücksprung mit RETI-Befehl zwecks
           ; Freigabe der Interruptkaskade

```

Die Befehle EXX und EXAF (Zweitregistersatz) sind nur bedingt zu verwenden (vgl. Abschnitt 6.3.)!

Die Initialisierung des Interrupts kann nach folgender Vorschrift in jeder Prioritätsebene erfolgen:

```

LD HL,ABC          ;Startadresse des Interruptprogramme
LD (0D00H+IVE),HL ;in der Startadressentabelle verein-
                  ;baren. Der Speicherbereich der Start-
                  ;adrestabelle ist durch den Inhalt des
                  ;I-Registers der CPU mit 0D00H bis
                  ;0DFFH festgelegt. Das I-Register darf
                  ;nicht verändert werden.
                  ;Achtung! Der RAM Bereich 0D00H bis
                  ;0DFFH ist für den Anwender nur teil-
                  ;weise freigegeben.

LD A,IVE           ;Interruptvector des Peripheriebau-
                  ;steins

OUT CTC           ;Ausgabe auf den Peripheriebau-
                  ;stein (bei CTC nur Kanal 0!)

LD A,Steuerwort 1 ;Ausgabe der weiteren Steuerworte
OUT CTC
LD A, Steuerwort 2
OUT CTC

```

Die hier vorgeschriebene Reihenfolge der Programmierung muß unbedingt eingehalten werden, wenn mit dem Auftreten höher priorisierter Interrupts zu rechnen ist.

Es muß darauf geachtet werden, daß die Interruptkaskade geschlossen ist (Steckreihenfolge der Karten).

Belegung des RAM-Bereichs 0DXX

0D00...0D05	nicht verändern!
0D06...0D07	Interruptstartadresse BREAK-Interrupt
0D08...0D0F	nicht verändern!
0D10...0D17	Anzeigebereiche Grafik
0D18...0D1B	TAB-Position (vgl. Abschnitt 5.3.)
0D1C...0D2B	nicht verändern!
0D30...0DEF	Anwendervorzugsbereich für Interruptstartadressen
0DF0...0DFE	nicht verändern!

Beispiel für ein zyklisches Programm:

Dieses Beispiel schreibt den gesamten Zeichenvorrat des BildwiederholSpeichers (BWS) von 21H bis 5FH in nachfolgend beschriebener Weise auf das Display.

Es sind zwei Interruptprogramme formuliert. In einem Initialisierungsteil ist der CTC-Schaltkreis der ZRE programmiert. Es werden zunächst nacheinander alle Displaypositionen mit einem Zeichen des BWS beschrieben (pro Interrupt ein Zeichen).

Ist der gesamte Bildschirm beschrieben, wird das zweite Interruptprogramm angesprungen und alle Bildschirmpositionen werden wieder gelöscht (pro Interrupt ein Zeichen). Da 2 Interruptprogramme abgearbeitet werden, muß jeweils die entsprechende Startadresse in der Startadrestabelle vereinbart werden.

Ist der gesamte Zeichenvorrat ausgeschrieben, beginnt der Zyklus von Neuem.

```

ANF:LD HL,0C00H ; Beginn des BildwiederholSpeichers
    LD (BWS),HL ; Merkwelle
    LD A,21H ; erstes darzustellendes Zeichen
    LD (ZEL),A
    LD HL,INT ; Startadr. Interruptprogramm
    LD (0D38H),HL ; in StartadreStabelle vereinbaren
    LD A,38H ; Interruptvektor
    OUT 80H ; auf Kanal 0 der ZRE ausgeben
    LD A,0A7H ; Kanalsteuerwort
    OUT 80H
    LD A,0B0H ; Zeitkonstante
    OUT 80H
AN1:JR AN1 ; Warteschleife
INT:EI ; Interruptprogramm 1
    PUSH AF
    PUSH HL
    LD HL,(BWS) ; Position des BWS laden
    LD A,(ZEL) ; A=Zeichen
    LD (HL),A ; Zeichen auf Position schreiben
    INC L ; BS-Adresse erhohen
    JRNZ IN1 ; BS voll beschrieben (L=0)?
    PUSH HL ; ja
    LD HL,ZWE ; Startadr. 2. Interruptprogramm
    LD (0D38H),HL ; vereinbaren
    POP HL
IN1:LD (BWS),HL ; aktuelle Position merken
    POP HL
    POP AF
    RETI
ZWE:BI ; Interruptprogramm 2
    PUSH AF
    PUSH HL
    LD HL,(BWS) ; Position
    LD A,20H ; Leerzeichen
    LD (HL),A ; LZ schreiben
    INC L ; Position erhohen

```

```

JRNZ ZW1          ; BS gelöscht?
PUSH HL          ; ja
LD HL,INT        ; Adr. Interruptprogr. 1 eintragen
LD (0D36H),HL
POP HL
LD A,(ZEL)
INC A            ; nachfolgendes Zeichen einstellen
CMP 60H         ; letztes Zeichen?
JRNZ ZW2        ; nein, Sprung
LD A,21H        ; ja, wieder erstes Zeichen laden
ZW2:LD (ZEL),A  ; aktuelles Zeichen auf Merkwelle
                    laden
ZW1:LD (BWS),HL ; Position merken
POP HL
POP AF
RETI
BWS:EXT 0D000H
ZEL:EXT 0D002H  ; auf freie Speicherzelle ver-
                    einbaren

```

Unterbrechungsprogramm

Die BREAK-Taste kann zur Auslösung eines hochpriorisierten Interrupts benutzt werden. Das dazugehörige Programm wird als Interruptprogramm formuliert. Die Initialisierung geschieht folgendermaßen:

```

LD HL, BRK      ; Startadresse Interrupt
LD (0D06H),HL  ; Adresse ist festgelegt!
LD A,0C7H      ; CTC als Zähler initialisieren
OUT 0F3H       ; Adresse ist festgelegt
LD A,01H       ; Zeitkonstante 1
OUT 0F3H       ; CTC (Kanal 3)

```

Der verwendete CTC befindet sich auf der Steckereinheit DPL. Die anderen Kanäle sind intern belegt. Dieser CTC-Kanal wird vom Schrittestprogramm verwendet und ist bei Aufruf des Objektcodeeditors belegt. Er muß danach gegebenenfalls neu initialisiert werden.

6.2. Verwendung von NMI und RST

Intern wird der nichtmaskierbare Interrupt nicht verwendet. Das Betriebssystem realisiert die NMI-Startadresse 6004H. Damit ist er dem Anwender zugänglich.

Die RST-Befehle sind auf folgenden Startadressen ebenfalls anwenderfrei:

<u>Auslösung</u>	<u>Startadresse (hex.)</u>
RESET	6001 , wenn auf 6000 = 00
NMI	6004
RST1	6007
RST2	600A
RST3	600D
RST4	6010
RST5	6013
RST6	6016
RST7	2C00 (verwendet vom Objektcodeeditor)

6.3. Bildschirm- und Tastaturinterrupt

Bildschirm und Tastatur werden über einen hochpriorisierten Interrupt bedient. Das ist intern notwendig. Bei Sperren dieser Interrupts bleibt der Bildschirm dunkel und die Tastatur kann nicht abgefragt werden.

Dieser Interrupt benötigt ca. 50 % der Rechenzeit des Prozessors. Das bedeutet, daß Anwenderprogramme nur halb so schnell ablaufen, entsprechend 1,2 MHz Taktfrequenz.

Ist der Bildschirm nicht auf Grafik geschaltet, so beträgt die längste Unterbrechung des Anwenderprogramms durch den Bildschirminterrupt 0,2 ms. Bei Einschalten der Grafik wird für max. 2,7 ms unterbrochen und die Interruptbelastung steigt bis auf 72 % (4 analoge Kurvenzüge).

Die Echtzeitfähigkeit des Gerätes MC 80 wird für Echtzeitanforderungen im 10 ms-Bereich und bei ausreichender Rechenzeit dadurch nicht beschränkt.

Für höhere Zeitanforderungen kann ein Zwei- oder Mehrprozessorsystem mit loser Kopplung eingesetzt werden. Dabei verbleiben im MC 80-Kern die Echtzeitprogramme der Bedienung und hierarchisch höheren Prozeßführung.

Der Displayinterrupt belegt den Zweitregistersatz. Damit steht er dem Anwender nicht zur Verfügung, wenn er den Displayinterrupt nicht abschaltet.

Abschalten des Displayinterrupt:

Unter Verzicht auf den Dialog über Bildschirm und Tastatur ist es auch möglich, zeitlich begrenzte Anwenderprogramme ohne Unterbrechung (Echtzeitbetrieb im μ s-Bereich) ablaufen zu lassen.

Um die volle Rechenzeit der CPU zur Verfügung zu stellen, sind dazu die höchstpriorisierten DPL-Interrupts abgeschaltet. Damit stehen den Anwenderinterrupts keine Einschränkungen entgegen.

Für das An- und Abschalten des Display sind die folgenden Programmteile unbedingt einzuhalten:

```
AUS:  DI                ; Abschalten des Display
      LD A,041H
      OUT 0F0H
      OUT 0F2H
      LD A,00FH
      OUT 0F7H
      LD A,080H
      OUT 0F6H
      LD A,020H
      OUT 0F4H
      EXX
      EXAF
      PUSH AF
      PUSH BC
      PUSH DE
      PUSH HL
      EI
      .
      .
      .

EIN:  DI                ; Einschalten des Display
      POP HL
      POP DE
      POP BC
      POP AF
      EXAF
      EXX
      LD A,0C1H
      OUT 0F0H
      OUT 0F2H
      EI
```

Nach dem Abschalten des Displays können die zeitkritischen Anwenderprogramme als Hauptprogramm oder über Interruptsteuerung ablaufen. Nach Ende des Anwenderprogrammes oder bei Bedarf ist der Bildschirm wieder zuzuschalten.

Mit dem Abschalten des Display ist gleichzeitig die Tastatur (außer RESET und BK) wirkungslos. Besteht der Wunsch, die Taste BREAK zur Rückkehr in den Anzeigemodus zuzulassen, so ist die Initialisierung für ein Unterbrechungsprogramm nach Abschnitt 6.1. durchzuführen.

6.4. Beispiel

Im MC 80 soll eine Echtzeituhr realisiert werden.

Mit dem Kommando "UHR" soll diese Uhr gestartet werden. Die Uhr soll in einer Interruptebene nicht sichtbar laufen. Es sollen 20 ms, Sekunden, Minuten und Stunden gezählt werden. Die Information soll für andere Programme auf 4 freien RAM-Zellen entnehmbar sein. Das Kommando "ZEIT" soll die Anzeige der Zeit auf dem Bildschirm bewirken. Die Zeitanzeige soll bei beliebigem Tastendruck anhalten und sonst mitlaufen. Die Taste "OFF" bewirkt ein Verlassen des Programms. Die "BREAK"-Taste soll ein Rückstellen der Uhr bewirken.

Lösung

Es wird ein 20 ms-Interrupt mit dem auf der ZRE K 2521 befindlichen CTC-Schaltkreis realisiert. Mit Vorteller 256 und Teilerfaktor 192 ergeben sich 20 ms bei einer Taktfrequenz von 2,4576 MHz mit einer Abweichung von - 0,4 %. Dieser Interrupt incrementiert den ersten Zähler (Byte auf Adresse 0D40H) bis auf 49. Die nächste Zählzelle (0D41H) wird bis auf 59 incrementiert und zählt die Sekunden, die Zählzellen 0D42H und 0D43H zählen Minuten und Stunden. Der Zeitinterrupt wird vom Programm, Uhr, als Kommando aufrufbar, initialisiert.

Das Programm ANZ dient dem Ausschreiben des Inhaltes der Zeitzellen mit einem Begleittext nach dem geforderten Algorithmus. Die Startadressen der Programme UHR und ANZ werden mit dem Namen "UHR" und "ZEIT" in der Menütabelle vereinbart. Der Interrupt der "BREAK"-Taste wird im Programm UHR initialisiert. Der Interrupt realisiert ein Rücksetzen der Zellen. Jedoch ist hier ein Softwarefehler eingebaut. Der BREAK-Interrupt (Programm USI) ist höher priorisiert als der Zeitzählinterrupt UHI.

Unterbricht der BREAK-Interrupt den Zeitzählinterrupt, so ist die Funktion des BREAK-Interrupts hinfällig, da vom fortgesetzten Zeitzählinterrupt die Zeitzählzellen unkorrigiert beschrieben werden.

Es muß hier eine Synchronisationssemaphore eingeführt werden oder der BREAK-Interrupt muß einen niedriger priorisierten Interrupt initialisieren, der die eigentliche Funktion ausführt. Der beschriebene Fehler tritt zufällig mit geringer Wahrscheinlichkeit auf und ist somit nicht augenscheinlich.

Das Programm "UHR" muß ab der Adresse C000 niedergelegt werden.

```

UHR LD HL,0           ; Zählzellen rücksetzen
    LD (0D40H),HL
    LD (0D42H),HL
    LD HL,UHI        ; Startadresse Zeitzählinterrupt
    LD (0D38H),HL / (0D0EH)*
    LD A,38H / (08H)* ; Interruptvektor
    OUT 80H / (0F8H)* ; CTC Kanal 0 auf ZRE
    LD A,0A7H        ; Zeitgeber 20 ms
    OUT 80H / (0FBH)*
    LD A,0C0H        ; Zeitkonstante 192
    OUT 80H / (0FBH)*
    LD HL,USI        ; Startadresse BREAK-Interrupt
    LD (0D06H),HL
    LD A,0C7H        ; Zähler 1 Impuls
    OUT 0F3H         ; BREAK-CTC auf STE DPL
    LD A,1
    OUT 0F3H
    JMP BSK          ; Sprung zur BS-Menüverwaltung
UHI EI              ; Freigabe höher prioris. Interrupts
    PUSH AF          ; benutzte Register retten
    PUSH HL
    PUSH DE
    LD HL,(0D40H)    ; Zählzelleninhalte laden
    LD DE,(0D42H)
    LD A,L            ; Incrementieren 20 ms-Zähler
    ADD 1
    DAA              ; dezimal zählen

```

*Klammerwerte gelten für ZRE K 2523

```

LD L,A
CMP 50H ; Zähler voll?
JRC UH1 ; nein
LD L,0 ; ja: auf 0 und nächste Zelle incre-
          mentieren
LD A,H ; Sekundenzähler
ADD 1
DAA
LD H,A
CMP 60H
JRC UH1
LD H,0
LD A,E ; Minutenzähler
ADD 1
DAA
LD E,A
CMP 60H
JRC UH1
LD E,0
LD A,D ; Stundenzähler
ADD 1
DAA
LD D,A
CMP 24H
JRC UH1
LD D,0

UH1: LD (0D40H),HL ; Zählzellen rüclladen
      LD (0D42H),DE
      POP DE ; Registerinhalte wieder herstellen
      POP HL
      POP AF
      RETI ; Rücksprung

      ; BREAK-Interrupt

USI: EI
      PUSH HL
      LD HL,0 ; Zählzellen auf 0 setzen

```

LD (0D40H),HL
 LD (0D42H),HL
 POP HL
 RETI

ANZ:	LD DE,0200H	; Zeitanzeige
	LD HL,TTA	; Bildschirmzeile 3
	LD A,(0D43H)	; Texttabelle
	CALL HBS	; Stundenzähler
	CALL AZT	; ausschreiben
	CALL AZT	; Text dazu ausschreiben
	LD A,(0D42H)	; Minutenzähler
	CALL HBS	
	CALL AZT	
	LD A,(0D41H)	; Sekundenzähler
	CALL HBS	
	CALL AZT	
	LD A,(0D40H)	; 20 ms-Zähler
	ADD A	
	DAA	
	CALL HBS	
	CALL AZT	
AN1:	CALL TST	; Tastatur abfragen
	CMP 0	; keine Taste?
	JRZ ANZ	; ja: Anzeige
	CMP 16H	; OFF?
	JRNZ AN1	; nein: Warten
	RET	; ja: Rücksprung
		; Text ausschreiben
AZT:	LD A,(HL)	
	AND 7FH	
	CALL BSS	
	INC B	
	INC HL	
	BIT 7,(HL)	
	JRZ AZT	
	RET	

```

TTA:  DB 0BAH           ; Texttabelle
      DB 0A0H           ; " "
      DM 'UHRLL'       ; "UHRLL"
      DB 0ACH           ; " , "
      DB 0A0H           ; "LSEC "
      DM 'SEC'         ;
      DB 0FFH         ; Tabellenende

HBS:  EXT 0300H        ; Adreßvereinbarung
TXS:  EXT 0465H
TST:  EXT 0013H
BSS:  EXT 000BH
BSK:  EXT 04ACH

```

Menütabelle:

```

FFEF : FF           ; Tabellenende als Kennbyte
FFF0 : FF           ; beliebige Adresse Low
FFF1 : FF           ; beliebige Adresse High
FFF2 : D4           T   (letztes Textzeichen)
FFF3 : 49           I
FFF4 : 45           B
FFF5 : 5A           Z   (ZEIT)
FFF6 : 80           Kennbyte
FFF7 : 7D           Adresse Low
FFF8 : C0           Adresse High
FFF9 : D2           R   (letztes Textzeichen)
FFFA : 48           H
FFFB : 55           U   (UHR)
FFFC : 80           Kennbyte
FFFD : 00           Adresse Low
FFFE : C0           Adresse High
FFFF : 00           Tabellenanfang

```

A N L A G E 1

Befehlsliste des Objektcodeeditors des MC 80

Erläuterungen:

- r ist eines der CPU-Register A, B, C, D, E, H, L,
- s ist ein 8-Bit-Speicherplatz entsprechend der angegebenen Bedingung,
- ss, dd, pp, qq sind 16-Bit-Speicherplätze entsprechend der angegebenen Bedingung,
- ii ist eines der Indexregister IX, IY,
- n ist ein 8-Bit-Wert im Bereich 0...255,
- d ist ein 8-Bit-Wert von -128...+127, Zweierkomplement,
- e ist eine Sprungdistanz von -126...+129 oder eine symbolische Adresse, die von dieser Sprungdistanz gekennzeichnet wird. Im RAM erfolgt die Abspeicherung von e-2,
- nn ist eine 16-Bit-Konstante oder ein symbolischer Name (Marke)

Abspeicherung: zuerst niederwertiger, dann höherwertiger Teil,
 Index L bzw. H bezeichnet den niederwertigen bzw. höherwertigen Teil eines Zweibyte-Speicherplatzes.

Flags:	Flagbelegungen:
C Carry-Flag Cy;	0 Flag wird rückgesetzt,
Z Zero-Flag,	1 Flag wird gesetzt,
P Paritäts/Überlaufflag P/V	. Flag wird nicht beeinflusst,
S Vorzeichenflag,	! Flag wird entsprechend der Operation gesetzt,
N Additions-Subtraktions- flag,	X nicht belegt,
H Halb-Byte-Übertrags- flag	P Parität setzt P/V V Überlauf setzt P/V.

Flaganordnung im F-Register:

S, Z, X, H, X, P, N, C

Mit Unterstreichungen sind in den Operationscode verschiedener Befehle die Teile gekennzeichnet, die sich in den O/I-Belegungen des davor angegebenen Befehls unterscheiden.

a) 8-Bit-Ladebefehle

Mnemonic	Symbol. Op.	Flags CZPSNH	Op. code	Zykl. M/T	Erläut.
LD r_1, r_2	$r_1 \leftarrow r_2$	0I $r_1 r_2$	1/4	r_1, r_2 Reg.
LD r, n	$r \leftarrow n$	00 r IIO - n -	2/7	000 B 00I C
LD $r, (HL)$	$r \leftarrow (HL)$	0I r IIO	2/7	0I0 D
LD $r, (11+d)$	$r \leftarrow (11+d)$	IIIIIIIOI 0I r IIO - d -	4/19	0II E I00 H I0I L III A
LD (HL), r	$(HL) \leftarrow r$	0IIIO r	2/7	
LD $(11+d), r$	$(11+d) \leftarrow r$	IIIIIIIOI 0IIIO r - d -	5/19	1 Reg. 11 0 IX I IY
LD (HL), n	$(HL) \leftarrow n$	00II0II0 - n -	3/10	
LD $(11+d), n$	$(11+d) \leftarrow n$	IIIIIIIOI 00II0II0 - d - - n -	5/19	
LD A, (BC)	$A \leftarrow (BC)$	0000I0I0	2/7	
LD A, (DE)	$A \leftarrow (DE)$	000II0I0	2/7	
LD A, (nn)	$A \leftarrow (nn)$	00IIII0I0 - n - - n -	4/13	
LD (BC), A	$(BC) \leftarrow A$	000000I0	2/7	
LD (DE), A	$(DE) \leftarrow A$	000I00I0	2/7	
LD (nn), A	$(nn) \leftarrow A$	00II00I0 - n - - n -	4/13	
LD A, I	$A \leftarrow I$..!100	IIIOII0I 0I0I0III	2/9	P/V-Flag wird entspr. IFF gesetzt
LD A, R	$A \leftarrow R$..!100	IIIOII0I 0I0I0III	2/9	
LD I, A	$I \leftarrow A$	IIIOII0I 0I000III	2/9	
LD R, A	$R \leftarrow A$	0I000III 0I000III	2/9	

b) 16-Bit-Ladebefehle

Mnemonic	Symbol. Op.	Flags CZPSNH	Op. code	Zykl. M/T	Erläut.
LD dd, nn	dd ← nn	00dd000I - n - - n -	3/10	dd <u>Reg. paar</u> 00 BC 01 DE
LD ii, nn	ii ← nn	IIIIIII0I 00I0000I - n - - n -	4/14	I0 HL II SP
LD HL, (nn)	L ← (nn) H ← (nn+1)	00I0I0I0 - n - - n -	5/16	pp <u>Reg. paar</u> 00 BC 01 DE
LD pp, (nn)	pp _H ← (nn+1) pp _L ← (nn)	IIII0I0I 0IddI0II - n - - n -	6/20	II SP
LD ii, (nn)	ii _H ← (nn+1) ii _L ← (nn)	IIiIIII0I 00I0I0I0 - n - - n -	6/20	i <u>Reg. ii</u> 0 IX I IY
LD (nn), HL	(nn+1) ← H (nn) ← L	00I0000I0 - n - - n -	5/16	
LD (nn), pp	(nn+1) ← pp _H (nn) ← pp _L	IIII0I0I 0Idd00II - n - - n -	6/20	
LD (nn), ii	(nn+1) ← ii _H (nn) ← ii _L	IIiIIII0I 00I0000I0 - n - - n -	6/20	
LD SP, HL	SP ← HL	IIIIII00I	1/6	
LD SP, ii	SP ← ii	IIiIIII0I IIIIII00I	2/10	
PUSH qq	(SP-2) ← qq _L (SP-1) ← qq _H	IIqq0I0I	3/11	qq <u>Reg. paar</u> 00 BC 01 DE
PUSH ii	(SP-2) ← ii _L (SP-1) ← ii _H	IIiIIII0I IIII00I0I	4/15	I0 HL II AF
POP qq	qq _H ← (SP+1) qq _L ← (SP)	IIqq000I	3/10	PUSH: SP Sp-2
POP ii	ii _H ← (SP+1) ii _L ← (SP)	IIiIIII0I IIII0000I	4/14	POP: SP SP+2

c) Registertausch, Blockladen und Blocksuchen

Mnemonic	Symbol.Op.	Flags CZPSNH	Op.code	Zykl. M/T	Erläut.
EX DE HL	DE ← HL	IIIIOIOII	1/4	
EXAF	AF ← AF'	0000I000	1/4	} für Anwender im Normalfall nicht nutzbar
EXX	BC ← BC' DE ← DE' HL ← HL'	IIIOII00I	1/4	
EX (SP), HL	H ← (SP+1) L ← (SP)	III000II	5/19	
EX (SP), IX	IX _H ← (SP+1) IX _L ← (SP)	IIOIIIIOI III000II	6/23	
EX (SP), IY	IY _H ← (SP+1) IY _L ← (SP)	IIIIIIIOI III000II	6/23	
LDI	(DE) ← (HL) DE ← DE+1 HL ← HL+1 BC ← BC-1	..1.00	III0II0I IOI00000	4/16	BC=0 setzt P-Flag
LDIR	wie LDI Wiederh. bis BC=0	..0.00	III0II0I IOII0000	5/21 4/16	wenn BC≠0 wenn BC=0
LDD	(DE) ← (HL) DE ← DE-1 HL ← HL-1 BC ← BC-1	..1.00	III0II0I IOI0I000	4/16	
LDDR	wie LDD Wiederh. bis BC=0	..0.00	III0II0I IOIIII000	5/21 4/16	wenn BC≠0 wenn BC=0
CPI	A ← (HL) HL ← HL+1 BC ← BC-1	.IIII!	III0II0I IOI0000I	3/16	BC=0 setzt P-Flag
CPIR	wie CPI Wiederh. bis BC=0 oder A=(HL)	.IIII!	III0II0I IOII000I	3/21 3/16	wenn BC≠0 u. A≠HL wenn BC=0 oder A=(HL)
CPD	A ← (HL) HL ← HL-1 BC ← BC-1	.IIII!	III0II0I IOI0I00I	3/16	
CPDR	wie CPD Wiederh. bis BC=0 oder A=(HL)	.IIII!	III0II0I IOIIII00I	3/21 3/16	wenn BC≠0 u. A≠(HL) wenn BC=0 oder A=(HL)

d) Arithmetische und logische Operationen

Mnemonic	Symbol. Op.	Flags	Op. code	Zykl.	Erläut.
		CZPSNH		M/T	
ADD r	A ← A+r	!!V!0!	I0000 r	1/4	r Reg.
ADD n	A ← A+n	!!V!0!	I1000II0 - n -	2/7	000 B 001 C 010 D 011 E
ADD (HL)	A ← A+(HL)	!V!0!	I0000II0	2/7	011 E
ADD (i1+d)	A ← A+(i1+d)	!!V!0!	II111101 I0000II0 - d -	5/19	100 H 101 L 111 A
ADC s	A ← A+s+CY	!!V!0!	001		s kann sein;
SUB s	A ← A-s		010		A, B, C, D, E, H, L (HL), (IX+d),
SBC s	A ← A-s-CY	!!V!1!	011		(IX+d) oder n
AND s	A ← A∧s	0!P!11	100		wie beim ADD- Befehl
OR s	A ← A∨s	0!P!00	110		
XOR s	A ← A⊕s	0!P!00	101		Die angeg. Bits
CMR s	A ← s	!!V!1!	111		ersetzen die
INC r	r ← r+1	!!V!0!	00 r 100	1/4	gekennz. im
INC (HL)	(HL) ← (HL)+1	!!V!0!	00110100	3/11	ADD-B.
INC (i1+d)	(i1+d) ← (i1+d)+1	!!V!0!	II111101 00110100 - d -	6/23	
DEC m	m ← m-1	!!V!1!	101		wie INC oben
ADD HL, ss	HL ← HL+ss	!...0X	00ss1001	3/11	ss Reg. paar
ADC HL, ss	HL ← HL+ss+CY	!!V!0X	11101101 01ss1010	4/15	00 BC 01 DE 10 HL 11 SP
SBC HL, ss	HL ← HL-ss-CY	!!V!1X	11101101 01ss0010	4/15	
ADD i1, pp	i1 ← i1+pp	!...0X	II111101 00pp1001	4/15	pp Reg. paar 00 BC 01 DE 10 i1 11 SP
INC ss	ss ← ss+1	00ss0011	1/6	ss wie ADD HL, ss
INC i1	i1 ← i1+1	II111101 00100011	2/10	
DEC ss	ss ← ss-1	00ss1011	1/6	
DEC i1	i1 ← i1-1	II111101 00101011	2/10	

e) Rot-, Schiebe-, BCD-, Akku.- und Flag-Operationen

Mnemonic	Symbol.Op.	Flags	Op.code	Zykl.	Erläuterungen	
		CZPSNH		M/T		
RLCA		!...00	00000III	1/4	Akkumulator rotieren	
RLA		!...00	000I0III	1/4		
RRCA		!...00	0000IIII	1/4		
RRA		!...00	000IIIII	1/4		
RLC r		!!P!00	II00I0II	2/8	r Reg.	
RLC (HL)			0000 r	000	B	
RLC (11+d)			II00I0II	4/15	00I	C
			00000IIO		0I0	D
			III1II0I	6/23	0II	E
			II00I0II		I00	H
			- d -		I0I	L
			00000IIO		III	A
RL m		!!P!00	<u>0I0</u>		Befehlsformat Register und Zyklen wie bei RLC m.	
RRC m		!!P!00	<u>00I</u>		Die unterstr. Bits sind aus- zutauschen!	
RR m		!!0!00	<u>0II</u>			
SLA m		!!P!00	<u>I00</u>		m kann sein: r, (HL), (IX+d), (IY+d)	
SRA m		!!P!00	<u>IOI</u>		<u>i</u> Reg. <u>ii</u>	
SRL m		!!P!00	<u>III</u>		0 IX	
					I IY	
RLD		!!P!00	III0II0I	5/18		
RRD		!!P!00	0II0IIII			
	A (HL)	!!P!00	III0II0I	5/18		
			0II00III			
DAA	BCD-Korrektur	!!P!!	00I00III	1/4	nach Add./Sub.	
CPL	A ← \bar{A}II	00I0IIII	1/4	Bitweises Kompl	
NEG	A ← 0 - A	!!V!I!	II0II0I	2/8	Zweierkompl.	
CCF	CY ← $\bar{C}Y$	1...0X	00IIIIII	1/4	Carry-Flag neg.	
SCF	CY ← 1	I...00	00II0III	1/4	Carry-Flag setzen	

f) Bit-Befehle, NOP und Interruptverarbeitung

Mnemonic	Symbol. Op.	Flags CZPSNH	Op. code	Zykl. M/T	Erläut.	
BIT b,r	$Z \leftarrow \overline{r}_b$.IXXOI	II00IOII OI b r	2/8	b	bit
					000	0
					00I	1
BIT b, (HL)	$Z \leftarrow \overline{(HL)}_b$.IXXOI	II00IOII OI b IIO	3/12	OIO	2
					OII	3
BIT b, (ii+d)	$Z \leftarrow \overline{(ii+d)}_b$.IXXOI	III1II0I II00IOII - d - OI b IIO	5/20	I00	4
					IOI	5
					IIO	6
					III	7
SET b,r	$r_b \leftarrow I$	II00IOII II b r	2/8	r	Reg.
					000	B
SET b, (HL)	$(HL)_b \leftarrow I$	II00IOII II b IIO	4/15	00I	C
					OIO	D
SET b, (ii+d)	$(ii+d)_b \leftarrow I$	III1II0I II00IOII - d - II b IIO	6/23	OII	E
					I00	H
					IOI	L
					III	A
RES b,m	$m_b \leftarrow 0$ wie SET	IO Für II einsetzen		1	Reg. ii
					0	IX
					I	IY
NOP	-	00000000	1/4	keine Op.	
HALT	CPU-Stop	OIII0II0	1/4	Weiterar- beit bei Interrupt	
DI	$IFF \leftarrow \emptyset$	IIII00II	1/4	Interrupt sperrern	
EI	$IFF \leftarrow I$	IIIIIOII	1/4	Interrupt freigeb.	
IMO	Interrupt- mode setzen	IIIOII0I OIO00IIO	2/8	Mode 0	
IM1	"	IIIOII0I OIOIOIIO	2/8	Mode 1	
IM2	"	IIIOII0I OIOIIII0	2/8	Mode 2 Vektorin- terrupt	
RETI	$PC_L \leftarrow (SP)$ $PC_H \leftarrow (SP+1)$ $SP \leftarrow SP+2$	IIIOII0I OIO0II0I	4/14	Rückspr. aus Int.- Progr.	
RETN	wie RETI $IFF2 \leftarrow IFF1$	IIIOII0I OIO00I0I	4/14	Rückspr. aus NMI- Progr.	

g) Sprünge, Unterprogrammaufruf und -rücksprung

Mnemonic	Symbol. Op.	Flags CZPSNH	Op. code	Zykl. M/T	Erläut.
JR #e	PC ← PC+e	000II000 - e-2 -	3/12	relat. Sprung *)
JRcc #e	PC ← PC+e wenn Bed. erf. wenn Bed. nicht erf.: Kein Sprung	00Icc000 - e-2 -	2/12 2/7	wenn Beding. *) erfüllt wenn kein Spr. cc Bed. cc 00 NZ Z=0 01 Z Z=1 10 NC CY=0 11 C CY=1
JMP nn	PC ← nn	II0000II - n - - n -	3/10	1 Reg. 11 0 IX I IY
JPcc nn	PC ← nn	IIcc00IO - n - - n -	3/10	ccc Bed. cc 000 NZ Z=0 001 Z Z=1 010 NC CY=0 011 C CY=1 100 PO P/V=0 101 PE P/V=1 110 P S=0 111 M S=I *)
JMP (HL)	PC ← HL	III0I00I	1/4	010 NC CY=0 011 C CY=1
JMP (ii)	PC ← ii	III1110I IIIOI00I	2/8	100 PO P/V=0 101 PE P/V=1
DJNZ #e	B ← B-1 wenn B≠0: PC ← PC+e wenn B=0 kein Sprung	000I0000 - e-2 -	2/8	110 P S=0 111 M S=I *) wenn B=0
				3/13	wenn B≠0
CALL nn	(SP-1) ← PC _H (SP-2) ← PC _L SP ← SP-2 PC ← nn	II00II0I - n - - n -	5/17	
CAcc nn	wie CALL wenn Bed. erf.	IIcc0I00 - n - - n -	5/17 3/10	wenn Beding. erf., sonst Auf- ruf nächster Befehl
RET	PC _L ← (SP) PC _H ← (SP+1) SP _H ← SP+2	II00I00I	3/10	
Rec	wie RET wenn Bed. erf.	IIcc0000	3/11 1/5	wenn Beding. erf., sonst Auf- ruf nächster Befehl
RST p	(SP-1) ← PC _L (SP-2) ← PC _H SP ← SP-2 PC _H ← 0 PC _L ← p	IItttIII	3/11	ttt p ttt p 000 0 100 4 001 1 101 5 010 2 110 6 011 3 111 7

*) Bei Sprung_L zu einer symbol. Adresse wird kein "n" angegeben.

h) Ein- und Ausgabe

Mnemonic	Symbol.Op.	Flags CZPSNH	Op.code	Zykl. M/T	Erläut.
IN n	A ← (n)	II0II0II - n -	3/11	n ist niederwert. Adreßteil
OUT n	(n) ← A	II0I00II - n -	3/11	A ist höherwert. Adreßteil
IN r	r ← (C)	.!P!0!	IIII0I0I OI r 000	3/12	C ist niederwert. Adreßteil
OUT r	(C) ← r	IIII0I0I OI r 00I	3/12	B ist höherwert. Adreßteil
INF	Flags! ← (C)	.!P!0!	IIII0I0I 0II10000	3/12	r Reg. 000 B
INI	(HL) ← (C) B ← B-1 HL ← HL+1	.+XXIX	IIII0I0I IOI000IO	4/16	00I C 0IO D 0II E IO0 H
INIR	wie INI Rep. bis B=0	.IXXIX	IIII0I0I IOI100IO	4/21 4/16	IOI L III A
IND	(HL) ← (C) B ← B-1 HL ← HL-1	.+XXIX	IIII0I0I IOI0IOIO	4/16	+...Zero-Flag wird von B=0 gesetzt
INDR	wie IND Rep. bis B=0	.IXXIX	IIII0I0I IOIIIOIO	4/21 4/16	wenn B≠0 wenn B=0
OUTI	(C) ← (HL) B ← B-1 HL ← HL+1	.+XXIX	IIII0I0I IOI000II	4/16	
OTIR	wie OUTI Rep. bis B=0	.IXXXI	IIII0I0I IOII00II	5/21 4/16	wenn B≠0 wenn B=0
OUTD	(C) ← (HL) B ← B-1 HL ← HL-1	.+XXIX	IIII0I0I IOI0I0II	4/16	
OTDR	wie OUTD Re. bis B=0	.IXXIX	IIII0I0I IOIIIOII	5/21 4/16	wenn B≠0 wenn B=0

1) Pseudooperationen

Mnemonic	Erläuterung
DB nn	definiere Byte mit dem Wert nn
DW nnnn	definiere Doppelbyte mit dem Wert nnnn
DM 'aabc'	definiere Zeichenkette mit den Zeichen aabc (ASCII-Code)
END	Verlassen der Quellcode-Eingabe
MAR:EXT nnnn	definiere Marke mit der externen Adresse nnnn

Pseudooperationen werden nicht reassembliert.

Tabelle CB

RLC B	RL B	SLA B		BIT 0,B	BIT 2,B	BIT 4,B	BIT 6,B	RES 0,B	RES 2,B	RES 4,B	RES 6,B	SET 0,B	SET 2,B	SET 4,B	SET 6,B
RLC C	RL C	SLA C		BIT 0,C	BIT 2,C	BIT 4,C	BIT 6,C	RES 0,C	RES 2,C	RES 4,C	RES 6,C	SET 0,C	SET 2,C	SET 4,C	SET 6,C
RLC D	RL D	SLA D		BIT 0,D	BIT 2,D	BIT 4,D	BIT 6,D	RES 0,D	RES 2,D	RES 4,D	RES 6,D	SET 0,D	SET 2,D	SET 4,D	SET 6,D
RLC E	RL E	SLA E		BIT 0,E	BIT 2,E	BIT 4,E	BIT 6,E	RES 0,E	RES 2,E	RES 4,E	RES 6,E	SET 0,E	SET 2,E	SET 4,E	SET 6,E
RLC H	RL H	SLA H		BIT 0,H	BIT 2,H	BIT 4,H	BIT 6,H	RES 0,H	RES 2,H	RES 4,H	RES 6,H	SET 0,H	SET 2,H	SET 4,H	SET 6,H
RLC L	RL L	SLA L		BIT 0,L	BIT 2,L	BIT 4,L	BIT 6,L	RES 0,L	RES 2,L	RES 4,L	RES 6,L	SET 0,L	SET 2,L	SET 4,L	SET 6,L
RLC (HL)	RL (HL)	SLA (HL)		BIT 0,(HL)	BIT 2,(HL)	BIT 4,(HL)	BIT 6,(HL)	RES 0,(HL)	RES 2,(HL)	RES 4,(HL)	RES 6,(HL)	SET 0,(HL)	SET 2,(HL)	SET 4,(HL)	SET 6,(HL)
RLC A	RL A	SLA A		BIT 0,A	BIT 2,A	BIT 4,A	BIT 6,A	RES 0,A	RES 2,A	RES 4,A	RES 6,A	SET 0,A	SET 2,A	SET 4,A	SET 6,A
RRC B	RR B	SRA B	SRL B	BIT 1,B	BIT 3,B	BIT 5,B	BIT 7,B	RES 1,B	RES 3,B	RES 5,B	RES 7,B	SET 1,B	SET 3,B	SET 5,B	SET 7,B
RRC C	RR C	SRA C	SRL C	BIT 1,C	BIT 3,C	BIT 5,C	BIT 7,C	RES 1,C	RES 3,C	RES 5,C	RES 7,C	SET 1,C	SET 3,C	SET 5,C	SET 7,C
RRC D	RR D	SRA D	SRL D	BIT 1,D	BIT 3,D	BIT 5,D	BIT 7,D	RES 1,D	RES 3,D	RES 5,D	RES 7,D	SET 1,D	SET 3,D	SET 5,D	SET 7,D
RRC E	RR E	SRA E	SRL E	BIT 1,E	BIT 3,E	BIT 5,E	BIT 7,E	RES 1,E	RES 3,E	RES 5,E	RES 7,E	SET 1,E	SET 3,E	SET 5,E	SET 7,E
RRC H	RR H	SRA H	SRL H	BIT 1,H	BIT 3,H	BIT 5,H	BIT 7,H	RES 1,H	RES 3,H	RES 5,H	RES 7,H	SET 1,H	SET 3,H	SET 5,H	SET 7,H
RRC L	RR L	SRA L	SRL L	BIT 1,L	BIT 3,L	BIT 5,L	BIT 7,L	RES 1,L	RES 3,L	RES 5,L	RES 7,L	SET 1,L	SET 3,L	SET 5,L	SET 7,L
RRC (HL)	RR (HL)	SRA (HL)	SRL (HL)	BIT 1,(HL)	BIT 3,(HL)	BIT 5,(HL)	BIT 7,(HL)	RES 1,(HL)	RES 3,(HL)	RES 5,(HL)	RES 7,(HL)	SET 1,(HL)	SET 3,(HL)	SET 5,(HL)	SET 7,(HL)
RRC A	RR A	SRA A	SRL A	BIT 1,A	BIT 3,A	BIT 5,A	BIT 7,A	RES 1,A	RES 3,A	RES 5,A	RES 7,A	SET 1,A	SET 3,A	SET 5,A	SET 7,A
0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

Tabellen DD		und FD		Tab.DD/CB+FD/CB	
Erstes Byte DD für ii=IX, FD für ii=IY	09 ADD ii,BC	A6	AND (ii+d)	zweites Byte CB	06 RLC (ii+d)
	19 ADD ii,DB	AE	XOR (ii+d)		0E RRC (ii+d)
	21 LD ii,nn	B6	OR (ii+d)		16 RL (ii+d)
	22 LD (nn),ii	BE	CMP (ii+d)		1E RR (ii+d)
	23 INC ii	CB	Tab.DD/CB		26 SLA (ii+d)
	29 ADD ii,ii	E1	POP ii		2E SRA (ii+d)
	2A LD ii,(nn)				3E SRL (ii+d)
	2B DEC ii	E3	EX (SP),ii		46 BIT 0,(ii+d)
	34 INC (ii+d)	E5	PUSH ii		4E BIT 1,(ii+d)
	35 DEC (ii+d)	E9	JMP (ii)		56 BIT 2,(ii+d)
	36 LD (ii+d),n	F9	LD SP,ii		5E BIT 3,(ii+d)
	39 ADD ii,SP				66 BIT 4,(ii+d)
	46 LD B,(ii+d)				6E BIT 5,(ii+d)
	4E LD C,(ii+d)				76 BIT 6,(ii+d)
	56 LD D,(ii+d)				7E BIT 7,(ii+d)
	5E LD E,(ii+d)				86 RES 0,(ii+d)
	66 LD H,(ii+d)				8E RES 1,(ii+d)
	6E LD L,(ii+d)				96 RES 2,(ii+d)
	70 LD (ii+d),B				9E RES 3,(ii+d)
	71 LD (ii+d),C				A6 RES 4,(ii+d)
72 LD (ii+d),D			AE RES 5,(ii+d)		
73 LD (ii+d),E			B6 RES 6,(ii+d)		
74 LD (ii+d),H			BE RES 7,(ii+d)		
75 LD (ii+d),L			C6 SET 0,(ii+d)		
77 LD (ii+d),A			CE SET 1,(ii+d)		
7E LD A,(ii+d)			D6 SET 2,(ii+d)		
86 ADD (ii+d)			DE SET 3,(ii+d)		
8E ADC (ii+d)			E6 SET 4,(ii+d)		
96 SUB (ii+d)			EE SET 5,(ii+d)		
9E SBC (ii+d)			F6 SET 6,(ii+d)		
			FE SET 7,(ii+d)		
				Erstes Byte DD(IX) oder FD(IY),	

Tabelle ED

IN B	IN D	IN H	IN F	LDI	LDIR	0
OUT B	OUT D	OUT H		OP1	OP1R	1
SBC HL,BC	SBC HL,DE	SBC HL,HL	SBC HL,SP	INI	INIR	2
LD (nn),BC	LD (nn),DE		LD (nn),SP	OUTI	OTIR	3
NEG						4
RETN						5
IMO	IM1					6
LD I,A	LD A,I	RRD				7
IN C	IN E	IN L	IN A	LDD	LDDR	8
OUT C	OUT E	OUT L	OUT A	OPD	OPDR	9
ADC HL,BC	ADC HL,DE	ADC HL,HL	ADC HL,SP	IND	INDR	A
LD BC,(nn)	LD DE,(nn)		LD SP,(nn)	OUTD	OTDR	B
						C
RETI						D
	IM2					E
		RLD				F
4	5	6	7	A	B	

VEB Elektronik Gera

G A B - N a c h w e i s

gemäß ASVO vom 01.12.1977

Benennung	Microcomputer MC 80
Hersteller	VEB Elektronik Gera, Betriebs- teil B
Entwicklungsstand	K 5/0

**1. Hauptmerkmale der gesundheitsschutz- und arbeitsschutz-
sowie brandschutztechnischen Lösung****1.1. Mechanisch-konstruktive Ausführung**

Das Gerät MC 80 ist als Auftischgerät ausgeführt. Die Aufstellung am Arbeitsplatz obliegt damit weitgehend den Anforderungen des Anwenders.

Das Gehäuse weist abgerundete Ecken und Kanten auf und ist frei von störenden Vorsprüngen.

Lösbare Teile sind die Tastatur und der Löschkammereinschub. Das Abnehmen der Tastatur erfordert sicheres Zufassen.

Mechanisch bewegte Teile treten nur im Magnetbandlaufwerk auf. Der Antrieb erfolgt im Normalfall bei geschlossener Klappe. Das Drehmoment ist gering und ungefährlich.

Die Tastatur garantiert eine günstige Handauflage, die Ermüdungserscheinungen beim Eingeben vorbeugt. Der Aufwand für die Tastenbetätigung ist gering und entspricht der elektronischen Schreibmaschine.

Für die Bildröhre ist Implosionsschutz durch die Metallarmierung gegeben, so daß bei sachgemäßer Montage und Handhabung keine spontane Implosion auftritt.

1.2. Elektrische Ausführung

Das Gerät ist in Schutzklasse I (mit Schutzleiteranschluß) ausgeführt und nur an Netzen mit Schutzmaßnahme nach TGL 200-0602/02 mit Schutzleiteranschluß betreibbar.

Bei angesteckter Tastatur, geschlossener Löschkammer und abgedeckter Programmierfassung wird der Schutzgrad IP 20 erreicht.

Damit ist ein Berühren spannungsführender Teile ausgeschlossen. Durch die Einhaltung der Kriech- und Luftstrecken und die Leitungsführung ist auch eine gefährdende Näherung an spannungsführende Teile nicht möglich.

Für die zum Betreiben der Bildröhre benötigte Hochspannung von 9 bis 12 kV sind im Gerät besondere Abschirmungs- und Sicherheitsmaßnahmen getroffen worden durch

- Käfig für Hochspannungstransformator
- TGL-gerechte Isolierung der Anodenleitung
- Anodenclip mit Plastschutzkappe

Diese Maßnahmen kommen dem Service zugute.

Nach Abnahme der Abdeckung ist die Programmierfassung zugänglich. Bei gestecktem EPROM-Schaltkreis ist hier eine Berührung von Kleinspannungen möglich. Während des Betriebes treten max. 27 V, potentialgetrennt vom Netz, auf.

Die Einhaltung der Schutzklasse I wird durch folgende Maßnahmen gesichert

- Schutzleiteranschluß der Gehäuseteile über Kabel oder Rahmen und Schraubverbindung mit Zehnerscheibe, die sich in die Oberfläche eindrückt.
Die Schraubverbindungen für den Schutzleiteranschluß werden in der Dokumentation genannt.
- Schutzleiterprüfung mit $I = 15 \text{ A}$ entspr. TGL 200-0602/02
- Isolationsprüfung mit 1,5 kV, 50 Hz zwischen Netz und Gehäuse verbunden mit Sekundärseite.

Am Netzschalter und im Hochspannungsteil ist Funkenbildung möglich. Ein Explosionsschutz für explosive Gase ist damit nicht gegeben.

Beim Stecken ist der Steckverbinder für den Bediener nicht zugänglich.

1.3. Optische Beanspruchung

Die Farbgebung des Gerätes ist neutral und unauffällig. Als Datensichtanzeige wurde ein grüner Bildschirm gewählt, der günstige Augenverträglichkeit gewährleistet. Es wird ein ruhig stehendes Bild erzeugt. Die Helligkeit der Bildschirmanzeige ist einstellbar.

1.4. Röntgenstrahlung

Die Röntgenstrahlungs-dosis ist im Arbeitsabstand (ca. 0,6 m) weit unterhalb dem zulässigen Maximalwert von 0,75 mR/h. Der Hersteller der Bildröhre gibt im Abstand von 5 cm einen Maximalwert von 0,5 mR/h an.

1.5. Ozongasbildung

Die UV-Strahlung der Löschkammer ($d_{\max} = 250 \text{ mm}$) gelangt nicht direkt nach außen.

Mit der UV-Strahlung einher geht die Ozongasbildung. Für Ozongas gelten folgende Werte:

Geruchsschwelle: $0,02 \text{ mg/m}^3$

MAK-Wert: $0,2 \text{ mg/m}^3$

Im Arbeitsbereich (Abstand ca. 30 cm) wurden unter ungünstigen Bedingungen (6 Std. Brenndauer, Kammer halb geöffnet, kleiner Raum mit geringer Luftzirkulation), Meßwerte bis etwa $0,05 \text{ mg/m}^3$ ermittelt.

2. Verbleibende Gefährdungen und Erschwernisse, sowie erforderliche Verhaltensmaßnahmen zur Verhütung schädlicher Auswirkungen

Abweichungen von den Rechtsvorschriften treten nicht auf. Eingriffe in das Gerät sind durch den Anwender im Bereich des Mikrorechners (Aufnahme 1) zulässig. Das Gerät darf jedoch nur bei gezogenem Netzstecker geöffnet werden.

Zum Zwecke der Prüfung und Reparatur darf das geöffnete Gerät nur von unterwiesenem Fachpersonal betrieben werden.

Beim Transport des Gerätes ist die Tastatur und das Netzkabel abzunehmen. Abnehmen und Anstecken der Tastatur hat auf einer ebenen Tischplatte zu erfolgen.

Zur Vermeidung des Ozongeruches ist die Löschkammer geschlossen zu betreiben und für gute Lüftung zu sorgen.

Wird seitens des Bedieners ein größerer Abstand zum Bildschirm gewünscht, so kann das Gerät mit abgesetzter Tastatur betrieben werden. Hierzu ist für den 58-poligen Steckverbinder ein Kabel (Plastschlauchleitung) mit Griffschalen anzufertigen.

Das Gerät darf nicht in Räumen mit explosiven Medien betrieben werden.

Die für den Anwender wichtigen Verhaltensmaßnahmen werden in der Bedienungsanleitung ausgewiesen.

3. Gutachten und Dokumente

- Meßprotokoll über Ozonbildung von der Arbeitshygiene-
inspektion Gera (Schreiben vom 14.10.82)
- TGL 7705 Allgemeine technische Bedingungen für Bild-
wiedergaberöhren

Gera, den 10.05.83

gez. Dr. Brose
Direktor WUT

Stellungnahme der Schutzgütekommision:

Die Anforderungen der ASVO § 3 (1) und (2) sind erfüllt.

Reg.-Nr. 132/83

Gera, den 10.05.83

gez. i.V. Milker
Vors.d.Kommission

Beschreibung
TKO-Test "TKO 2"

1. Ziel und Umfang der Prüfung

Mit Hilfe des Testprogrammes TKO 2 soll die Funktionsfähigkeit des Mikrocomputers MC 80.22 nachgewiesen werden. Die Funktionsgruppen Magnetbandeinheit, Tastatur und EPROM-Programmierung werden einmalig, die Speicherbaugruppen und das Display zyklisch getestet.

2. Komponenten

2.1. Magnetband

Der Magnetbandtest beginnt mit dem Schreiben von 3FH Blöcken eines zufällig ausgewählten Bereiches. Diese Aufzeichnung wird anschließend gelesen und verglichen. Bei ordnungsgemäßer Schreib- und Lesefunktion wird zum Schluß die Bandendabschaltung und der Kassettenauswurf getestet.

Die Bedienung des Laufwerkes erfolgt ohne Bedienerkommandos (Ausnahme: Auswurf). Der gesamte Magnetbandtest wird protokolliert.

2.2. Uhr

Mit Beginn der Tastaturprüfung wird eine Uhr gestartet, die ständig in der rechten oberen Bildecke Stunden, Minuten und Sekunden anzeigt. Der TKO-Test ist für eine Dauer von 100 Stdn. ausgelegt; nach dieser Zeit und Beendigung des Prüfzyklus ROM-RAM-Display erscheint die Ausschrift 'ENDE'.

2.3. Tastatur

Die Tastatur wird auf Schlüsse und Unterbrechungen geprüft. Dazu wird auf dem Display die jeweils zu prüfende Taste blinkend und alle gedrückten Tasten (z. B. auch bei Schluß) statisch angezeigt. Nur bei Erkennung der einzeln bestätigten, richtigen Taste wird der Test mit der folgenden Taste fortgesetzt. Die Funktion der BREAK-Taste wird mitgeprüft, die der RESET-Taste nicht.

2.4. EPROM-Programmierung

Zu diesem Test ist ein Prüf-EPROM zu stecken. Der Inhalt dieses EPROM wird gelesen und mit dem Soll-Inhalt verglichen. Dadurch werden Fehler auf Adressen-, Daten- und Steuerleitungen und fehlende Betriebsspannungen erkannt. Eine Kontrolle der Programmierung erfolgt nicht.

2.5. Zyklussteuerung

Die Zyklussteuerung ruft nacheinander die Komponenten ROM, RAM und Display auf und zählt die Zyklen.

2.6. ROM

Die Module S 0.1 - S 0.4, S 1.1. - S 1.5, S 2, S 3.1 - S 3.7 und S 4 werden durch Vergleich ihrer CRC-Werte mit dem Sollwert auf Vollständigkeit und Richtigkeit geprüft. Die Bildung des Prüffrestes erfolgt in einem Zyklus zu jedem EPROM 10 mal, so daß auch instabiles Verhalten erkannt wird. Fehler werden durch '*' hinter dem Prüffrest angezeigt. Bei den Geräten ohne BASIC sind die CRC-Werte für S 3 und S 4.1 - S 4.7 ungültig.

2.7. RAM

Es wird der Operativspeicher K 3525 auf Adresse C000H-FFFFH geprüft. Dazu wird jeweils 1 Bit im gesamten Bereich registriert, der Prüffrest über diese 16 KByte gebildet und mit dem Sollwert verglichen (Fehleranzeige wie bei ROM). Da während dieser Prüfung das auf dem RAM befindliche Testprogramm nicht genutzt werden kann, läßt die Zyklussteuerung die Komponenten RAM und Uhr (Interruptserviceroutine) auf den RAM-Bereich der ZRE um.

2.8. Display

Mit Hilfe der Komponente Display (2 Bilder) ist eine visuelle Überprüfung der Bildgeometrie, -linearität und des DA-Wandlers möglich.

3. Testhilfsmittel

- 1 Kassette mit TKO-Testprogramm
- 1 Test-EPROM mit Bitmuster (HL) : = L

Bedienungsanleitung

zum TKO-Test "TKO 2"

Das TKO-Testprogramm für den MC 80.22 ist 40 Blöcke lang. Es befindet sich auf der Kassette ab Block 0001 und ist in den RAM-Bereich E000H bis FFFH zu laden. Der Start erfolgt aus dem Menü.

TEST 01:

MAGNETBAND

Zum Test der Magnetbandfunktionen Schreiben, Lesen und Spulen verbleibt die Kassette mit dem Testprogramm im Gerät. Alle Funktionen, außer der Kontrolle des Kassettenauswurfes, werden automatisch ohne Bedienerkommandos ausgeführt; ein Abbruch des Testes ist nur mit 'RESET' möglich. Die aktuelle Laufwerkfunktion wird unterhalb der Versuchsanzahl durch \rightarrow , \leftarrow , RD, WR, CUE bzw. Leerzeichen für Stop gezeigt. Beim Betätigen des Kassettenauswurfes beginnt nach dem Schließen ein neuer Testzyklus.

SCHREIBEN

Nach dem Start des TKO-Testes aus dem Menü beginnt die Aufzeichnung von 3 FH Blöcke ab Block 0041H; die RAM-Adresse wird zufällig zwischen C000H-C07FH gewählt.

Im Fehlerprotokoll steht für jeden zu schreibenden Block ein Punkt, der bei erfolgter Aufzeichnung verschwindet. Die erste Position im Fehlerprotokoll ist für den zu suchenden Block 0040H reserviert; hier wird die Anzahl der Spulversuche bis zum Erreichen des Fangbereiches angezeigt (z.B. 4 S) steht statt 'S' ein 'C', wird ein Sektor mit CRC gelesen. Sollte nach 9 Spulversuchen Block 0040 nicht gefunden worden sein, erscheint die Ausschrift 'Block 0040 fehlt' oder bei fehlender Bandmarke auch 'FDAT' und nach kurzer Zeit wird ein neuer Versuch gestartet.

LESEN

Nach der Aufzeichnung erfolgt das Lesen der 3FH Blöcke in den RAM ab Adresse E080H. Das Fehlerprotokoll wird wie beim Schreiben benutzt. Für den Block 0041 sind i.a. 3-4 Spulversuche erforderlich, so daß im Fehlerprotokoll die Ausschrift 3 S bzw. 4 S erscheint. Geht beim Lesen die Synchronisation verloren oder ist ein Block nicht ordnungsgemäß aufgezeichnet worden, wird dies im Fehlerprotokoll ebenfalls mit der Anzahl der Spulversuche und dem 'S' angezeigt.

VERGLEICH

Sind alle 3FH Blöcke eingelesen, erfolgt ein Vergleich der RAM-Bereiche E080-EFFF mit C0...CF... Bei Vergleichsfehlern (z. B. wenn die Aufzeichnung von vorausgegangenen Tests gelesen wurde, weil das Schreiben nicht funktioniert), wird im Fehlerprotokoll die Anzahl Fehler pro Block (max. 80H) angezeigt und anschließend ein neuer Testzyklus begonnen.

KONTROLLE BANDENDABSCHALTUNG UND KASSETTENAUFWURF

Die Kassette wird im schnellen Rücklauf zum Anfang gespult. Bei ordnungsgemäßer Funktion der Bandendabschaltung wird der Bediener zur Entnahme der Kassette aufgefordert.

TEST 02:

TASTATUR

Bei diesem Test werden die einzelnen Tasten reihenweise fortlaufend überprüft und auf dem Bildschirm angezeigt. Gleichzeitig startet beim Tastaturtest eine Uhr, die die Dauer des Testes bis maximal 100 h anzeigt. Das Zeichen der jeweils zu testenden Taste blinkt auf dem Bildschirm. Wenn die Taste in Ordnung ist, verlischt nach dem Betätigen das Zeichen auf dem Bildschirm und das nächste blinkt. Nach dem Drücken der Umschalttaste () muß mittels einer anderen UC-Taste wieder zurückgeschaltet werden. Bei auftretenden Fehlern müssen diese erst beseitigt werden, bevor die nächste Taste geprüft werden kann. Wenn mehrere Tasten Schluß untereinander haben, leuchten alle entsprechenden Zeichen auf dem Bildschirm, blinken aber nicht. Wenn die Tastatur fehlerfrei ist, folgt nach prüfen der letzten Taste (ENTER) der nächste Test.

DAUER 00:02:30

TEST 03:

EPROM - PROGRAMMIERUNG

EPROM - STECKEN

ENTER DRÜCKEN

Ein mit einem definierten Bitmuster programmierter EPROM wird auf die Fassung gesteckt, eingelesen und mit einem Speicherbereich verglichen. Wenn der Vergleich keinen Fehler ergibt, wird der zyklische Dauertest gestartet.

Ist ein leerer oder falscher EPROM gesteckt bzw. die Fassung defekt, erscheint die Ausschrift "EPROM LEER" bzw. "EPROM FEHLER".

Bei der Ausschrift "EPROM DEFEKT"? ist zu kontrollieren, ob überhaupt ein EPROM gesteckt und ob er richtig gesteckt ist. Wenn die Programmierung in Ordnung ist, folgen 5 Tests, die zyklisch abgearbeitet werden.

ZYKLUS 0001

DAUER: 00:29:19

TEST 11: MODUL

S 0.1 - S 1.5

0.1 8022H

1.2 53ECH

0.2 D03PH

1.3 FFA7H

0.3 8203H

1.4 61ECH

0.4 20BPH

1.5 1048H

1.1 45DDH

ZYKLUS 0001

DAUER 00:29:28

TEST 12: MODUL

S 2 - S 4.7

2 0E06H

3 C31AH

4.1 0D6DH

4.5 75C4H

4.2 A63CH

4.6 AF6PH

4.3 B853H

4.7 367DH

4.4 A75EH

ZYKLUS 0001

DAUER 00:29:55

TEST 13: RAM C000H - FFFFH

	/BIT PRÜFSUMME		/BIT PRÜFSUMME
-	046DH	4	0A3AH
0	EA9DH	5	19F1H
1	C9ACH	6	3F55H
2	8FCEH	7	721DH
3	030AH	-	046DH

- zwei verschiedene Grafikdarstellungen

Bei diesen Tests, die in der Schleife immer wieder abgearbeitet werden, muß auf fehlerhafte Prüfsummen (hinter der jeweiligen Prüfsumme) geachtet werden.

Der Test läuft 100 h lang und wird dann am Ende eines Zyklus abgebrochen. Auf dem Bildschirm blinkt dann "ENDE".