

SIEMENS

Experimentier- computer ECB85

Bedienungsanleitung 11.81

Herausgegeben von

Siemens AG, Bereich Bauelemente, Balanstraße 73, 8000 München 80

Für die angegebenen Schaltungen, Beschreibungen und Tabellen wird keine Gewähr bezüglich der Freiheit von Rechten Dritter übernommen.

Liefermöglichkeiten und technische Änderungen vorbehalten.

Fragen der Technik, Preise und Liefermöglichkeiten richten Sie bitte an unsere Zweigniederlassungen im Inland. Abteilung VB oder an unsere Landesgesellschaften im Ausland (siehe Geschäftsstellenverzeichnis).

SIEMENS

Bedienungsanleitung
Ausgabe 11.81

ECB85
Experimentiercomputer

Experimentiercomputer ECB85

Die in dieser Schrift beschriebene Baugruppe ist unter der folgenden Bestell-Nummer lieferbar:

Typ	Funktion	Bestell-Nr.
ECB85-S710	Experimentiercomputer	B88385-A-S710

Mit den Angaben werden die Bauelemente spezifiziert, nicht Eigenschaften zugesichert.

Liefermöglichkeiten und technische Änderungen vorbehalten.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts ist nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadensersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung.

(C) Siemens AG 1981

I n h a l t

Vorwort	5
1. Allgemeine Grundlagen	9
1.1 Vorbereiten des Anwender-Programms	10
1.2 Eingeben und Testen des Anwender-Programms	14
1.3 Testen der Hardware	15
1.4 Das Magnetbandgerät als Speicher	16
1.5 Das Arbeiten mit EPROMs	17
2. Beschreibung des ECB85	18
2.1 Hardware des Geräts	19
2.2 Software des Geräts	23
2.3 Überblick über die Kommandos	26
2.4 Das automatische Umrechnen von Adressen	28
2.5 Das Prinzip der Schattenregister	30
2.6 Das Unterbrechen von Programmen	31
3. Allgemeines zum Arbeiten mit dem ECB85	33
3.1 Lageplan der Hardware	34
3.2 Anschließen der Betriebsspannung	37
3.3 Anschließen des Magnetbandgeräts	38
3.4 Belegung des Adreßraums	39
3.5 Programm-Beispiel	41
3.6 Inbetriebnahme des Geräts	46
3.7 Prinzipien der Kommunikation	47
3.8 Verwendete Symbolik des Anzeigefelds	51
3.9 Einleiten eines Kommandos	56
4. Kommandos mit Bezug auf den Speicher	57
4.1 Speicher auslesen und beschreiben (Kommando 0)	58
4.2 Speicherbereich mit Konstante füllen (Kommando 1)	62
4.3 Speicherinhalt übertragen (Kommando 2)	66
4.4 Adressen umrechnen (Kommando 3)	71
4.5 EPROMs beschreiben (Kommando 0 bis 2)	76

5.	Kommandos zum Testen der Hardware	79
5.1	Über Ausgabe-Kanal ausgeben (Kommando 4)	81
5.2	Über Eingabe-Kanal eingeben (Kommando 5)	84
6.	Kommandos zum Testen der Software	87
6.1	Register auslesen und beschreiben (Kommando 6)	89
6.2	Programm ohne Unterbrechungspunkte starten (Kommando 7)	92
6.3	Programm mit Unterbrechungspunkten starten (Kommando 8)	96
6.4	Programm im Einzelschritt abarbeiten (Kommando 9)	102
6.5	Programm-Wahlschalter und I-Taste	106
7.	Kommandos mit Bezug auf das Magnetbandgerät	110
7.1	Information auf Magnetband schreiben (Kommando A)	111
7.2	Information von Magnetband lesen (Kommando B)	115
8.	Peripherie für Anwender-Programme	119
8.1	Arbeiten mit dem Tasten- und Anzeigefeld	120
8.2	Das Anschließen von Anwender-Hardware	128
8.3	Arbeiten mit dem Baustein SAB8155	134
8.4	Arbeiten mit dem Baustein SAB8085A	151
9.	Technische Unterlagen und Arbeitshilfen	167
9.1	Hardware-Unterlagen	168
9.2	Software-Unterlagen	176
9.3	Übersetzungstabelle für die Assembler-Befehle	222
9.4	Zusammenstellung der Kommandos	227

Vorwort

Die Computertechnik hat in den wenigen Jahrzehnten ihres Bestehens einen stürmischen Aufschwung genommen. Es erscheint heute selbstverständlich, daß Produktionsanlagen und Kraftwerke von Computern gesteuert, Karteien und Konten von Computern verwaltet werden. Computer waren jedoch trotz aller technischen Fortschritte bis vor wenigen Jahren recht umfangreiche und teure Geräte, aufgebaut aus zahlreichen Einzelteilen. Ihr Einsatz beschränkte sich daher auf Anwendungen, bei denen große Datenmengen zu verarbeiten sind.

Mit der Entwicklung von Mikrocomputern, Geräten aus einem oder wenigen hochintegrierten Halbleiter-Bausteinen, hat sich die Situation grundlegend geändert. Ihre geringe Größe und ihr niedriger Preis machen sie geeignet, sowohl in Maschinen der industriellen Technik als auch in Geräten des täglichen Bedarfs Funktionen zu übernehmen, die bisher von anwendungsspezifisch entwickelten Steuer-, Meß- und Regelschaltungen ausgeführt wurden. Vereinzelt werden Mikrocomputer sogar schon dazu benutzt, die Abläufe in Waschmaschinen, Küchenherden, Fernsehgeräten und Autos zu steuern.

Im Zuge dieser Entwicklung werden immer mehr Menschen mit dem Mikrocomputer in Berührung kommen. Konstrukteure in der Industrie sehen in zunehmendem Maße die Notwendigkeit, sich in diese neue Technik einzuarbeiten. Auch interessierte Laien suchen Möglichkeiten, sich mit dem Mikrocomputer vertraut zu machen. Dabei darf nicht übersehen werden, daß es sich bei Mikrocomputern um technisch komplizierte Systeme handelt, deren zweckmäßiger Einsatz Vorkenntnisse voraussetzt. Mikrocomputer haben, ebenso wie Großcomputer, eine vom Anwendungsfall unabhängige Schaltung und erhalten ihre speziellen Eigenschaften erst durch die Programmierung. Wer diese neue Technik anwenden will, muß sich daher mit zwei Gebieten auseinandersetzen: der Schaltungstechnik (Hardware) und der Programmierung (Software).

Es gibt heute eine Reihe verschiedener Mikrocomputer-Systeme, die sich in Aufbau und Programmierung unterscheiden. Unter den besonders vielseitigen

8-Bit-Typen sind das System 8080 und dessen modernerer aufwärts-kompatibler Nachfolger 8085 am weitesten verbreitet und daher für den potentiellen Anwender besonders interessant.

Als Hilfe bei der Einarbeitung in die neue Technik erfreuen sich Experimentiercomputer besonderer Beliebtheit, da sie ohne langwierige Aufbau- und Testarbeiten sofort die Ausführung selbst geschriebener Programme erlauben. Der ECB85 ist ein solches Gerät auf der Basis des Systems 8085. Er wird fertig aufgebaut und geprüft einschließlich Tastenfeld-Eingabe und Ziffernfeld-Ausgabe geliefert und ist nach Anschließen einer vom Benutzer bereitzustellenden Spannungsquelle von +5V betriebsbereit.

Als Lern- und Übungsgerät für die Mikrocomputer-Systeme 8080 und 8085 erlaubt der ECB85 zunächst das Eingeben, Ändern, Testen und Ausführen von Programmen. Zum Abspeichern und Wieder-Einlesen von Programmen und Daten kann über eine Normbuchse ein Tonbandgerät oder Kassettenrekorder angeschlossen werden, und der eingebaute EPROM-Programmierer ermöglicht das Beschreiben von EPROM-Bausteinen. Über die Hardware-Schnittstelle des ECB85 können außerdem periphere Schaltungen angesteuert werden, so daß sich die Funktion eines Programms auch in der realen Umwelt überprüfen läßt.

Durch seine besondere Leistungsfähigkeit eignet sich der ECB85 über seine Eigenschaft als Lernhilfe hinaus noch für eine Reihe professioneller Anwendungsfälle. So kann der in den ECB85 eingebaute EPROM-Programmierer sehr gut dazu benutzt werden, um "vor Ort" den Inhalt von EPROMs zu ändern. Dies ist z.B. dann erwünscht, wenn ein auf einem professionellen Entwicklungssystem erstelltes Programm am Platz des Einsatzes durch dort ermittelte spezielle Konstanten ergänzt werden muß. Ein anderes Anwendungsgebiet ist die Realisierung kleiner Maschinensteuerungen in geringer Stückzahl. Für diesen Einsatz ist es vorteilhaft, daß der ECB85 neben dem eigentlichen Mikrocomputer auch ein Tasten- und Anzeigefeld

als Kommunikationsinstrument mit dem Anwender und ein Lochrasterfeld für den Aufbau spezieller Interface-Schaltungen enthält. Außerdem ermöglichen der auf dem ECB85 vorhandene Programm-Wahlschalter und die Interrupt-Taste interessante Betriebsarten.

Die vorliegende Bedienungsanleitung erklärt die Benutzung des ECB85 und gibt Hinweise für den zweckmäßigen Einsatz. Auf den Befehlsvorrat und die Schaltungstechnik der Systeme 8080 und 8085 kann diese Schrift jedoch nicht eingehen. Wer sich darüber informieren möchte, sei auf das allgemeine Mikrocomputer-Schrifttum von Siemens verwiesen.

1. Allgemeine Grundlagen

Diese Bedienungsanleitung für den ECB85 setzt die Kenntnis der Assembler-Programmiersprache der Systeme 8080 und 8085 voraus, die man sich anhand der entsprechenden Siemens-Themenbücher aneignen kann. Darüber hinaus empfehlen sich bei der Anwendung des ECB85 bestimmte Vorgehensweisen, die in den folgenden Kapiteln behandelt werden. Auf die speziellen Eigenschaften des ECB85 wird dabei noch nicht eingegangen.

Bevor ein Anwender-Programm überhaupt in den ECB85 eingegeben werden kann, sind gewisse Vorarbeiten erforderlich, denen das folgende Kapitel gewidmet ist. Daran anschließend wird in allgemeiner Form das Eingeben und Testen eines Programms mit Hilfe des ECB85 behandelt. Ein weiteres Kapitel befaßt sich mit dem Testen eventuell angeschlossener externer Hardware. Schließlich wird noch die Anwendung des Magnetbandgeräts als externer Speicher behandelt und das Arbeiten mit EPROMs erläutert.

1.1 Vorbereiten des Anwender-Programms

Die Befehle des Anwender-Programms werden beim ECB85 im Hexadezimal-Code ein- und ausgegeben. Das bedeutet jedoch nicht, daß der Benutzer auf dieser Ebene arbeiten sollte. Um Fehler zu vermeiden, empfiehlt es sich, das Programm zunächst in Assembler-Sprache zu schreiben. Bei der Assembler-Darstellung läßt sich die Funktion der Befehle viel besser überblicken als bei einem reinen Zahlen-Code.

Die Aufgabe der Programmierung besteht darin, mit einem gegebenen beschränkten Befehlsvorrat eine bestimmte vorgegebene Funktion zu erfüllen. Die zweckmäßige Verwendung der Befehle erfordert Erfahrung und Übung, die am besten mit dem Durcharbeiten von Programm-Beispielen erworben wird.

Programme können recht komplex werden. Es empfiehlt sich daher, die Arbeit gut zu planen. Ein sehr brauchbares Hilfsmittel sind die Programm-Ablaufpläne. Man kann mit einem ganz groben Plan anfangen und ihn in weiteren Teilplänen verfeinern. Erst danach wird mit der eigentlichen Programmierung in Assembler-Sprache begonnen.

Das Übersetzen vom Assembler- in den Hexadezimal-Code (Assemblierung), das meistens auf einem Computer durch das Assembler-Übersetzungsprogramm ausgeführt wird, kann auch von Hand mit Hilfe der alphabetisch geordneten Übersetzungstabelle in Kapitel 9.3 vorgenommen werden. Bei der Übersetzung von Hand ist planmäßiges Vorgehen besonders wichtig, um unnötige Arbeit zu vermeiden. Ein größeres Programm sollte nicht auf einmal übersetzt werden, da ein Fehler am Programmanfang die Überarbeitung des ganzen Programms erforderlich macht. Man teilt es in kleine überschaubare Blöcke mit übersichtlich definierten Schnittstellen ein und übersetzt und testet immer nur einen Block. Bei einem Fehler muß eventuell der ganze Block überarbeitet werden, da sich beim Einfügen oder Weglassen von Befehlen alle darauf folgenden Sprungadressen ändern. Nach dem Test des Blocks kommt der adreßmäßig nächste Block an die Reihe.

Zum Übersetzen der Assembler-Befehle von Hand verwendet man am besten ein Formular mit vier Spalten:

1. 1 Schreibstelle für die Zahl der Bytes pro Befehl
2. 4 Schreibstellen für die hexadezimalen Adressen des ersten Bytes der Befehle
3. 6 Schreibstellen für die Hexadezimal-Darstellung der Befehle
4. Der Rest für die Assembler-Darstellung der Befehle und die Kommentare.

Zunächst werden in Spalte 4 dieses Formulars die Assembler-Befehle geschrieben. Benutzt man neben den Assembler-Symbolen nur Hexadezimal-Zahlen und symbolische Adressen, so geht die Übersetzung des Assembler-Programms in die Hexadezimal-Darstellung folgendermaßen vor sich:

- a) Nach der Tabelle in Kapitel 9.3 wird zu den Assembler-Befehlen die Zahl der Bytes in Spalte 1 und das erste Byte der Hexadezimal-Darstellung in Spalte 3 eingetragen.
- b) Beginnend mit der Anfangsadresse werden mit Hilfe der Zahl der Bytes in Spalte 1 die Adressen der ersten Bytes der Befehle in Spalte 2 eingeschrieben.
- c) Aus den Angaben in Spalte 2 und 4 wird eine Tabelle mit den symbolischen Adressen und den zugehörigen Zahlenwerten aufgestellt.
- d) Die zweiten und dritten Bytes der Hexadezimal-Darstellung, falls vorhanden, werden mit Hilfe der Tabelle der symbolischen Adressen in Spalte 3 angefügt. Es ist dabei zu beachten, daß in der Hexadezimal-Darstellung auf das erste Byte eines Befehls zunächst das höherwertige und dann das niederwertige Byte der Zahl folgt.

Zum besseren Verständnis ist ein ausgefülltes Übersetzungsformular als Beispiel in Tabelle 1.1-1 angefügt. Dieses Beispiel wird in einem späteren Kapitel noch näher erläutert werden. An dieser Stelle soll schon allgemein darauf hingewiesen werden, daß in dieser Bedienungsanleitung

Zahlen in Zusammenhang mit Programmen grundsätzlich hexadezimal zu verstehen sind, weshalb auf eine besondere Kennzeichnung verzichtet wird. In Assembler-Programmen werden die Zahlen trotzdem, wo erforderlich, durch ein angehängtes H als hexadezimal gekennzeichnet, damit die Konventionen der Assembler-Sprache eingehalten werden.

Tabelle 1.1-1

Beispiel für die Durchführung der manuellen Assemblierung

Spalte 1	Spalte 2	Spalte 3	Spalte 4
Anzahl der Bytes pro Befehl	Hexadezimal-Adresse des ersten Bytes	Hexadezimal-Darstellung der Befehle	Assembler-Darstellung der Befehle
1 Schreibstelle	4 Schreibstellen	6 Schreibstellen	
2	07EE	0E08	ANWMUL: MVI C,8
1	07F0	7A	MOV A,D
3	07F1	210000	LXI H,0
1	07F4	54	MOV D,H
1	07F5	29	MUL1: DAD H
1	07F6	17	RAL
3	07F7	D2FB07	JNC MUL2
1	07FA	19	DAD D
1	07FB	0D	MUL2: DCR C
3	07FC	C2F507	JNZ MUL1
1	07FF	C9	RET

Tabelle der symbolischen Adressen: ANWMUL = 07EE
MUL1 = 07F5
MUL2 = 07FB

1.2 Eingeben und Testen des Anwender-Programms

Nach dem Übersetzen des Anwender-Programms von Assembler- in Maschinensprache wird es mit Hilfe bestimmter Kommandos in den Schreib-/Lese-Speicher des ECB85 eingegeben. Neben dem sequentiellen Schreiben von Befehlen und Daten in den Speicher ist auch das Kontrollieren des Speicherinhalts und das gezielte Überschreiben einzelner Speicherzellen möglich. Außerdem kann der Inhalt ganzer Blöcke im Speicher verschoben werden, was die Korrektur bei versehentlich weggelassenen oder mehrfach eingegebenen Bytes erleichtert.

Nachdem das Programm in gewünschter Weise im Speicher steht, kann es getestet werden. Hierfür bietet der ECB85 Kommandos zum Beschreiben und Auslesen von Registern und zum Starten von Programmen mit dem wahlweisen Setzen von bis zu zwei Unterbrechungspunkten. Normalerweise geht der Programmtest in drei Schritten vor sich:

1. Eingeben von Zahlenwerten in die Register des Mikrocomputers.
2. Starten des Programms bei einer vorgegebenen Adresse und Anhalten bei einer vorgegebenen Adresse.
3. Feststellen der Reaktion (bei Steuerungsaufgaben) bzw. Ausgeben von Zahlenwerten aus den Registern des Mikrocomputers (bei Berechnungen).

Ein Vergleich der durch das Programm erhaltenen Ergebnisse mit den theoretisch zu erwartenden Ergebnissen zeigt, ob das Programm richtig gearbeitet hat. Um Fehler auszuschließen, muß in den meisten Fällen der Test mit verschiedenen Werten wiederholt werden.

1.3 Testen der Hardware

Mikrocomputer-Anwendungen können sehr komplex werden, und entsprechend aufwendig ist dann auch die Suche nach eventuellen Fehlern. Beim Testen ist systematisches Vorgehen dringend notwendig, um den Zeitaufwand nicht allzu stark anwachsen zu lassen. Es wurde schon darauf hingewiesen, daß das Programm aus diesem Grund zum Testen in kleine Blöcke zerlegt werden sollte. Ein anderer empfehlenswerter Grundsatz ist das getrennte Testen von Hard- und Software. Dieser Fall tritt beim ECB85 natürlich nur dann ein, wenn externe Hardware angeschlossen ist.

Für den Hardware-Test bietet der ECB85 zwei Kommandos, über die Datenwörter an Ausgabe-Kanäle direkt ausgegeben und Datenwörter von Eingabe-Kanälen eingelesen werden können. Mit diesen Kommandos läßt sich die einwandfreie Funktion der Hardware schnell prüfen. Da bei Mikrocomputer-Anwendungen die Intelligenz im allgemeinen im Programm liegt und sich die Peripherie-Hardware auf elementare Funktionen beschränkt, ist die Zahl der erforderlichen Prüfungen gering. Ein Hardware-Test ist in jedem Fall lohnend, da man sich dann beim Software-Test umso besser auf das Programm konzentrieren kann.

1.4 Das Magnetbandgerät als Speicher

Die in den ECB85 eingegebenen Anwender-Programme und Daten stehen zunächst im flüchtigen Schreib-/Lese-Speicher und gehen beim Ausschalten der Betriebsspannung verloren. Das gleiche tritt auch beim Überschreiben des Speichers mit anderer Information ein. Wird die alte Information später noch benötigt und will man sich das wiederholte manuelle Eingeben ersparen, muß sie auf ein nicht-flüchtiges Speichermedium übertragen werden.

Als bequeme und billige Methode der Datensicherung ist beim ECB85 das Abspeichern auf Magnetband vorgesehen. Sie empfiehlt sich besonders für Programme, deren Bearbeitung noch nicht abgeschlossen ist oder die relativ selten benötigt werden. Als Aufnahme- und Wiedergabegerät für den ECB85 kann ein Tonbandgerät oder Kassettenrekorder einfacher Art verwendet werden. Der Anschluß erfolgt über ein Kabel und die entsprechende Normbuchse des ECB85. Die Aufnahme und Wiedergabe der Information läuft mit einer Geschwindigkeit von etwa 55 Byte/s. Damit finden z.B. auf einer Kassette von 2 x 30 Minuten Spielzeit knapp 200 KByte Platz.

1.5 Das Arbeiten mit EPROMs

Der ECB85 erlaubt die Verwendung von EPROM-Bausteinen in seinem Speicher. EPROMs sind Festwertspeicher-Bausteine, die mit UV-Licht gelöscht und unter besonderen Bedingungen elektrisch beschrieben werden können. Das Löschen wird am bequemsten in einem speziellen EPROM-Löschgerät vorgenommen, das im wesentlichen aus einer UV-Lampe und einer Schaltuhr besteht. Nach dem Löschen enthält der Baustein in allen Bit-Positionen Einsen. Beim elektrischen Beschreiben im ECB85 werden an gewünschten Bit-Positionen Nullen eingeschrieben. Dazu muß auf dem Gerät der Spannungswandler eingeschaltet werden, der aus der Betriebsspannung die für das Beschreiben von EPROMs erforderliche höhere Spannung erzeugt. Ein Rückändern von Nullen in Einsen ist auf elektrischem Wege nicht möglich.

Im ECB85 können EPROMs der Typen SAB2758 (Kapazität 1 KByte) und SAB2716 (Kapazität 2 KByte) verwendet werden. Zum Lesen und elektrischen Beschreiben werden die Bausteine in eine dafür vorgesehene Fassung auf dem ECB85 gesteckt. Zu diesem Steckplatz kann die Zentraleinheit lesend und schreibend zugreifen. Beim Lesen gibt der Baustein in normaler Weise seine Information ab. Das Beschreiben ist nur möglich, wenn der schon erwähnte Spannungswandler eingeschaltet ist.

2. Beschreibung des ECB85

In den folgenden zwei Kapiteln wird die zum ECB85 gehörende Hard- und Software so weit beschrieben, wie Kenntnisse darüber für den zweckmäßigen Einsatz des Geräts erforderlich sind. Eine ausführliche Erläuterung wird bewußt weggelassen, da das genaue Verständnis erhebliche Vorkenntnisse erfordert, mit denen sich der Anfänger auf diesem Gebiet zunächst noch nicht belasten sollte. Für Fortgeschrittene sind der Stromlaufplan der Hardware und das Listing des Monitor-Programms, aus denen die genaue Arbeitsweise zu ersehen ist, in Teil 9 angefügt.

In einem weiteren Kapitel werden die Kommandos mit ihren Funktionen und ihrer Anwendung kurz erklärt. Darauf folgen Erläuterungen zu einigen in Kommandos benutzten Prinzipien, deren Zweck und Vorteil für manchen zunächst nicht ersichtlich sein mag. Es handelt sich dabei um das automatische Umrechnen von Adressen in Befehlen nach dem Verschieben im Speicher, um das Prinzip der Schattenregister und um das beim ECB85 verwendete Verfahren zum Unterbrechen von Programmen bei vorgegebenen Adressen.

2.1 Hardware des Geräts

Der ECB85 enthält auf einer Leiterplatte mit den Abmessungen 234 mm x 320 mm einen kompletten Mikrocomputer mit Zentraleinheit, Speicher, paralleler Ein-/Ausgabe und Zähler, zusätzlich noch ein Tasten- und Anzeigefeld, einen EPROM-Programmierer und einen Umsetzer zum Anschließen eines Magnetbandgeräts. Für den Aufbau von Anwender-Schaltungen ist ein Lochrasterfeld vorhanden, in dessen Nähe die Anwender-Schnittstelle an einem Anschlußfeld zur Verfügung steht. Die Funktion des ECB85 wird anhand des Blockschaltbilds 2.1-1 erläutert.

Die Zentraleinheit ist mit dem Baustein SAB8085A realisiert. Sie steuert über den Bus alle Funktionseinheiten des ECB85. Die Zentraleinheit wird mit Hilfe der Rücksetz-Automatik (RC-Glied) automatisch beim Einschalten der Betriebsspannung oder manuell beim Drücken der R-Taste an Adresse 0 gestartet. Drei direkte Interrupt-Eingänge des Bausteins SAB8085A stehen über Treiber an Anschlüssen der Anwender-Schnittstelle zur Verfügung. Über die I-Taste kann auch manuell Interrupt ausgelöst werden. Die Einzelbit-Ein- und Ausgänge SID und SOD des SAB8085A können über den SID-SOD-Multiplexer wahlweise an Anschlußpunkte der Anwender-Schnittstelle oder an den Umsetzer für das Magnetbandgerät geschaltet werden. Der durchgeschaltete Weg wird durch einen Schiebeschalter, den sogenannten Programm-Wahlschalter, bestimmt. Dieser Schalter hat seinen Namen von einer weiteren Funktion: Die Zentraleinheit kann durch entsprechende Software abfragen, in welcher Stellung sich der Programm-Wahlschalter befindet. Abhängig davon wird zu einem von zwei möglichen Programmen verzweigt.

Der Speicher besteht aus 1,25 KByte Schreib-/Lese-Speicher (RAM) und zwei Steckplätzen für löschbare Festwertspeicher-Bausteine (EPROM). Der Schreib-/Lese-Speicher dient zum Speichern von Programmen in der Experimentier- und Testphase sowie zum Ablegen von Zwischenergebnissen. Die beiden EPROM-Steckplätze können wahlweise mit Bausteinen SAB2758 (1 KByte) oder SAB2716 (2 KByte) ausgerüstet werden. Der EPROM-Steckplatz Nr. 1 ist bei Lieferung mit einem EPROM SAB2716 bestückt, der ein Monitor-Programm

enthält. Der EPROM-Steckplatz Nr. 2 ist mit einer Spezialfassung ausgerüstet, die ein Einstecken und Ziehen des EPROM-Bausteins ohne mechanische Beanspruchung ermöglicht. Der Platz eignet sich daher besonders gut für selbst-programmierte EPROMs, die öfter gewechselt werden.

Der EPROM-Programmierer ist für die Bausteine SAB2758 und SAB2716 eingerichtet und benutzt die EPROM-Spezialfassung (EPROM-Steckplatz Nr. 2), die vorstehend erwähnt wurde. Die zum Programmieren erforderliche Zusatzspannung wird mit Hilfe eines Spannungswandlers aus der Betriebsspannung erzeugt. Der Spannungswandler kann durch einen Schiebeschalter, den sogenannten Programmierschalter, ein- und ausgeschaltet werden. Bei jedem schreibenden Zugriff der Zentraleinheit zu diesem EPROM-Steckplatz wird ein Datenwort eingeschrieben. Dabei hält eine spezielle Logikschaltung die Zentraleinheit für die Zeitdauer an, die zum Einschreiben eines Bytes erforderlich ist. Die Zeit wird von einem Ausgangssignal des Zähler-Teils abgeleitet. Bei jedem Schreibzyklus leuchtet eine Leuchtdiode, die sogenannte Programmierlampe, kurz auf.

Die parallele Ein-/Ausgabe bietet 22 Anschlüsse, die zur Verwendung in Anwender-Schaltungen an Anschlußpunkte der Anwender-Schnittstelle geführt sind. Die Anschlüsse lassen sich in Gruppen zu 6 bzw. 8 beliebig auf Ein- und Ausgabe programmieren, wobei wahlweise Betrieb ohne und mit Rückmeldung (Handshake-Betrieb) möglich ist.

Der Zähler ist fest mit dem internen Takt verbunden und kann softwaremäßig im Teilverhältnis programmiert werden. Das Ausgangssignal des Zählers ist zum EPROM-Programmierer geführt und dient dort zum Festlegen der Schreibzeit. Zusätzlich steht das Signal auch an einem Anschluß der Anwender-Schnittstelle zur Verfügung. Es kann dazu benutzt werden, Anwender-Schaltungen zu steuern oder Programm-Unterbrechungen (Interrupt) auszulösen.

Schreib-/Lese-Speicher, Ein-/Ausgabe und Zähler lassen sich durch Einstecken eines Mehrfunktions-Bausteins SAB8155 in eine vorhandene Fassung

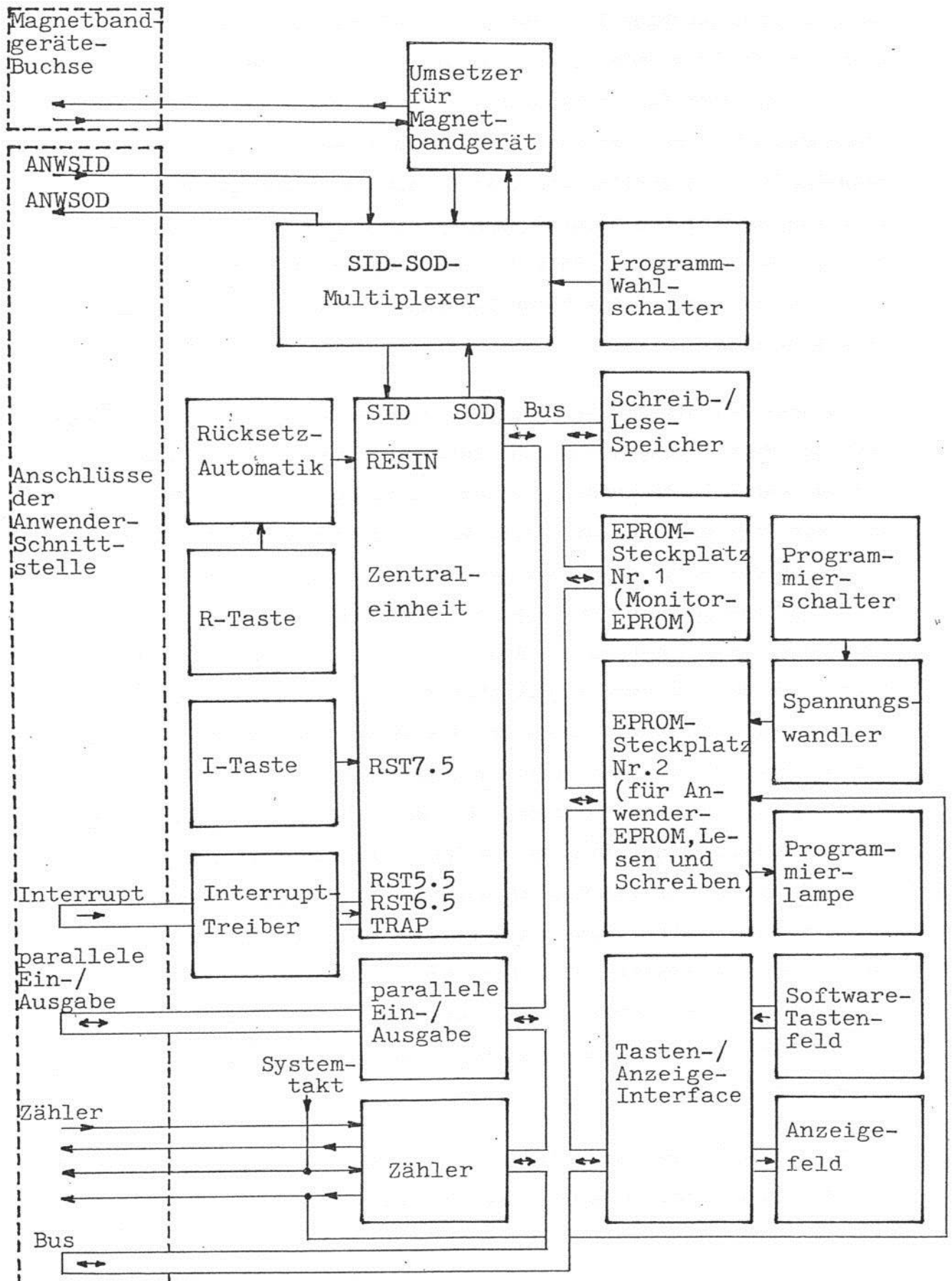
erweitern. Damit stehen zusätzlich 0,25 KByte Schreib-/Lese-Speicher, 22 Ein-/Ausgabe-Anschlüsse und 1 Zähler zur Verfügung. Die zusätzlichen externen Signale sind ausschließlich an Anschlüsse der Anwender-Schnittstelle geführt.

Das Tasten- und das Anzeigefeld ermöglichen die Kommunikation zwischen Anwender und Mikrocomputer ohne zusätzliche externe Geräte. Dies setzt jedoch das Monitor-Programm oder ein entsprechendes selbst entwickeltes Anwender-Programm voraus. Das Tastenfeld enthält neben der bereits erwähnten R- und I-Taste zum hardwaremäßigen Beeinflussen der Zentraleinheit noch 20 Tasten mit Software-Funktion. Diese Software-Tasten werden im Rahmen des Monitor-Programms abgefragt und dienen zu dessen Steuerung. Das Anzeigefeld besteht aus 8 Sieben-Segment-Anzeigen, die ebenfalls durch das Monitor-Programm gesteuert werden. Zwischen Tasten-/Anzeigefeld und Mikrocomputer-Bus liegt das entsprechende Interface.

Der Umsetzer für das Magnetbandgerät erlaubt die Verwendung eines normalen Tonbandgeräts oder Kassettenrekorders als externer Speicher mit unbegrenzter Kapazität. Das Magnetbandgerät wird über ein Kabel an die dafür vorgesehene Normbuchse auf dem ECB85 angeschlossen.

Das Lochrasterfeld bietet die Möglichkeit, kleine Anwender-Schaltungen direkt auf der Leiterplatte des ECB85 aufzubauen. Zur Ansteuerung stehen die Signale von Interrupt-System, Ein-/Ausgabe, Zähler und Bus an Anschlüssen der Anwender-Schnittstelle zur Verfügung.

Blockschaltbild des ECB85 (Beispiel: MIKROCOMPUTER)



2.2 Software des Geräts

Seine Leistungsfähigkeit erhält der ECB85 mit durch ein Monitor-Programm von 2 KByte Umfang, das in einem EPROM mitgeliefert wird. Dieses Programm benutzt für die Kommunikation mit dem Anwender das Tastenfeld und Anzeigefeld auf der Baugruppe, so daß keine zusätzlichen Ein-/Ausgabe-Geräte erforderlich sind. Natürlich kann man mit einem so kleinen Betriebsprogramm nicht auf Assembler-Ebene arbeiten, sondern muß die Befehle im Hexadezimal-Code eingeben. Davon abgesehen bietet das Monitor-Programm jedoch eine Reihe leistungsfähiger Kommandos, die die Arbeit sehr erleichtern.

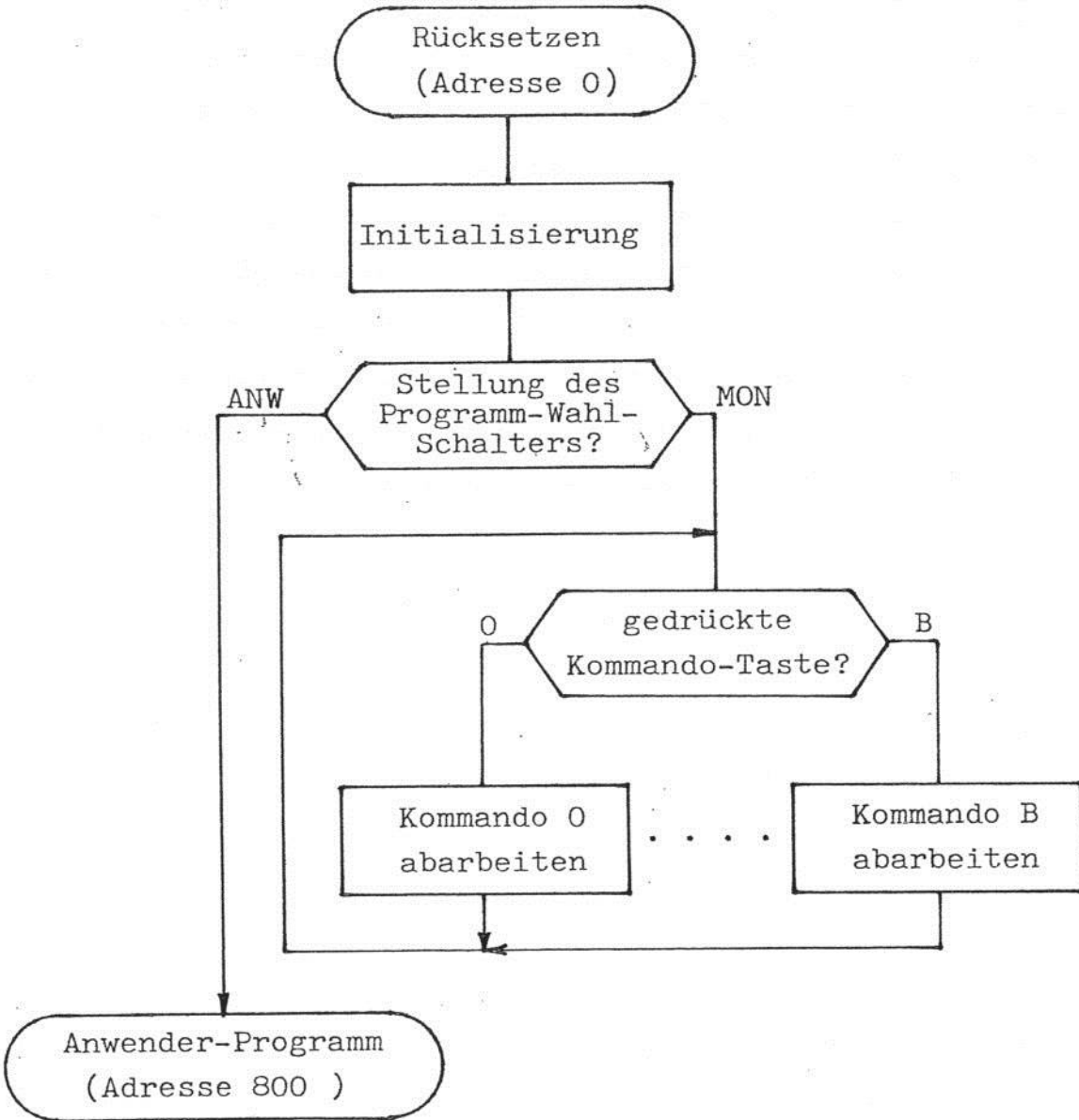
Ein grober Ablaufplan des Monitor-Programms ist in Bild 2.2-1 angegeben. Nach dem Rücksetzen, das durch Einschalten der Betriebsspannung oder Drücken der R-Taste erfolgt, werden zunächst im Initialisierungs-Teil alle vom Monitor-Programm benutzten programmierbaren Bausteine im ECB85 auf die vorgesehene Betriebsart eingestellt. Anschließend wird der Programm-Wahlschalter abgefragt. Befindet er sich in der Stellung ANW (dabei sind zugleich die Anschlüsse SID und SOD des SAB8085A auf Anschlüsse der Anwender-Schnittstelle geschaltet), wird sofort zu Adresse 800 gesprungen. Dies ist die Anfangsadresse des EPROM-Steckplatzes Nr. 2 in den man einen anwenderspezifisch programmierten EPROM-Baustein stecken kann. Befindet sich der Schalter dagegen in der Stellung MON (dabei sind zugleich die Anschlüsse SID und SOD des SAB8085A an den Umsetzer für das Magnetbandgerät geschaltet), wird das Monitor-Programm fortgesetzt. Zunächst wird dann gewartet, daß mit Drücken einer Taste ein Kommando aufgerufen wird. Anschließend wird das Kommando ausgeführt. Einzelheiten zu den einzelnen Kommandos werden an dieser Stelle nicht gebracht, da in späteren Kapiteln noch genau darauf eingegangen wird.

Das Monitor-Programm läßt die Interrupts RST1.... RST6, RST5.5, RST6.5 und TRAP zu und bewirkt dafür jeweils Weitersprünge zu um 800 höheren

Adressen. Diese Adressen liegen im Bereich des EPROM-Steckplatzes Nr. 2, der für Anwender-Programme vorgesehen ist. Der Interrupt RST0 kann nicht allgemein verwendet werden, da sein Sprungziel gleich ist mit der Startadresse des Monitor-Programms, und das Sprungziel des Interrupt RST7 wird für eine spezielle Aufgabe des Monitor-Programms verwendet. Der Interrupt RST7.5 ist ebenfalls zugelassen, wird aber anders als die übrigen Interrupts behandelt. Ausgelöst wird ein Interrupt RST7.5 durch das Drücken der I-Taste. Beim Auftreten dieses Interrupts rettet die Zentraleinheit den Inhalt aller Register in einem speziellen Bereich des Schreib-/Lese-Speichers und steht dann für Eingabe von Monitor-Kommandos zur Verfügung. Auf diese Weise kann man Anwender-Programme zu einem gewünschten Zeitpunkt unterbrechen und später an der Unterbrechungsstelle wieder fortsetzen. Auf Einzelheiten wird noch später eingegangen.

Bild 2.2-1

Grober Ablaufplan des Monitor-Programms



2.3 Überblick über die Kommandos

Das Monitor-Programm des ECB85 kann auf Anweisung des Anwenders 11 Funktionen ausführen. Diese Funktionen werden durch Kommandos, die über das Tastenfeld eingegeben werden, eingeleitet. Bei den Kommandos werden vier Gruppen unterschieden: Kommandos mit Bezug auf den Speicher, Kommandos zum Testen der Hardware, Kommandos zum Testen der Software und Kommandos mit Bezug auf das Magnetbandgerät.

Die Kommandos mit Bezug auf den Speicher erlauben das Auslesen und Beschreiben des internen Speichers. Dabei wird die Adresse auf Wunsch jeweils automatisch inkrementiert (um 1 erhöht) oder dekrementiert (um 1 verringert). Außerdem kann ein Speicherbereich mit einer Konstanten gefüllt und der Inhalt eines Speicherbereichs in einen anderen Bereich übertragen werden. Ein weiteres Kommando bietet eine erhebliche Erleichterung im Zusammenhang mit dem Übertragen von Speicherinhalten. Nach einer Verschiebung stimmen die Adressen von Sprungbefehlen, die Bezug auf den verschobenen Speicherbereich haben, nicht mehr. Die manuelle Umrechnung der Sprungadressen ist aufwendig und fehleranfällig. Das Monitor-Programm des ECB85 umfaßt jedoch ein Kommando, durch das die Befehle in einem beliebigen Speicherbereich darauf geprüft werden, ob sie Sprünge zu dem verschobenen Speicherbereich enthalten. Bei allen Befehlen, für die das zutrifft, wird die Sprungadresse automatisch umgerechnet. - Alle Kommandos dieser Gruppe können auch zum Programmieren eines EPROMs in EPROM-Steckplatz Nr. 2 benutzt werden, wenn gleichzeitig der zugehörige Spannungswandler eingeschaltet ist.

Die Kommandos zum Testen der Hardware erlauben das Eingeben von Daten über Eingabe-Kanäle und das Ausgeben über Ausgabe-Kanäle. Bei wiederholter Ausführung des Kommandos bleibt die Kanal-Adresse konstant, so daß man bequem die Auswirkung unterschiedlicher Daten erkennen kann. Mit diesen Kommandos ist ein schneller Test der peripheren Hardware möglich.

Mit den Kommandos zum Testen der Software besteht zunächst die Möglichkeit, die Register der Zentraleinheit auszulesen und zu beschreiben. Dieses Kommando wird benötigt, um die Zentraleinheit vor einem Programmtest

in einen definierten Zustand versetzen und nach einem Programmtest die Ergebnisse kontrollieren zu können. Weitere Kommandos dienen zum Prüfen von Anwender-Programmen. Sie ermöglichen das Starten von Programmen ohne und mit Anhalten bei vorgebbaren Unterbrechungspunkten. Vor dem Starten und nach dem Unterbrechen lassen sich die Register kontrollieren und beliebig laden. Auf diese Weise kann man die Funktion des Programms prüfen. Als Besonderheit des Monitor-Programms ist zu erwähnen, daß Unterbrechungspunkte auch im EPROM gesetzt werden dürfen, da jeder Befehl einzeln unter Monitor-Kontrolle abläuft. Weiterhin lassen sich die Programme im Einzelschritt abarbeiten, wobei mit einem Tastendruck jeweils der nächste Befehl freigegeben und jedes Register zwischendurch angesehen und beschrieben werden kann.

Die Kommandos mit Bezug auf das Magnetbandgerät dienen zum Abspeichern von Programmen auf Tonband mit einem normalen Tonbandgerät oder Kassettenrekorder und zum Wieder-Einlesen von Tonband. Da die Programme im Schreib-/Lese-Speicher des ECB85 bei Ausschalten der Betriebsspannung verloren gehen, wird man sie zweckmäßigerweise vorher auf Band abspeichern und später bei Bedarf wieder einlesen. Auf diese Weise wird die wiederholte Eingabe über das Tastenfeld eingespart.

Zur schnellen Orientierung über die verfügbaren Kommandos wird in Kapitel 9.4 eine Zusammenstellung gebracht.

2.4 Das automatische Umrechnen von Adressen

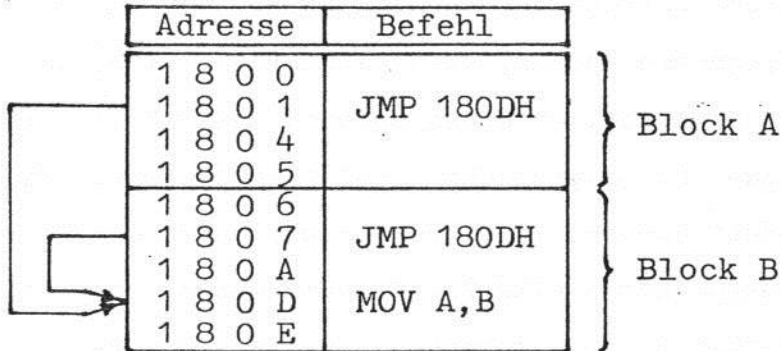
Nach Übertragen von Information, die zu einem Anwender-Programm gehört, aus einem Speicherbereich in einen anderen, kann es zu Störungen im Ablauf dieses Programms kommen, wenn nicht zusätzliche Änderungen vorgenommen werden. Dies soll an einem Beispiel erläutert werden. Bild 2.4-1a zeigt zwei Programmblöcke A und B im Speicher, die beide Sprungbefehle zu einer Adresse in Block B haben. Wird nun Block B in der Adressenlage entsprechend Bild 2.4-1b verschoben, so zeigen die Sprungbefehle weiterhin zu den Adressen in der alten Lage von Block B. Das ist jedoch unerwünscht, denn der Adreßbereich, in dem sich früher Block B befand, soll für andere Zwecke benutzt werden, und durch die Verschiebung darf die logische Folge der Befehle nicht verändert werden. Um dies zu erreichen, müssen in allen Programmblöcken, die Sprungziele in dem verschobenen Block haben, die Sprungadressen entsprechend der Verschiebung umgerechnet werden. Dies ist im Beispiel sowohl für Block A als auch für Block B erforderlich. Danach ergeben sich die Sprungziele richtig wie in Bild 2.4-1c dargestellt.

Eine manuelle Ausführung dieser Umrechnung ist zeitraubend und fehleranfällig. Der ECB85 bietet nun ein Kommando, das diese Aufgabe erledigt. Einzugeben sind dazu die Anfangs- und Endadresse des verschobenen Bereichs in der Ursprungslage und die Anfangsadresse in der verschobenen Lage. Weiterhin werden Anfangs- und Endadresse des Bereichs, in dem die Sprungadressen umgerechnet werden sollen, benötigt. Das Kommando ist auf alle Speicherbereiche anzuwenden, die Sprungbefehle zu dem verschobenen Speicherbereich enthalten.

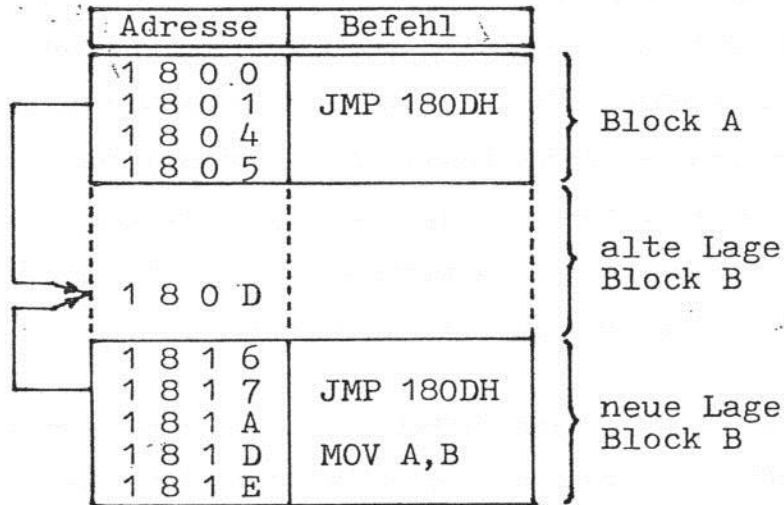
Bild 2.4-1

Verschieben von Programmblöcken

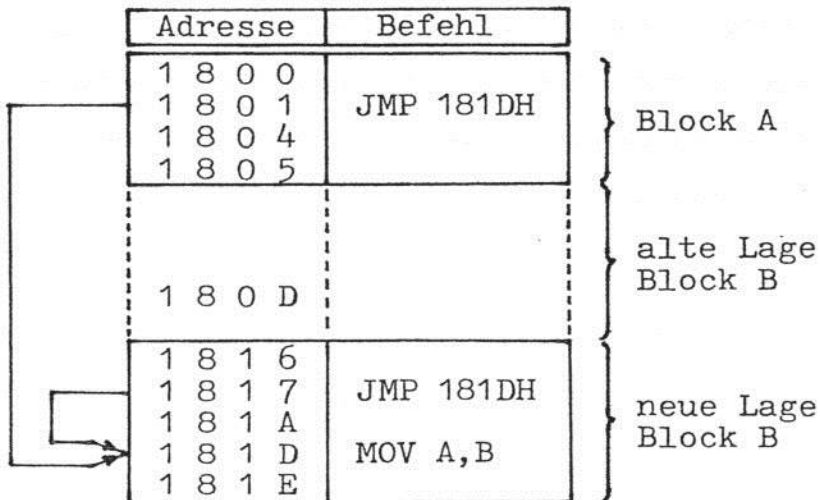
a) ursprüngliche Lage



b) verschobene Lage mit falschen Sprungzielen



c) verschobene Lage mit richtigen Sprungzielen



2.5 Das Prinzip der Schattenregister

Um ein zu testendes Programm mit definierten Anfangsbedingungen starten und nach Ausführung die Auswirkung kontrollieren zu können, muß der Anwender Zugriff zu den Registern der Zentraleinheit haben. Bei den üblichen Mikroprozessor-Bausteinen, so auch beim SAB8085A des ECB85, ist jedoch ein Zugriff zu den Registern durch reine Hardware-Maßnahmen nicht möglich. Vielmehr bedeutet das Eingeben oder Auslesen eines Registerinhalts immer schon ein kleines Mikrocomputer-Programm. Außerdem kann der Inhalt der Register zerstört werden, wenn die Zentraleinheit im Rahmen des Monitor-Programms anschließend Operationen ausführt.

Um diese Schwierigkeiten zu umgehen, wurden beim ECB85 sogenannte "Schattenregister" eingeführt. Dies sind festgelegte Speicherzellen im Schreib-/Lese-Speicher, die den einzelnen Registern der Zentraleinheit zugeordnet sind. Mit den Kommandos werden nicht direkt die Register der Zentraleinheit angesprochen, sondern diese Schattenregister. Der Inhalt der Schattenregister erhält seine eigentliche Funktion erst in Zusammenhang mit den Kommandos, die ein Abarbeiten von Anwender-Programmen auslösen. Vor dem Starten eines Anwender-Programms mit dem entsprechenden Kommando lädt das Monitor-Programm automatisch die Register der Zentraleinheit aus den Schattenregistern, und nach dem Unterbrechen eines Anwender-Programms durch einen gesetzten Unterbrechungspunkt oder nach Ausführung eines Einzelschritts speichert das Monitor-Programm automatisch die Inhalte der Register der Zentraleinheit in die Schattenregister ab.

Diese Besonderheiten sollte sich der Anwender bei Arbeiten mit den Registern stets vor Augen halten. Die über das entsprechende Kommando "in ein Register eingeschriebenen" Daten stehen also noch nicht im Register, sondern werden erst beim nächsten Starten eines Anwender-Programms als Registerinhalt wirksam. Und die mit dem zugehörigen Kommando "aus einem Register ausgelesenen" Daten sind nicht die tatsächlich in dem Register stehenden Daten, sondern bedeuten den Inhalt, den das Register bei der letzten Programmunterbrechung hatte bzw. der mit dem erstgenannten Kommando "in das Register eingeschrieben" wurde.

2.6 Das Unterbrechen von Programmen

Beim Testen von Programmen möchte man die Ausführung oft an einer definierten Stelle unterbrechen, um die Registerinhalte zu prüfen und neu zu setzen. Deswegen gibt es beim ECB85 ein Kommando zum Starten von Anwender-Programmen, bei dem bis zu zwei Unterbrechungsadressen vorgegeben werden können. Wird eine dieser Adressen erreicht, so wird die Programmausführung abgebrochen und in das Monitor-Programm zurückgesprungen. Zwei Unterbrechungsadressen sind erforderlich, wenn hinter einer Programmverzweigung, bei der man den tatsächlichen Weg des Programms nicht kennt, eine Unterbrechung sichergestellt werden soll.

In Zusammenhang mit diesen Adressen sind noch einige Anmerkungen zu machen. Üblicherweise werden bei Monitor-Programmen Programmunterbrechungen zu Testzwecken so realisiert, daß das Monitor-Programm an die entsprechenden Adressen im Anwender-Programm vorübergehend Befehle einschreibt, die einen Rücksprung ins Monitor-Programm bewirken. Damit ist eine Programmunterbrechung auf einfache Weise möglich. Diese Methode hat jedoch den Nachteil, daß man keine Unterbrechungspunkte in Programmen setzen kann, die sich in EPROM-Speichern befinden, da in diese im normalen Betrieb keine Daten eingeschrieben werden können. Beim ECB85 werden die Unterbrechungspunkte anders realisiert, um auch die Möglichkeiten für Programmunterbrechungen im EPROM zu bieten. Die Befehle werden unter Monitor-Kontrolle einer nach dem anderen in einen reservierten Bereich des Schreib-/Lese-Speichers kopiert und dort ausgeführt. Nach Ausführung jedes Befehls wird geprüft, ob die Unterbrechungsadresse erreicht ist, und in diesem Fall wird die Ausführung des Anwender-Programms beendet. Das Programm läuft nach diesem Verfahren natürlich langsamer ab, da das Monitor-Programm für jeden Anwender-Befehl eine ganze Reihe von Operationen ausführen muß. Bei der Prüfung von Programmen spielt dies jedoch zumeist keine wesentliche Rolle. Um daneben aber auch die Möglichkeit des Echtzeit-Betriebs von Anwender-Programmen zu bieten, gibt es auch ein Kommando ohne Unterbrechungspunkte.

Ein weiteres Kommando erlaubt die Einzelschritt-Abarbeitung von Anwender-Programmen. Dabei wird mit einem Tastendruck die Ausführung jeweils eines Befehls freigegeben. Vorher und nachher kann man die Inhalte aller Register ansehen und ändern, so daß damit ein sehr detaillierter Test des Anwender-Programms möglich ist. Die Ausführung der Befehle erfolgt in gleicher Weise wie beim Programmstart mit Unterbrechungspunkten.

3. Allgemeines zum Arbeiten mit dem ECB85

In den folgenden Kapiteln werden Fragen der Bedienung des ECB85 behandelt, die für alle Kommandos in gleicher Weise von Bedeutung sind. Dazu gehört im Bereich der Hardware die Lage der für den Anwender wichtigen Bedien- und Anzeigeelemente, die Forderungen an die Stromversorgung und der Anschluß eines Magnetbandgeräts. Für die Software wird ein Plan der Belegung von Speicher- und Ein-/Ausgabe-Adreßraum angegeben und ein Programm-Beispiel gebracht, das später zur Demonstration der Kommandos dienen soll. Schließlich werden generelle Erläuterungen zu Inbetriebnahme und Kommunikation mit dem Gerät gegeben und das Einleiten eines Kommandos allgemein behandelt.

3.1 Lageplan der Hardware

Alle Bauteile des ECB85 befinden sich auf einer geätzten Leiterplatte mit den Abmessungen 234 mm x 320 mm. Die obere Hälfte belegt die Mikrocomputer-Elektronik, das rechte untere Viertel enthält im wesentlichen das Tasten- und Anzeigefeld, und das linke untere Viertel wird von einem Lochrasterfeld eingenommen. Bild 3.1-1 zeigt die Lage aller Hardware-Elemente, die für den Anwender von besonderer Bedeutung sind. Sie werden im folgenden kurz besprochen.

Zum Anschließen der Betriebsspannung 5V sind zwei Buchsen vorhanden. Die Buchse für den Minuspol (2) ist auf der Leiterplatte mit "0V" bezeichnet, die Buchse für den Pluspol (1) ist auf der Leiterplatte durch "+5V" kenntlich gemacht.

Im Mittelpunkt des Mikrocomputers steht der Prozessor-Baustein SAB8085A (22), der steckbar eingebaut ist. Der EPROM-Steckplatz Nr. 1 (21) ist mit dem Monitor-EPROM bestückt. Der Baustein SAB8155 Nr. 1 (20) befindet sich auf einer Fassung, für den Erweiterungs-Baustein SAB8155 Nr. 2 ist eine weitere Fassung (19) vorgesehen. Beim Stecken der genannten Bausteine ist auf die richtige Lage zu achten. Bild 3.1-1 zeigt für die einzelnen Bausteine die richtige Lage der Markierungskerbe. Schließlich ist noch der Programm-Wahlschalter (6) zu erwähnen.

Dem EPROM-Programmierer ist der EPROM-Steckplatz Nr. 2 (5), der Programmierschalter (7) und die Programmierlampe (9) zugeordnet. Der EPROM-Steckplatz ist mit einer Spezialfassung ausgerüstet, die ein Ziehen und Stecken des Bausteins ohne mechanische Beanspruchung erlaubt. Beim Stecken von EPROMs ist darauf zu achten, daß die Markierungskerbe die im Bild angegebene Lage hat.

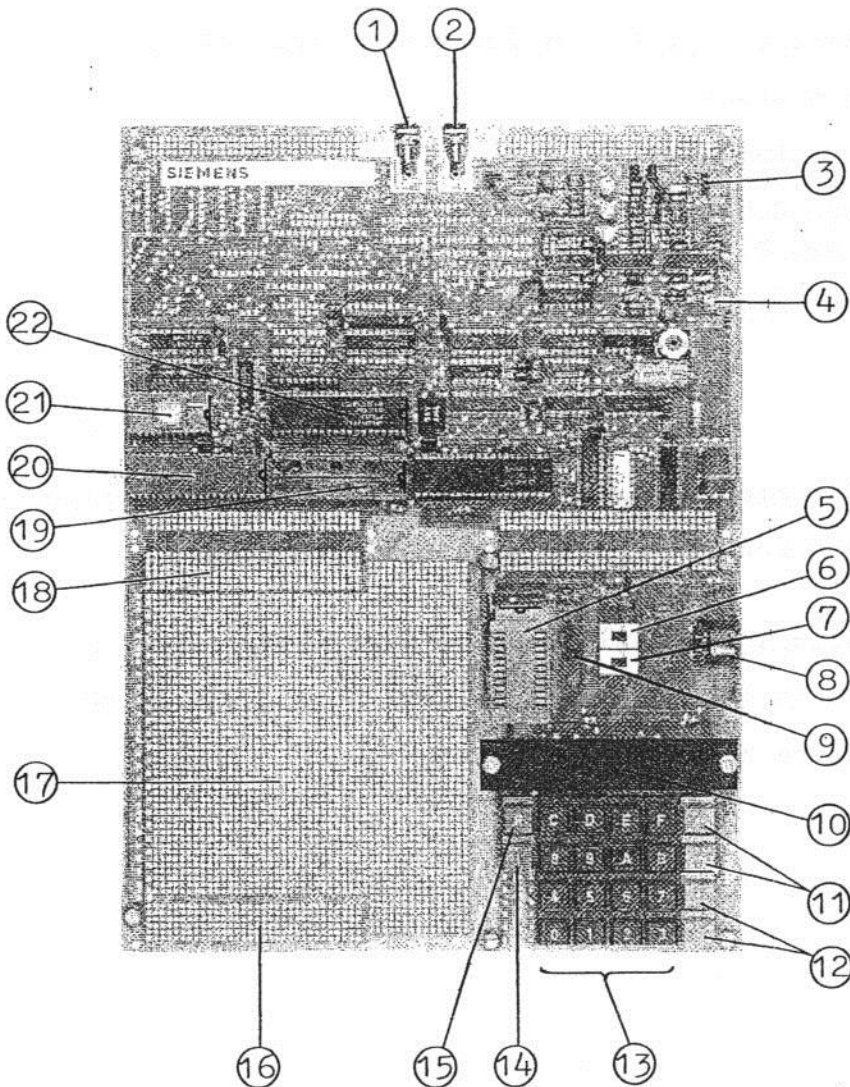
Das Tasten-/Anzeigefeld besteht aus der R-Taste (15), der I-Taste (14), den 16 Hexadezimal-Tasten (13), den beiden Schreibmarken-Steuertasten (11), den beiden Abschlußtasten (12) und der 8-stelligen Anzeige (10).

Für den Anschluß eines Magnetbandgeräts ist die Anschlußbuchse (8) vorgesehen. Die Eingangsempfindlichkeit kann am Potentiometer (4) eingestellt werden. Dabei läßt sich die Spannung am Meßpunkt MP2 (3) kontrollieren.

Zum Aufbau von Anwender-Schaltungen ist das Lochrasterfeld (17) vorgesehen. Die Mikrocomputer-Signale stehen dabei an den Anschlüssen der Anwender-Schnittstelle (18) zur Verfügung. Zum Einlöten eines Peripherie-Steckers ist ein Lötfeld (16) vorgesehen.

Bild 3.1-1

Lageplan der für den Anwender wichtigen Bauelemente



Zuführung für Betriebsspannung

- ② Buchse für Minuspol
- ① Buchse für Pluspol

Umsetzer für Magnetbandgerät

- ⑧ Buchse für Magnetbandgerät
- ④ Potentiometer für Eingangsempfindlichkeit
- ③ Meßpunkt MP2 für Kontrolle des Eingangspiegels

EPROM-Programmierer

- ⑤ EPROM-Steckplatz Nr.2
- ⑦ Programmierschalter
- ⑨ Programmierlampe

Mikrocomputer-Elektronik

- ②② SAB8085A EPROM-Steckplatz Nr.1 mit Monitor-EPROM
- ②① SAB8155 Nr.1 Steckplatz für SAB8155 Nr.2
- ⑥ Programm Wahlschalter

Anwenderfeld

- ①⑦ Lochrasterfeld
- ①⑧ Anschlüsse der Anwenderschnittstelle
- ①⑥ Lötfield für Peripheriestecker

Tasten-/Anzeigefeld

- ①⑤ R-Taste
- ①④ I-Taste
- ①③ Hexadezimal-Tasten
- ①① Schreibmarken-Steuertasten
- ①② Abschluß-tasten
- ①⑩ Anzeigefeld

3.2 Anschließen der Betriebsspannung

Der ECB85 benötigt lediglich eine von extern zuzuführende Betriebsspannung von 5V mit einer Toleranz von $\pm 5\%$. Zur Zuführung sind zwei Buchsen vorhanden. Die Stromaufnahme beträgt:

ECB85 in Grundausstattung	700 mA
zusätzlich für einen Baustein SAB2716 (EPROM-Steckplatz Nr. 2)	105 mA
zusätzlich für einen Baustein SAB8155 (SAB8155 Nr. 2)	180 mA
Summe	<u>985 mA</u>

Die Stromaufnahme erhöht sich entsprechend, wenn zusätzliche periphere Schaltungen z.B. auf dem Lochrasterfeld angeschlossen sind.

Vor Anlegen der Betriebsspannung sollte genau kontrolliert werden, ob die Höhe und Polarität richtig ist, da durch falsche Polung und zu hohe Spannungen (über 7V) Bausteine zerstört werden können.

3.3 Anschließen des Magnetbandgeräts

Zum Abspeichern von Programmen auf Magnetband und zum Einlesen von Programmen von Magnetband kann ein Magnetbandgerät über die auf dem ECB85 vorhandene Normbuchse angeschlossen werden. Das Verbindungskabel zwischen dem Magnetbandgerät und dem ECB85 muß jeweils die Anschlußstifte gleicher Nummer untereinander verbinden (nicht gekreuzt).

Als Magnetbandgerät ist ein Tonbandgerät oder ein Kassettenrekorder geeignet. Es werden nur geringe Anforderungen hinsichtlich Tonqualität und Übertragungsbandbreite gestellt. Dagegen ist ein Bandzählwerk vorteilhaft, um Anfangs- und Endpunkt von Programmen auf dem Band leicht wiederfinden zu können.

Das Schreiben auf Band erfolgt mit Hilfe der zwei festen Frequenzen 2,4 kHz und 3,6 kHz. Die Information wird wie bei serieller asynchroner Übertragung ausgegeben, wobei jedem Byte ein Anlaufschritt mit der Dauer einer Bit-Zeit vorangeht und ein Sperrschritt mit der Dauer zweier Bit-Zeiten folgt. Die Übertragungsrate beträgt 600 Baud, entsprechend etwa 55 Byte pro Sekunde. Am Signalausgang der Normbuchse steht das Signal mit einem Pegel von 400 mV_{SS} zur Verfügung. Wenn das Magnetbandgerät mit einer Aussteuerungs-Automatik ausgestattet ist, sind bei der Aufnahme auf Band keine Einstellungen erforderlich. Besitzt es jedoch eine manuelle Einstellung, so ist diese auf maximal (ohne Übersteuerung) mögliche Aussteuerung einzustellen.

Beim Einlesen der Information vom Magnetbandgerät werden die Frequenz-Signale wieder in logische Pegel verwandelt und zu Bytes gruppiert. Der ECB85 ist bei Auslieferung so eingestellt, daß am Signaleingang ein Pegel von 300 - 400 mV_{SS} erforderlich ist. Sollte das Magnetbandgerät eine andere Ausgangsspannung liefern, muß die Eingangsempfindlichkeit an dem in Bild 3.1-1 bezeichneten Potentiometer so eingestellt werden, daß am Meßpunkt MP2 ein Sinussignal von 1,5 ... 2,5 V_{SS} vorhanden ist.

3.4 Belegung des Adreßraums

Beim ECB85 gibt es zwei Adreßräume: Den Speicher-Adreßraum, der mit speicherbezogenen Befehlen angesprochen wird, und den Ein-/Ausgabe-Adreßraum, zu dem man mit ein-/ausgabebezogenen Befehlen (IN, OUT) Zugriff hat.

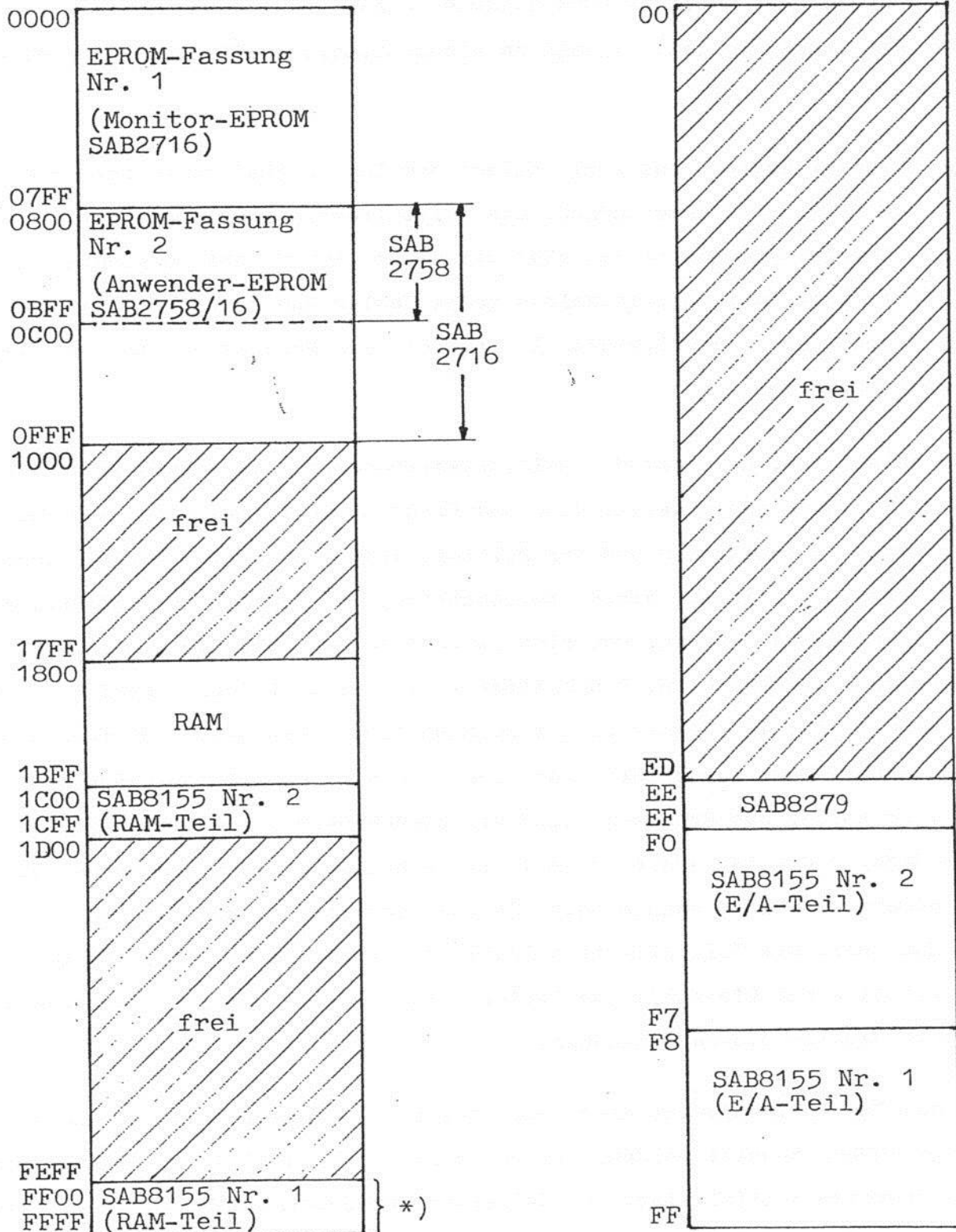
Der Speicher-Adreßraum umfaßt 64 K ($64 \times 1024 = 65536$) Adressen. Davon ist ein Teil mit Speicher-Bausteinen ausgebaut. Bild 3.4-1a zeigt den Adreßplan. Der Bereich 0000 bis 07FF ist dem EPROM-Steckplatz Nr. 1 zugeordnet, in dem normalerweise der Monitor-EPROM steckt. Daran schließt sich der Bereich des EPROM-Steckplatzes Nr. 2 an, der je nach verwendetem EPROM-Typ von 0800 bis 0BFF (SAB2758) oder von 0800 bis 0FFF (SAB2716) geht. Der Bereich 1800 bis 1BFF ist dem Schreib-/Lese-Speicher auf der Baugruppe zugeordnet. Ebenfalls Schreib-/Lese-Speicher liegt zwischen FF00 bis FFFF, dieser ist im Mehrfunktions-Baustein SAB8155 enthalten. Der in Bild 3.4-1a gekennzeichnete obere Teil dieses Speicherbereichs ist für das Monitor-Programm reserviert und darf in Anwender-Programmen nicht benutzt werden. Wenn ein zweiter Baustein SAB8155 in die dafür auf dem ECB85 vorhandene freie Fassung eingesteckt ist, steht außerdem noch RAM zwischen 1C00 und 1CFF zur Verfügung. Der nicht belegte Teil des Adreßraums kann für Erweiterungen über die an den Anschlüssen der Anwender-Schnittstelle verfügbaren Busleitungen benutzt werden.

Der Ein-/Ausgabe-Adreßraum umfaßt 256 Adressen. Davon wird ein Teil für die Ein-/Ausgabe-Bausteine auf der Baugruppe benutzt. Bild 3.4-1b zeigt die Belegung dieses Adreßraums. Der Bereich zwischen EE und EF ist dem Tasten-Anzeige-Steuer-Baustein SAB8279 zugeordnet. Der Adreßbereich F8 bis FF gehört zum Mehrfunktions-Baustein SAB8155. Wenn ein zweiter Baustein SAB8155 in die dafür vorhandene freie Fassung eingesteckt ist, ist auch noch der Bereich zwischen F0 und F7 belegt. Der nicht belegte Teil des Adreßraums kann für Erweiterungen über die an den Anschlüssen der Anwender-Schnittstelle verfügbaren Busleitungen benutzt werden.

Bild 3.4-1
Adreßraum

a) Speicher-Adreßraum

b) Ein-/Ausgabe-Adreßraum



*) Der RAM-Bereich zwischen Adresse FFB3 und FFFF ist für das Monitor-Programm reserviert und darf in Anwender-Programmen nicht benutzt werden.

3.5 Programm-Beispiel

Damit der Anwender gleich von Anfang an ein geeignetes Testobjekt zur Verfügung hat, an dem die Kommandos des ECB85 demonstriert werden können, enthält der Monitor-EPROM zwischen Adresse 07E3 und 07FF ein einfaches Programm-Beispiel. Es soll an dieser Stelle ausführlich besprochen werden.

Das Beispiel besteht aus zwei Teilen: dem Multiplikations-Programm ANWMUL und dem Aufruf-Programm ANWRUF. Das Multiplikations-Programm hat die Form eines Unterprogramms, es ist also mit einem CALL-Befehl aufzurufen. Es multipliziert zwei vorzeichenlose ganze Zahlen von je 8 Bit Länge in den Registern D und E und liefert als Ergebnis ein Produkt von 16 Bit Länge im Registerpaar H&L.

Der Ablaufplan, nach dem das Multiplikations-Programm intern arbeitet, ist in Bild 3.5-1a dargestellt. Wenn das Programm gestartet wird, muß der Multiplikand in Register D und der Multiplikator in Register E bereitstehen. Das Programm definiert einen Stellenzähler, mit dem die Multiplikationsschritte gezählt werden, und eine Zwischensumme, über die schrittweise das Ergebnis aufgebaut wird. Die Multiplikation erfolgt über 8 bedingte Additionen des Multiplikators zu der Zwischensumme. Die n-te Addition ($n = 1 \dots 8$) wird jeweils nur ausgeführt, wenn das n-te Bit des Multiplikanden den Wert 1 hat. Am Anfang des Programms wird die Zwischensumme auf 0 gesetzt. Sie wird vor jeder bedingten Addition um 1 Bit nach links verschoben, wobei als niederwertigstes Bit 0 nachgezogen wird. Am Ende stellt die Zwischensumme das Produkt dar. Die Multiplikation läuft im Prinzip genauso ab, wie man es auch schriftlich mit Bleistift und Papier machen würde. Das ausgeführte Programm ist in Tabelle 3.5-2a angegeben.

Um das Multiplikationsprogramm auch testen zu können, kann das Aufruf-Programm ANWRUF benutzt werden. Der Ablaufplan ist in Bild 3.5-1b angegeben. Das Programm initialisiert den Kellerspeicherzeiger, gibt Interrupt frei, ruft das Multiplikations-Programm ANWMUL auf und schließt mit einem HLT-Befehl ab. Die Interrupt-Freigabe soll dabei ermöglichen, mit der I-Taste

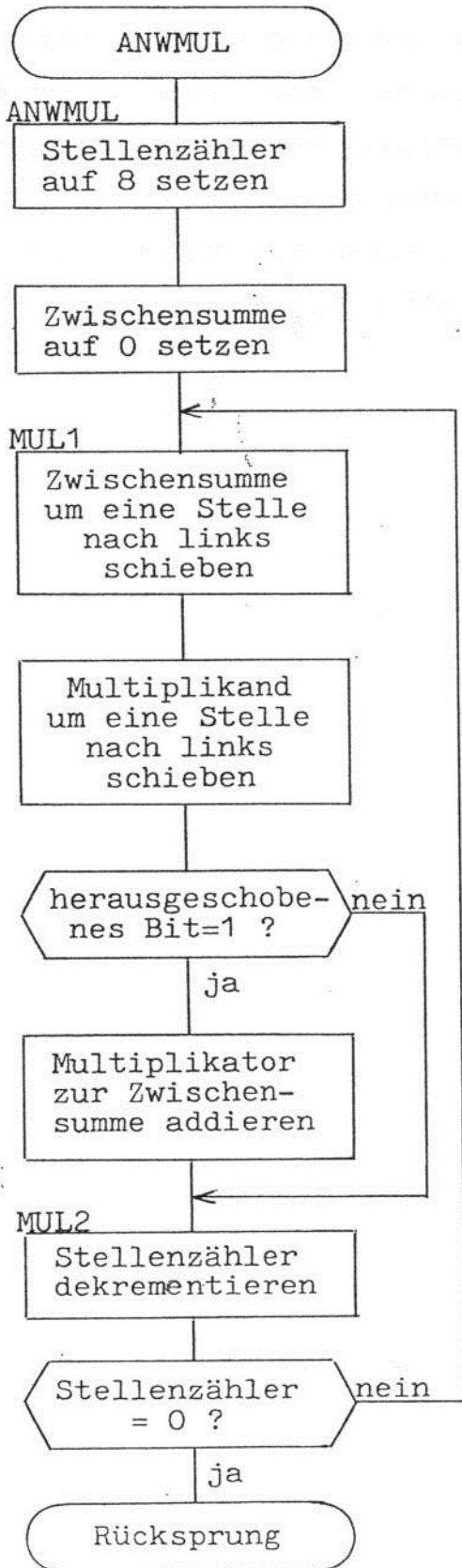
den Halt-Zustand abubrechen. Das ausgeführte Programm ist in Tabelle 3.5-2b angegeben.

Um die Anwendung des Programms zu zeigen, wird im folgenden ein Zahlenbeispiel gebracht. Bild 3.5-3a zeigt die auftretenden Größen, die benutzten Register und Beispiele für die Zahlen. Die Zahlen sind zum besseren Verständnis zunächst dezimal angegeben und außerdem in die beim Bearbeiten mit dem ECB85 erforderliche hexadezimale Darstellung gebracht. Bild 3.5-3b zeigt die einzelnen Schritte, die bei Anwendung des Programms auszuführen sind. Die tatsächliche Ausführung wird später noch erläutert.

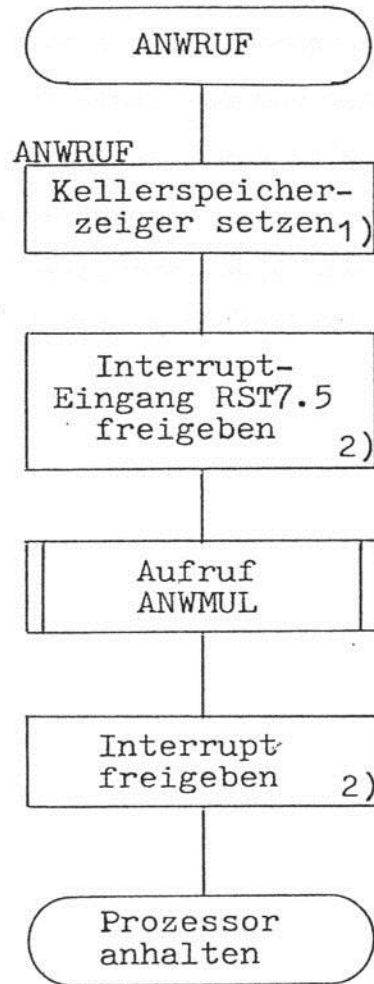
Bild 3.5-1

Ablaufplan des Anwender-Beispiels

a) Multiplikations-
Programm ANWMUL



b) Aufruf-Programm
ANWRUF



1) unbedingt erforderlich in allen Programmen, die Unterprogramme aufrufen (CALL)

2) erforderlich, um das Programm mit der I-Taste abbrechen zu können

Tabelle 3.5-2

Programme des Anwender-Beispiels

a) Multiplikations-Programm ANWMUL

Hexadezi- mal-Adresse	Hexadezi- mal-Befehl	Assembler-Befehl	Kommentar
07EE	0E08	ANWMUL: MVI C,8H	;Stellenzähler auf 8 ; setzen
07FO	7A	MOV A,D	
07F1	210000	LXI H,0	;Zwischensumme auf 0 ; setzen
07F4	54	MOV D,H	
07F5	29	MUL1: DAD H	;Zwischensumme um eine ; Stelle nach links ; schieben
07F6	17	RAL	;Multiplikand um eine ; Stelle nach links ; schieben
07F7	D2FB07	JNC MUL2	;herausgeschobenes Bit= ; =1? nein -Sprung
07FA	19	DAD D	;ja -Multiplikator zur ; Zwischensumme ad- ; dieren
07FB	0D	MUL2: DCR C	;Stellenzähler dekre- ; mentieren ;Stellenzähler=0?
07FC	C2F507	JNZ MUL1	;nein -Sprung
07FF	C9	RET	;ja -Rücksprung

b) Aufruf-Programm ANWRUF

Hexadezi- mal-Adresse	Hexadezi- mal-Befehl	Assembler-Befehl	Kommentar
07E3	31B2FF	ANWRUF: LXI SP,FFB2	;Kellerspeicherzei- ger setzen
07E6	3EOB	MVI A,OBH	;Interrupt-Eingang- ; RST7.5 freigeben
07E8	30	SIM	
07E9	CDEE07	CALL ANWMUL	;Aufruf ANWMUL
07EC	FB	EI	;Interrupt freigeben
07ED	76	HLT	;Prozessor anhalten

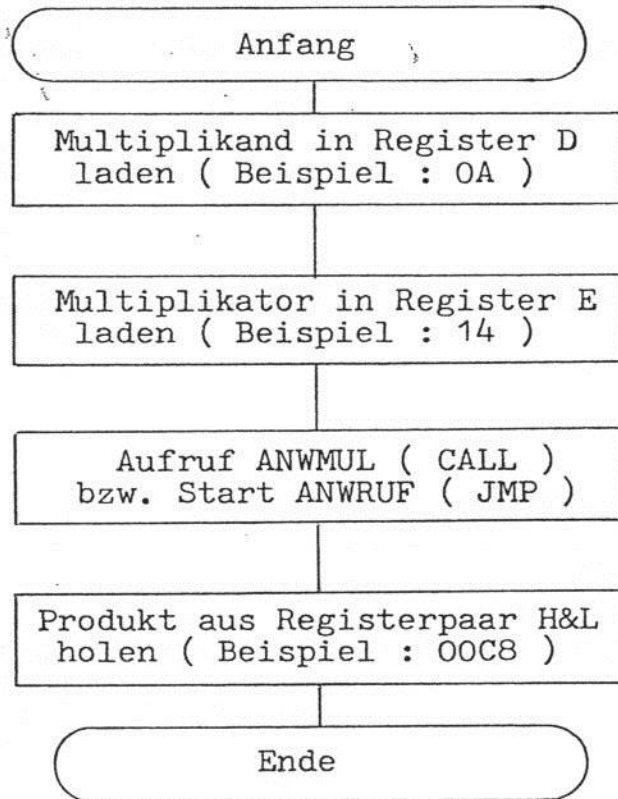
Bild 3.5-3

Zahlenbeispiel für das Unterprogramm ANWMUL

a) Register und Zahlen

Größe	Multiplikand	Multiplikator	Produkt
Register	D	E	H&L
Zahlen-Beispiel (dezimal)	10	20	200
Zahlen-Beispiel (hexadezimal)	<i>EINGABE</i> 0A	14	<i>ERGEBNIS</i> 00C8

b) Schritte bei der Anwendung



3.6 Inbetriebnahme des Geräts

Beim Anlegen der Betriebsspannung wird der ECB85 durch seine Rücksetz-Automatik selbsttätig rückgesetzt und bei Adresse 0 gestartet. Die gleiche Wirkung kann auch zu jedem späteren Zeitpunkt durch Drücken der R-Taste erreicht werden. Da die Adresse 0 dem EPROM-Steckplatz Nr. 1 zugeordnet ist und an dieser Stelle bei Lieferung der Monitor-EPROM steckt, wird durch das Rücksetzen das Monitor-Programm gestartet.

Will man mit dem Monitor-Programm arbeiten, muß beim Rücksetzen der Programm-Wahlschalter die Stellung "MON" haben. Ist dies nicht der Fall, so ist er in diese Stellung zu bringen und anschließend noch einmal die R-Taste zu drücken. Außerdem sollte als Ausgangsstellung für die folgende Beschreibung der Programmierschalter die der Bezeichnung "PRGR" entgegengesetzte Stellung einnehmen. Die Funktion der beiden genannten Schalter in den anderen Stellungen wird noch später behandelt. Zunächst ist immer von der beschriebenen Grundstellung auszugehen.

3.7 Prinzipien der Kommunikation

Nach dem Starten des Monitor-Programms können Kommandos eingegeben werden. Durch die Kommandos teilt der Anwender dem ECB85 die gewünschte Operation mit. Ein Kommando besteht aus dem Kommando-Symbol, Parametern und Abschlußzeichen. Das Kommando-Symbol wählt die Funktion aus, die Parameter bezeichnen die genauen Bedingungen. Kommando-Symbol und Parameter werden im allgemeinen durch Abschlußzeichen beendet und als gültig erklärt. Wenn alle erforderlichen Eingaben erfolgt sind, beginnt der ECB85 mit der Ausführung des Kommandos.

Die Eingabe und Ausführung der Kommandos erfolgt im Dialog zwischen Anwender und Mikrocomputer. Der Anwender gibt seine Anweisungen über das Tastenfeld ein, der Mikrocomputer teilt durch Ausgabe über das Anzeigefeld laufend mit, welche Eingabe im Augenblick erwartet wird und ob das Kommando ordnungsgemäß abläuft.

Das Tastenfeld umfaßt 2 Tasten mit Hardware-Funktion und 20 Tasten mit Software-Funktion. Die Hardware-Tasten sind mit R (Rücksetzen) und I (Interrupt) bezeichnet. Sie wirken direkt auf die Hardware des Mikrocomputers ein und haben im einzelnen folgende Funktion:

- a) Die R-Taste dient zum Rücksetzen des ECB85. Sie wird hauptsächlich dazu benutzt, die Zentraleinheit aus einem unübersichtlichen oder unerwünschten Zustand wieder in einen definierten Ausgangszustand zu bringen. Rücksetzen bedeutet im wesentlichen den Programmstart bei Adresse 0. Wenn im EPROM-Steckplatz Nr. 1 mit der Anfangsadresse 0 der Monitor-EPROM steckt (Zustand bei Auslieferung), wird durch Rücksetzen das gerade laufende Programm abgebrochen und das Monitor-Programm gestartet.
- b) Die I-Taste dient zum Auslösen eines Interrupt RST7.5 (Adresse 3C). Das Monitor-Programm des ECB85 enthält zu diesem Interrupt ein spezielles Interrupt-Programm, das die Registerinhalte im Schreib-/Lese-Speicher sicherstellt und anschließend die Eingabe von Kommandos erlaubt. Das Sicherstellen der Registerinhalte ermöglicht das spätere lückenlose Fortsetzen.

des unterbrochenen Anwender-Programms. Die zweckmäßige Benutzung der I-Taste wird noch im weiteren Verlauf dieser Bedienungsanleitung erläutert.

Die Software-Tasten sind mit 0...9, A...F, ←, →, ↑ und ↓ bezeichnet. Das Drücken dieser Tasten hat nur dann Auswirkungen, wenn die Tasten im Rahmen eines Programms abgefragt werden. Hier interessiert speziell die Funktion, die den Tasten durch das Monitor-Programm zugeordnet wird. Dabei kann man die Software-Tasten in folgende drei Gruppen einteilen:

- a) Die Hexadezimaltasten 0...9 und A...F dienen zum Eingeben von Hexadezimalzahlen. Beim ECB85 werden Kommando-Symbole, Maschinenbefehle, Adressen und Daten einheitlich als Hexadezimalzahlen dargestellt.
- b) Die Abschlußtasten ↑ und ↓ dienen hauptsächlich zum Abschließen bei der Eingabe von Kommando-Symbolen und Parametern. In einigen Fällen haben sie außerdem noch Nebenwirkungen wie z.B. die automatische Inkrementierung und Dekrementierung von Zahlen. Manchmal werden sie außerdem für ganz andere Zwecke benutzt wie z.B. das Freigeben eines auszuführenden Befehls im Anwender-Programm. Darauf wird im Einzelfall noch genau eingegangen.
- c) Die Steuertasten ← und → dienen zum Verschieben der sogenannten Schreibmarke, deren Zweck noch erläutert werden wird.

Das Anzeigefeld besteht aus acht Sieben-Segment-Anzeigen. Es dient einerseits zur Kontrolle der vom Anwender über das Tastenfeld vorgenommenen Eingaben, andererseits werden damit Ergebnisse angezeigt. Das Anzeigefeld ist in drei Bereiche mit unterschiedlichen Aufgaben gegliedert, wobei die Grenzen zum Teil fließend sind. Sie haben folgende Funktionen:

- a) Die am weitesten links liegende Anzeigestelle heißt Kommandofeld. Diese Stelle ist ausschließlich dafür reserviert, das eingegebene Kommando-Symbol anzuzeigen. Der an der rechten Seite dieser Anzeigestelle vorhandene Punkt ist stets erleuchtet.

- b) Die mittleren Anzeigestellen bilden das Adreßfeld. Die Grenze nach rechts ist je nach Erfordernissen unterschiedlich. Im Adreßfeld wird entweder ein Parametername (2 Zeichen), der Name eines Registerpaares (2 Zeichen), eine Speicher-Adresse (4-stellige Zahl) oder eine Ein-/Ausgabe-Adresse (2-stellige Zahl) angezeigt.
- c) Die rechten Anzeigestellen bilden das Datenfeld. Die Grenze nach links ist je nach Erfordernissen unterschiedlich. Im Datenfeld werden entweder Speicher-Adressen (4-stellig), Ein-/Ausgabe-Adressen (2-stellig), Daten im Zusammenhang mit einer Speicherzelle oder einem Ein-/Ausgabekanal (2-stellig) oder Daten in Zusammenhang mit einem Registerpaar (4-stellig) angezeigt. Zwischen Adreß- und Datenfeld steht immer ein Gleichheitszeichen. Dieses bedeutet jedoch hier nicht Gleichheit, sondern Zuordnung.

Bei der Eingabe von Kommandos in den ECB85 werden konsequent zwei Prinzipien eingehalten: Die Bereitstellung von Vorschlagwerten (Default-Werte) bei der Parameter-Eingabe und die Verwendung einer Schreibmarke (Cursor) bei jeder Art von Eingabe. Der Vorschlagwert soll dem Anwender die Eingabe von Parametern erleichtern. Wenn der ECB85 einen Parameter anfordert, so macht er grundsätzlich für den Zahlenwert einen Vorschlag im Anzeigefeld. Dieser Zahlenwert ist je nach Zweckmäßigkeit in der speziellen Situation eine feste Zahl oder der letzte im gleichen Zusammenhang verwendete Wert. Wenn der Anwender den Vorschlagwert als gültige Zahl für den angeforderten Parameter erklären will, braucht er lediglich eine der Abschlußtasten zu drücken. Er kann jedoch auch den Vorschlagwert als Ganzes oder in einzelnen Stellen ändern und erst dann abschließen.

Die Schreibmarke ist eine besondere Markierung einer Stelle des Anzeigefelds. Falls in der entsprechenden Stelle ein Zeichen angezeigt wird, äußert sich die Schreibmarke durch Blinken des Zeichens (abwechselnd hell und dunkel). Ist die Anzeigestelle jedoch leer, wird die Schreibmarke durch Blinken des unteren waagerechten Segments (Segment d) dargestellt.

Durch die Schreibmarke zeigt der ECB85 an, an welcher Stelle der Anzeige die nächste eingegebene Hexadezimalziffer erscheint. Aus der Stellung der Schreibmarke ergibt sich zugleich die Funktion der nächsten Eingabe. Die Schreibmarke kann über das Anzeigefeld verschoben werden, wobei es folgende Möglichkeiten gibt:

a) Programmgesteuerte Verschiebung

Wenn das Monitor-Programm die Eingabe eines neuen Parameters erwartet, zeigt der ECB85 einen Vorschlagwert an, und die Schreibmarke springt auf die linke Stelle dieser Zahl. Dies ist die Aufforderung an den Anwender, einen Zahlenwert, beginnend mit der höchstwertigen Ziffer, einzugeben.

b) Verschieben durch Zifferneingabe

Mit dem Drücken einer Zifferntaste wird die Ziffer, die in der durch die Schreibmarke gekennzeichneten Stelle steht, durch die eingegebene Ziffer ersetzt. Im allgemeinen springt zugleich die Schreibmarke automatisch auf die rechts anschließende Ziffer. Auf diese Weise kann der Anwender eine im Anzeigefeld stehende Zahl von links nach rechts überschreiben. Wenn die Schreibmarke auf der am weitesten rechts stehenden Stelle der Anzeige, an der ein Überschreiben zugelassen ist, steht, bleibt sie auch beim weiteren Eingeben von Ziffern stehen, und mit den eingegebenen Ziffern wird immer die gleiche Stelle überschrieben.

c) Verschieben durch die Steuertasten

Mit der Steuertaste → kann die Schreibmarke um eine Stelle nach rechts, mit der Steuertaste ← um eine Stelle nach links verschoben werden. Damit besteht die Möglichkeit, in einem Vorschlagwert einzelne Stellen selektiv zu ändern. Dabei werden automatisch Anzeigestellen übersprungen, an denen ein Überschreiben nicht zulässig ist. Außerdem werden in diesem Zusammenhang die ganz linke und ganz rechte Anzeigestelle als benachbart behandelt, d.h. man kann sich die Anzeigestellen wie auf einem Kreis angeordnet vorstellen. So läßt sich mit der Taste → die Schreibmarke von der ganz rechten in die ganz linke und mit der Taste ← von der ganz linken in die ganz rechte Anzeigestelle verschieben.

3.8 Verwendete Symbolik des Anzeigefelds

In diesem Kapitel wird die Symbolik erläutert, die der ECB85 für die Rückmeldung an den Anwender über das Anzeigefeld benutzt. Dabei kann man, wie schon beschrieben, nach dem Ort der Anzeige unterscheiden zwischen Kommando-, Adreß- und Datenfeld.

Im Kommandofeld zeigt der ECB85 das Kommando-Symbol an, das beim nachfolgenden Drücken einer der Abschlußtasten wirksam wird. Die zugelassenen Kommando-Symbole zusammen mit ihrer Funktion sind in Tabelle 3.8-1 zusammengestellt.

Die Anzeige im Adreßfeld hat je nach Erfordernis eine der folgenden Bedeutungen:

a) Name eines angeforderten Parameters

Die benutzten Symbole sind in Tabelle 3.8-2a zusammengestellt. Die Bedeutung wird in Zusammenhang mit den einzelnen Kommandos behandelt.

b) Name eines Registerpaars

Die benutzten Symbole sind in Tabelle 3.8-2b zusammengestellt. Im wesentlichen handelt es sich um die bekannten Register des Mikroprozessor-Bausteins SAB8085A. Das Zustandswort (F) ist zwar für das System 8085 genau festgelegt, doch wird es wenig benutzt und ist daher in Bild 3.8-1 kurz erläutert. Das Interrupt-Statuswort ist speziell für den ECB85 definiert und sonst nicht gebräuchlich. Es besteht aus zwei Bytes und soll anhand von Bild 3.8-3b erläutert werden. Das niederwertige Byte enthält das Datenwort, das nach dem letzten Unterbrechen eines Anwender-Programms mit Hilfe des Befehls RIM ausgelesen wurde. Außerdem hat das in dem Wort enthaltene Bit D3 noch die Nebenfunktion, daß vor dem nächsten Starten eines Anwender-Programms mit dem entsprechenden Kommando nach diesem Bit Interrupt freigegeben oder gesperrt wird. Das höherwertige Byte des Interrupt-Statusworts enthält ein Datenwort, das vor dem nächsten Starten eines Anwender-Programms mit dem entsprechenden Kommando mit Hilfe des Befehls SIM ausgegeben werden soll. Die Bedeutung des Interrupt-Statusworts wird noch in Zusammenhang mit den einzelnen Kommandos erläutert.

c) Adresse

Wenn im Adreßfeld eine Zahl steht, handelt es sich bei einer vierstelligen Zahl um eine Speicher-Adresse und bei einer zweistelligen Zahl um eine Ein-/Ausgabe-Adresse.

Der ECB85 bietet als Erleichterung für den Anwender die Möglichkeit, Adressen mit der Abschlußtaste ↓ automatisch zu inkrementieren (um 1 erhöhen) und mit der Taste ↑ zu dekrementieren (um 1 verringern). Dabei sind Adressen im weitesten Sinn aufzufassen, also Speicher-Adressen, Ein-/Ausgabe-Adressen oder Registernamen. Ob im Einzelfall die genannte Möglichkeit besteht, hängt von dem jeweiligen Kommando ab. An dieser Stelle sollen nur die Besonderheiten gebracht werden, nach denen das Inkrementieren und Dekrementieren vor sich geht:

a) Speicher-Adressen

Beim Inkrementieren und Dekrementieren sind die Adressen 0000 und FFFF als benachbart zu betrachten. Bild 3.8-4a zeigt die Verhältnisse schematisch.

b) Ein-/Ausgabe-Adressen

Beim Inkrementieren und Dekrementieren sind die Adressen 00 und FF als benachbart zu betrachten. Bild 3.8-4b zeigt die Verhältnisse schematisch.

c) Register-Namen

Für das Inkrementieren und Dekrementieren gilt die in Bild 3.8-4c angegebene Reihenfolge der Registerpaare. Dabei sind die Registerpaare AF und IS als benachbart zu betrachten.

Tabelle 3.8-1

Symbole im Kommandofeld und ihre Funktion

□	Speicher auslesen und beschreiben
□	Speicherbereich mit Konstante füllen
□	Speicherinhalt übertragen
□	Adressen umrechnen
□	über Ausgabe-Kanal ausgeben
□	über Eingabe-Kanal eingeben
□	Register auslesen und beschreiben
□	Programm ohne Unterbrechungspunkte starten
□	Programm mit Unterbrechungspunkten starten
□	Programm im Einzelschritt abarbeiten
□	Information auf Magnetband schreiben
□	Information von Magnetband lesen

Tabelle 3.8-2

Symbole im Adreßfeld und ihre Funktion

a) Symbole für Parameternamen

AA	Anfangsadresse
EA	Endadresse
BA	Bestimmungsadresse
SA	Startadresse
H1	Unterbrechungsadresse 1
H2	Unterbrechungsadresse 2
AU	Anfangsadresse des umzurechnenden Bereichs
EU	Endadresse des umzurechnenden Bereichs
PA	Peripherie-Adresse
db	Datenbyte

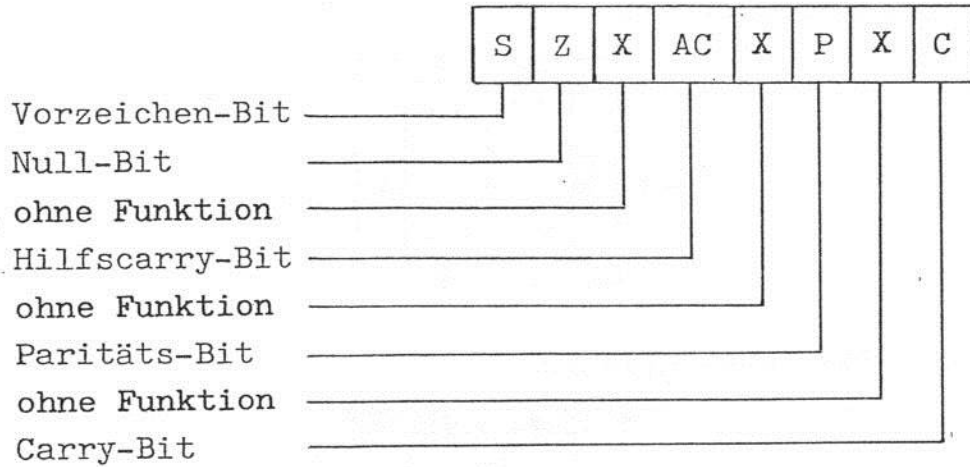
b) Symbole für Registerpaar-Namen

AF	Akkumulator & Zustandswort(Flag)
BC	Registerpaar B & C
DE	Registerpaar D & E
HL	Registerpaar H & L
PC	Programmzähler (program counter)
SP	Kellerspeicher-Zeiger (stack pointer)
IS	Interrupt-Status

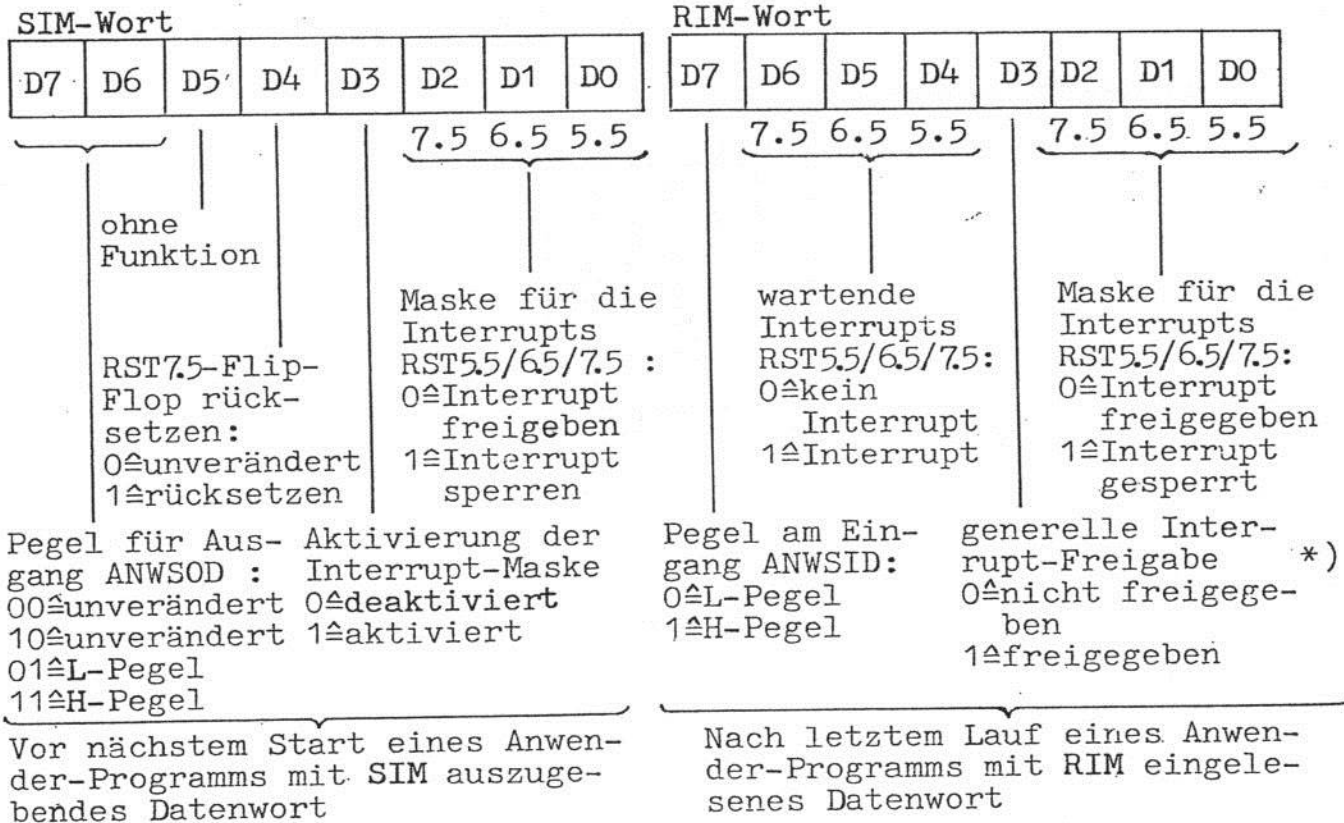
Bild 3.8-3

Bedeutung der Registernamen

a) Zustandswort (F)



b) Interrupt-Statuswort (IS)

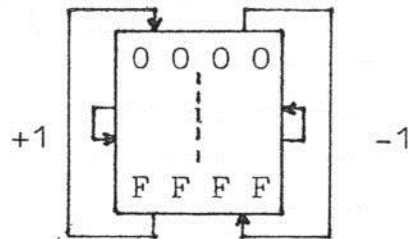


*) Dieses Bit hat beim ECB85 außerdem noch die Nebenfunktion, daß beim nächsten Start eines Anwender-Programms entsprechend diesem Bit Interrupt generell gesperrt bzw. freigegeben wird.

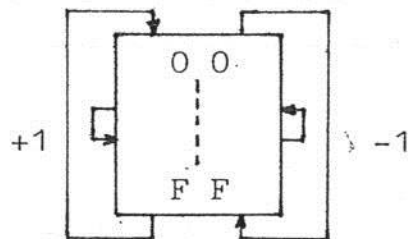
Bild 3.8-4

Inkrementierung und Dekrementierung von Adressen

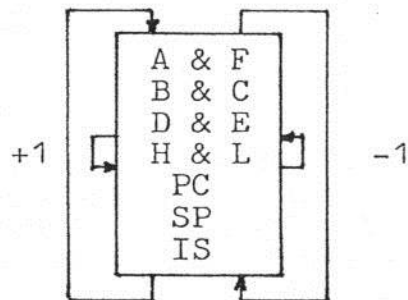
a) Speicher-Adressen



b) Ein-/Ausgabe-Adressen



c) Register-Namen



3.9 Einleiten eines Kommandos

Nach dem Einschalten der Betriebsspannung oder Drücken der R-Taste ist das Anzeigefeld dunkel bis auf den Punkt des Kommandofelds (linke Anzeigestelle). Die Schreibmarke steht im Kommandofeld. Der ECB85 erwartet nun die Eingabe eines der Kommando-Symbole 0...9, A, B. Das Drücken der nicht für Kommando-Symbole zugelassenen Tasten wird ignoriert und hat keine Auswirkungen. Wenn ein gültiges Kommando-Symbol über das Tastenfeld eingegeben ist, erscheint es im Kommandofeld. Falls dann noch weitere gültige Kommando-Symbole folgen, wird das Kommandofeld mit den neuen Eingaben überschrieben. Das angezeigte Kommando-Symbol wird erst durch Drücken einer der Abschlußtasten ↑ oder ↓ wirksam. Der weitere Ablauf hängt von dem jeweiligen Kommando ab.

Ein laufendes Kommando läßt sich jederzeit durch Drücken der R-Taste abbrechen. Anschließend kann dann ein neues Kommando eingegeben werden. Ein Kommando läßt sich aber auch abbrechen, indem man die Schreibmarke mit einer der Schreibmarken-Steuertasten (← , →) in das Kommandofeld bringt und das dort stehende Kommando-Symbol durch ein neues überschreibt. Sobald ein Kommando-Symbol eingegeben ist, werden Adreß- und Datenfeld dunkel. Mit dem Drücken einer der Abschlußtasten wird das Kommando-Symbol wirksam. Der Unterschied zwischen den beiden Möglichkeiten des Kommando-Abbruchs besteht hauptsächlich darin, daß beim Drücken der R-Taste der Programm-Wahlschalter abgefragt, Interrupt für Anwender-Programme gesperrt und gewissen Vorschlagwerten für Parameter feste Werte zugeordnet werden, während das im anderen Fall nicht geschieht.

4. Kommandos mit Bezug auf den Speicher

Die in den folgenden Kapiteln behandelte Gruppe enthält vier Kommandos, die das Einschreiben neuer Information oder das Ändern vorhandener Information im Speicher des ECB85 ermöglichen. Geändert werden kann sowohl die Information selbst als auch ihr Ort im Speicher. Neben dem Auslesen und Beschreiben einzelner Speicherzellen ist auch das Füllen ganzer Speicherbereiche mit einer Konstanten, das Übertragen von Information zwischen verschieden gelegenen Speicherbereichen und das automatische Umrechnen der Adressen in Sprungbefehlen möglich.

Information kann nicht nur in den Schreib-/Lese-Speicher eingeschrieben werden, sondern auch in einen im EPROM-Steckplatz Nr. 2 steckenden Speicherbaustein, wenn sich der Programmierschalter in der Stellung PRGR befindet. Obwohl dafür die gleichen Kommandos wie beim Beschreiben von Schreib-Lese-Speicher benutzt werden, ist diese Möglichkeit in einem besonderen Kapitel dargestellt, da dabei einige Besonderheiten zu beachten sind.

4.1 Speicher auslesen und beschreiben (Kommando 0)

Funktion

Das Kommando 0 dient zum Auslesen von Information aus dem Speicher und wahlweise zum Einschreiben neuer Information. Die Speicheradressen können dabei gegenüber dem letzten Wert entweder automatisch inkrementiert und dekrementiert oder manuell in beliebiger Weise geändert werden. Die Information läßt sich ebenfalls beliebig ändern. Im Rahmen dieses Kommandos erscheinen zu einem Zeitpunkt jeweils eine Speicheradresse und die an dieser Stelle stehende Information im Anzeigefeld.

Ablauf

Für den nachfolgend beschriebenen normalen Ablauf des Kommandos wird vorausgesetzt, daß das Einschreiben von Information nur bei Adressen versucht wird, unter denen sich funktionsfähiger Schreib-/Lese-Speicher befindet.

Die Schreibmarke muß sich zu Beginn im Kommandofeld befinden. Es wird das Kommando-Symbol 0 eingegeben und eine beliebige der Abschlußtasten gedrückt. Darauf erscheint im Adreßfeld eine 4-stellige Speicheradresse und im Datenfeld der zugehörige 2-stellige Speicherinhalt. Die anfangs angezeigte Speicheradresse ist ein Vorschlagwert, der nach dem Einschalten des Geräts eine zufällige Zahl und im weiteren Verlauf der Wert der zuletzt im Zusammenhang mit den Kommando 0 verwendeten Speicheradresse ist. Die Schreibmarke steht zu Anfang auf der linken Stelle des Adreßfelds.

Wenn die Schreibmarke im Adreßfeld steht, kann die Adresse mit Hilfe der Zifferntasten und eventueller Benutzung der Schreibmarken-Steuertasten beliebig geändert werden. Nach jeder Änderung der Adresse erscheint sofort der zugehörige Speicherinhalt im Datenfeld.

Zum Ändern des Speicherinhalts ist die Schreibmarke in das Datenfeld zu verschieben. Dies kann auf zwei Arten erfolgen: Nach dem Überschreiben der rechten Stelle der Adresse verschiebt sich die Schreibmarke automatisch auf die linke Stelle des Datenfelds; die gleiche Stellung kann aber auch mit der Schreibmarken-Steuertaste erreicht werden. Das Überschreiben des

Speicherinhalts erfolgt in bekannter Weise mit den Zifferntasten und eventueller Benutzung der Schreibmarken-Steuertasten. Nach jeder Änderung einer Ziffer des Datenworts wird der neue Wert sofort eingeschrieben.

Bei diesem Kommando werden die Adressen und Daten nicht mit den Abschluß-tasten beendet. Diese Tasten haben stattdessen eine andere Funktion. Unabhängig von der augenblicklichen Stellung der Schreibmarke im Adreß- und Datenfeld führt das Drücken der Taste ↓ dazu, daß die im Adreßfeld stehende Adresse inkrementiert (um 1 erhöht) wird und im Datenfeld der zugehörige Speicherinhalt erscheint. Mit dem Drücken der Taste ↑ wird die im Adreßfeld stehende Adresse dekrementiert (um 1 verringert) und ebenfalls im Datenfeld der zugehörige Speicherinhalt angezeigt. Nach jedem Drücken der Taste ↑ oder ↓ befindet sich die Schreibmarke auf der linken Stelle des Datenfelds.

Das Kommando 0 ist vom Prinzip her endlos und kann nur durch Eingabe eines neuen Kommandos (dazu Schreibmarke in das Kommandofeld verschieben) oder Drücken der R-Taste abgebrochen werden.

Besonderheiten

- 1) Nach dem Einschalten des Geräts stehen im Schreib-/Lese-Speicher Zufallswerte, die dann bei Anwendung des Kommandos angezeigt werden.
- 2) Das Monitor-Programm liest bei Ausführung des Kommandos 0 sofort nach jedem Einschreiben in den Speicher die Information wieder aus (Kontrolllesen) und zeigt die ausgelesene Information im Datenfeld der Anzeige an. Eine über das Tastenfeld als Information eingegebene Ziffer erscheint deshalb nur dann im Datenfeld, wenn sie sich auch in den Speicher einschreiben ließ, d.h. wenn sich unter der Adresse funktionsfähiger Schreib-Lese-Speicher befindet. In allen anderen Fällen, z.B. bei nicht ausgebautem Speicher oder EPROM unter der Adresse, erscheint im Datenfeld der Anzeige die beim Kontrolllesen vorgefundene Information.

Beispiel

In den Schreib-/Lese-Speicher soll bei Adresse 1800 das Datenwort 31 (z.B. als Operationscode für den Befehl LXI SP) eingegeben werden. Anschließend ist der Inhalt unter der folgenden Adresse 1801 festzustellen. Der Ablauf ist in Tabelle 4.1-1 dargestellt.

Tabelle 4.1-1

Beispiel für das Kommando 0

Eingabe		Ausgabe	
Taste	Bedeutung	Anzeigefeld 1)	Bedeutung
		(Schreibmarke im Kommandofeld)	Bereitschaft zur Eingabe eines Kommando-Symbols
0	Kommando-Symbol	0. Δ	eingegebenes Kommando-Symbol
↓	Abschluß des Kommando-Symbols	0. X X X X = Y Y Δ	Kommando-Symbol, Speicher-Adresse und Speicherinhalt
1	neue Ziffer für erste Stelle der Speicher-Adresse	0. 1 X X X = Y Y Δ	Kommando-Symbol, Speicher-Adresse (geändert) und Speicherinhalt
8	neue Ziffer für zweite Stelle der Speicher-Adresse	0. 1 8 X X = Y Y Δ	- " -
0	neue Ziffer für dritte Stelle der Speicher-Adresse	0. 1 8 0 X = Y Y Δ	- " -
0	neue Ziffer für vierte Stelle der Speicher-Adresse	0. 1 8 0 0 = Y Y Δ	- " -
3	neue Ziffer für erste Stelle des Speicherinhalts	0. 1 8 0 0 = 3 Y Δ	Kommando-Symbol, Speicher-Adresse und Speicherinhalt (geändert)
1	neue Ziffer für zweite Stelle des Speicherinhalts	0. 1 8 0 0 = 3 1 Δ	- " -
↓	Inkrementieren der Speicher-Adresse	0. 1 8 0 1 = Y Y Δ	Kommando-Symbol, Speicher-Adresse (erhöht), Speicherinhalt

- 1) Δ Stellung der Schreibmarke
 XXXX Vorschlagwert für Speicher-Adresse
 YY alter Speicherinhalt unter der angezeigten Speicher-Adresse

4.2 Speicherbereich mit Konstante füllen (Kommando 1)

Funktion

Das Kommando 1 dient zum Einschreiben des gleichen Informationsworts (Konstante) an alle Adressen eines Speicherbereichs. Die Anfangs- und Endadresse des zu beschreibenden Speicherbereichs sowie das einzuschreibende Informationswort sind als Kommando-Parameter einzugeben.

Ablauf

Für den nachfolgend beschriebenen normalen Ablauf des Kommandos wird vorausgesetzt, daß sich in dem ganzen bezeichneten Bereich funktionsfähiger Schreib-/Lese-Speicher befindet.

Die Schreibmarke muß sich zu Beginn im Kommandofeld befinden. Es wird das Kommando-Symbol 1 eingegeben und eine beliebige der Abschlußtasten gedrückt. Darauf erscheint im Adreßfeld das Symbol für die Anfangsadresse (AA) und im Datenfeld die Zahl 0000 als Vorschlagwert für die Adresse. Die Schreibmarke steht zu Anfang auf der linken Stelle des Datenfelds.

Der Vorschlagwert der Adresse kann jetzt durch Überschreiben geändert werden. Mit Drücken einer beliebigen der Abschlußtasten wird der Wert als gültig übernommen. Darauf erscheint im Adreßfeld das Symbol für die Endadresse (EA) und im Datenfeld die Zahl 0000 als Vorschlagwert für die Adresse. Eingabe eines Werts und Abschluß erfolgen in gleicher Weise. Darauf erscheint im Adreßfeld das Symbol für das Datenbyte (db) und im Datenfeld die Zahl 00 als Vorschlagwert für das Informationswort. Nach Eingabe eines Werts und Abschluß ist die Parameter-Eingabe beendet.

Anschließend führt das Gerät das Kommando aus. Dabei wird laufend im Adreßfeld die gerade bearbeitete Adresse und im Datenfeld das Informationswort angezeigt. Nach Ausführung des Kommandos steht die Schreibmarke im Kommandofeld und Adreß- und Datenfeld sind dunkel.

Besonderheiten

- 1) Wenn die eingegebene Anfangsadresse größer ist als die Endadresse, werden die beiden Teilbereiche [Anfangsadresse ... FFFF] und [0000 ... Endadresse] beschrieben.

- 2) Das Monitor-Programm liest bei Ausführung des Kommandos sofort nach jedem Einschreiben in den Speicher die Information wieder aus (Kontrolllesen) und zeigt die ausgelesene Information im Datenfeld an. Das über das Tastenfeld eingegebene Informationswort erscheint deshalb nur dann im Datenfeld, wenn es sich auch in den Speicher einschreiben ließ, d.h. wenn sich unter der Adresse funktionsfähiger Schreib-/Lese-Speicher befindet. In allen anderen Fällen, z.B. bei nicht ausgebautem Speicher oder EPROM unter der Adresse, erscheint im Datenfeld der Anzeige die beim Kontrolllesen vorgefundene Information.

- 3) Das Monitor-Programm prüft jedesmal, ob das eingeschriebene und das beim Kontrolllesen ausgelesene Datenwort identisch sind. Beim Auftreten eines Fehlers wird das Einschreiben wiederholt, so lange das Kontrolllesen eine falsche Information liefert. Im Adreßfeld steht dann die Fehleradresse und im Datenfeld das dort beim Kontrolllesen vorgefundene Datenwort. Die Schreibmarke ist in diesem Fall nicht vorhanden. Das Kommando läßt sich in dieser Situation nur mit der R-Taste abbrechen.

Beispiel

Im Schreib-/Lese-Speicher soll der Bereich zwischen der Anfangsadresse 1810 und der Endadresse 18FF mit der Konstanten 76 gefüllt werden. Der Ablauf ist in Tabelle 4.2-1 dargestellt.

Tabelle 4.2-1

Beispiel für das Kommando 1

Eingabe		Ausgabe	
Taste	Bedeutung	Anzeigefeld ¹⁾	Bedeutung
		(Schreibmarke im Kommandofeld)	Bereitschaft zur Eingabe eines Kommando-Symbols
1	Kommando-Symbol	1. Δ	einggegebenes Kommando-Symbol
↓	Abschluß des Kommando-Symbols	1. A A = 0 0 0 0 Δ	Kommando-Symbol, Symbol für Anfangsadresse, Vorschlagwert für Anfangsadresse
1	neue Ziffer für erste Stelle der Anfangsadresse	1. A A = 1 0 0 0 Δ	Kommando-Symbol, Symbol für Anfangsadresse, Anfangsadresse (geändert)
8	neue Ziffer für zweite Stelle der Anfangsadresse	1. A A = 1 8 0 0 Δ	- " -
1	neue Ziffer für dritte Stelle der Anfangsadresse	1. A A = 1 8 1 0 Δ	- " -
↓	Abschluß der Anfangsadresse	1. E A = 0 0 0 0 Δ	Kommando-Symbol, Symbol für Endadresse, Vorschlagwert für Endadresse
1	neue Ziffer für erste Stelle der Endadresse	1. E A = 1 0 0 0 Δ	Kommando-Symbol, Symbol für Endadresse, Endadresse (geändert)
B	neue Ziffer für zweite Stelle der Endadresse	1. E A = 1 B 0 0 Δ	- " -
F	neue Ziffer für dritte Stelle der Endadresse	1. E A = 1 B F 0 Δ	- " -
F	neue Ziffer für vierte Stelle der Endadresse	1. E A = 1 B F F Δ	- " -

(Fortsetzung)

Tabelle 4.2-1
(Fortsetzung)

Eingabe		Ausgabe	
Taste	Bedeutung	Anzeigefeld 1)	Bedeutung
↓	Abschluß der Endadresse	1. d b = 0 0 Δ	Kommando-Symbol, Symbol für Datenbyte, Vorschlagwert für Datenbyte
7	neue Ziffer für erste Stelle des Datenbytes	1. d b = 7 0 Δ	Kommando-Symbol, Symbol für Datenbyte, Datenbyte (geändert)
6	neue Ziffer für zweite Stelle des Datenbytes	1. d b = 7 6 Δ	- " -
↓	Kommando-Abschluß	1. 1 * * * = 7 6	Kommando-Symbol, laufende Anzeige der Speicher-Adresse, Datenbyte
		1. Δ	Kommando ordnungsgemäß ausgeführt

1) Δ Stellung der Schreibmarke
 * in dieser Stelle laufende Anzeige

4.3 Speicherinhalt Übertragen (Kommando 2)

Funktion

Das Kommando 2 dient zum Übertragen (Kopieren) von Information aus einem Speicher-Ursprungsbereich in einen Speicher-Zielbereich. Der Speicher-Ursprungsbereich wird durch seine Anfangsadresse und Endadresse bezeichnet, der Zielbereich durch seine Anfangsadresse, die in diesem Fall Bestimmungsadresse heißt. Diese drei Zahlen sind als Parameter einzugeben.

Ablauf

Für den nachfolgend beschriebenen normalen Ablauf des Kommandos wird vorausgesetzt, daß von der Bestimmungsadresse bis zu einer um die Länge des Übertragenen Bereichs größeren Adresse funktionsfähiger Schreib-/Lese-Speicher vorhanden ist. Außerdem darf die Anfangsadresse nicht größer als die Endadresse sein und der Zielbereich die Adresse FFFF nicht überschreiten.

Die Schreibmarke muß sich zu Beginn im Kommandofeld befinden. Es wird das Kommando-Symbol 2 eingegeben und eine beliebige der Abschlußtasten gedrückt. Darauf erscheint im Adreßfeld das Symbol für die Anfangsadresse des Ursprungsbereichs (AA) und im Datenfeld die Zahl 0000 als Vorschlagwert für die Adresse. Die Schreibmarke steht zu Anfang auf der linken Stelle des Datenfelds.

Der Vorschlagwert der Adresse kann jetzt durch Überschreiben geändert werden. Mit Drücken einer beliebigen der Abschlußtasten wird der Wert als gültig übernommen. Darauf erscheint im Adreßfeld das Symbol für die Endadresse des Ursprungsbereichs (EA) und im Datenfeld die Zahl 0000 als Vorschlagwert für die Adresse. Eingabe eines Werts und Abschluß erfolgen in gleicher Weise. Dann wird die Bestimmungsadresse, der das Symbol bA und der Vorschlagwert 0000 zugeordnet ist, eingegeben. Mit dem zugehörigen Drücken einer der Abschlußtasten ist die Parameter-Eingabe beendet.

Bei der Ausführung des Kommandos geht das Monitor-Programm je nach Relation der Adressen unterschiedlich vor. Wenn die Bestimmungsadresse kleiner als die Anfangsadresse des Ursprungsbereichs ist, wird zunächst der Inhalt der

Anfangsadresse zur Bestimmungsadresse übertragen, und dann die weiteren Adressen in aufsteigender Adressenfolge bearbeitet. Ist dagegen die Bestimmungsadresse größer als die Anfangsadresse, wird zunächst der Inhalt der Endadresse des Ursprungsbereichs zur obersten Adresse des Zielbereichs übertragen und dann die weiteren Adressen in abnehmender Adressenfolge. Auf diese Weise wird sichergestellt, daß auch bei sich überlappendem Ursprungs- und Zielbereich die Daten übertragen werden, ehe die entsprechende Speicherzelle überschrieben wird.

Während das Kommando abläuft, wird laufend im Adreßfeld der Anzeige die gerade bearbeitete Zieladresse und im Datenfeld das Informationswort angezeigt. Nach Ausführung des Kommandos steht die Schreibmarke im Kommandofeld und Adreß- und Datenfeld sind dunkel.

Besonderheiten

- 1) Wenn die eingegebene Anfangsadresse größer ist als die Endadresse oder beim Beschreiben des Zielbereichs die Adresse FFFF überschritten wird, ist ein ordnungsgemäßer Ablauf des Kommandos nicht sichergestellt.
- 2) Das Monitor-Programm liest bei Ausführung des Kommandos sofort nach jedem Einschreiben in den Speicher die Information wieder aus (Kontrolllesen) und zeigt sie im Datenfeld an. Das aus dem Ursprungsbereich stammende Informationswort erscheint deshalb nur dann im Datenfeld, wenn es sich auch in den Speicher einschreiben ließ, d.h. wenn sich unter der Adresse funktionsfähiger Schreib-/Lese-Speicher befindet. In allen anderen Fällen, z.B. bei nicht ausgebautem Speicher oder EPROM, erscheint im Datenfeld der Anzeige die beim Kontrolllesen vorgefundene Information.
- 3) Das Monitor-Programm prüft jedesmal, ob das eingeschriebene und das beim Kontrolllesen ausgelesene Datenwort identisch sind. Beim Auftreten eines Fehlers wird das Einschreiben wiederholt, so lange das Kontrolllesen eine falsche Information liefert. Im Adreßfeld steht dann die Fehleradresse und im Datenfeld das dort beim Kontrolllesen vorgefundene Datenwort. Die Schreibmarke ist in diesem Fall nicht vorhanden. Das Kommando läßt sich daher nur mit der R-Taste abbrechen.

Beispiel

Es soll der Inhalt des Speichers zwischen der Anfangsadresse 0000 und der Endadresse 00FF (Anfang des Monitor-Programms) in den Speicherbereich ab Adresse 1800 übertragen werden. Der Ablauf ist in Tabelle 4.3-1 dargestellt.

Tabelle 4.3-1

Beispiel für das Kommando 2

Eingabe		Ausgabe	
Taste	Bedeutung	Anzeigefeld ¹⁾	Bedeutung
		(Schreibmarke im Kommandofeld)	Bereitschaft zur Eingabe eines Kommando-Symbols
2	Kommando-Symbol	2. Δ	eingegebenes Kommando-Symbol
↓	Abschluß des Kommando-Symbols	2. A A = 0 0 0 0 Δ	Kommando-Symbol, Symbol für Anfangsadresse des Ursprungsbereichs, Vorschlagwert für Anfangsadresse
↓	Abschluß der Anfangsadresse des Ursprungsbereichs	2. A A = 0 0 0 0 Δ	Kommando-Symbol, Symbol für Endadresse des Ursprungsbereichs, Vorschlagwert für Endadresse
→	Übernahme der ersten Stelle der Endadresse des Ursprungsbereichs	2. E A = 0 0 0 0 Δ	Kommando-Symbol, Symbol für Endadresse des Ursprungsbereichs, Wert der Endadresse
→	Übernahme der zweiten Stelle der Endadresse des Ursprungsbereichs	2. E A = 0 0 0 0 Δ	- " -
F	neue Ziffer für dritte Stelle der Endadresse des Ursprungsbereichs	2. E A = 0 0 F 0 Δ	Kommando-Symbol, Symbol für Endadresse des Ursprungsbereichs, Wert der Endadresse (geändert)
F	neue Ziffer für vierte Stelle der Endadresse des Ursprungsbereichs	2. E A = 0 0 F F Δ	- " -
↓	Abschluß der Endadresse des Ursprungsbereichs	2. b A = 0 0 0 0 Δ	Kommando-Symbol, Symbol für Bestimmungsadresse, Vorschlagwert für Bestimmungsadresse

(Fortsetzung)

Tabelle 4.3-1
(Fortsetzung)

Eingabe		Ausgabe	
Taste	Bedeutung	Anzeigefeld ¹⁾	Bedeutung
1	neue Ziffer für erste Stelle der Bestimmungsadresse	2. b A = 1 0 0 0 Δ	Kommando-Symbol, Symbol für Bestimmungsadresse, Bestimmungsadresse (geändert)
8	neue Ziffer für zweite Stelle der Bestimmungsadresse	2. b A = 1 8 0 0 Δ	- " -
↓	Kommando-Abschluß	2. 1 8 ; * * = * *	Kommando-Symbol, laufende Anzeige der Speicheradresse des Zielbereichs und derer Speicherinhalt
		2. Δ	Kommando ordnungsgemäß ausgeführt

- 1) Δ Stellung der Schreibmarke
* in dieser Stelle laufende Anzeige

4.4 Adressen umrechnen (Kommando 3)

Funktion

Das Kommando 3 dient zum Umrechnen der Sprungadressen in 8080/8085-Sprungbefehlen, wenn die Sprungbefehle in einem vorgegebenen Bereich liegen und die Sprungadressen in einen ebenfalls vorgegebenen Bereich zeigen.

Im allgemeinen wird dieses Kommando im Anschluß an Kommando 2 verwendet, mit dem die Übertragung des Inhalts aus dem einen in einen anderen Speicherbereich erfolgt. Kommando 3 führt dieses Übertragen nicht aus, doch sind ihm durch Eingabe der bei Kommando 2 verwendeten Parameter Anfangsadresse, Endadresse und Bestimmungsadresse die Kennwerte der Verschiebung mitzuteilen.

Der Speicherbereich, in dem die Sprungadressen umzurechnen sind, wird durch Eingabe der "Anfangsadresse des umzurechnenden Bereichs" und der "Endadresse des umzurechnenden Bereichs" bezeichnet. Umgerechnet werden in diesem Bereich alle Sprungadressen, die in den Ursprungsbereich des verschobenen Informationsblocks zeigen, in der Weise, daß sie danach in den Zielbereich des verschobenen Informationsblocks zeigen.

Ablauf

Für den nachfolgend beschriebenen normalen Ablauf des Kommandos wird vorausgesetzt, daß sich im umzurechnenden Bereich funktionsfähiger Schreib-/Lese-Speicher befindet und daß die Anfangsadresse des umzurechnenden Bereichs auf den Operationscode eines Befehls zeigt. Außerdem darf die Anfangsadresse des umzurechnenden Bereichs nicht größer sein als die Endadresse.

Die Schreibmarke muß sich zu Beginn im Kommandofeld befinden. Es wird das Kommando-Symbol 3 eingegeben und eine beliebige der Abschlußtasten gedrückt. Darauf erscheint im Adreßfeld das Symbol für die Anfangsadresse des Ursprungsbereichs (AA) und im Datenfeld als Vorschlagwert nach vorangehendem Rücksetzen die Zahl 0000 und später der zuletzt mit Kommando 2 oder 3 in diesem Zusammenhang eingegebene Wert. Die Schreibmarke steht zu Anfang an der linken Stelle des Datenfelds.

Der Vorschlagswert der Adresse kann jetzt durch Überschreiben geändert werden. Mit Drücken einer beliebigen der Abschlußtasten wird der Wert als gültig übernommen. In gleicher Weise wird anschließend die Endadresse des Ursprungsbereichs (Symbol EA) und die Bestimmungsadresse (Symbol bA) eingegangen, wobei für den Vorschlagswert Entsprechenden gilt.

Danach erscheint im Adreßfeld das Symbol für die Anfangsadresse des umzurechnenden Bereichs (AU) und im Datenfeld die Zahl 0000 als Vorschlagswert. Nach der Eingabe wird die Endadresse des umzurechnenden Bereichs (EU) mit dem Vorschlagswert 0000 angefordert. Nach Eingabe und Abschluß wird das Kommando ausgeführt. Während der Ausführung bleibt die letzte Anzeige erhalten, wobei jedoch die Schreibmarke verschwindet.

Nach Ausführung des Kommandos steht die Schreibmarke im Kommandofeld und Adreß- und Datenfeld sind dunkel.

Besonderheiten

- 1) Mit Hilfe des Kommandos werden nur Sprungadressen in unbedingten und bedingten Sprungbefehlen (unbedingte und bedingte JMP, CALL, RET) umgerechnet. Wenn dagegen Zahlen zunächst als Daten auftreten und die Verwendung als Adressen erst später ersichtlich wird (z.B. durch PCHL-Befehl), erfolgt in diesem Fall keine Umrechnung.
- 2) Wenn für einen Adreßbereich die eingegebene Anfangsadresse größer ist als die Endadresse, ist ein ordnungsgemäßer Ablauf des Kommandos nicht sichergestellt.
- 3) Das Kommando kann nur dann ordnungsgemäß ablaufen, wenn sich im umzurechnenden Bereich funktionsfähiger Schreib-/Lese-Speicher befindet. Eine Kontrolle der eingeschriebenen Information erfolgt jedoch bei diesem Kommando nicht, so daß auch im Fehlerfall keine Meldung erscheint.
- 4) Die Umrechnung der Adressen läuft nur dann ordnungsgemäß ab, wenn die Anfangsadresse des umzurechnenden Bereichs auf den Operationscode eines Befehls zeigt.

- 5) Die Endadresse des umzurechnenden Bereichs darf auf ein beliebiges Byte eines Befehls zeigen. Dieser Befehl wird als letzter umgerechnet.
- 6) Die Ausführungen des Kapitels 2.4 sind zu beachten.

Beispiel

Es wird vorausgesetzt, daß das Anwender-Beispiel im Monitor-Programm (siehe Kapitel 3.5) mit Kommando 2 aus dem Ursprungsbereich zwischen der Anfangsadresse (AA) 07E3 und der Endadresse (EA) 07FF zu einem Zielbereich übertragen worden ist, der mit der Bestimmungsadresse (bA) 1800 beginnt. In seinem neuen Bereich zwischen der Adresse (AU) 1800 und der Adresse (EU) 181C sollen in Sprungbefehlen alle Sprungadressen umgerechnet werden, die sich auf den übertragenen Speicherbereich beziehen. Der Ablauf ist in Tabelle 4.4-1 dargestellt.

Tabelle 4.4-1

Beispiel für das Kommando 3

Eingabe		Ausgabe	
Taste	Bedeutung	Anzeigefeld 1)	Bedeutung
		(Schreibmarke im Kommandofeld)	Bereitschaft zur Eingabe eines Kommando-Symbols
3	Kommando-Symbol	3. Δ	einggegebenes Kommando-Symbol
↓	Abschluß des Kommando-Symbols	3. A A = 0 7 E 3 Δ	Kommando-Symbol, Symbol für Anfangsadresse des Ursprungsbereichs, Vorschlagwert für Anfangsadresse
↓	Abschluß der Anfangsadresse des Ursprungsbereichs (Vorschlagwert übernommen)	3. E A = 0 7 F F Δ	Kommando-Symbol, Symbol für Endadresse des Ursprungsbereichs, Vorschlagwert für Endadresse
↓	Abschluß der Endadresse des Ursprungsbereichs (Vorschlagwert übernommen)	3. b A = 1 8 0 0 Δ	Kommando-Symbol, Symbol für Bestimmungsadresse, Vorschlagwert für Bestimmungsadresse
↓	Abschluß der Bestimmungsadresse (Vorschlagwert übernommen)	3. A U = 0 0 0 0 Δ	Kommando-Symbol, Symbol für Anfangsadresse des umzurechnenden Bereichs, Vorschlagwert für Anfangsadresse
1	neue Ziffer für erste Stelle der Anfangsadresse des umzurechnenden Bereichs	3. A U = 1 0 0 0 Δ	Kommando-Symbol, Symbol für Anfangsadresse des umzurechnenden Bereichs, Anfangsadresse (geändert)
8	neue Ziffer für zweite Stelle der Anfangsadresse des umzurechnenden Bereichs	3. A U = 1 8 0 0 Δ	- " -
↓	Abschluß der Anfangsadresse des umzurechnenden Bereichs (dritte und vierte Stelle des Vorschlagwertes übernommen)	3. E U = 0 0 0 0 Δ	Kommando-Symbol, Symbol für Endadresse des umzurechnenden Bereichs, Vorschlagwert für Endadresse

(Fortsetzung)

Tabelle 4.4-1
(Fortsetzung)

Eingabe		Ausgabe	
Taste	Bedeutung	Anzeigefeld ¹⁾	Bedeutung
1	neue Ziffer für erste Stelle der Endadresse des umzurechnenden Bereichs	3. E U = 1 0 0 0 Δ	Kommando-Symbol, Symbol für Endadresse, Endadresse (geändert)
8	neue Ziffer für zweite Stelle der Endadresse	3. E U = 1 8 0 0 Δ	- " -
1	neue Ziffer für dritte Stelle der Endadresse	3. E U = 1 8 1 0 Δ	- " -
C	neue Ziffer für vierte Stelle der Endadresse	3. E U = 1 8 1 C Δ	- " -
↓	Kommando-Abschluß	3. Δ	Kommando ausgeführt

1) Δ Stellung der Schreibmarke

Adresse 0200 - 0FFF
bei 2746 (2KB)

4.5 EPROMs beschreiben (Kommando 0 bis 2)

Mit Hilfe der Kommandos 0, 1 und 2 können auch EPROMs beschrieben werden. Allerdings lassen sich auf diese Weise nur gespeicherte 1-Zustände in 0-Zustände umwandeln. Das Beschreiben der EPROMs erfolgt mit einer Geschwindigkeit von etwa 11 Byte pro Sekunde. Im allgemeinen wird ein EPROM-Baustein vor jedem Programmieren mit Hilfe von UV-Licht in den 1-Zustand gebracht, wie es in Kapitel 1.5 bereits erwähnt wurde. Man kann jedoch einen EPROM auch ähnlich wie ein Notizbuch Stück für Stück vollschreiben und bei Bedarf wieder vollkommen löschen. Damit ergibt sich eine Alternative zum Abspeichern auf Magnetband.

Mit dem ECB85 lassen sich die EPROM-Bausteine SAB2758 (1 KByte) und SAB2716 (2 KByte) beschreiben. Sie unterscheiden sich lediglich in der Kapazität, eine Umschaltung des Geräts ist nicht erforderlich. Zum Stecken und Ziehen von EPROMs in Steckplatz Nr. 2 ist die Fassung zu entriegeln, indem der Spannhebel senkrecht gestellt wird. Nach dem Stecken muß durch Verriegeln der Fassung Kontakt hergestellt werden, was durch Umlegen des Spannungshebels erfolgt. Beim Einstecken des Bausteins ist auf die richtige Lage der Markierungskerbe zu achten, wie sie in Bild 3.1-1 angegeben ist.

(oben)

Beim Stecken und Ziehen eines EPROM-Bausteins und Umlegen des Spannungshebels entstehen häufig auf dem Datenbus des ECB85 Störspannungen, die den Ablauf des Monitor-Programms beeinflussen und zum Überschreiben der im Schreib-/Lese-Speicher enthaltenen Information führen können. Dies läßt sich verhindern, wenn man während der Zeit, in der bei eingestecktem Baustein die Fassung nicht verriegelt ist, die R-Taste gedrückt hält. 1

Bevor der EPROM beschrieben wird, ist die Programmierspannung einzuschalten, indem der Programmierschalter in die Stellung PRGR gebracht wird. Nach Ende der Programmierung sollte die Programmierspannung wieder ausgeschaltet werden, um den EPROM nicht irrtümlich falsch zu beschreiben.

Nach diesen Vorbereitungen läßt sich der EPROM genauso wie Schreib-/Lese-Speicher beschreiben. Nur mit dem Unterschied, daß sich ein von 1 auf 0

geändertes Bit auf elektrische Weise nicht mehr zurück ändern läßt. Bei einem Programmierfehler ist vielmehr der ganze Baustein mit UV-Licht zu löschen. Da das Beschreiben einer Speicherzelle im EPROM wesentlich längere Zeit erfordert als im Schreib-/Lese-Speicher, dehnt der ECB85 bei einem schreibenden Zugriff zu dem EPROM-Steckplatz Nr. 2 automatisch den Zyklus. Außerdem blinkt in jedem Schreibzyklus, der sich auf die Adressen des EPROM-Steckplatzes Nr. 2 bezieht, die Programmierlampe kurz auf. Dies ist auch dann der Fall, wenn die Programmierspannung nicht eingeschaltet ist. Durch das im Rahmen der Kommandos durchgeführte Kontrolllesen im Anschluß an das Schreiben wird geprüft und gemeldet, ob die Information richtig im EPROM steht.

Da bei einer fehlerhaft in den EPROM eingegebenen Information der ganze Baustein wieder gelöscht werden muß, empfiehlt sich im allgemeinen nicht die Anwendung des Kommandos 0 zu dessen Beschreiben. Günstiger ist es, wenn zunächst die Information in den Schreib-/Lese-Speicher mit Hilfe des Kommandos 0 eingegeben und anschließend kontrolliert wird. Ist alles in Ordnung, kann der ganze Informationsblock mit Kommando 2 in den EPROM-Bereich übertragen werden. Falls dies erforderlich ist, kann auch ein ganzer Bereich des EPROM durch das Kommando 1 mit einer Konstanten gefüllt werden. Die Anwendung des Kommandos 0 auf den EPROM ist dann zweckmäßig, wenn nur einzelne Speicherzellen beschrieben werden sollen.

Normalerweise wird man ein Programm vor dem Übertragen in den EPROM im Schreib-/Lese-Speicher testen. In vielen Fällen soll das in den EPROM übertragene Programm in einem anderen Adreßbereich laufen. Zu diesem Zweck müssen die Adressen in Sprungbefehlen mit Hilfe des Kommandos 3 umgerechnet werden. Da sich jedoch beim Beschreiben von EPROMs nur 1-Informationen in 0-Informationen umwandeln lassen, ist nach dem Übertragen der Information diese Änderung nicht mehr möglich. Deswegen ist in Zusammenhang mit EPROMs die Reihenfolge der Kommandos 2 und 3 zu vertauschen:

- 1) Es sei angenommen, daß das Programm zunächst im Schreib-/Lese-Speicher zwischen Adresse AA und EA steht und dort ablauffähig ist,

d.h. die für diesen Adreßbereich richtigen Sprungadressen in den Sprungbefehlen hat. Das Programm soll in den EPROM übertragen werden und dort in einem Adreßbereich ab Adresse bA ablauffähig sein.

- 2) Als erstes wird das Programm im Schreib-/Lese-Speicher zwischen Adresse AA und EA mit Hilfe des Kommandos 3 so umgerechnet, als ob es vorangehend zu Adresse bA verschoben worden wäre. Bei der Umrechnung sind im Bereich AA bis EA die Teile auszunehmen, die Daten enthalten.
- 3) Danach wird der Inhalt des Bereichs zwischen AA und EA mit Hilfe des Kommandos 2 zur Adresse bA übertragen.

Will man einen EPROM auf dem ECB85 kopieren, so muß dies in zwei Schritten erfolgen. Zunächst steckt man den EPROM, der die Information enthält, in die Fassung des EPROM-Programmiers und überträgt mit Kommando 2 seinen Inhalt in den Schreib-/Lese-Speicher ab Adresse 1800. Anschließend wird dieser EPROM gezogen und stattdessen ein gelöschter EPROM eingesteckt. Nach Einschalten der Programmiervspannung kann mit Kommando 2 die Information vom Schreib-/Lese-Speicher in den EPROM-Baustein übertragen werden. Hat der zu kopierende Informationsblock eine Länge bis zu 1 KByte - entsprechend dem größten zusammenhängenden Block an Schreib-/Lese-Speicher im ECB85 - kann das Übertragen als Ganzes erfolgen. Bei größeren Blöcken muß man hintereinander in zwei Durchläufen Teilblöcke kopieren. In allen genannten Fällen ist, wie schon erwähnt, beim Stecken und Ziehen des EPROMs die R-Taste zu drücken.

Die auf dem ECB85 in EPROMs geschriebenen Programme kann man, falls die Adressen in Sprungbefehlen auf den vorgesehenen Arbeitsbereich abgestimmt sind, im EPROM-Steckplatz Nr. 1 (nach Ziehen des Monitor-EPROM) bzw. Nr. 2 des ECB85 oder auch auf anderen Mikrocomputern ablaufen lassen.

5. Kommandos zum Testen der Hardware

Zum Testen der externen Anwender-Hardware stehen zwei Kommandos zur Verfügung. Mit dem einen kann ein Datenwort über einen Eingabe-Kanal eingelesen, mit dem anderen ein Datenwort über einen Ausgabe-Kanal ausgegeben werden. Diese Funktionen scheinen auf den ersten Blick viel mit dem Lesen und Beschreiben von Speicherzellen gemeinsam zu haben. Es bestehen jedoch wesentliche Unterschiede in der Anwendung:

- 1) Beim Speicher bezieht sich das Lesen und Schreiben unter der gleichen Adresse immer auf die gleiche Speicherzelle, weshalb Lesen und Schreiben im Kommando 0 zusammengefaßt sind. Bei Ein- und Ausgabe-Kanälen kann sich jedoch das Lesen und Schreiben unter der gleichen Adresse auf unterschiedliche Register beziehen, die in keiner Beziehung zueinander stehen, so daß Lesen und Schreiben zweckmäßigerweise mit unterschiedlichen Kommandos realisiert werden.
- 2) Beim Speicher interessiert oft der Inhalt einer Speicherzelle in Zusammenhang mit den benachbarten Speicherzellen, deshalb werden beim Kommando 0 die Abschlußtasten zum automatischen Inkrementieren und Dekrementieren der Adresse benutzt. Bei Ein- und Ausgabe-Kanälen besteht jedoch zwischen benachbarten Adressen im allgemeinen keine funktionale Beziehung. Es ist bei ihnen oft zweckmäßig, zur Prüfung oder Lokalisierung eines Fehlers einen Schreib- oder Lesevorgang mehrmals auf der gleichen Adresse zu wiederholen. Deshalb bewirken im Rahmen der auf Ein-/Ausgabe-Kanäle bezogenen Kommandos die Abschlußtasten zweckmäßigerweise keine Änderung der Adresse.
- 3) Bei Speicherzellen hat die Wiederholung des Einschreibens und Auslesens der gleichen Adresse keine weiteren Folgen. Dagegen ist zu beachten, daß bei manchen programmierbaren Ein-/Ausgabe-Bausteinen die Wiederholung einer Schreib- oder Lese-Operation Auswirkungen hat. Außerdem muß bei programmierbaren Ein-/Ausgabe-Bausteinen ein Kanal im allgemeinen erst für die vorgesehene Betriebsart programmiert werden, bevor eine

Daten-Ein- oder -Ausgabe wirksam wird. Dies ist z.B. bei den Bausteinen SAB8155 auf dem ECB85 der Fall.

Die Kommandos zum Ausgeben über einen Ausgabe-Kanal und zum Eingeben über einen Eingabe-Kanal werden in den beiden folgenden Kapiteln behandelt. Die Kommandos berücksichtigen die vorstehend aufgeführten Gesichtspunkte.

5.1 Über Ausgabe-Kanal ausgeben (Kommando 4)

Funktion

Das Kommando 4 dient zum Ausgeben von Information über einen Ausgabe-Kanal. Der Ausgabe-Kanal wird durch die Peripherie-Adresse bezeichnet, außerdem ist das auszugebende Datenbyte erforderlich. Diese beiden Größen sind als Parameter einzugeben.

Ablauf

Die Schreibmarke muß sich zu Beginn im Kommandofeld befinden. Es wird das Kommando-Symbol 4 eingegeben und eine beliebige der Abschlußtasten gedrückt. Darauf erscheint im Adreßfeld das Symbol für die Peripherie-Adresse (PA) und im Datenfeld als Vorschlagwert die Zahl 00. Die Schreibmarke steht zu Anfang auf der linken Stelle des Datenfelds.

Der Vorschlagwert der Adresse kann jetzt durch Überschreiben geändert werden. Mit Drücken einer beliebigen der Abschlußtasten wird der Wert als gültig übernommen.

Danach erscheint im Adreßfeld die vorher eingegebene Adresse und im Datenfeld als Vorschlagwert für das auszugebende Datenbyte nach vorangegangenem Rücksetzen die Zahl 00 und später der zuletzt in Kommando 4 oder 5 für das Datenbyte aufgetretene Wert. Die Schreibmarke steht zu Anfang auf der linken Stelle des Datenfelds. Der Vorschlagwert des Datenbytes kann jetzt durch Überschreiben geändert werden. Es kann jetzt auch mit der Schreibmarken-Steuertaste die Schreibmarke in das Adreßfeld verschoben und die Adresse überschrieben werden. Das Überschreiben der Adresse oder des Datenbytes führt noch nicht zur Ausgabe. Erst mit dem Drücken einer der Abschlußtasten wird an die im Adreßfeld stehende Peripherie-Adresse das im Datenfeld stehende Datenbyte genau einmal ausgegeben. Die Adresse bleibt dabei unverändert und die Schreibmarke steht danach immer auf der linken Stelle des Datenfelds.

Das Kommando 4 ist vom Prinzip her endlos und kann nur durch Eingabe eines neuen Kommandos (dazu Schreibmarke in das Kommandofeld verschieben) oder Drücken der R-Taste abgebrochen werden.

Besonderheiten

- 1) Das Monitor-Programm prüft nicht, ob unter der angegebenen Adresse auch tatsächlich ein Ausgabe-Kanal vorhanden ist. Ein Ausgeben an nicht vorhandene Kanäle hat keine Auswirkungen.
- 2) Bei Ausgabe an den vom Monitor-Programm benutzten Ein-/Ausgabe-Baustein SAB8279 führt das Monitor-Programm zwar die Ausgabe aus, stellt danach aber sofort wieder den alten Zustand des Bausteins her, da dieser für den ordnungsgemäßen Ablauf des Monitor-Programms erforderlich ist. Daher haben diese Ausgaben praktisch keine Auswirkung.
- 3) Bei Ausgabe an den von Monitor-Programm für EPROM-Programmierung mitbenutzten Ein-/Ausgabe-Baustein SAB8155 Nr. 1 führt das Monitor-Programm die Ausgabe aus, stellt aber danach den alten Zustand des Bausteins wieder her, wenn mit einem Kommando in eine Speicherzelle im Adreßbereich zwischen 0000 und 0FFF eingeschrieben wird. In diesem Fall hat die Ausgabe durch den Anwender praktisch keine Auswirkung.

Beispiel

Der Peripherie-Baustein SAB8155 Nr. 1 soll so programmiert werden, daß alle Kanäle (A, B, C) für Ausgabe benutzt werden können. Zu diesem Zweck ist an Adresse F8 das Kommandowort OF zu übergeben (siehe Kapitel 8.3). Der dafür erforderliche Ablauf ist in Tabelle 5.1-1 dargestellt.

5.2 Über Eingabe-Kanal eingeben (Kommando 5)

Funktion

Das Kommando 5 dient zum Eingeben von Information über einen Eingabe-Kanal. Der Eingabe-Kanal wird durch die Peripherie-Adresse bezeichnet, die als Parameter einzugeben ist.

Ablauf

Die Schreibmarke muß sich zu Beginn im Kommandofeld befinden. Es wird das Kommando-Symbol 5 eingegeben und eine beliebige der Abschlußtasten gedrückt. Darauf erscheint im Adreßfeld das Symbol für die Peripherie-Adresse (PA) und im Datenfeld als Vorschlagwert die Zahl 00. Die Schreibmarke steht zu Anfang auf der linken Stelle des Datenfelds.

Der Vorschlagwert der Adresse kann jetzt durch Überschreiben geändert werden. Mit Drücken einer beliebigen der Abschlußtasten wird der Wert als gültig übernommen und genau ein Lese-Zugriff ausgeführt.

Danach erscheint im Adreßfeld die vorher eingegebene Adresse und im Datenfeld das ausgelesene Datenwort. Die Adresse kann jetzt beliebig überschrieben werden. Da ein Ändern des Datenworts über das Tastenfeld nicht sinnvoll ist, läßt sich im Rahmen dieses Kommandos die Schreibmarke nicht ins Datenfeld verschieben. Das Überschreiben der Adresse führt noch nicht zum Einlesen. Erst mit dem Drücken einer der Abschlußtasten wird ein Datenwort von der im Adreßfeld stehenden Peripherie-Adresse genau einmal eingelesen und im Datenfeld angezeigt. Die Adresse bleibt dabei unverändert und die Schreibmarke steht danach immer auf der linken Stelle des Adreßfelds.

Das Kommando 5 ist vom Prinzip her endlos und kann nur durch Eingabe eines neuen Kommandos (dazu Schreibmarke in das Kommandofeld verschieben) oder Drücken der R-Taste abgebrochen werden.

Besonderheiten

Das Monitor-Programm prüft nicht, ob unter einer Adresse auch tatsächlich ein Eingabe-Kanal vorhanden ist. Das Einlesen von einem nicht ausgebauten Kanal liefert als Datenbyte einen Zufallswert.

Beispiel

Durch das Rücksetzen werden automatisch im Baustein SAB8155 Nr. 1 alle Kanäle auf Eingabe geschaltet. Deshalb ist es möglich, ohne besondere Vorbereitungen von einem dieser Kanäle ein Datenwort einzulesen.

In dem folgenden Beispiel soll von Kanal A des Bausteins SAB8155 Nr. 1 ein Datenwort eingelesen werden. Die zugehörige Peripherie-Adresse ist F9 (siehe Kapitel 8.3). Der Ablauf ist in Tabelle 5.2-1 dargestellt. Wenn die Peripherie-Anschlüsse offen sind, läßt sich über das zu erwartende Datenbyte nichts voraussagen.

Tabelle 5.2-1

Beispiel für das Kommando 5

Eingabe		Ausgabe	
Taste	Bedeutung	Anzeigefeld ¹⁾	Bedeutung
		(Schreibmarke im Kommandofeld)	Bereitschaft zur Eingabe eines Kommando-Symbols
5	Kommando-Symbol	5. Δ	einggegebenes Kommando-Symbol
↓	Abschluß des Kommando-Symbols	5. P A = 0 0 Δ	Kommando-Symbol, Symbol für Peripherie-Adresse, Vorschlagwert für Peripherie-Adresse
F	neue Ziffer für erste Stelle der Peripherie-Adresse	5. P A = F 0 Δ	Kommando-Symbol, Symbol für Peripherie-Adresse, Peripherie-Adresse (geändert)
9	neue Ziffer für zweite Stelle der Peripherie-Adresse	5. P A = F 9 Δ	- " -
↓	Abschluß der Peripherie-Adresse (Kommando-Ausführung, Lesen der Information)	5. F 9 = Y Y Δ	Kommando-Symbol, Peripherie-Adresse, eingelesenes Datenbyte

- 1) Δ Stellung der Schreibmarke
 YY das im Augenblick der Kommando-Ausführung von den Eingängen eingelesene Datenbyte

6. Kommandos zum Testen der Software

Zum Testen von Programmen stehen vier Kommandos zur Verfügung. Eines davon hat eine Hilfsfunktion und dient dazu, die Schattenregister vor einem Programmablauf mit vorgegebenen Werten zu laden und nach einem Programmablauf zu kontrollieren. Drei Kommandos sind zum Starten von Programmen vorgesehen. Dabei kann man zwischen Programmablauf ohne Unterbrechungspunkte, mit bis zu zwei Unterbrechungspunkten und im Einzelschritt-Verfahren unterscheiden. Diese Kommandos werden in den anschließenden Kapiteln behandelt.

Ein weiteres Kapitel ist dem Programm-Wahlschalter und der I-Taste gewidmet. Diese Elemente haben zwar nicht direkt mit den Kommandos zu tun, sie nehmen jedoch auf den Ablauf des Monitor-Programms Einfluß und haben ähnliche Wirkungen wie Kommandos.

Zu der nachfolgend behandelten Gruppe von Kommandos sind noch einige allgemeine Erläuterungen für die Behandlung von Interrupts zu geben. Zunächst soll der Begriff des Interrupt-Status definiert werden. Darunter ist die in Flip-Flops in der Zentraleinheit gespeicherte Steuer-Information zu verstehen, die sich auf die folgenden beiden Punkte bezieht:

1) Generelle Interrupt-Freigabe/-Sperrung

Dabei handelt es sich um die übergeordnete Festlegung, ob Interrupts grundsätzlich von der Zentraleinheit angenommen werden können oder nicht. Einfluß haben hierbei das Rücksetzen der Zentraleinheit (Sperrung) sowie die Befehle EI (Freigabe) und DI (Sperrung).

2) Individuelle Interrupt-Freigabe/-Sperrung

Dabei handelt es sich um die zusätzliche Möglichkeit, die Interrupts RST5.5, RST6.5 und RST7.5 selektiv zu maskieren (sperrung). Einfluß haben hierbei das Rücksetzen der Zentraleinheit (sperrung) sowie der Befehl SIM (sperrung und freigabe).

Beim ECB85 muß zwischen dem aktuellen Interrupt-Status und dem Schatten-Interrupt-Status unterschieden werden. Unter dem aktuellen Interrupt-Status

ist der tatsächliche Zustand der Zentraleinheit bezüglich Interrupt zu verstehen. Von ihm hängt es ab, ob eine im Augenblick eintreffende Interrupt-Anforderung angenommen wird oder nicht. Der Schatten-Interrupt-Status ist dagegen lediglich im Interrupt-Statuswort des Schattenbereichs abgelegt. Wie allgemein für Information im Schattenbereich gilt (siehe Kapitel 2.5), wird er erst bei Starten eines Anwender-Programms mit bestimmten Kommandos zum aktuellen Interrupt-Status. Nach dem Rücksetzen des ECB85 schreibt das Monitor-Programm in den Schattenbereich "generelle Interrupt-Sperre". Der Schatten-Interrupt-Status wird in Zusammenhang mit bestimmten Kommandos durch die Befehle des Anwender-Programms beeinflusst. Einzelheiten dazu werden in den folgenden Kapiteln gebracht.

6.1 Register auslesen und beschreiben (Kommando 6)

Funktion

Das Kommando 6 dient zum Auslesen von Information aus den Schattenregistern und zum Einschreiben neuer Information. Der Inhalt der Schattenregister erhält seine Bedeutung erst in Zusammenhang mit den Kommandos 7, 8 und 9, wie es in Kapitel 2.5 prinzipiell erläutert wurde. Der Inhalt der Register wird jeweils paarweise behandelt. Mit Hilfe der Abschlußtasten kann auf ein in der vorgegebenen Reihenfolge benachbartes Register übergegangen werden (siehe Kapitel 3.8).

Ablauf

Die Schreibmarke muß sich zu Beginn im Kommandofeld befinden. Es wird das Kommando-Symbol 6 eingegeben und eine beliebige der Abschlußtasten gedrückt. Darauf erscheint im Adreßfeld der Name eines Registerpaares. Nach dem Einschalten der Betriebsspannung ist es ein zufälliges, danach das in Zusammenhang mit den Kommandos 6, 8 oder 9 zuletzt betrachtete Registerpaar. Im Datenfeld steht der Inhalt als 4-stellige Zahl, der im Schattenregister hierzu gespeichert ist. Dieser Inhalt ist nach dem Einschalten ein Zufallswert, später der durch das Monitor-Programm in den Schattenbereich eingeschriebene Wert. Die Schreibmarke befindet sich auf der linken Stelle des Datenfelds. Der Registerinhalt läßt sich in beliebiger Weise überschreiben. Nach jedem Eingeben einer Ziffer wird der neue Wert sofort eingeschrieben, ohne daß zuvor eine der Abschlußtasten gedrückt werden muß. Die Schreibmarke kann nicht in das Adreßfeld gebracht werden, so daß sich dieses auch nicht überschreiben läßt.

Bei diesem Kommando werden die Daten nicht mit den Abschlußtasten beendet. Diese Tasten haben stattdessen eine andere Funktion. Wenn die Schreibmarke im Datenfeld steht, führt ein Drücken der Taste ↓ dazu, daß im Adreßfeld der Name des folgenden Registerpaares (Bild 3.8-4c) und im Datenfeld sein Inhalt angezeigt wird. Mit dem Drücken der Taste ↑ erscheint im Adreßfeld der Name des vorangehenden Registerpaares und im Datenfeld sein Inhalt. Nach jedem Drücken einer Abschlußtaste steht die Schreibmarke wieder auf der linken Stelle des Datenfelds.

Das Kommando 6 ist vom Prinzip her endlos und kann nur durch Eingabe eines neuen Kommandos (dazu Schreibmarke in das Kommandofeld verschieben) oder Drücken der R-Taste abgebrochen werden.

Besonderheiten

keine

Beispiel

Es soll der Programmzähler im Schattenbereich ausgelesen und mit der Zahl 1800 überschrieben werden. Danach ist der Inhalt des Registerpaars IS anzuzeigen. Der Ablauf ist in Tabelle 6.1-1 dargestellt.

Tabelle 6.1-1

Beispiel für das Kommando 6

Eingabe		Ausgabe	
Taste	Bedeutung	Anzeigefeld 1)	Bedeutung
		(Schreibmarke im Kommandofeld)	Bereitschaft zur Eingabe eines Kommando-Symbols
6	Kommando-Symbol	6. Δ	einggegebenes Kommando-Symbol
↓	Abschluß des Kommando-Symbols	6. X X = Y Y Y Y Δ	Kommando-Symbol, Symbol für ein Registerpaar, Inhalt dieses Registerpaars
↓...↓	mehrfacher Wechsel des Registerpaars bis zur Anzeige des Programmzählers	6. P C = Y Y Y Y Δ	Kommando-Symbol, Symbol für Programmzähler, Inhalt des Programmzählers
1	neue Ziffer für erste Stelle des Programmzählers	6. P C = 1 Y Y Y Δ	Kommando-Symbol, Symbol für Programmzähler, Inhalt des Programmzählers (geändert)
8	neue Ziffer für zweite Stelle des Programmzählers	6. P C = 1 8 Y Y Δ	- " -
0	neue Ziffer für dritte Stelle des Programmzählers	6. P C = 1 8 0 Y Δ	- " -
0	neue Ziffer für vierte Stelle des Programmzählers	6. P C = 1 8 0 0 Δ	- " -
↓	Übergang auf das folgende Registerpaar	6. S P = Y Y Y Y Δ	Kommando-Symbol, Symbol für Kellerpeicherzeiger, Inhalt des Kellerpeicherzeigers
↓	- " -	6. I S = Y Y Y Y Δ	Kommando-Symbol, Symbol für Interrupt-Statusregister, Interrupt-Status

- 1) Δ Stellung der Schreibmarke
 XX Symbol eines zufälligen Registerpaars
 YYYY aktueller Inhalt des angezeigten Registerpaars

6.2 Programm ohne Unterbrechungspunkte starten (Kommando 7)

Funktion

Das Kommando 7 dient zum Starten von Anwender-Programmen im Echtzeitbetrieb ohne die Möglichkeit der Eingabe von Unterbrechungspunkten. Die Startadresse ist als Parameter vorzugeben. Die Register werden vor dem Starten von Anwender-Programmen aus dem Schattenbereich geladen.

Ablauf

Für den nachfolgenden normalen Ablauf des Kommandos wird vorausgesetzt, daß als Startadresse die Adresse des Operationscodes eines Befehls gewählt wird und daß das Anwender-Programm so geartet ist, daß nicht in undefinierte Speicherbereiche gesprungen wird.

Die Schreibmarke muß sich zu Beginn im Kommandofeld befinden. Es wird das Kommando-Symbol 7 eingegeben und eine beliebige der Abschlußtasten gedrückt. Darauf erscheint im Adreßfeld das Symbol für Startadresse (SA) und im Datenfeld als Vorschlagwert der im Schattenregister für den Programmzähler stehende Wert. Die Schreibmarke steht auf der linken Stelle des Datenfelds.

Der Wert kann beliebig überschrieben werden. Nach jeder Änderung wird der neue Wert sofort in den Schattenbereich übertragen. Nach Drücken einer beliebigen der Abschlußtasten lädt das Monitor-Programm die Register der Zentraleinheit aus dem Schattenbereich und startet das Programm bei der vorgegebenen Adresse. Auf dem Anzeigefeld bleiben das Symbol für den Programmzähler und der Wert der Startadresse stehen, soweit das Anwender-Programm nicht die Programmierung des Bausteins SAB8279 ändert. Die Schreibmarke verschwindet jedoch. Im Allgemeinen kann ein mit diesem Kommando gestartetes Programm nur mit der R- oder I-Taste abgebrochen werden.

Besonderheiten

- 1) Wie schon erwähnt, wird als Vorschlagwert für die Startadresse der Inhalt des Programmzählers im Schattenbereich genommen. Falls vorher mit Kommando 8 oder 9 (folgende Kapitel) gearbeitet oder ein Programm mit der I-Taste abgebrochen wurde (Kapitel 6.5) und anschließend der Programmzähler im Schattenbereich nicht mit Kommando 6 geändert wurde,

steht dort die Unterbrechungsadresse. Wenn man in diesen Fällen den Vorschlagwert als Startadresse für Kommando 7 akzeptiert, wird das Programm an der Stelle fortgesetzt, an der es unterbrochen wurde. Entsprechend werden auch alle alten Registerinhalte verwendet.

- 2) Nach dem Rücksetzen des ECB85 und während des Arbeitens im Monitor-Programm steht der aktuelle Interrupt-Status auf Sperre (generell und individuell bezogen auf einzelne Interrupt-Eingänge). Außerdem setzt das Monitor-Programm den Schatten-Interrupt-Status beim Rücksetzen auf Sperre. Vor dem Starten eines Anwender-Programms mit Kommando 7 lädt das Monitor-Programm den aktuellen Interrupt-Status aus dem Schatten-Interrupt-Status. Der aktuelle Interrupt-Status ist also dann der gleiche, wie er beim letzten Abbrechen eines Programms mit Kommando 8 oder 9 vorhanden war bzw. der sich aus dem Ändern des Interrupt-Statusworts im Schattenbereich mit Kommando 6 ergibt.
- 3) Wenn ein Anwender-Programm, das mit Kommando 7 gestartet wurde, mit der R-Taste abgebrochen wird, speichert das Monitor-Programm nicht die Registerinhalte im Schattenbereich ab, so daß sie auch nicht mit Kommando 6 kontrolliert werden können.
- 4) Wenn man ein mit Kommando 7 gestartetes Programm unterbrechen und danach die Registerinhalte ansehen sowie den aktuellen Interrupt-Status in den Schattenbereich übertragen will, muß dazu die I-Taste benutzt werden. Bedingung dafür ist jedoch, daß zu dem Zeitpunkt der Interrupt RST7.5, der der I-Taste zugeordnet ist, freigegeben ist. Nähere Einzelheiten dazu werden in Kapitel 6.5 gebracht.
- 5) Wenn als Startadresse eines Anwender-Programms nicht die Adresse des Operationscodes eines Befehls angegeben wird, sondern die Adresse des Datenteils eines Befehls, kann es zur Zerstörung von Programm und Daten im Schreib-/Lese-Speicher kommen. Das gleiche ist möglich, wenn im Laufe der Programm-Abarbeitung in undefinierte Speicherbereiche gesprungen

wird. In diesem Fall hilft nur das Abbrechen mit der R-Taste und erneute Einschreiben der Information.

- 6) Anwender-Programme, die nicht in einer Endlos-Schleife enden, sollten mit einem HLT-Befehl abgeschlossen werden, um undefinierte Sprünge nach ihrer Abarbeitung zu vermeiden.

Beispiel

Das im Monitor-EPROM zwischen Adresse 07E3 und 07FF (Kapitel 3.5) stehende Anwender-Beispiel soll mit Kommando 7 gestartet werden. Die Startadresse ist 07E3. Das Beispiel besteht aus einem Unterprogramm für eine Multiplikation und einem Aufruf-Programm. Das Aufruf-Programm schließt mit einem HLT-Befehl ab, so daß nach dessen Abarbeitung ein definierter Zustand besteht.

Vor dem Starten können die in Tabelle 3.5-3 angegebenen Zahlenwerte für die Register mit Hilfe von Kommando 6 in den Schattenbereich eingeschrieben werden. Den Ablauf des anschließend einzugebenden Kommandos 7 zeigt Tabelle 6.2-1. Wie schon erwähnt, befindet sich die Zentraleinheit nach Abarbeitung des Programms im HALT-Zustand, ohne daß die zu diesem Zeitpunkt vorhandenen Registerinhalte in den Schattenbereich übertragen worden sind. Im Vorgriff auf Kapitel 6.5 soll schon erwähnt werden, daß man jetzt durch Drücken der I-Taste das Programm unterbrechen und die Registerinhalte in den Schattenbereich bringen kann, in dem man sie mit Kommando 6 kontrollieren kann. Das richtige Ergebnis des Programms kann für die vorgegebenen Eingangsgrößen der Tabelle 3.5-3 entnommen werden.

Tabelle 6.2-1

Beispiel für das Kommando 7

Eingabe		Ausgabe	
Taste	Bedeutung	Anzeigefeld 1)	Bedeutung
		(Schreibmarke im Kommandofeld)	Bereitschaft zur Eingabe eines Kommando-Symbols
7	Kommando-Symbol	7. Δ	einggegebenes Kommando-Symbol
↓	Abschluß des Kommando-Symbols	7. S A = Y Y Y Y Δ	Kommando-Symbol, Symbol für Startadresse, Vorschlagwert für Startadresse
0	neue Ziffer für erste Stelle der Startadresse	7. S A = 0 Y Y Y Δ	Kommando-Symbol, Symbol für Startadresse, Startadresse (geändert)
7	neue Ziffer für zweite Stelle der Startadresse	7. S A = 0 7 Y Y Δ	- " -
E	neue Ziffer für dritte Stelle der Startadresse	7. S A = 0 7 E Y Δ	- " -
3	neue Ziffer für vierte Stelle der Startadresse	7. S A = 0 7 E 3 Δ	- " -
↓	Kommando-Ausführung	7. S A = 0 7 E 3	Anwender-Programm bei Adresse 07E3 gestartet

1) Δ Stellung der Schreibmarke
 YYYY Vorschlagwert für Startadresse

6.3 Programm mit Unterbrechungspunkten starten (Kommando 8)

Funktion

Das Kommando 8 dient zum Starten von Anwender-Programmen, wobei die Möglichkeit der Eingabe von bis zu zwei Unterbrechungspunkten besteht. Im Gegensatz zur normalerweise benutzten Realisierung von Unterbrechungspunkten verwendet der ECB85 ein Verfahren, bei dem auch im EPROM-Bereich gesetzte Unterbrechungspunkte wirksam werden. Ein mit Kommando 8 gestartetes Anwender-Programm läuft gegenüber den Echtzeit-Verhältnissen mit wesentlich reduzierter Geschwindigkeit ab, da jeder Befehl einzeln unter Kontrolle des Monitor-Programms ausgeführt wird (siehe Kapitel 2.6).

Als Parameter sind die Startadresse und die Adressen von bis zu zwei Unterbrechungspunkten einzugeben. Die Register der Zentraleinheit werden vor dem Starten aus dem Schattenbereich geladen. Nach Erreichen eines Unterbrechungspunkts wird der Inhalt der Register der Zentraleinheit abgespeichert.

Ablauf

Für den nachfolgend beschriebenen normalen Ablauf des Kommandos wird vorausgesetzt, daß als Adressen für Start- und Unterbrechungspunkte die Adressen von Operationscodes von Befehlen eingegeben werden und daß das Anwender-Programm so geartet ist, daß nicht in undefinierte Speicherbereiche gesprungen wird.

Die Schreibmarke muß sich zu Beginn im Kommandofeld befinden. Es wird das Kommando-Symbol 8 eingegeben und eine beliebige der Abschlußtasten gedrückt. Darauf erscheint im Adreßfeld das Symbol für die Startadresse (SA) und im Datenfeld als Vorschlagwert der im Schattenregister für den Programmzähler stehende Wert. Die Schreibmarke steht auf der linken Stelle des Datenfelds.

Der Wert kann beliebig überschrieben werden. Nach jeder Änderung wird der neue Wert sofort in den Programmzähler im Schattenbereich übertragen. Nach Drücken einer beliebigen der Abschlußtasten erscheint im Adreßfeld das Symbol für die erste Unterbrechungsadresse (H1) und im Datenfeld ein Vorschlagwert. Dieser ist nach dem Rücksetzen 0000 und später der zuletzt mit

Kommando 8 für die erste Unterbrechungsadresse eingegebene Wert. Die Schreibmarke steht auf der linken Stelle des Datenfelds. Nach Überschreiben und Abschließen mit einer beliebigen der Abschlußtasten erscheint im Adreßfeld das Symbol für die zweite Unterbrechungsadresse (H2) und im Datenfeld ein Vorschlagwert. Dieser ist nach dem Rücksetzen 0000 und später der zuletzt mit Kommando 8 für die zweite Unterbrechungsadresse eingegebene Wert. Die Schreibmarke steht auf der linken Stelle des Datenfelds. Nach Überschreiben und Abschließen mit einer beliebigen der Abschlußtasten ist die Parameter-Eingabe beendet.

Das Monitor-Programm nimmt die Schreibmarke weg und gibt auf dem Adreßfeld das Symbol PC aus. Dann lädt es die Register der Zentraleinheit aus dem Schattenbereich und startet das Anwender-Programm. Vor Abarbeitung jedes neuen Befehls erscheint die Adresse des Operationscodes im Datenfeld. Falls einer der gesetzten Unterbrechungspunkte erreicht worden ist, wird das Anwender-Programm abgebrochen. Der augenblickliche Inhalt der Register der Zentraleinheit wird in den Schattenbereich abgespeichert. Auf der Anzeige erscheint im Datenfeld die erreichte Unterbrechungsadresse, und die Schreibmarke steht im Kommandofeld. Wird keiner der Unterbrechungspunkte angetroffen, läuft das Programm immer weiter und läßt sich dann nur mit der R-Taste abbrechen.

Wenn das Anwender-Programm ordnungsgemäß über das Erreichen eines Unterbrechungspunktes abgebrochen wurde, können auf einfache Weise im Rahmen des Kommandos 8 die Register-Inhalte im Schattenbereich ausgelesen und geändert werden. Zu diesem Zweck ist die Schreibmarke mit den Schreibmarken-Steuertasten in das Datenfeld zu verschieben. Jetzt kann der Inhalt des angezeigten Registers kontrolliert und überschrieben und mit den Abschlußtasten ↓ und ↑ auf das folgende bzw. vorangehende Registerpaar übergegangen werden. Die Vorgehensweise ist dabei identisch mit Kommando 6.

Besonderheiten

- 1) Als Vorschlagwert für die Startadresse wird der Inhalt des Programmzählers im Schattenbereich genommen. Falls vorher mit Kommando 8 oder 9

(folgendes Kapitel) gearbeitet oder ein Programm mit der I-Taste abgebrochen wurde (Kapitel 6.5) und anschließend der Programmzähler im Schattenbereich nicht mit Kommando 6 geändert wurde, steht dort die Unterbrechungsadresse. Wenn man in diesen Fällen den Vorschlagwert als Startadresse für Kommando 8 akzeptiert, wird das Programm an der Stelle fortgesetzt, an der es unterbrochen wurde. Entsprechend werden auch alle alten Registerinhalte verwendet.

- 2) Wenn als Startadresse eines Anwender-Programms nicht die Adresse des Operationscodes eines Befehls angegeben wird, sondern die Adresse des Datenteils eines Befehls, kann es zur Zerstörung von Programm und Daten im Schreib-/Lese-Speicher kommen. Das gleiche ist möglich, wenn im Laufe der Programm-Abarbeitung in undefinierte Speicherbereiche gesprungen wird. In diesem Fall hilft nur das Abbrechen mit der R-Taste und erneute Einschreiben der Information.
- 3) Nach dem Starten eines Anwender-Programms mit Kommando 8 wird der erste Befehl ausgeführt, ohne daß das Antreffen einer der Unterbrechungsadressen geprüft wird. Erst ab dem zweiten Befehl ist ein Unterbrechen möglich. Diese Festlegung erlaubt es, bei Unterbrechungspunkten in Programmschleifen auf einfache Weise jeweils einen Schleifendurchlauf auszulösen, wobei die Vorschlagwerte für die Parameter einfach als gültig akzeptiert werden können (beim Unterbrechen wird die Unterbrechungsadresse zum Vorschlagwert für die neue Startadresse).
- 4) Die Möglichkeit zur Vorgabe von zwei Unterbrechungspunkten soll sicherstellen, daß bei Programmverzweigungen unabhängig vom tatsächlichen Programmablauf ein Unterbrechungspunkt wirksam wird.
- 5) Wird nur ein Unterbrechungspunkt benötigt, so kann man beide Unterbrechungsadressen gleich machen oder für eine von ihnen eine Zahl eingeben, die nie zutrifft.

- 6) Bei einer Unterbrechung wird der Befehl, auf den die Unterbrechungsadresse zeigt, nicht mehr ausgeführt.
- 7) Wenn als Unterbrechungsadresse nicht die Adresse des Operationscodes eines Befehls eingegeben wird, wird dieser Unterbrechungspunkt nicht wirksam.
- 8) Während Anwender-Programme, die mit Kommando 8 gestartet wurden, ablaufen, werden keine Interrupts angenommen. Befehle EI und DI im Anwender-Programm haben keinen Einfluß auf den aktuellen Interrupt-Status. Sie bewirken jedoch, daß der Schatten-Interrupt-Status entsprechend geändert wird.
- 9) Wenn ein Anwender-Programm, das mit Kommando 8 gestartet wurde, mit der R-Taste abgebrochen wird, speichert das Monitor-Programm nicht die Registerinhalte im Schattenbereich ab, so daß sie auch nicht mit Kommando 6 kontrolliert werden können.

Beispiel

Vom Anwender-Beispiel im Monitor-EPROM soll das Multiplikations-Unterprogramm (Kapitel 3.5) geprüft werden. Seine Startadresse ist 07EE, der Rücksprungbefehl steht in Adresse 07FF.

Es wird vorausgesetzt, daß die Werte der Eingangsparameter mit Kommando 6 entsprechend Tabelle 3.5-3 geladen sind. Für Kommando 8 wird als Startadresse 07EE eingegeben, für die erste Unterbrechungsadresse 07FF (der dort stehende Befehl RET wird nicht mehr ausgeführt) und für die zweite Unterbrechungsadresse der Vorschlagwert 0000 (dieser Unterbrechungspunkt wird nie erreicht und ist damit unwirksam). Nach Erreichen des Unterbrechungspunktes soll das Ergebnis mit dem in Tabelle 3.5-3 angegebenen Ergebnis verglichen werden. Den Ablauf des Kommandos 8 zeigt Tabelle 6.3-1.

Tabelle 6.3-1

Beispiel für das Kommando 8

Eingabe		Ausgabe	
Taste	Bedeutung	Anzeigefeld 1)	Bedeutung
		(Schreibmarke im Kommandofeld)	Bereitschaft zur Eingabe eines Kommando-Symbols
8	Kommando-Symbol	8. Δ	einggegebenes Kommando-Symbol
↓	Abschluß des Kommando-Symbols	8. S A = Y Y Y Y Δ	Kommando-Symbol, Symbol für Startadresse, Vorschlagwert für Startadresse
0	neue Ziffer für erste Stelle der Startadresse	8. S A = 0 Y Y Y Δ	Kommando-Symbol, Symbol für Startadresse, Startadresse (geändert)
7	neue Ziffer für zweite Stelle der Startadresse	8. S A = 0 7 Y Y Δ	- " -
E	neue Ziffer für dritte Stelle der Startadresse	8. S A = 0 7 E Y Δ	- " -
E	neue Ziffer für vierte Stelle der Startadresse	8. S A = 0 7 E E Δ	- " -
↓	Abschluß der Startadresse	8. H 1 = 0 0 0 0 Δ	Kommando-Symbol, Symbol für erste Unterbrechungsadresse, Vorschlagwert für erste Unterbrechungsadresse
→	Übernahme der ersten Stelle des Vorschlagwertes für erste Unterbrechungsadresse	8. H 1 = 0 0 0 0 Δ	- " -
7	neue Ziffer für zweite Stelle der ersten Unterbrechungsadresse	8. H 1 = 0 7 0 0 Δ	Kommando-Symbol, Symbol für erste Unterbrechungsadresse, erste Unterbrechungsadresse (geändert)

(Fortsetzung)

Tabelle 6.3-1
(Fortsetzung)

Eingabe		Ausgabe	
Taste	Bedeutung	Anzeigefeld 1)	Bedeutung
F	neue Ziffer für dritte Stelle der ersten Unterbrechungsadresse	8. H 1 = 0 7 F 0 Δ	Kommando-Symbol, Symbol für erste Unterbrechungsadresse, erste Unterbrechungsadresse (geändert)
F	neue Ziffer für vierte Stelle der ersten Unterbrechungsadresse	8. H 1 = 0 7 F F Δ	- " -
↓	Abschluß der ersten Unterbrechungsadresse	8. H 2 = 0 0 0 0 Δ	Kommando-Symbol, Symbol für zweite Unterbrechungsadresse, Vorschlagwert für zweite Unterbrechungsadresse
↓	Übernahme des Vorschlagswerts für zweite Unterbrechungsadresse, Kommando-Ausführung (Programm starten)	8. P C = 0 7 F F Δ 2)	Kommando-Symbol, Symbol für Programmzähler, erreichte Unterbrechungsadresse
←	Verschieben der Schreibmarke auf die letzte Stelle des Datenfeldes	8. P C = 0 7 F F Δ	- " -
↑	Wechsel des Registerpaars bis zu H&L	8. H L = Z Z Z Z	Kommando-Symbol, Symbol für Registerpaar H&L, der zuletzt erreichte Inhalt des Registerpaars H&L

- 1) Δ Stellung der Schreibmarke
 YYYY Vorschlagwert für Startadresse
 ZZZZ aktueller Inhalt des angezeigten Registerpaars

- 2) Während der Abarbeitung des Programms wird im Adreßfeld laufend die Adresse des abgearbeiteten Befehls angezeigt. Das Programm ist jedoch in diesem Fall so kurz, daß davon praktisch nichts wahrzunehmen ist.

6.4 Programm im Einzelschritt abarbeiten (Kommando 9)

Funktion

Das Kommando 9 dient zum Abarbeiten von Anwender-Programmen im Einzelschritt-Verfahren. Durch einen Tastendruck wird jeweils die Abarbeitung des nächsten Befehls freigegeben. Vor Ausführung des Befehls werden die Register der Zentraleinheit aus dem Schattenbereich geladen und nach Ausführung des Befehls werden die Registerinhalte der Zentraleinheit im Schattenbereich abgespeichert. Auf diese Weise können alle Registerinhalte geändert und kontrolliert werden. Durch zweckmäßige Festlegung der Vorschlagwerte für die verschiedenen Parameter wurde erreicht, daß sich die Kommandos 6, 7, 8 und 9 lückenlos aneinander anfügen.

Ablauf

Die Schreibmarke muß sich zu Beginn im Kommandofeld befinden. Es wird das Kommando-Symbol 9 eingegeben und eine beliebige der Abschlußtasten gedrückt. Darauf erscheint im Adreßfeld nach dem Einschalten der Name eines zufälligen und später der Name des zuletzt im Zusammenhang mit einem der Kommandos 6, 8 oder 9 behandelten Registerpaares, und im Datenfeld steht der zugehörige Inhalt aus dem Schattenbereich. Die Schreibmarke steht im Kommandofeld.

Will man ein anderes Registerpaar ansehen oder den Inhalt eines Registerpaares ändern, so ist die Schreibmarke mit den Schreibmarken-Steuertasten in das Datenfeld zu verschieben. Jetzt läßt sich der Inhalt des angezeigten Schattenregisters beliebig überschreiben. Nach der Eingabe jeder neuen Ziffer wird der neue Wert sofort in den Schattenbereich übertragen, ohne daß eine Abschlußtaste gedrückt werden muß. Wenn die Schreibmarke im Datenfeld steht, können die Registerpaare mit Hilfe der Tasten ↑ und ↓ wie bei Kommando 8 gewechselt und bei Wunsch geändert werden. Ist man damit fertig, ist die Schreibmarke wieder in das Kommandofeld zu schieben.

Wenn das Kommando-Symbol 9 im Kommandofeld steht und die Schreibmarke auf das Kommandofeld zeigt, wird mit jedem Drücken einer der Abschlußtasten ↑ oder ↓

die Abarbeitung genau eines Befehls ab der im Schattenspeicher stehenden Programmzähler-Adresse freigegeben. Zu diesem Zweck lädt das Monitor-Programm die Register der Zentraleinheit aus dem Schattenbereich, führt einen Befehl aus und speichert den Inhalt der Register der Zentraleinheit wieder im Schattenbereich ab. Außerdem wird anschließend der Name des vorher gewählten Registerpaares und sein neuer Inhalt angezeigt.

Auf die beschriebene Weise kann man ein Anwender-Programm Befehl für Befehl durchgehen und dabei den Inhalt eines frei wählbaren Registerpaares verfolgen. Zwischendurch lassen sich jedoch auch die Inhalte anderer Register kontrollieren und ändern, wenn man die Schreibmarke wieder ins Datenfeld schiebt.

Das Kommando 9 ist vom Prinzip her endlos und kann nur durch Eingabe eines neuen Kommandos (dazu Schreibmarke in das Kommandofeld verschieben) oder Drücken der R-Taste beendet werden.

Besonderheiten

- 1) Die Ausführung des Kommandos 9 durch das Monitor-Programm läuft ähnlich ab wie die des Kommandos 8. Der Unterschied ist lediglich, daß bei Kommando 9 als Unterbrechungsbedingung die vollzogene Ausführung eines Befehls genommen wird.
- 2) Während ein Befehl mit Kommando 9 abgearbeitet wird, werden keine Interrupts angenommen. Befehle EI und DI im Anwender-Programm haben keinen Einfluß auf den aktuellen Interrupt-Status. Sie bewirken jedoch, daß der Schatten-Interrupt-Status entsprechend geändert wird.

Beispiel

Vom Anwender-Beispiel im Monitor-EPR0M sollen die ersten zwei Befehle des Multiplikations-Unterprogramms (Kapitel 3.5) ausgeführt werden. Die Anfangsadresse ist 07EE. Nach Ausführung des ersten Befehls soll das Registerpaar B&C, nach Ausführung des zweiten Befehls das Registerpaar A&F kontrolliert werden. Den Ablauf des Kommandos zeigt Tabelle 6.4-1.

Tabelle 6.4-1

Beispiel für das Kommando 9

Eingabe		Ausgabe	
Taste	Bedeutung	Anzeigefeld ¹⁾	Bedeutung
		(Schreibmarke im Kommandofeld	Bereitschaft zur Eingabe eines Kommando-Symbols
9	Kommando-Symbol	9. Δ	eingegebenes Kommando-Symbol
↓	Abschluß des Kommando-Symbols	9. X X = Y Y Y Y Δ	Kommando-Symbol, Symbol für Registerpaar, Inhalt dieses Registerpaars
→	Verschieben der Schreibmarke in das Datenfeld	9. X X = Y Y Y Y Δ	- " -
↓...↓	Wechsel des Registerpaars bis zum Programmzähler	9. P C = Y Y Y Y Δ	Kommando-Symbol, Symbol für Programmzähler, der zuletzt erreichte Stand des Programmzählers
0	neue Ziffer für erste Stelle des Programmzählers	9. P C = 0 Y Y Y Δ	Kommando-Symbol, Symbol für Programmzähler, Inhalt des Programmzählers (geändert)
7	neue Ziffer für zweite Stelle des Programmzählers	9. P C = 0 7 Y Y Δ	- " -
E	neue Ziffer für dritte Stelle des Programmzählers	9. P C = 0 7 E Y Δ	- " -
E	neue Ziffer für vierte Stelle des Programmzählers	9. P C = 0 7 E E Δ	- " -
→	Verschieben der Schreibmarke ins Kommandofeld	9. P C = 0 7 E E Δ	Kommando-Symbol, Symbol für Programmzähler, Inhalt des Programmzählers
↓	Kommando-Ausführung (einen Anwender-Befehl ausführen)	9. P C = 0 7 F 0 Δ	Kommando-Symbol, Symbol für Programmzähler, Inhalt des Programmzählers (geändert)

(Fortsetzung)

Tabelle 6.4-1
(Fortsetzung)

Eingabe		Ausgabe	
Taste	Bedeutung	Anzeigefeld 1)	Bedeutung
→	Verschieben der Schreibmarke ins Datenfeld	9. P C = 0 7 F 0 Δ	Kommando-Symbol, Symbol des Programmzählers, Inhalt des Programmzählers
↑...↑	Wechsel des Registerpaars bis zu B&C	9. B C = Y Y Y Y Δ	Kommando-Symbol, Symbol für Registerpaar B&C, Inhalt des Registerpaars B&C
↓...↓	Wechsel des Registerpaars bis zu PC	9. P C = 0 7 F 0 Δ	Kommando-Symbol, Symbol für Programmzähler, Inhalt des Programmzählers
←	Verschieben der Schreibmarke ins Kommandofeld	9. P C = 0 7 F 0 Δ	- " -
↓	Kommando-Ausführung (einen Anwender-Befehl ausführen)	9. P C = 0 7 F 1 Δ	Kommando-Symbol, Symbol für Programmzähler, Inhalt des Programmzählers (geändert)
→	Verschieben der Schreibmarke ins Datenfeld	9. P C = 0 7 F 1 Δ	Kommando-Symbol, Symbol für Programmzähler, Inhalt des Programmzählers
↓...↓	Wechsel des Registerpaars bis zu A&F	9. A F = Y Y Y Y Δ	Kommando-Symbol, Symbol für Akkumulator und Zustandsregister, Inhalt dieser Register

- 1) Δ Stellung der Schreibmarke
 XX Symbol für Registerpaar
 YYYY aktueller Inhalt der angezeigten Register

6.5 Programm-Wahlschalter und I-Taste

Der Programm-Wahlschalter und die I-Taste des ECB85 erlauben interessante Betriebsarten, die in diesem Kapitel behandelt werden sollen. Der Programm-Wahlschalter ist ein Schiebeschalter, dessen zwei Stellungen auf der Leiterplatte mit "MON" (Monitor-Programm) und "ANW" (Anwender-Programm) bezeichnet sind. Der Schalter hat zugleich eine Hardware- und eine Software-Funktion. Hardwaremäßig schaltet er in der Stellung MON den Eingang SID und den Ausgang SOD des SAB8085A an den Umsetzer für das Magnetbandgerät, in der Stellung ANW schaltet er die gleichen Anschlüsse des SAB8085A über Treiber auf die Anschlüsse ANWSID und ANWSOD der Anwender-Schnittstelle. Bezüglich der Software-Funktion kann die Zentraleinheit außerdem die Stellung des Schalters abfragen. Das Monitor-Programm des ECB85 führt eine solche Abfrage durch und verzweigt abhängig von der Schalterstellung. Darauf wird noch im folgenden näher eingegangen.

Die I-Taste ist ein Tastenschalter, der an den Interrupt-Eingang RST7.5 des Bausteins SAB8085A angeschlossen ist. Wenn die Zentraleinheit Interrupt für diesen Eingang freigegeben hat, führt ein Drücken der I-Taste zu einem Interrupt-Sprung zu Adresse 3C. Das Monitor-Programm enthält in dieser Adreßlage ein spezielles Interrupt-Programm, das nachfolgend noch näher erläutert wird.

Die Software-Funktion des Programm-Wahlschalters und der I-Taste, die sich durch das Monitor-Programm des ECB85 ergeben, sind in Bild 6.5-1 schematisch dargestellt. Bei Einschalten der Betriebsspannung oder Drücken der R-Taste wird das Programm bei Adresse 0000 gestartet. Dort beginnt das Monitor-Programm, wenn der EPROM-Steckplatz Nr. 1 den Monitor-EPROM enthält. Das Monitor-Programm fragt die Stellung des Programm-Wahlschalters ab. Befindet er sich in der Stellung MON, so wird weiter im Monitor-Programm gearbeitet, und der Anwender kann den ECB85 über die Kommandos steuern. Befindet sich der Schalter jedoch in Stellung ANW, so wird zu Adresse 0800 gesprungen. Dies ist die Anfangsadresse des EPROM-Steckplatzes Nr. 2. Wenn in dieser Fassung ein mit

einem Anwender-Programm bestückter EPROM steckt, wird dieses Programm abgearbeitet.

Neben diesem selbsttätigen Sprung ins Anwender-Programm besteht jedoch auch die Möglichkeit, aus dem Monitor-Programm mit Hilfe des Kommandos 7 zu einer beliebigen Adresse eines Anwender-Programms zu springen.

Es ist auch als umgekehrter Weg der manuell ausgelöste Sprung aus einem beliebigen Programm in das Monitor-Programm möglich. Zu diesem Zweck ist die I-Taste vorgesehen. Falls der Interrupt RST7.5 freigegeben ist, wird bei Drücken der I-Taste das laufende Programm abgebrochen und zu Adresse 3C gesprungen, die im Bereich des Monitor-EPROM liegt. Das Monitor-Programm speichert bei Einsprung auf diese Adresse die Register-Inhalte in den Schattenbereich ab und bringt die beim Auftreten des Interrupt im Kellerspeicher gerettete Unterbrechungsadresse in den Programmzähler im Schattenbereich. Daher ist es möglich, das mit der I-Taste unterbrochene Programm mit Kommando 7, 8 oder 9 fortzusetzen. Ein eventuelles Prellen der I-Taste hat keine Auswirkungen, da nach Annahme des Interrupt dieser automatisch gesperrt wird und in diesem Zustand bleibt, so lange im Monitor-Programm gearbeitet wird.

Das beschriebene Konzept eröffnet interessante Möglichkeiten für den Einsatz in Meß-, Steuer- und Regel-Einrichtungen, die automatisch arbeiten und trotzdem manuell beeinflussbar sein sollen. Es ergeben sich folgende Betriebsarten:

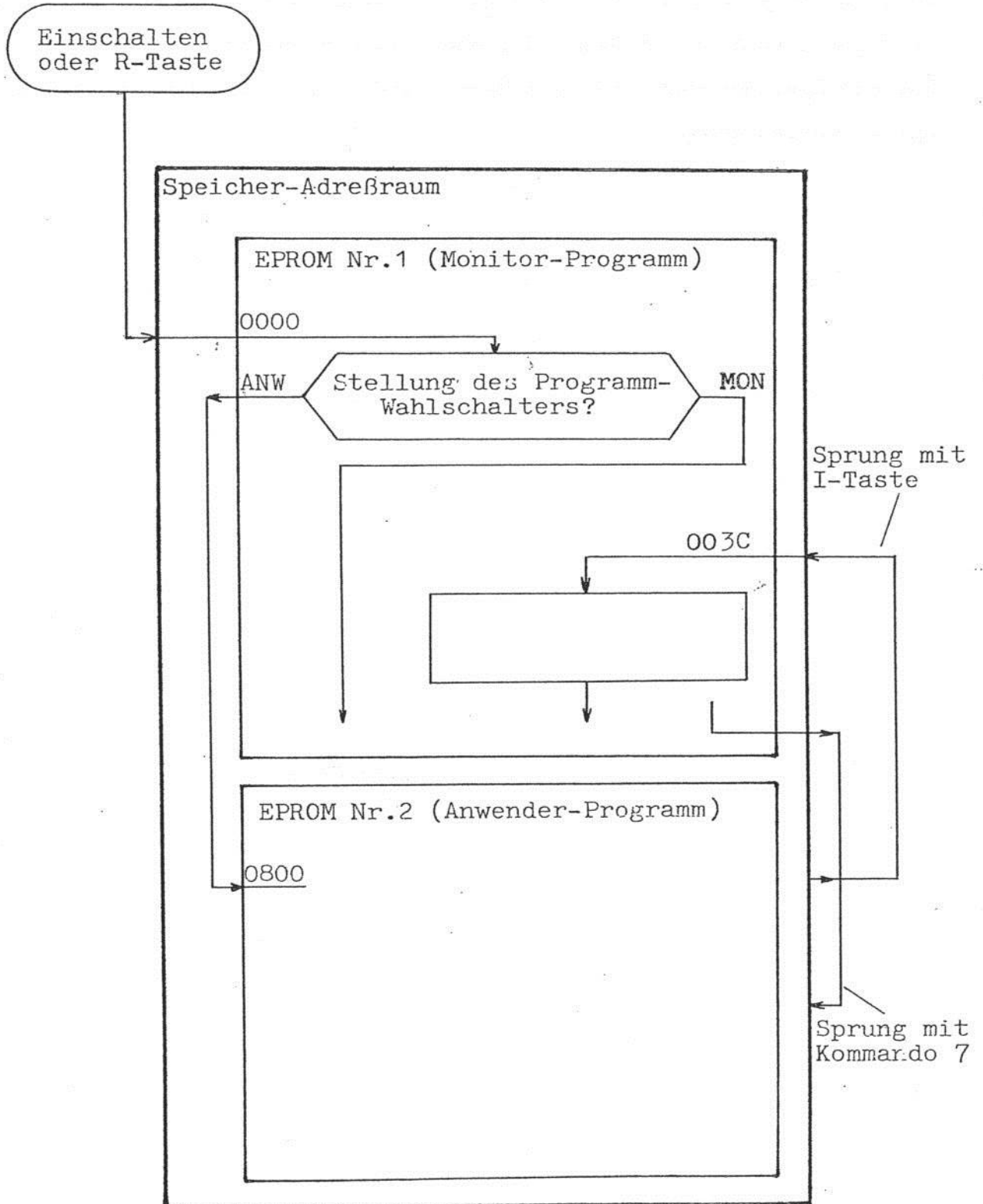
- 1) Manuelles Starten von Anwender-Programmen unter Monitor-Kontrolle: Im ^{ECB} ECB85 muß der Monitor-EPROM stecken, der Programm-Wahlschalter steht auf MON. In dieser Betriebsart wird beim Einschalten der Betriebsspannung oder Drücken der R-Taste in das Monitor-Programm gesprungen. Von hier kann man über die Eingabe eines Kommandos in das Anwender-Programm gelangen. Das Anwender-Programm kann mit Hilfe der I-Taste wieder verlassen werden. Es ist auf diese Weise beliebig oft ein Wechsel zwischen Monitor- und Anwender-Programm möglich.

- 2) Automatisches Starten von Anwender-Programmen unter Monitor-Kontrolle:
Im ECB85 muß der Monitor-EPROM und ein Anwender-EPROM stecken, der Programm-Wahlschalter steht auf ANW. In dieser Betriebsart wird bei Einschalten der Betriebsspannung oder Drücken der R-Taste in das Anwender-Programm gesprungen. Dies ist dann sinnvoll, wenn der ECB85 zur automatischen Steuerung von Maschinen eingesetzt wird. Aus dem Anwender-Programm kann man jederzeit mit Hilfe der I-Taste in das Monitor-Programm springen. Dies ist z.B. erforderlich, wenn Steuer-Parameter geändert werden sollen. Danach kann wieder in das Anwender-Programm zurückgesprungen werden. Es ist auf diese Weise beliebig oft ein Wechsel zwischen Anwender- und Monitor-Programm möglich.

Der Einsatz der I-Taste wurde bereits im Beispiel zu Kommando 7 gezeigt. Als Anwender-Programm diente dort das Anwender-Beispiel von Kapitel 3.5. In diesem Programm wird als Voraussetzung für die Benutzung der I-Taste speziell der Interrupt-Eingang RST7.5 und außerdem allgemein Interrupt freigegeben (Aufruf-Programm ANWRUF). Das Programm-Beispiel endet mit einem HLT-Befehl. Im Beispiel des Kapitel 6.2 wird das Anwender-Beispiel mit Kommando 7 gestartet, und nach Ausführung befindet sich die Zentraleinheit entsprechend dem letzten Befehl im HALT-Zustand. Bei Drücken der I-Taste werden die Inhalte der Register einschließlich aktuellem Interrupt-Status im Schattenbereich abgespeichert und können dann mit Kommando 6 kontrolliert werden.

Bild 6.5-1

Die Möglichkeiten für das Starten des Monitor- und eines Anwender-Programms



7. Kommandos mit Bezug auf das Magnetbandgerät

In Zusammenhang mit dem ECB85 kann ein Magnetbandgerät (Tonbandgerät oder Kassettenrekorder) der unteren Preisklasse als externer Speicher benutzt werden. Der Anschluß wurde bereits in Kapitel 3.3 behandelt. Das Monitor-Programm umfaßt je ein Kommando, über das ein Übertragen von Information aus dem Speicher des ECB85 zum Magnetband bzw. in umgekehrter Richtung veranlaßt werden kann.

7.1 Information auf Magnetband schreiben (Kommando A)

Funktion

Das Kommando A dient zum Schreiben von Information aus dem Speicher des ECB85 auf Magnetband. Der Speicherbereich im ECB85, aus dem die Information übertragen werden soll, wird durch die Anfangsadresse und die Endadresse bezeichnet. Die Anfangsadresse enthält das erste, die Endadresse das letzte übertragene Byte. Anfangs- und Endadresse werden im Rahmen dieses Kommandos als Parameter eingegeben. Die Adressen werden nicht mit auf das Magnetband übertragen.

Ablauf

Für den nachfolgend beschriebenen normalen Ablauf wird vorausgesetzt, daß entsprechend den Angaben in Kapitel 3.3 das Magnetbandgerät an den ECB85 angeschlossen und der Pegel richtig eingestellt ist.

Die Schreibmarke muß sich zu Beginn im Kommandofeld befinden. Es wird das Kommando-Symbol A eingegeben und eine beliebige der Abschlußtasten gedrückt. Darauf erscheint im Adreßfeld das Symbol für die Anfangsadresse des Speicherbereichs (AA) und im Datenfeld als Vorschlagwert die Zahl 0000. Die Schreibmarke steht auf der linken Stelle des Datenfelds.

Der Vorschlagwert der Adresse kann jetzt durch Überschreiben geändert werden. Mit Drücken einer beliebigen der Abschlußtasten wird der Wert als gültig übernommen. Dann erscheint im Adreßfeld das Symbol für die Endadresse des Speicherbereichs (EA) und im Datenfeld als Vorschlagwert die Zahl 0000. Die Schreibmarke steht auf der linken Stelle des Datenfelds. Der Vorschlagwert der Adresse kann durch Überschreiben geändert werden. Vor dem Drücken der Abschlußtaste muß das Magnetbandgerät in der Betriebsart Aufnahme gestartet werden. Mit dem Drücken einer beliebigen der Abschlußtasten des ECB85 wird die Übertragung sofort gestartet.

Während der Übertragung wird laufend im Adreßfeld die gerade bearbeitete Adresse des ECB85 und im Datenfeld die zugehörige Information angezeigt.

Die Schreibmarke ist nicht vorhanden. Nachdem das letzte Byte übertragen ist, befindet sich die Schreibmarke im Kommandofeld, wo das Kommando-Symbol A angezeigt wird. Adreß- und Datenfeld sind dunkel.

Besonderheiten

- 1) Es empfiehlt sich, vor und hinter dem beschriebenen Bandabschnitt ausreichend freien Platz zu lassen, um das Einlesen mit Kommando B zu erleichtern.
- 2) Wenn die eingegebene Anfangsadresse größer ist als die Endadresse, wird die Information aus den beiden Teilbereichen [Anfangsadresse ... FFFF] und [0000 ... Endadresse] ausgegeben.

Beispiel

Es soll der Inhalt des Speichers des ECB85 zwischen der Anfangsadresse 1800 und der Endadresse 1842 auf Magnetband abgespeichert werden. Den Ablauf des Kommandos zeigt Tabelle 7.1-1.

Tabelle 7.1-1

Beispiel für das Kommando A

Eingabe		Ausgabe	
Taste	Bedeutung	Anzeigefeld ¹⁾	Bedeutung
		(Schreibmarke im Kommandofeld)	Bereitschaft zur Eingabe eines Kommando-Symbols
A	Kommando-Symbol	A. Δ	einggegebenes Kommando-Symbol
↓	Abschluß des Kommando-Symbols	A. A A = 0 0 0 0 Δ	Kommando-Symbol, Symbol für Anfangsadresse, Vorschlagwert für Anfangsadresse
1	neue Ziffer für erste Stelle der Anfangsadresse	A. A A = 1 0 0 0 Δ	Kommando-Symbol, Symbol für Anfangsadresse, Anfangsadresse (geändert)
8	neue Ziffer für zweite Stelle der Anfangsadresse	A. A A = 1 8 0 0 Δ	- " -
↓	Abschluß der Anfangsadresse	A. E A = 0 0 0 0 Δ	Kommando-Symbol, Symbol für Endadresse, Vorschlagwert für Endadresse
1	neue Ziffer für erste Stelle der Endadresse	A. E A = 1 0 0 0 Δ	Kommando-Symbol, Symbol für Endadresse, Endadresse (geändert)
8	neue Ziffer für zweite Stelle der Endadresse	A. E A = 1 8 0 0 Δ	- " -
4	neue Ziffer für dritte Stelle der Endadresse	A. E A = 1 8 4 0 Δ	- " -
2	neue Ziffer für vierte Stelle der Endadresse	A. E A = 1 8 4 2 Δ	- " -
Magnetbandgerät in der Betriebsart Aufnahme starten			
↓	Abschluß der Endadresse	A. 1 8 * * = * *	Kommando-Symbol, laufende Anzeige der Speicheradresse und derer Speicherinhalt

(Fortsetzung)

Tabelle 7.1-1
(Fortsetzung)

Eingabe		Ausgabe	
Taste	Bedeutung	Anzeigefeld 1)	Bedeutung
		A. Δ	Kommando-Symbol (Kommando ordnungs- gemäß ausgeführt)
Magnetbandgerät kann abgeschaltet werden			

- 1) Δ Stellung der Schreibmarke
* in dieser Stelle laufende Anzeige

7.2 Information von Magnetband lesen (Kommando B)

Funktion

Das Kommando B dient zum Übertragen von Information, die vorher mit Kommando A auf Magnetband geschrieben worden ist, zurück in den Speicher des ECB85. Da mit Kommando A die Adreßlage nicht mit auf das Magnetband ausgegeben wurde, müssen bei Kommando B die Anfangs- und Endadresse des im ECB85 zu beschreibenden Bereichs als Parameter eingegeben werden. Das Monitor-Programm schreibt nach Starten des Kommandos das erste eintreffende Byte an die Anfangsadresse und das folgende an die jeweils nächste Adresse. Wenn der bezeichnete Speicherbereich vollgeschrieben ist, ist die Ausführung des Kommandos beendet.

Ablauf

Für den nachfolgend beschriebenen normalen Ablauf wird vorausgesetzt, daß entsprechend den Angaben in Kapitel 3.3 das Magnetbandgerät an den ECB85 angeschlossen und der Pegel richtig eingestellt ist. Das Magnetband soll so weit zurückgespult sein, daß der Tonkopf auf dem freien Vorspann vor dem zu lesenden Bandabschnitt steht. Außerdem soll zwischen der angegebenen Anfangs- und Endadresse im ECB85 funktionsfähiger Schreib-/Lese-Speicher vorhanden sein.

Die Schreibmarke muß sich zu Beginn im Kommandofeld befinden. Es wird das Kommando-Symbol B eingegeben und eine beliebige der Abschlußtasten gedrückt. Darauf erscheint im Adreßfeld das Symbol für die Anfangsadresse des zu beschreibenden Speicherbereichs im ECB85 (AA) und im Datenfeld als Vorschlagwert die Zahl 0000. Die Schreibmarke steht auf der linken Stelle des Datenfelds.

Der Vorschlagwert kann jetzt durch Überschreiben geändert werden. Mit Drücken einer beliebigen der Abschlußtasten wird der Wert als gültig übernommen. Dann erscheint im Adreßfeld das Symbol für die Endadresse des zu beschreibenden Speicherbereichs im ECB85 (EA) und im Datenfeld als Vorschlagwert die Zahl 0000. Die Schreibmarke steht auf der linken Stelle des Datenfelds.

Der Vorschlagwert kann durch Überschreiben geändert werden. Mit Drücken einer beliebigen der Abschlußtasten wird der Wert als gültig übernommen. Danach wartet der ECB85 auf Daten von Magnetband. Das Magnetbandgerät ist jetzt in der Betriebsart Wiedergabe zu starten. Während der Datenübertragung wird im Adreßfeld die gerade bearbeitete Adresse des ECB85 und im Datenfeld die zugehörige Information angezeigt. Die Schreibmarke ist während dieser Zeit nicht vorhanden. Nachdem die letzte Adresse in dem bezeichneten Speicherbereich des ECB85 beschrieben ist, befindet sich die Schreibmarke im Kommandofeld, wo das Kommando-Symbol B angezeigt wird. Adreß- und Datenfeld sind dunkel. Das Magnetbandgerät kann jetzt abgeschaltet werden.

Besonderheiten

- 1) Information kann nur unter Adressen im ECB85 eingeschrieben werden, für die funktionsfähiger Schreib-/Lese-Speicher vorhanden ist. Das Monitor-Programm prüft jedesmal durch Kontrolllesen das richtige Einschreiben. Bei erstmaligem Eintreten eines Fehlers wird das Kommando abgebrochen. Im Adreßfeld erscheint dann die Fehleradresse und im Datenfeld die beim Kontrolllesen vorgefundene Information. Die Schreibmarke ist nicht vorhanden. Dieser Zustand kann daher nur mit der R-Taste abgebrochen werden.
- 2) Solange keine neuen Daten vom Magnetbandgerät kommen, bleibt die alte Adresse mit den zugehörigen Daten stehen.
- 3) Wenn die eingegebene Anfangsadresse größer ist als die Endadresse, werden die beiden Teilbereiche des Speichers [Anfangsadresse ... FFFF] und [0000 ... Endadresse] vollgeschrieben. Da jedoch nicht der ganze Bereich mit Schreib-/Lese-Speicher ausgebaut ist, wird das Kontrolllesen mit Sicherheit irgendwann einen Fehler melden.

Beispiel

Ein Anwender-Programm soll vom Magnetband in den ECB85 zwischen Adresse 0 und 1842 eingegeben werden. Den Ablauf des Kommandos zeigt Tabelle 7.2-1.

Tabelle 7.2-1

Beispiel für das Kommando B

Eingabe		Ausgabe	
Taste	Bedeutung	Anzeigefeld ¹⁾	Bedeutung
		(Schreibmarke im Kommandofeld)	Bereitschaft zur Eingabe eines Kommando-Symbols
B	Kommando-Symbol	b. Δ	eingegebenes Kommando-Symbol
↓	Abschluß des Kommando-Symbols	b. A A = 0 0 0 0 Δ	Kommando-Symbol, Symbol für Anfangsadresse, Vorschlagwert für Anfangsadresse
1	neue Ziffer für erste Stelle der Anfangsadresse	b. A A = 1 0 0 0 Δ	Kommando-Symbol, Symbol für Anfangsadresse, Anfangsadresse (geändert)
8	neue Ziffer für zweite Stelle der Anfangsadresse	b. A A = 1 8 0 0 Δ	- " -
↓	Abschluß der Anfangsadresse	b. E A = 0 0 0 0 Δ	Kommando-Symbol, Symbol für Endadresse, Vorschlagwert für Endadresse
1	neue Ziffer für erste Stelle der Endadresse	b. E A = 1 0 0 0 Δ	Kommando-Symbol, Symbol für Endadresse, Endadresse (geändert)
8	neue Ziffer für zweite Stelle der Endadresse	b. E A = 1 8 0 0 Δ	- " -
4	neue Ziffer für dritte Stelle der Endadresse	b. E A = 1 8 4 0 Δ	- " -
2	neue Ziffer für vierte Stelle der Endadresse	b. E A = 1 8 4 2 Δ	- " -
↓	Kommando-Ausführung	b. E A = 1 8 4 2	Kommando-Symbol, Symbol für Endadresse, Endadresse
Magnetbandgerät in der Betriebsart Wiedergabe starten			

(Fortsetzung)

Tabelle 7.2-1
(Fortsetzung)

Eingabe		Ausgabe	
Taste	Bedeutung	Anzeigefeld 1)	Bedeutung
		b. 1 8 * * = * *	Kommando-Symbol, laufende Anzeige der Speicheradres- se und derer Spei- cherinhalt
		b. Δ	Kommando-Symbol, (Kommando ordnungs- gemäß ausgeführt)
Magnetbandgerät kann abgeschaltet werden			

- 1) Δ Stellung der Schreibmarke
* in dieser Stelle laufende Anzeige

8. Peripherie für Anwender-Programme

Der ECB85 muß mit seiner Umgebung Information austauschen, wenn seine Arbeit einen praktischen Sinn haben soll. Falls sein Kommunikationspartner ein Mensch ist, kann die Verbindung am einfachsten über das eingebaute Tasten-/Anzeigefeld erfolgen. Das folgende Kapitel befaßt sich damit, wie diese Möglichkeit in Anwender-Programmen bequem genutzt werden kann. Die Kommunikation mit Maschinen und Geräten erfolgt dagegen im allgemeinen über elektrische Signale. Zu diesem Zweck enthält der ECB85 ein Lochrasterfeld zum Aufbau von Anwender-Schaltungen und ein Anschlußfeld, auf dem die Signale der Anwender-Schnittstelle zur Verfügung stehen. Eines der folgenden Kapitel beschäftigt sich damit, wie die Signale der Schnittstelle angewendet werden. In zwei weiteren Kapiteln wird das Arbeiten mit den Bausteinen SAB8155 und SAB8085A in Hinblick auf die Anwender-Schnittstelle behandelt.

8.1 Arbeiten mit dem Tasten- und Anzeigefeld

Alle Tastenschalter des Tastenfelds mit Ausnahme der R- und I-Taste sowie die acht Stellen des Anzeigefelds sind über den Baustein SAB8279 an den internen Bus angekoppelt. Der Baustein erfordert eine spezielle Programmierung. In diesem Kapitel werden die grundsätzliche Arbeitsweise des SAB8279 sowie die im Monitor-EPROM stehenden Unterprogramme für das Einlesen vom Tastenfeld und das Ausgeben an das Anzeigefeld behandelt, so daß der Anwender mit diesen Kenntnissen in seinen Programmen das Tasten- und Anzeigefeld benutzen kann.

Der Baustein SAB8279 enthält zwei Funktionseinheiten. Die erste dient zu Abfragen der Tasten, die zweite zum Steuern der Anzeige. Der Baustein läßt sich softwaremäßig durch Übergabe von Kommandowörtern auf unterschiedliche Betriebsarten programmieren. Die Zentraleinheit kann jederzeit seinen Zustand über ein Statuswort abfragen. Außerdem werden noch die Datenwörter vom Tastenfeld und die Datenwörter zum Anzeigefeld zwischen SAB8279 und Zentraleinheit übertragen. Die im ECB85 für diese verschiedenen Informationen benutzten Adressen sind in Tabelle 8.1-1 zusammengestellt.

Der Tastenteil des SAB8279 ordnet den gedrückten Tasten bestimmte Tastenwörter zu. Der für den ECB85 geltende Zusammenhang ist in Tabelle 8.1-2 zusammengestellt. Das Tastenwort ist die Information, die für die Zentraleinheit vom Tastenfeld verfügbar ist. Die Verwendung der Tasten in Anwender-Programmen ist beliebig und nicht auf die vom Monitor-Programm vorgesehene Funktion beschränkt. Bei den Tastenwörtern ist zu beachten, daß die beiden höchstwertigen Bits undefiniert sind und deshalb vor Bewertung des Tastenworts ausgeblendet werden müssen.

Der Anzeigeteil des SAB8279 enthält einen Pufferspeicher für jede Anzeigestelle. Die Zentraleinheit kann in diesen Pufferspeicher Anzeigewörter eingeben, und die entsprechende Anzeige bleibt so lange stehen, wie der Inhalt des Pufferspeichers nicht geändert wird. Der für den ECB85 geltende Zusammenhang zwischen den von der Zentraleinheit übergebenen Anzeigewörtern und dem Segmentmuster der Anzeige ist in Bild 8.1-3 gezeigt. Zu beachten ist

dabei, daß die Anzeigewörter rein formal angeben, welche Segmente einer Anzeigestelle hell und dunkel sein sollen. Es läßt sich damit jede gewünschte Kombination von beleuchteten Segmenten herstellen. Die Anzeigestelle, an der das auszugebende Zeichen erscheint, wird als Zahl zwischen 0 und 7 angegeben. Dabei ist der am weitesten links liegenden Anzeigestelle die 0 zugeordnet.

Der Baustein SAB8279 erlaubt vielfältige Betriebsarten, die so komplex sind, daß eine vollständige Beschreibung den Rahmen einer Bedienungsanleitung sprengen würde. Die genaue Arbeitsweise und Anwendung kann dem Datenblatt des Bausteins entnommen werden. Um jedoch dem Anwender ohne diese Kenntnisse die Anwendung des Bausteins zu ermöglichen, enthält der Monitor-EPROM einige Unterprogramme mit einfachen Schnittstellen. Sie werden mit CALL-Befehlen aufgerufen und nehmen dem Anwender die Programmierung des Bausteins ab.

Voraussetzung für diese Unterprogramme ist, daß der SAB8279 vorher durch Software initialisiert worden ist. Die Initialisierung findet automatisch beim Starten des Monitor-Programms statt. Wenn man also den EC885 bei gestecktem Monitor-EPROM einschaltet oder rücktsetzt, wird der SAB8279 richtig initialisiert.

Es ist zu beachten, daß das Monitor-Programm nach Drücken der R- oder I-Taste (z.B. zum Unterbrechen eines Anwender-Programms), bei Erreichen eines im Rahmen von Kommando 8 gesetzten Unterbrechungspunktes oder nach Ausführung eines Befehls im Einzelschritt mit Kommando 9 den Baustein SAB8279 in bestimmter Weise programmiert. Dies ist erforderlich, da der Baustein im Monitor-Programm für den Dialog zwischen Anwender und Mikrocomputer benötigt wird. Eine in einem Anwender-Programm vorgenommene Programmierung wird daher zerstört, wenn zwischenzeitlich Kommandos in der oben angegebenen Art verwendet werden.

Für das Arbeiten mit dem Tasten-/Anzeigefeld über den SAB8279 stehen drei Unterprogramme zur Verfügung. Alle für ihre Anwendung wichtigen Angaben

sind in Tabelle 8.1-4 zusammengestellt. Der dort jeweils angegebene Name ist nur eine Kurzbezeichnung und hat sonst keine Bedeutung. Alle Unterprogramme befinden sich im Monitor-EPROM und werden mit CALL-Befehlen aufgerufen, zu denen in der Tabelle 8.1-4 jeweils auch der Hexadezimalcode angegeben ist. Vor Aufruf eines Unterprogramms müssen die Eingangsparameter in den bezeichneten Registern bereitgestellt werden, und nach Rücksprung aus dem Unterprogramm stehen die Ausgangsparameter in den angegebenen Registern zur Verfügung.

Mit dem Unterprogramm ANWEIN wird das Tastenwort einer gedrückten Taste eingelesen. Mit dem Unterprogramm ANWAZL läßt sich das gesamte Anzeigefeld löschen, was normalerweise am Anfang eines Programmdurchlaufs erforderlich ist. Das Unterprogramm UMW führt eine Codeumwandlung zwischen Tasten- und Anzeigewort durch, so daß man zu einer gedrückten Taste die passende Anzeige erhält. Schließlich erlaubt das Unterprogramm ANWAUS, an eine beliebige Stelle des Anzeigefelds ein gewünschtes Segmentmuster auszugeben.

Die Benutzung des Tasten- und Anzeigefelds in Anwender-Programmen soll an einem einfachen Programm-Beispiel demonstriert werden. Das Programm erwartet in einer Endlos-Folge die Eingabe von Paaren einstelliger Hexadezimal-Zahlen über das Tastenfeld. Dabei bedeutet die erste Zahl eines Paares die Stelle der Anzeige und die zweite Zahl das dort anzuzeigende Zeichen. Nach der Eingabe jedes Wertepaares erfolgt die Anzeige. Der Vorgang kann beliebig oft wiederholt werden. Alle angezeigten Zeichen bleiben erhalten, bis sie durch neue Eingaben überschrieben werden.

Das Programm soll anhand des Ablaufplans in Bild 8.1-5a besprochen werden. Da in dem Programm mit Unterprogrammen gearbeitet werden soll, ist zunächst dem Kellerspeicherzeiger ein Wert zuzuweisen (initialisieren). Daraufhin wird das Anzeigefeld durch Aufruf des Unterprogramms ANWAZL gelöscht. Es folgt eine Endlosschleife, die den Rest des Programms umfaßt. In dieser Schleife wird durch zweimaligen Aufruf des Unterprogramms ANWEIN je ein

Tastenwort für die Anzeigestelle und das anzuzeigende Zeichen eingelesen. Vom ersten Tastenwort für die Anzeigestelle werden nur die letzten drei Bits verwendet, so daß das Drücken der in diesem Zusammenhang nicht sinnvollen Tasten 8...9, A...F, ←, →, ↑ und ↓ so wirkt, als ob eine der Tasten 0...7 gedrückt worden wäre. Das zweite Tastenwort wird durch Aufruf des Unterprogramms UMW in das zugehörige Anzeigewort umcodiert. Dabei führt das Drücken der Tasten ←, →, ↑ und ↓, deren Zeichen auf der Anzeige nicht darstellbar sind, bedingt durch das Unterprogramm UMW, zu Anzeigen im Bereich 0 bis 3. Als letzter Schritt in der Endlosschleife wird durch Aufruf des Unterprogramms ANWAUS die eingegebene Ziffer an die gewünschte Stelle des Anzeigefelds ausgegeben. Das vollständige Programm mit Adressen und Befehlen im Hexadezimalcode ist in Bild 8.1-5b angegeben.

Um das Programm in Betrieb zu nehmen, ist es mit Hilfe von Kommando 0 in den Schreib-/Lese-Speicher ab Adresse 1800 einzugeben und mit Kommando 7 zu starten. Das Anzeigefeld wird daraufhin dunkel. Es sind jetzt über das Tastenfeld zwei Hexadezimalziffern direkt hintereinander einzugeben, wobei die erste Ziffer im Bereich 0...7 und die zweite im Bereich 0...F liegen sollte. Danach erscheint an einer Stelle des Anzeigefelds eine Ziffer. Der Vorgang kann beliebig oft wiederholt werden. Angezeigte Ziffern bleiben so lange stehen, bis sie überschrieben werden.

Tabelle 8.1-1

Adressen des Bausteins SAB8279

Adresse	Information	
	Ausgabe zum SAB8279	Eingabe vom SAB8279
EE	Datenwort	Datenwort
EF	Kommandowort	Statuswort

Tabelle 8.1-2

Zusammenhang zwischen Taste und Tastenwort

Taste	Tastenwort*)	Taste	Tastenwort*)
0	00	A	0A
1	01	B	0B
2	02	C	0C
3	03	D	0D
4	04	E	0E
5	05	F	0F
6	06	←	10
7	07	→	11
8	08	↑	12
9	09	↓	13

*) Hexadezimal-Zahl. Die beiden höchstwertigen Bits des Tastenworts sind undefiniert; in dieser Tabelle wird vorausgesetzt, daß diese beiden Bits durch Software ausgeblendet sind (Befehl ANI 3FH).

Bild 8.1-3

Zusammenhang zwischen Anzeigewort und Segmentmuster

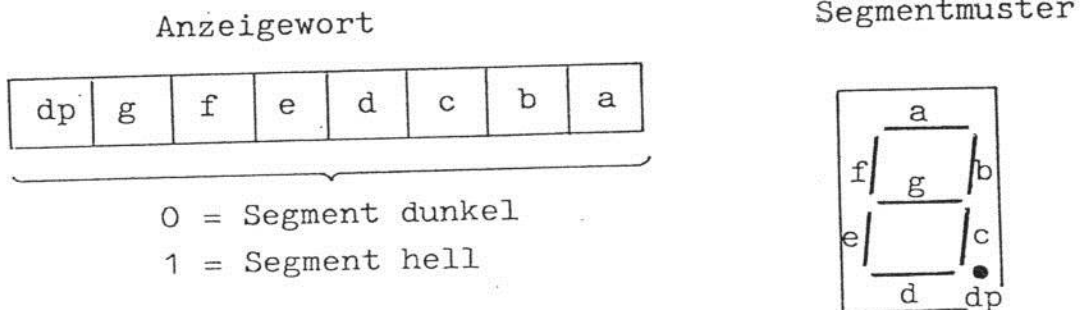


Tabelle 8.1-4

Unterprogramme im Monitor-EPROM zum Arbeiten mit dem Tasten-/Anzeigefeld

a) Einlesen von Tastenfeld

Name: ANWEIN

Ort: Monitor-EPROM, Adresse 07D1 bis 07E2

Aufruf:CALL 07D1 / CD D1 07

Funktion:Es wird gewartet, bis nach dem Aufruf dieses Unterprogramms das erste Mal eine Taste gedrückt wird. Das zugehörige Tastenwort(Tabelle 8.1-2) wird eingelesen und steht nach Rücksprung in einem Register.

Eingangsparameter:keine

Ausgangsparameter:Tastenswort in Register A

b) Löschen des Anzeigefelds

Name: ANWAZL

Ort: Monitor-EPROM, Adresse 07BC bis 07C6

Aufruf:CALL 07BC / CD BC 07

Funktion:Das Anzeigefeld wird gelöscht, so daß es danach dunkel ist.

Eingangsparameter:keine

Ausgangsparameter:keine

c) Umwandeln eines Tastenworts in das zugehörige Anzeigewort

Name: UMW

Ort: Monitor-EPROM, Adresse 05C9 bis 05E7

Aufruf:CALL 05C9 / CD C9 05

Funktion:Es werden die für die Hexadezimaltasten festgelegten Tastenwörter(Tabelle 8.1-2) umgewandelt in Anzeigewörter, so daß die zu den Tasten gehörenden Zeichen auf der Anzeige dargestellt werden können(entsprechend Bild 8.1-3).Von den eingegebenen Tastenwörtern werden nur die letzten 4 Bit verwendet.

Eingangsparameter:Tastenswort in Register A

Ausgangsparameter:Anzeigewort in Register A

(Fortsetzung

Tabelle 8.1-4
(Fortsetzung)

d) Ausgabe an Anzeigefeld

Name: ANWAUS

Ort: Monitor-EPROM, Adresse 07C7 bis 07D0

Aufruf: CALL 07C7 / CD C7 07

Funktion: Ausgabe des durch ein Anzeigewort (Bild 8.1-3) beschriebenen Zeichens an eine beliebige Anzeigestelle. Der Ort wird durch die Anzeigestelle bezeichnet, die bei der Anzeige links mit 0 anfängt und nach rechts wächst.

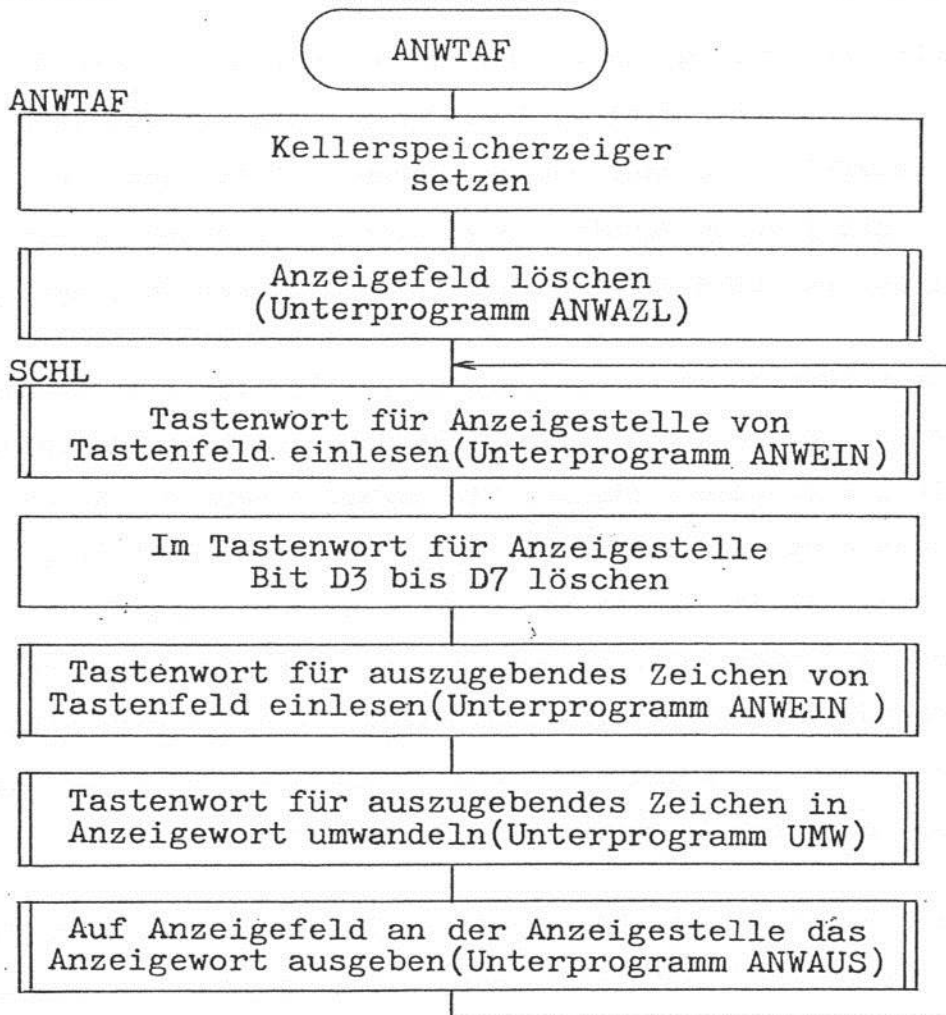
Eingangsparameter: Anzeigewort in Register A
Anzeigestelle in Register B

Ausgangsparameter: keine

Bild 8.1-5

Beispiel für die Anwendung der Tasten-/Anzeige-Unterprogramme

a) Ablaufplan



b) Programm

Hexadezi- mal-Adresse	Hexadezi- mal-Befehl	Assembler-Befehl	Kommentar
1800	31FF1B	ANWTAF: LXI SP, 1BFFH	;Kellerspeicherzei- ger setzen
1803	CDBC07	CALL ANWAZL	;Anzeigefeld löschen ;(Unterprogr. ANWAZL)
1806	CDD107	SCHL: CALL ANWEIN	;Tastenwort für Anzei- gestelle von Tasten- feld einlesen (Unter- programm ANWEIN)
1809	E607	ANI 07H	;Im Tastenwort für An- zeigestelle Bit D3 bis D7 löschen
180B	47	MOV B, A	;
180C	CDD107	CALL ANWEIN	;Tastenwort für auszu- gebendes Zeichen von Tastenfeld einlesen ;(Unterprogramm ANWEIN)
180F	CDC905 3b48	CALL UMW	;Tastenwort für auszu- gebendes Zeichen in Anzeigewort umwandeln ;(Unterprogramm UMW)
1812	CDC707	CALL ANWAUS	;Auf Anzeigefeld an der Anzeigestelle das An- zeigewort ausgeben ;(Unterprogramm ANWAUS)
1815	C30618	JMP SCHL	;Schleife

8.2 Das Anschließen von Anwender-Hardware

Der ECB85 enthält zum Verbinden mit der Peripherie ein Anschlußfeld mit der Anwender-Schnittstelle. Anwender-Schaltungen können direkt auf dem Lochrasterfeld aufgebaut werden. Durch Einlöten von Steckern kann auch eine leicht lösbare Verbindung zu extern liegenden Schaltungen hergestellt werden. Bild 8.2-1 zeigt die Lage dieser Bereiche auf dem ECB85.

Das Anschlußfeld der Anwender-Schnittstelle ist eine Gruppe von 3 x 32 Löt-löchern, an denen die für die Verwendung durch den Anwender vorgesehenen Signale zur Verfügung stehen. Die Bezeichnungen der Anschlüsse bestehen jeweils aus einem Buchstaben (a, b, c) und einer Zahl (1...32) und gehen in Bild 8.2-1 hervor. Von diesen Löchern kann man über Drähte eine Verbindung zum Lochrasterfeld herstellen. Es ist jedoch auch möglich, an diesen Platz eine 96-polige Federleiste (z.B. Siemens C 42334-A192-A503) einzulöten und eine Anwender-Baugruppe senkrecht einzustecken oder über ein Kabel eine Verbindung zu extern liegenden Schaltungen herzustellen.

Auf dem Lochrasterfeld können direkt Versuchs- bzw. Interface-Schaltungen aufgebaut werden. Seitlich stehen, wie in Bild 8.2-1 angegeben, 0V (Masse) und +5V an breiten Leiterbahnen zur Verfügung. An der außenliegenden Schmalseite des Lochrasterfelds befindet sich eine Gruppe von Löt-löchern, die im Bild als "Platz für Peripheriestecker" und "Anschlußfeld für Peripheriestecker" bezeichnet sind. In den Platz für den Peripheriestecker kann eine 96-polige Messerleiste (z.B. Siemens C 42334-A191-A501) eingelötet werden. Jeder Anschluß der Messerleiste ist dann mit einem Lötloch im Anschlußfeld für den Peripheriestecker verbunden, über das mit einem Draht die Verbindung zum Lochrasterfeld hergestellt werden kann. Der Peripheriestecker ist dazu bestimmt, die Anwender-Schaltungen auf dem Lochrasterfeld mit extern liegenden Geräten zu verbinden.

Nun zurück zum Anschlußfeld der Anwender-Schnittstelle. Die Signale und die Bezeichnung der zugeordneten Löt-löcher sind in Tabelle 8.2-2 zusammengestellt. Die Signale können in Versorgungsspannungen, Bussignale und Peripheriesignale eingeteilt werden. Zur Gruppe Versorgungsspannungen gehören

zwei Signale:

- 1) 0V - Bezugspunkt für die Spannung +5V und alle Signale
- 2) +5V - Anschluß zum Speisen der Anwender-Schaltungen aus der gemeinsamen Quelle.

Die Bus-Signale erlauben die Erweiterung des ECB85 durch weitere Speicher- und Peripherie-Bausteine. Auf die genaue Funktion, die Pegel und die Belastbarkeit sowie die Anwendung kann im Rahmen dieser Bedienungsanleitung nicht eingegangen werden. Wer die Bussignale verwenden möchte, sei auf den ausführlichen Stromlaufplan des ECB85 in Kapitel 9.1 und die Datenblätter der betroffenen Bausteine verwiesen. Im folgenden werden diese Signale nur aufgezählt und kurz charakterisiert:

- 1) A0 bis A15 - Ausgänge für Adressen.
- 2) D0 bis D7 - Ein-/Ausgänge für Daten.
- 3) ALE - Ausgang für Multiplex-Signal. Bei H-Pegel an diesem Ausgang handelt es sich bei der Information auf dem Datenbus um das niederwertige Adreßbyte, bei L-Pegel um das Datenbyte.
- 4) $\overline{IO/M}$ - Ausgang für Adressierungsmodus. Bei L-Pegel läuft ein Speicher-Zyklus, bei H-Pegel ein Ein-/Ausgabe-Zyklus.
- 5) \overline{WR} , \overline{RD} - Ausgänge für Schreib- und Leseimpuls.
- 6) ANWRDY - Eingang für Zyklus-Sperrung-/Freigabe. Dient zum Anpassen der Geschwindigkeit der Zentraleinheit an langsamere Speicher- und Ein-/Ausgabe-Bausteine.
- 7) \overline{RESIN} , RESOUT - Eingang für das Rücksetzsignal und Ausgang für das davon abgeleitete Ausgangssignal.
- 9) INTR, \overline{INTA} - Eingang und Ausgang zum Erweitern des Interrupt-Systems.

- 9) HOLD, HLDA - Eingang und Ausgang für direkten Speicherzugriff.
- 10) SO, S1 - Ausgänge für den Status der Zentraleinheit.

Die Peripherie-Signale erlauben die direkte Ansteuerung von Anwenderschaltungen. Diese Signale sind für den Anwender von besonderer Bedeutung und werden in den folgenden Kapiteln noch genau behandelt. An dieser Stelle sollen sie schon kurz charakterisiert werden:

- 1) P1A0 bis P1C5 (Grundausstattung) und P2A0 bis P2C5 (bei nachbestücktem zweiten Baustein SAB8155) - parallele Ein-/Ausgänge.
- 2) $\overline{\text{TIMO-1}}$, CLK (Grundausstattung) und $\overline{\text{TIMO-2}}$, $\overline{\text{TIMO-2}}$ (bei nachbestücktem zweiten Baustein SAB8155) - Zähler-Ein- und -Ausgänge sowie Taktausgang.
- 3) ANWSID, ANWSOD - Einzel-Ein- und -Ausgang des SAB8085A.
- 4) $\overline{\text{RST5.5}}$, $\overline{\text{RST6.5}}$, TRAP - direkte Interrupt-Eingänge des SAB8085A.

An dieser Stelle soll allgemein darauf hingewiesen werden, daß beim Löten an den Anschlüssen der Anwender-Schnittstelle wegen der auf dem ECB85 verwendeten MOS-Bausteine besondere Regeln zu beachten sind. Die MOS-Bausteine können durch unkontrolliert angelegte elektrische Spannungen leicht zerstört werden. Schon das Berühren der Anschlußstifte oder der Kunststoffgehäuse kann zu Zerstörung durch elektrostatische Ladungen führen. Um dies zu vermeiden, werden folgende Vorsichtsmaßnahmen empfohlen:

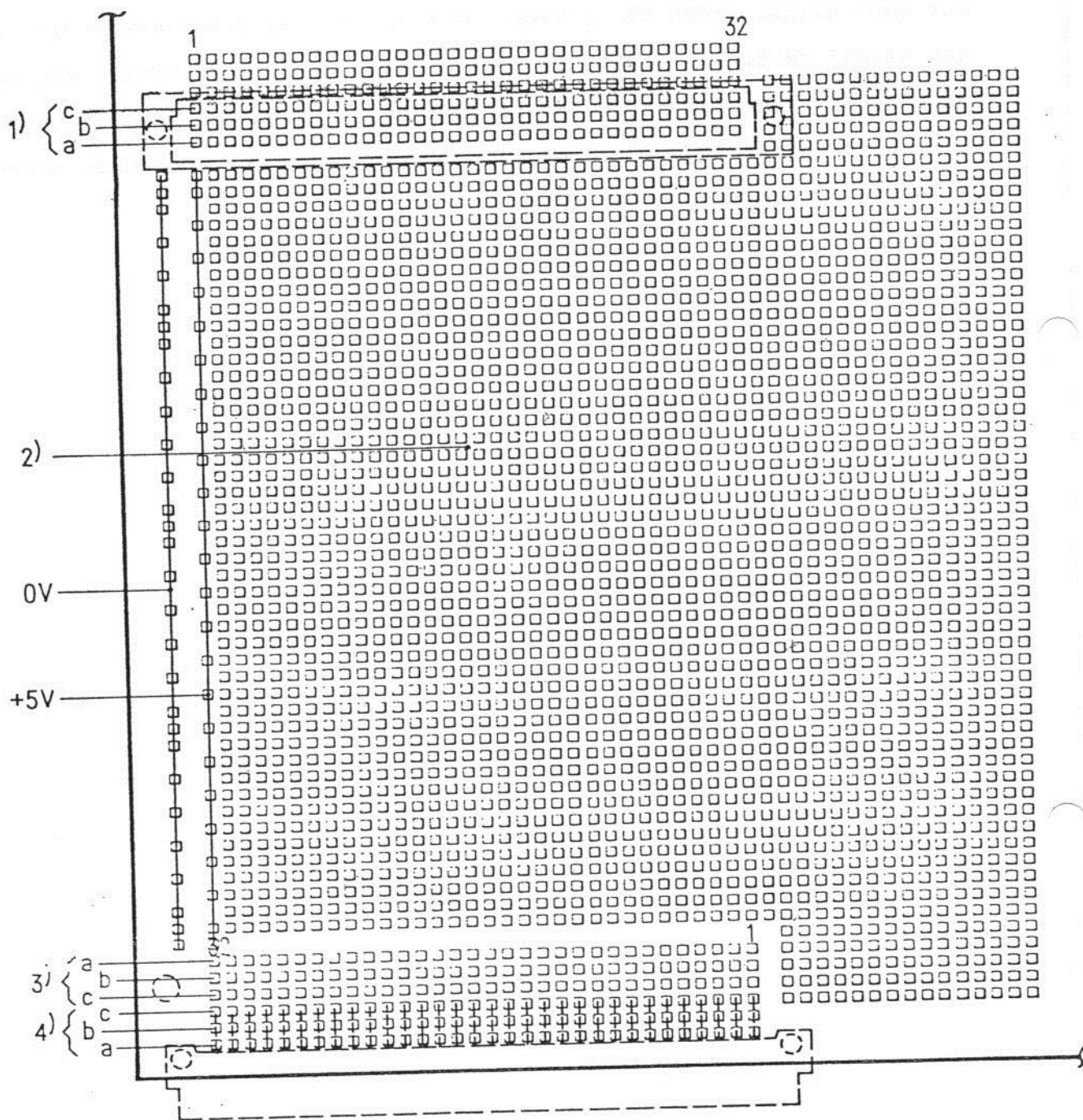
- 1) Bevor Sie leitende Teile oder Gehäuse von MOS-Bausteinen anfassen, bringen Sie Ihren Körper durch kurzes Berühren der Metallteile an der OV-Buchse mit der Hand auf gleiches Potential mit dem ECB85. Dies ist besonders wichtig nach Herumgehen auf Kunststoff-Teppichen, was zum elektrischen Aufladen des Körpers führen kann.
- 2) Verwenden Sie zum Löten entweder LötKolben ohne Trenntrafo, die geerdet sind, oder LötKolben mit Trenntrafo, die eine geerdete Schirmwicklung haben. Berühren Sie vor dem Löten mit der Lötspitze die Metallteile

der OV-Buchse, um LötKolben und ECB85 auf das gleiche Potential zu bringen.

Wer ganz sicher gehen will, kann zum Löten an der Anwender-Schnittstelle die beiden Bausteine SAB8155 ziehen. Dies geschieht am besten so, daß man die Klinge eines Schraubenziehers zwischen Fassung und Baustein schiebt und durch vorsichtiges Drehen und Weiterschieben des Schraubenziehers den Baustein gleichmäßig herausdrückt.

Bild 8.2-1

Lager und Bezeichnung der Anschlüsse der Anwender-Schnittstelle
und des Lochrasterfeldes



- 1) Anschlußfeld der Anwender-Schnittstelle
- 2) Lochrasterfeld
- 3) Anschlußfeld für Peripheriestecker
- 4) Platz für Peripheriestecker

Tabelle 8.2-2

Signale der Anwender-Schnittstelle

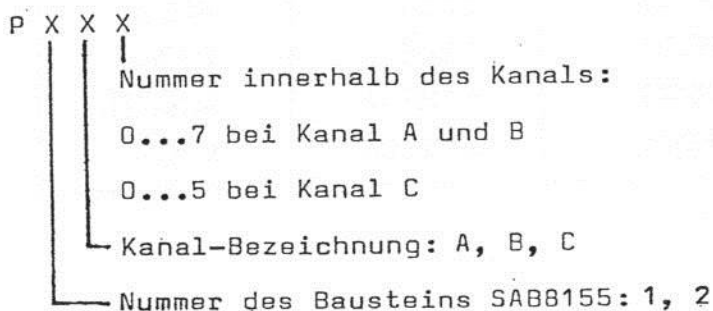
	a	b	c
1	0V	0V	0V
2	+5V	+5V	+5V
3	$\overline{\text{RST5.5}}$	P1A0	A1
4	D7	P1A1	D6
5	D5	P1A2	P1A5
6	A2	P1A3	P1A4
7	A3	D4	P1A6
8	A4	D3	P1A7
9	A5	D2	P1B0
10	P1B1	D1	D0
11	A6	P1B2	ALE
12	A7	P1B3	$\overline{\text{WR}}$
13	A0	P1B4	$\overline{\text{RST6.5}}$
14	$\overline{\text{RD}}$	P1B5	$\overline{\text{TIMO-1}}$
15	$\text{IO}/\overline{\text{M}}$	P1B6	P1C5
16	A8	P1B7	$\overline{\text{TRAP}}$
17	RESOUT	P1C0	INTR
18	CLK	P1C1	HOLD
19	P1C4	P1C2	P1C3
20	$\overline{\text{INTA}}$	S1	A9
21	P2A0	ANWRDY	A10
22	P2A1	$\overline{\text{TIMO-2}}$	A11
23	P2A2	A13	A12
24	P2A3	A15	A14
25	P2A4	S0	P2A5
26	P2A6	$\overline{\text{RESIN}}$	P2A7
27	P2B1	P2B0	P2B2
28	P2B3	P2C5	P2B4
29	P2B6	P2C4	P2B7
30	HLDA	P2B5	P2C0
31	P2C1	TIMI-2	ANWSID
32	P2C3	ANWSOD	P2C2

8.3 Arbeiten mit dem Baustein SAB8155

Der ECB85 enthält zwei Fassungen für Bausteine SAB8155, von denen die eine bei Lieferung bestückt ist und die andere nachbestückt werden kann. Die Bausteine werden durch die Nummern 1 und 2 unterschieden. Jeder Baustein umfaßt einen Schreib-/Lese-Speicher-, Ein-/Ausgabe- und Zähler-Teil. Der Schreib-/Lese-Speicher dieser Bausteine hat keine Anschlüsse zur Anwender-Schnittstelle und wurde bereits in Kapitel 3.4 behandelt. Wie Bild 8.3-1a zeigt, sind die 22 Ein-/Ausgabe-Anschlüsse (P1A0 bis P1C5, P2A0 bis P2C5) jedes Bausteins auf Anschlüsse der Anwender-Schnittstelle geführt. Der Eingang des Zählers im Baustein SAB8155 Nr. 1 liegt fest am internen Takt CLK von 3,072 MHz. Dieser Takt ist auch an der Anwender-Schnittstelle verfügbar. Der Ausgang des Zählers im Baustein SAB8155 Nr. 1 sowie Ein- und Ausgang des Zählers im Baustein SAB8155 Nr. 2 liegen ebenfalls auf Anschlüssen der Anwender-Schnittstelle. Die Pegel und Belastungen bzw. Belastbarkeiten aller genannten Ein- und Ausgänge sind in Bild 8.3-1b zusammengestellt.

Der Baustein SAB8155 enthält für Ein-/Ausgabe- und Zähler-Teil gemeinsam ein 8-Bit-Kommandoregister, das durch Übergabe eines Kommandoworts aus der Zentraleinheit geladen wird und die Funktion des Bausteins festlegt. Der Zustand beider Teile läßt sich jederzeit durch Einlesen eines gemeinsamen Statusworts in die Zentraleinheit abfragen. Außerdem sind dem Ein-/Ausgabe-Teil gesondert noch 3 und dem Zähler-Teil noch 2 Ein-/Ausgabe-Adressen zugeordnet. Die Adressen sind in Tabelle 8.3-2 angegeben.

Der Ein-/Ausgabe-Teil des Bausteins SAB8155 hat 22 periphere Anschlüsse. Sie sind in Gruppen zu sogenannten Kanälen zusammengefaßt: Kanal A und B bestehen aus je 8 und Kanal C aus 6 Anschlüssen. Die Signale sind folgendermaßen bezeichnet:



Die Kanäle können durch Steuerworte so geschaltet werden, daß sie Ein- oder Ausgänge sind. Die Kanäle A und B lassen sich außerdem ohne und mit Rückmeldung betreiben. Bei Betrieb mit Rückmeldung (Handshake-Betrieb) sind jedem Kanal noch 3 Steuersignale zugeordnet, wofür bei Wahl dieser Betriebsart jeweils 3 Anschlüsse von Kanal C benutzt werden. Für die Kanäle A und B ergeben sich somit folgende 4 Betriebsarten:

1) Eingabe ohne Rückmeldung:

Die Zentraleinheit liest den jeweiligen Augenblickswert an den peripheren Anschlüssen ein.

2) Eingabe mit Rückmeldung:

Die Zentraleinheit liest das Datenwort aus einem dem Kanal zugeordneten Zwischenregister im Baustein ein. In dieses Zwischenregister werden die an den peripheren Anschlüssen liegenden Daten mit dem Steuer-Eingangssignal "Daten übernehmen" übergeben. Der Peripherie steht ein Steuer-Ausgangssignal "Register voll" zur Verfügung, welches anzeigt, ob ein übergebenes Datenwort noch nicht von der Zentraleinheit aus dem Zwischenregister abgeholt ist. Außerdem gibt es noch das Steuer-Ausgangssignal "Interrupt anfordern", mit dessen Hilfe eine Programm-Unterbrechung angefordert werden kann, sobald die Peripherie ein Datenwort in das Zwischenregister übergeben hat.

3) Ausgabe ohne Rückmeldung:

Die Zentraleinheit übergibt das Datenwort in ein dem Kanal zugeordnetes Zwischenregister im Baustein, in dem es der Peripherie bis zu einer neuen Datenausgabe zur Verfügung steht.

4) Ausgabe mit Rückmeldung:

Die Funktion ist wie bei Ausgabe ohne Rückmeldung, doch gibt es zusätzlich noch 3 Steuersignale. Über das Steuer-Eingangssignal "Daten abgeholt" teilt die Peripherie mit, daß sie ein von der Zentraleinheit in das Zwischenregister übergebenes Datenwort ausgewertet hat und nicht mehr benötigt

Mit dem Steuer-Ausgangssignal "Daten bereit" wird der Peripherie mitgeteilt, daß ein neues, noch nicht von der Peripherie ausgewertetes Datenwort im Zwischenregister zur Verfügung steht. Außerdem gibt es noch das Steuer-Ausgangssignal "Interrupt anfordern", mit dessen Hilfe eine Programm-Unterbrechung angefordert werden kann, sobald die Peripherie ein Datenwort mit dem Steuer-Eingangssignal "Daten abgeholt" quittiert hat.

Vor dem Datenverkehr zwischen Zentraleinheit und Ein-/Ausgabe-Teil muß die Betriebsart der Kanäle durch Übergabe eines Kommandoworts festgelegt werden. In Bild 8.3-3 ist der Teil des Kommandoworts erläutert, der für die Betriebsart des Ein-/Ausgabe-Teils maßgeblich ist. Mit Bit D0 und D1 wird die Datenrichtung der Kanäle A und B festgelegt. Bit D2 und D3 bestimmen darüber, ob Kanal A und B ohne oder mit Rückmeldung betrieben und für welche Datenrichtung die eventuell von Kanal C noch übrigbleibenden Anschlüsse verwendet werden sollen. Mit Bit D4 und D5 wird, falls für den jeweiligen Kanal Betrieb mit Rückmeldung gewählt wurde, festgelegt, ob die Steuer-Ausgänge "Interrupt anfordern" deaktiviert bleiben (d.h. dauernd L-Pegel haben) oder aktiviert werden (d.h. Interrupt anfordern) sollen. In Tabelle 8.3-4 sind für die Bits D0 bis D3 alle möglichen Kombinationen und die damit verbundenen Betriebsarten zusammengestellt. Die Verwendung der Anschlüsse des Kanals C für Steuersignale und die Zeitbeziehungen gehen auf Bild 8.3-5 hervor. In beiden Darstellungen sind von den Anschluß-Bezeichnungen (z.B. P1A0) nur die letzten zwei Zeichen (z.B. A0) angegeben, da die Angaben in gleicher Weise für beide Bausteine SAB8155 gelten.

Der Datenverkehr zwischen Zentraleinheit und Ein-/Ausgabe-Teil erfolgt über Ein-/Ausgabe-Befehle. Hierbei ist dem Anschluß eines Kanals mit der Nummer n (A_n , B_n , C_n) das Bit im Datenwort mit der Nummer n (D_n) zugeordnet. Es bestehen folgende Besonderheiten:

- 1) Wenn Anschlüsse eines Kanals als Daten-Ausgänge benutzt werden, kann man mit Eingabe-Befehlen die an diese Anschlüsse ausgegebenen Daten auch wieder einlesen.

- 2) Wenn Anschlüsse eines Kanals als Daten-Eingänge benutzt werden, haben die mit Ausgabe-Befehlen an diese Anschlüsse übergebenen Daten keine Auswirkung.
- 3) Wenn Anschlüsse eines Kanals als Steuer-Ausgänge programmiert sind, haben die mit Eingabe-Befehlen von diesen Anschlüssen eingelesenen Daten keine Funktion.
- 4) Wenn Anschlüsse eines Kanals als Steuer-Eingänge benutzt werden, haben die mit Ausgabe-Befehlen an diese Anschlüsse ausgegebenen Daten keine Auswirkung.

Der Status des Ein-/Ausgabe-Teils kann jederzeit durch Einlesen eines Statusworts in die Zentraleinheit abgefragt werden. Die Aussage des Statusworts für den Ein-/Ausgabe-Teil geht aus Bild 8.3-6 hervor. Bit D1 und D4 ist zu entnehmen, ob eine Aktivität der Zentraleinheit erforderlich ist. Bit D2 und D5 zeigen, ob der Steuerausgang "Interrupt anfordern" durch das Kommandowort deaktiviert oder aktiviert ist. Über Bit D0 und D3 schließlich kann man den Pegel am Steuerausgang "Interrupt anfordern" abfragen.

Der Zähler-Teil des Bausteins SAB8155 hat je einen Eingang und Ausgang. Der Eingang des Zählers im Baustein Nr. 1 ist fest mit dem Systemtakt von 3,072 MHz verbunden. Der Ausgang des Zählers im Baustein Nr. 1 ist auf den Anschluß TIMO-1 der Anwender-Schnittstelle geführt. Vom Zähler im Baustein Nr. 2 liegt der Eingang an Anschluß TIMI-2 und der Ausgang an Anschluß TIMO-2 der Anwender-Schnittstelle.

Der Zähler-Teil umfaßt einen 14-Bit-Zähler, ein 16-Bit-Zählerregister, ein 16-Bit-Zwischenregister und eine Verknüpfungsschaltung für die Erzeugung des Zähler-Ausgangssignals. Das Zählerregister wird durch die Zentraleinheit mit einem 2 Bit umfassenden Mode und einem 14 Bit umfassenden Teilverhältnis geladen. Der Mode legt fest, welche Länge der Ausgangsimpuls haben und ob der Zähler einmalig oder periodisch arbeiten soll. Das

Teilverhältnis gibt die Zahl der Eingangsimpulse an, die zu einem vollständigen Zählerdurchlauf gehören. Der Zähler kann mit dem Kommandowort gestoppt und gestartet werden. Beim Starten wird der Inhalt des Zählerregisters in das Zwischenregister übernommen, die Betriebsart entsprechend dem Mode im Zwischenregister eingestellt und das Teilverhältnis aus dem Zwischenregister in den Zähler übernommen. Anschließend zählt der Zähler bei jeder ansteigenden Flanke des Eingangstakts um 1 herunter. Nach einer dem Teilverhältnis entsprechenden Zahl von Perioden des Eingangstakts ist ein Zählerdurchlauf beendet. Bei einmaligem Zählen wird nur ein Zählerdurchlauf ausgeführt, bei periodischem Zählen wird der Zähler nach jedem Zählerdurchlauf aus dem Zwischenregister geladen und zählt dann herunter. Es ist zu beachten, daß zwar jederzeit ein neuer Mode und ein neues Teilverhältnis in das Zählerregister übergeben werden können, daß sie jedoch erst beim nächsten Starten wirksam werden und bis dahin die vorher eingegebenen Werte wirksam bleiben.

Vor dem Starten des Zählers müssen Mode und Teilverhältnis von der Zentraleinheit übergeben werden. Dies erfolgt über zwei Ausgabebefehle mit den in Tabelle 8.3-2 angegebenen Adressen. Der Inhalt der ausgegebenen Datenwörter ist in Bild 8.3-7 angegeben. Eins der Datenwörter enthält den Mode und die höherwertigen 6 Bits, das andere die niederwertigen 8 Bits des Teilverhältnisses. Die Kodierung des Modes und die damit verbundene Funktion geht Bild 8.3-8 hervor.

Das Starten und Stoppen des Zählers erfolgt durch Ausgabe eines Kommandoworts von der Zentraleinheit. In Bild 8.3-9 ist der Teil des Kommandoworts erläutert, der für den Zähler-Teil maßgeblich ist. Die Zentraleinheit kann jederzeit durch Einlesen eines Statusworts abfragen, ob ein Zählerdurchlauf beendet ist. Der Teil des Statusworts, der dem Zähler zugeordnet ist, ist in Bild 8.3-10 erläutert.

Es ist zu beachten, daß das Monitor-Programm bei jedem mit den Kommandos 0, 1, 2 und B veranlaßten Einschreiben unter einer Adresse zwischen 0000 und 0FFF den Baustein SAB8155 Nr. 1 (Ein-/Ausgabe und Zähler) in bestimmter Weise programmiert (u.a. alle Kanäle auf Eingabe). Dies ist erforderlich,

da der Zähler des Bausteins für die EPROM-Programmierung (EPROM-Steckplatz Nr. 2) benötigt wird. Eine in einem Anwender-Programm vorgenommene Programmierung wird daher zerstört, wenn zwischenzeitlich Kommandos in der oben angegebenen Art verwendet werden.

Das Arbeiten mit dem Ein-/Ausgabe-Teil des Bausteins SAB8155 soll an einem ganz einfachen Beispiel demonstriert werden. Die Aufgabe besteht darin, über einen Eingang ein binäres Signal einzugeben und das gleiche binäre Signal über einen Ausgang auszugeben. Das Eingangssignal wird durch einen Schalter hergestellt und das Ausgangssignal durch eine Leuchtdiode angezeigt. Bild 8.3-11a zeigt die dafür erforderliche externe Schaltung, die über die Anwender-Schnittstelle angeschlossen wird. Als Eingang wird Anschluß P1A0, als Ausgang Anschluß P1B0 benutzt. Zwischen Schalter- und Lampenzustand besteht folgende Wirkungskette: Schalter geschlossen - L-Pegel an Eingang P1A0 - über Programm L-Pegel an Ausgang P1B0 - über Inverter-H-Pegel an der Leuchtdiode - Leuchtdiode dunkel. Bild 8.3-11b zeigt das Programm, mit dem der Pegel von Eingang P1A0 auf Ausgang P1B0 übertragen wird. Zunächst ist der Baustein SAB8155 Nr. 1 so zu initialisieren, daß Kanal A auf Eingabe und Kanal B auf Ausgabe geschaltet ist. Anschließend wird in einer Endlosschleife das Datenwort von Kanal A eingelesen und wieder an Kanal B ausgegeben. Eigentlich wird in diesem Fall lediglich das niederstwertige Bit des Datenworts benötigt, es ist jedoch nur byteweise Übertragung möglich. Bild 8.3-11c zeigt das ausgeführte Programm. Es kann mit Kommando 0 eingegeben und mit Kommando 7 gestartet werden. Nach dem Starten muß bei geschlossenem Schalter die Leuchtdiode dunkel und bei offenem Schalter die Leuchtdiode hell sein. Daß die Kopplung von Schalter und Leuchtdiode tatsächlich über das Programm erfolgt, kann man durch Abbrechen des Programms mit der R-Taste sehen. Danach reagiert die Leuchtdiode nicht mehr auf das Umlegen des Schalters.

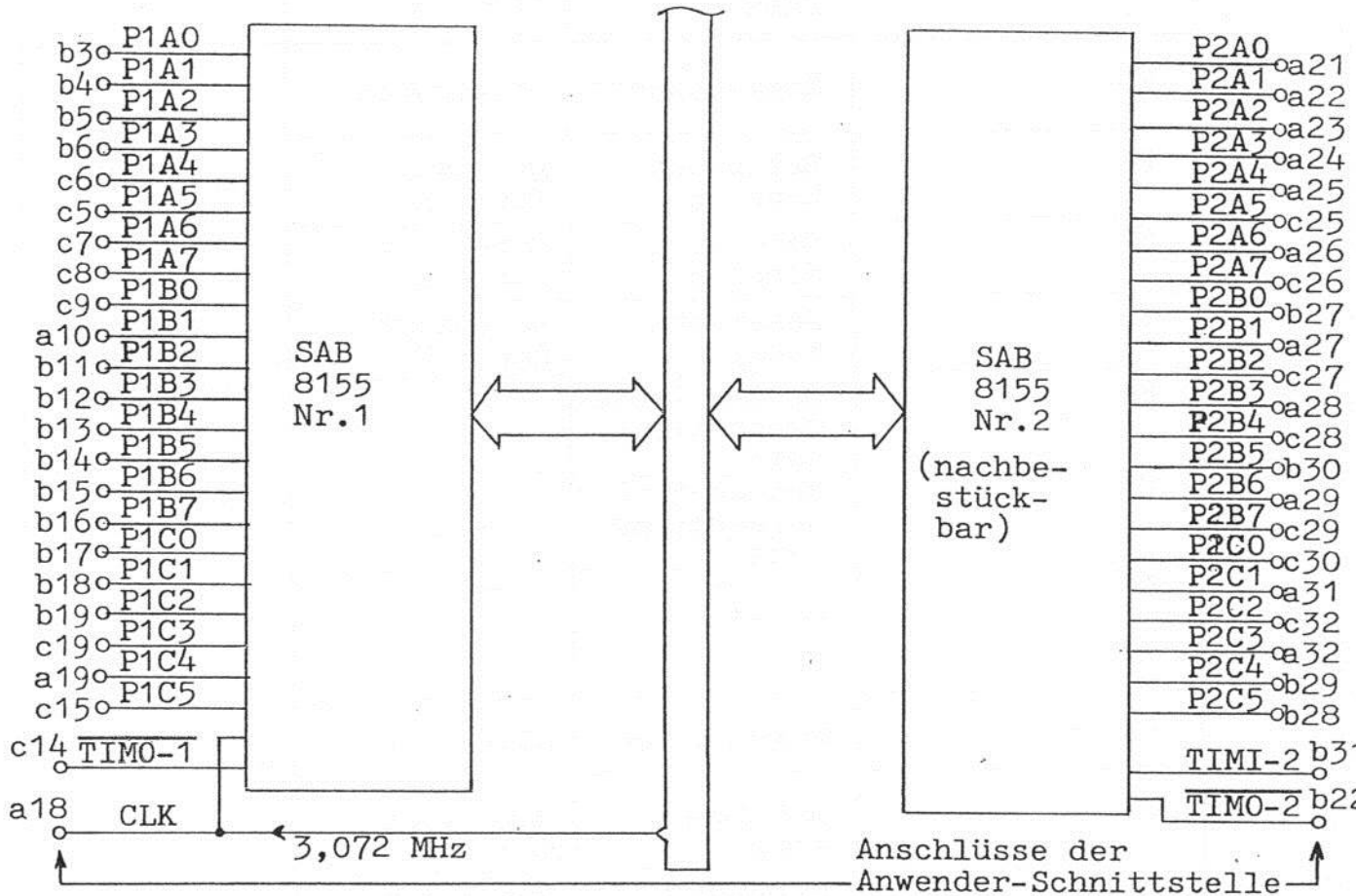
Ein weiteres Beispiel erläutert die Programmierung der Zähler. Der Zähler im Baustein Nr. 1 soll als programmierbarer Tongenerator verwendet werden. Dazu ist, wie in Bild 8.3-12a dargestellt, sein Ausgang über einen Spannungsteiler und ein Kabel an einen Phonostecker zu legen, der in die

Eingangsbuchse eines Verstärkers gesteckt wird. Der Ablaufplan des Programms ist in Bild 8.3-12b angegeben. Es besteht aus einer Endlos-schleife mit einer Reihe von Operationen. Zunächst wird auf das Drücken einer Hexadezimaltaste des Tastenfelds gewartet, und das Tastenwort wird eingelesen (Programm ANWEIN im Monitor-EPROM). Vom Tastenwort werden die niederwertigen vier Bits isoliert und als Bit D2 bis D5 des höherwertigen Bytes des Modeworts verwendet. Sie bilden damit die höchstwertigen Bits des Zähler-Teilverhältnisses. Mit Bit D6 und D7 des Modeworts wird der Zähler auf symmetrische Signale und periodischen Betrieb programmiert. Alle übrigen Bits des Modeworts sind 0. Das Modewort wird ausgegeben und anschließend durch Ausgabe des Kommandoworts der Zähler gestartet. Das vollständige Programm ist in Bild 8.3-12c angegeben. Das Programm ist mit Kommando 0 einzugeben und mit Kommando 7 zu starten. Nach jedem Drücken einer Taste des Tastenfelds ertönt der dieser Taste zugeordnete Ton. Die Frequenzen sind in Tabelle 8.3-12d angegeben.

Bild 8.3-1

Anschlüsse der parallelen Ein-/Ausgabe und der Zähler an der Anwender-Schnittstelle

a) Schaltung



b) statische Daten

		L-Pegel *)	H-Pegel *)
Eingänge	TIMI-2 P1A0...P1C5 P2A0...P2C5	$U \leq 0,8V$ $ I \leq 10\mu A$	$U \geq 2,0V$ $ I \leq 10\mu A$
Ausgänge	CLK TIMO-1, TIMO-2 P1A0...P1C5 P2A0...P2C5	$U \leq 0,45V$ bei $I = 1,28mA$ $U \leq 0,45V$ bei $I = 2mA$	$U \geq 2,4V$ bei $-I = 320\mu A$ $U \geq 2,4V$ bei $-I = 400\mu A$

*) Bei den Stromangaben für Ein- und Ausgänge bedeutet ein positives Vorzeichen, daß der Strom in den ECB85 hineinfließt und ein negatives Vorzeichen, daß der Strom herausfließt. Es wird dabei die technische Stromrichtung angenommen (entgegen dem Elektronenstrom).

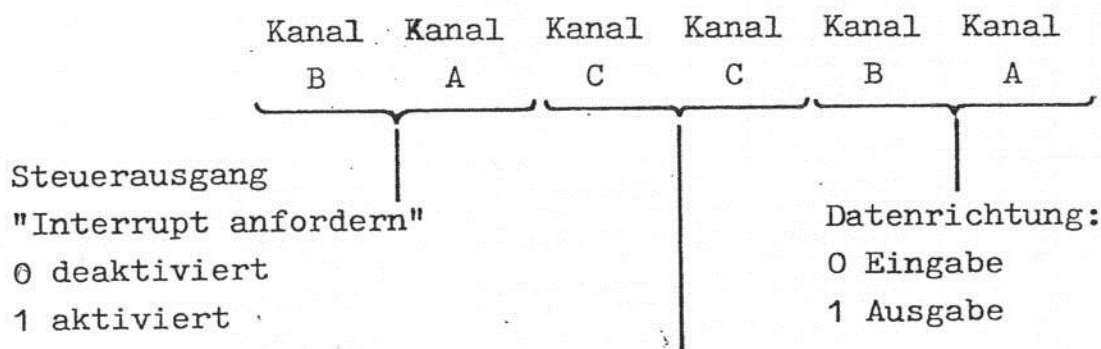
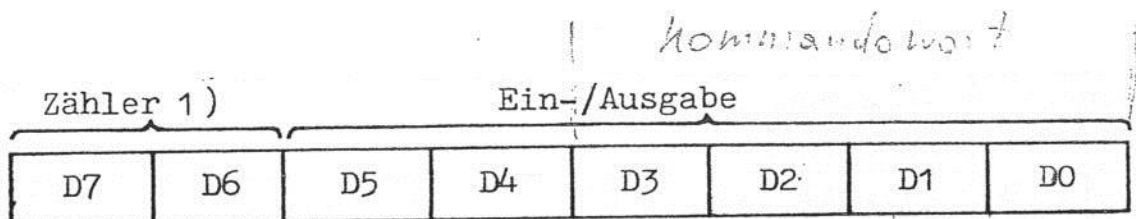
Tabelle 8.3-2

Adressen der Bausteine SAB8155 (Ein-/Ausgabe-Adreßraum)

Adresse <i>im E/A-Speicher- Bereich</i>	Information		Baustein
	Ausgabe zum SAB8155	Eingabe vom SAB8155	
F0	Kommandowort	Statuswort	SAB8155 Nr. 2 <i>nicht betrieben!</i>
F1	Datenwort Kanal A	Datenwort Kanal A	
F2	Datenwort Kanal B	Datenwort Kanal B	
F3	Datenwort Kanal C	Datenwort Kanal C	
F4	Modewort, nie- derwertiges Byte	-	
F5	Modewort, hö- herwertiges Byte	-	
F6	-	-	
F7	-	-	
F8	Kommandowort	Statuswort	SAB8155 Nr. 1
F9	Datenwort Kanal A	Datenwort Kanal A	
FA	Datenwort Kanal B	Datenwort Kanal B	
FB	Datenwort Kanal C	Datenwort Kanal C	
FC	Modewort, nie- derwertiges Byte	-	
FD	Modewort, hö- herwertiges Byte	-	
FE	-	-	
FF	-	-	

Bild 8.3-3

Kommandowort für den Baustein SAB8155, Ein-/Ausgabe-Teil ²⁾



Datenrichtung und Funktion:

00 Eingabe

01 Steuersignale für Kanal A + Ausgabe

10 Steuersignale für Kanal A und B

11 Ausgabe

- 1) Falls der Zähler nicht beeinflusst werden soll, ist $D6 = 0$ und $D7 = 0$ zu setzen. Dieser Teil des Kommando-Worts wird in Bild 8.3-9 dargestellt.
- 2) Der Begriff "Kommandowort" bezieht sich auf den Baustein und darf nicht mit "Kommando" des Monitor-Programms verwechselt werden.

Tabelle 8.3-4

Betriebsarten des Ein-/Ausgabe-Teils im Baustein SAB8155

Kommando- Wort				Verwendung der Anschlüsse			
D3	D2	D1	D0	A0...A7	C0... C2	C3... C5	B0...B7
0	0	0	0	Eingabe	Eingabe		Eingabe
0	0	0	1	Ausgabe	Eingabe		Eingabe
0	0	1	0	Eingabe	Eingabe		Ausgabe
0	0	1	1	Ausgabe	Eingabe		Ausgabe
0	1	0	0	Eingabe	Steuersignale 1)	Ausgabe	Eingabe
0	1	0	1	Ausgabe	Steuersignale 2)	Ausgabe	Eingabe
0	1	1	0	Eingabe	Steuersignale 1)	Ausgabe	Ausgabe
0	1	1	1	Ausgabe	Steuersignale 2)	Ausgabe	Ausgabe
1	0	0	0	Eingabe	Steuersignale 1)	Steuersignale 3)	Eingabe
1	0	0	1	Ausgabe	Steuersignale 2)	Steuersignale 3)	Eingabe
1	0	1	0	Eingabe	Steuersignale 1)	Steuersignale 4)	Ausgabe
1	0	1	1	Ausgabe	Steuersignale 2)	Steuersignale 4)	Ausgabe
1	1	0	0	Eingabe	Ausgabe		Eingabe
1	1	0	1	Ausgabe	Ausgabe		Eingabe
1	1	1	0	Eingabe	Ausgabe		Ausgabe
1	1	1	1	Ausgabe	Ausgabe		Ausgabe

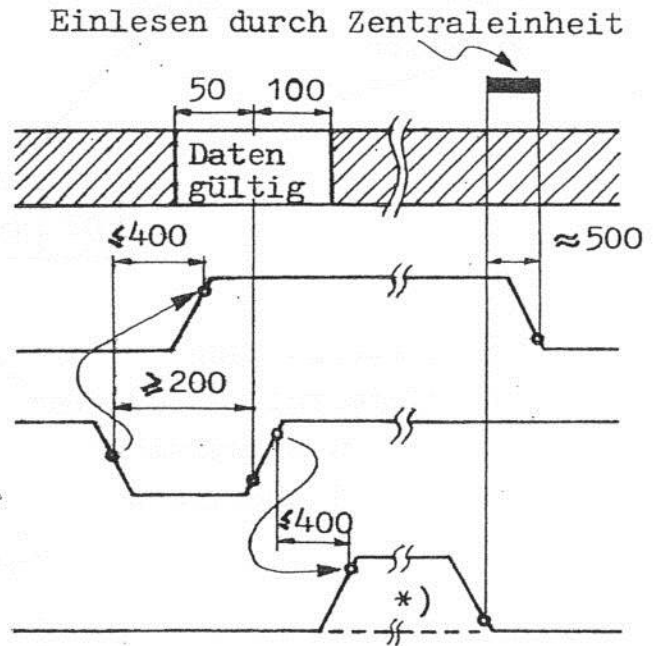
- 1) Steuersignale zu Eingabe über Kanal A,
nähere Angaben siehe Bild 8.3-5a
- 2) Steuersignale zu Ausgabe über Kanal A,
nähere Angaben siehe Bild 8.3-5b
- 3) Steuersignale zu Eingabe über Kanal B,
nähere Angaben siehe Bild 8.3-5a
- 4) Steuersignale zu Ausgabe über Kanal B,
nähere Angaben siehe Bild 8.3-5b

Bild 8.3-5

Ein-/Ausgabe-Teil des SAB8155, Betrieb mit Rückmeldung

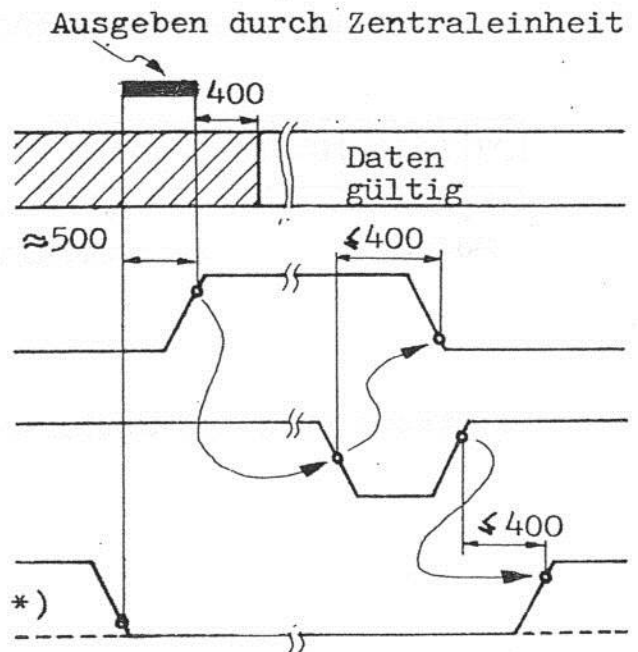
a) Eingabe mit Rückmeldung (Zeitangaben in ns)

Signal		Anschluß	
Funktion	Richtung	Kanal A	Kanal B
Daten	ein	A0...A7	B0...B7
Register voll	aus	C1	C4
Daten übernehmen	ein	C2	C5
Interrupt anfordern	aus	C0	C3



b) Ausgabe mit Rückmeldung (Zeitangaben in ns)

Signal		Anschluß	
Funktion	Richtung	Kanal A	Kanal B
Daten	aus	A0...A7	B0...B7
Daten bereit	aus	C1	C4
Daten abgeholt	ein	C2	C5
Interrupt anfordern	aus	C0	C3



*) ausgezogene Linie: { Kommando-Wort Bit D4 (für Kanal A) } = 1
 gestrichelte Linie: { bzw. D5 (für Kanal B) } = 0

Bild 8.3-6

Statuswort des Bausteins SAB8155, Ein-/Ausgabe-Teil

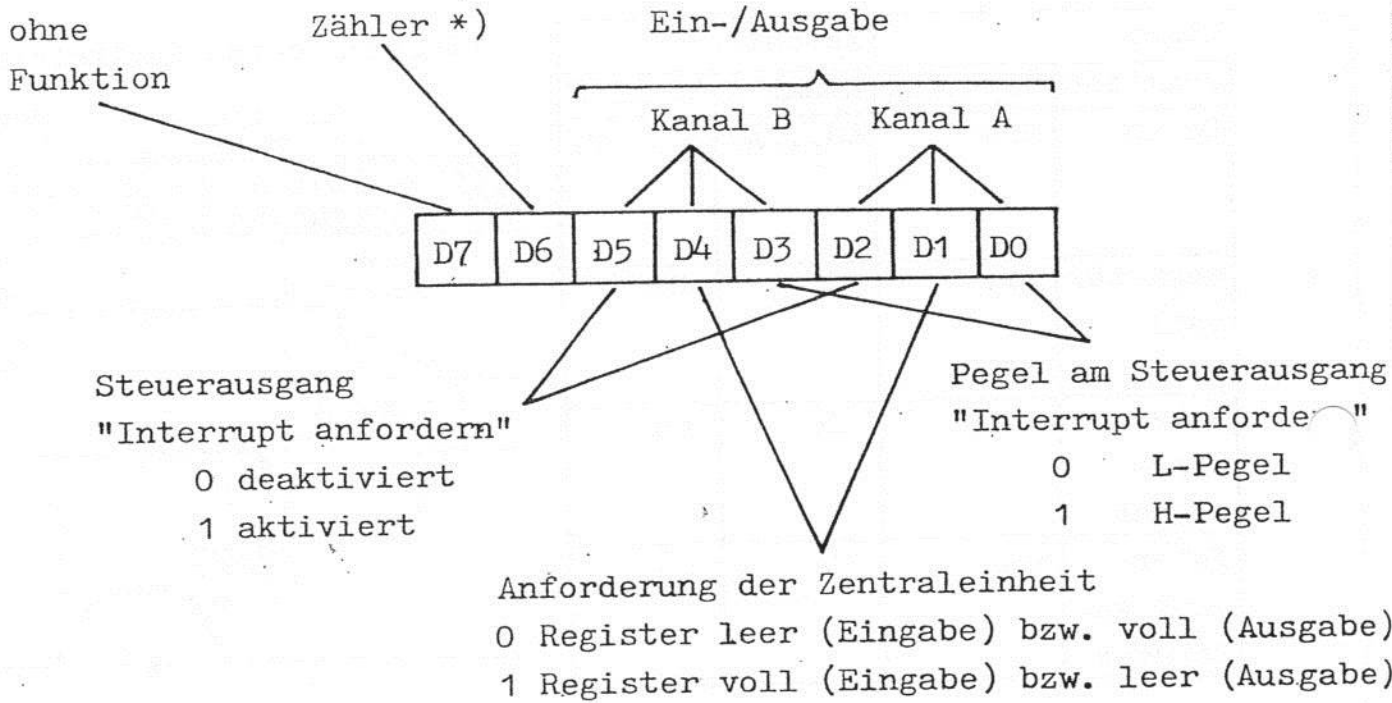
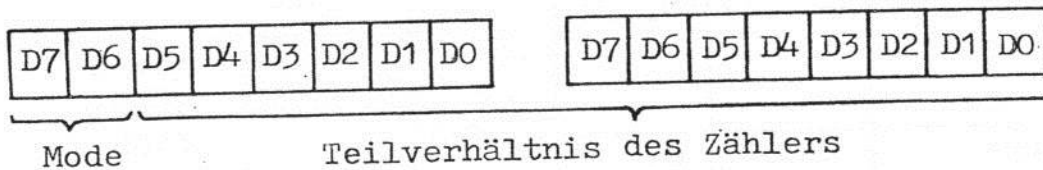


Bild 8.3-7

Modewort des Bausteins SAB8155



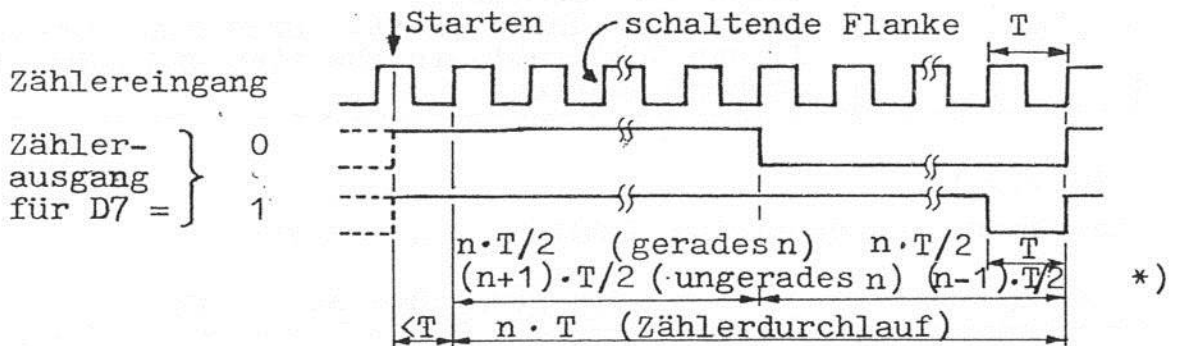
*) Dieser Teil des Statusworts wird in Bild 8.3-10 dargestellt.

Bild 8.3-8

Modes des Zählers im Baustein SAB8155 und Signale

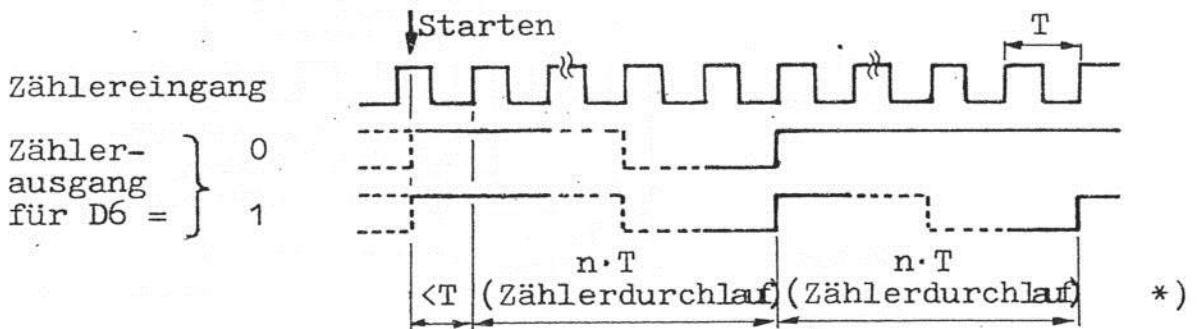
a) Funktion von Modebit D7

D7	Funktion
0	Zählerausgang während der ersten Hälfte des Zählerdurchlaufs auf H-Pegel, während der zweiten auf L-Pegel. Bei ungeradzahligem Teilverhältnis des Zählers ist der Zeitabschnitt mit H-Pegel am Zählerausgang um eine Periode des Eingangstakts länger.
1	Zählerausgang am Ende eines Zählerdurchlaufs für eine Periode des Eingangstakts auf L-Pegel, in der übrigen Zeit auf H-Pegel.



b) Funktion von Modebit D6

D6	Funktion
0	einmaliges Zählen
1	periodisches Zählen



*) $n = \text{Teilverhältnis des Zählers } (n = 2 \dots [2^{14} - 1])$

Bild 8.3-9

Kommandowort für den Baustein SAB8155, Zähler-Teil

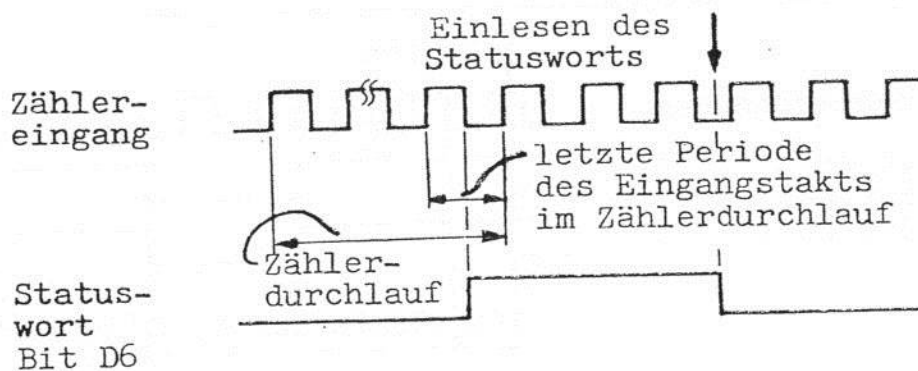
Zähler		Ein-/Ausgabe 1)					
D7	D6	D5	D4	D3	D2	D1	D0
0	0	Keine Funktion					
0	1	Zähler stoppen. Zähler-Ausgang bleibt auf H-Pegel bzw. geht auf H-Pegel, sobald der Zähler-Eingang H-Pegel hat.					
1	0	Zähler nach Ablauf des augenblicklichen Zählerdurchlaufs stoppen.					
1	1	Falls der Zähler steht: Zähler sofort mit Mode und Teilverhältnis aus dem zugehörigen Register laden und starten. Falls der Zähler läuft: Ende des augenblicklichen Zählerablaufs abwarten und dann wie vorstehend starten.					

Bild 8.3-10

Statuswort des Bausteins SAB8155, Zähler-Teil

ohne Funktion		Ein-/Ausgabe-Teil 2)					
D7	D6	D5	D4	D3	D2	D1	D0

Zähler-Status:
 Wird 1 mit der im Zählerdurchlauf letzten fallenden Flanke des Eingangstakts.
 Wird 0 nach dem Einlesen des Statusworts.

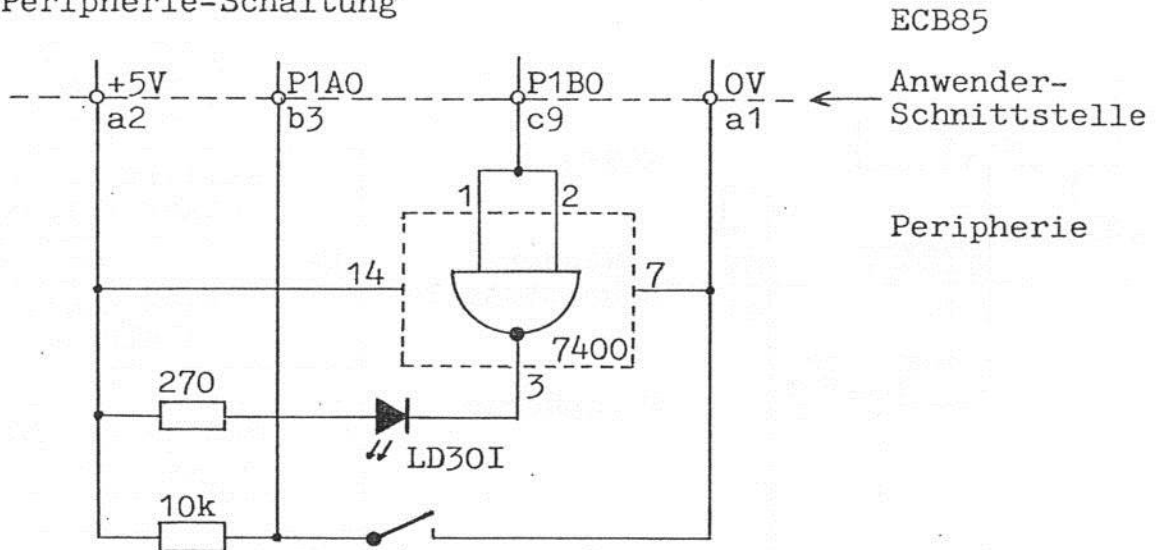


- 1) Falls der Ein-/Ausgabe-Teil nicht beeinflusst werden soll, muß in diesem Teil immer die für die gewünschte Betriebsart erforderliche Bit-Kombination übergeben werden. Hierzu Bild 8.3-3.
- 2) Dieser Teil des Statusworts wird in Bild 8.3-6 erläutert.

Bild 8.3-11

Demonstrationsbeispiel für die Ein-/Ausgabe

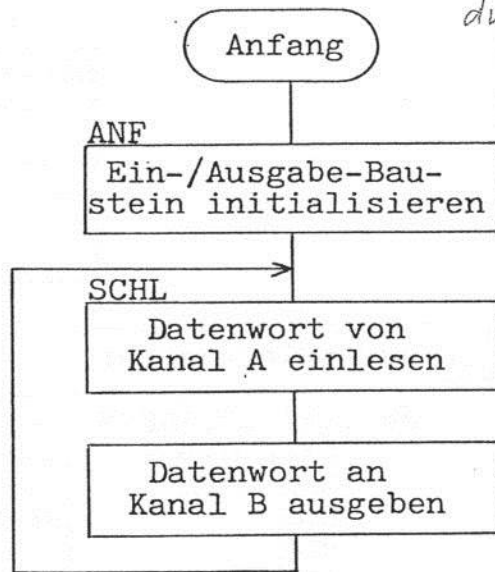
a) Peripherie-Schaltung



Schalter geschlossen u. Port A als Ausgang →

Zerstörung des E/A-Bausteins durch Kurzschluß

b) Ablaufplan des Programms



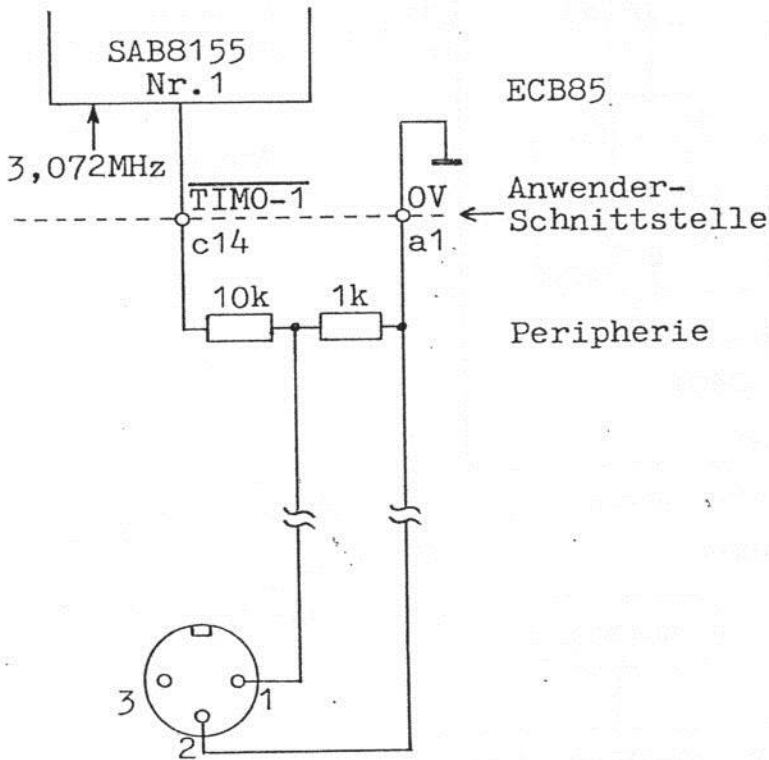
c) Programm

Hexadezimal-Adresse	Hexadezimal-Befehl	Assembler-Befehl	Kommentar
1800	3E02	ANF: MVI A,02H	;Ein-/Ausgabe-Baustein
1802	D3F8	OUT OF8H	; initialisieren
1804	DBF9	SCHL: IN OF9H	;Datenwort von Kanal A
1806	D3FA	OUT OFAH	;Datenwort an Kanal B
1808	C30418	JMP SCHL	; ausgeben ;Schleife

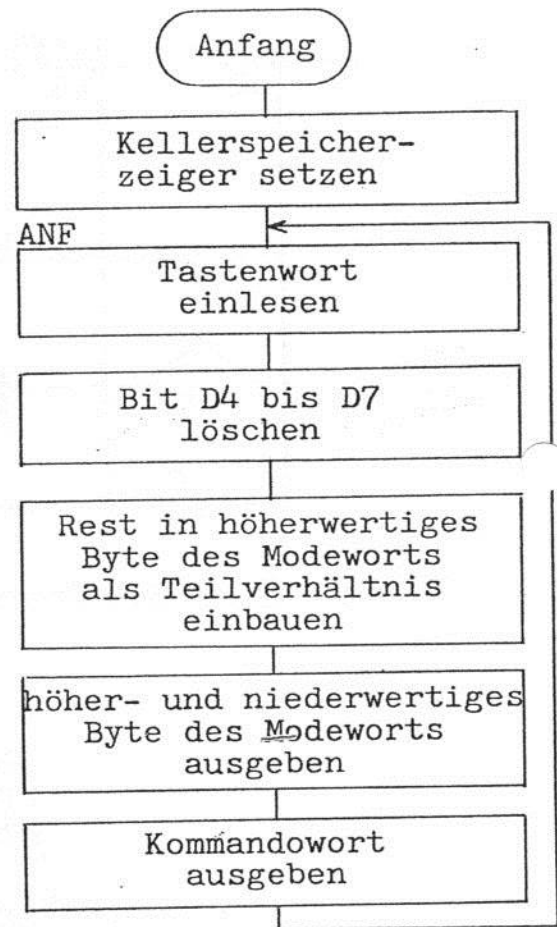
Bild 8.3-12

Demonstrationsbeispiel für den Zähler

a) Peripherie-Schaltung



b) Programm-Ablaufplan



c) Programm

Hexadezimal-Adresse	Hexadezimal-Befehl	Assembler-Befehl	Kommentar
1800	31FF1B	ANF:LXI SP,1BFFH	;Kellerspeicherzeiger setzen
1803	CDD107	CALL 07D1H	;Tastenwort einlesen
1806	E60F	ANI 0FH	;Bit D4 bis D7 löschen
1808	07	RLC	;Rest in höherwertiges Byte
1809	07	RLC	; des Modeworts einbauen
180A	C640	ADI 40H	
180C	D3FD	OUT OFDH	;höher-und niederwertiges
180E	3E00	MVI A,00	; Byte des Modeworts aus-
1810	D3FC	OUT OFCH	; geben
1812	3ECO	MVI A,0COH	;Kommandowort ausgeben
1814	D3F8	OUT OF8H	
1816	C30018	JMP ANF	;Schleife

d) Zusammenhang zwischen gedrückter Taste und Frequenz

Taste	1	2	3	4	5	6	7	...	n	...
Frequenz/Hz	3000	1500	1000	750	600	500	428	...	3kHz/n	...

8.4 Arbeiten mit dem Baustein SAB8085A

Der Baustein SAB8085A umfaßt die Zentraleinheit, eine kleine Ein-/Ausgabe-Einheit (SID und SOD) und die Interrupt-Steuerung. Die Zentraleinheit wurde bereits behandelt, so daß an dieser Stelle nicht mehr auf sie eingegangen werden muß. Die Einzel-Ein- und Ausgänge SID und SOD des Bausteins SAB8085A sind über nicht-invertierende Treiber an der Anwender-Schnittstelle verfügbar und heißen dort ANWSID (zu SID) und ANWSOD (zu SOD). Von den direkten Interrupt-Eingängen des Bausteins SAB8085A sind RST5.5, RST6.5 und TRAP über Inverter mit der Anwender-Schnittstelle verbunden. Die Signale heißen dort $\overline{\text{RST5.5}}$ (über Inverter zu RST5.5 am SAB8085A), $\overline{\text{RST6.5}}$ (entsprechend zu RST6.5) und $\overline{\text{TRAP}}$ (entsprechend zu TRAP). Im folgenden wird immer von den invertierten Signalen die Rede sein. Der Interrupt-Eingang RST7.5 ist über einen Inverter mit der I-Taste verbunden. Bild 8.4-1a zeigt einen Schaltungsauszug mit den aufgeführten Signalen. Die Pegel und Belastungen bzw. Belastbarkeiten aller genannten Ein- und Ausgänge gehen aus Tabelle 8.4-1b hervor.

Der augenblickliche Pegel am Eingang ANWSID kann über den Befehl RIM eingelesen werden. Mit dem Befehl SIM kann man den Ausgang ANWSOD setzen. Der Zustand am Ausgang bleibt bis zu einer neuen Ausgabe erhalten. Die softwaremäßige Behandlung wird im Anschluß an die Besprechung von Interrupt erläutert.

Über die Interrupt-Eingänge $\overline{\text{RST5.5}}$, $\overline{\text{RST6.5}}$ und $\overline{\text{TRAP}}$ können mit elektrischen Signalen Interrupt-Sprünge ausgelöst werden. Über die I-Taste kann man Interrupts RST7.5 bewirken. Den Interrupt-Eingängen sind bestimmte Sprungadressen im Bereich des EPROM-Steckplatzes Nr. 1 (Monitor-EPROM) zugeordnet. Der Monitor-EPROM enthält an diesen Stellen Sprungbefehle, die weiter in den Bereich des EPROM-Steckplatzes Nr. 2 führen. Tabelle 8.4-2 zeigt die verschiedenen Sprungziele. Die erste Spalte ist von Interesse, wenn der Monitor-EPROM gezogen und stattdessen ein Anwender-EPROM eingesteckt ist. Die zweite Spalte gilt bei eingestecktem Monitor-EPROM.

Die Auslösung von Interrupt-Sprüngen über die Eingänge RST5.5, RST6.5 und RST7.5 (I-Taste) kann durch Programm individuell freigegeben oder gesperrt (maskiert) werden. Außerdem ist mit Hilfe der Befehle EI (enable interrupt) und DI (disable interrupt) eine generelle Freigabe und Sperrung möglich. Im Gegensatz dazu gibt es für den Interrupt-Eingang TRAP weder eine individuelle Möglichkeit der Sperrung, noch hat die generelle Sperrung auf ihn einen Einfluß; er ist also immer wirksam. Die Interrupt-Eingänge haben zueinander eine festgelegte Priorität. Der Eingang mit der höheren Priorität wird zuerst bedient, wenn mehrere Interrupt-Anforderungen gleichzeitig anstehen. Die höchste Priorität hat der Eingang TRAP, dann folgen RST7.5 (I-Taste), RST6.5 und RST5.5. Bei Ausführung eines Interrupt-Sprungs sperrt die Zentraleinheit automatisch Interrupt. Im allgemeinen gibt man vor Rücksprung aus dem Interrupt-Bedienprogramm Interrupt wieder frei (Befehl EI).

Auch hardwaremäßig bestehen Unterschiede zwischen den Interrupt-Eingängen, die im Bild 8.4-3 zusammengestellt sind. Bei den Eingängen RST5.5 und RST6.5 ist nur der Pegel von Bedeutung: Dem Ruhezustand ist H-Pegel zugeordnet. Eine Interrupt-Anforderung besteht so lange, wie L-Pegel anliegt. Verschwindet der L-Pegel, ehe der Interrupt bedient ist - weil beispielsweise gerade ein Interrupt höherer Priorität bearbeitet wurde - geht die Interrupt-Anforderung verloren. Liegt dagegen auch nach Bedienung des Interrupts noch L-Pegel an, so führt das anschließend noch einmal zu einem Interrupt-Sprung. Im Gegensatz dazu beginnt für den Eingang RST7.5 der Zeitraum der Interrupt-Anforderung mit einer HL-Flanke (Drücken der I-Taste), wofür auch ein kurzer Impuls ausreicht, und endet entweder mit dem zugehörigen Interrupt-Sprung oder durch Rücksetzen mit Hilfe des Befehls SIM, auf den später eingegangen wird. Beim Eingang TRAP beginnt der Zeitraum der Interrupt-Anforderung ebenfalls mit einer HL-Flanke, er dauert aber nur so lange, bis entweder der L-Pegel am Interrupt-Eingang verschwindet oder der zugehörige Interrupt-Sprung ausgeführt wird. Da TRAP die höchste Priorität hat und nicht gesperrt werden kann, vergehen maximal etwa 6 μ s (Dauer des längsten Befehls) vom Beginn der Anforderung bis zum Interrupt-Sprung. Ebenso lang muß daher der TRAP-Impuls mindestens sein, damit der Interrupt mit Sicherheit bearbeitet wird. Nach einem durch TRAP

ausgelösten Interrupt-Sprung muß der Pegel mindestens kurzzeitig auf H gehen, damit ein neuer Interrupt ausgelöst werden kann. Die Speicherwirkung der Interrupt-Eingänge RST7.5 und TRAP wird durch zugeordnete Flip-Flops ermöglicht. Mit dem Rücksetzen der Zentraleinheit werden auch diese Flip-Flops rückgesetzt, so daß dann keine Interrupt-Anforderung besteht. Im unbeschalteten Zustand liegen alle Interrupt-Eingänge auf H-Pegel (offene TTL-Eingänge) und fordern daher keinen Interrupt an. In Bild 8.4-3 sind die Zeiträume dargestellt, in denen bei den verschiedenen Interrupt-Eingängen eine Interrupt-Anforderung besteht.

Für den Datenverkehr zwischen der Zentraleinheit auf der einen und der Interrupt-Steuerung und der Ein-/Ausgabe-Einheit auf der anderen Seite gibt es die beiden Befehle SIM und RIM. Der Befehl SIM (set interrupt mask) hat den Maschinencode 30 und beeinflusst abhängig vom Inhalt des Akkumulators den Datenausgang ANWSOD sowie die Interrupt-Steuerung. Es ergeben sich folgende Teilfunktionen, die auch in Bild 8.4-4a angegeben sind:

a) Steuern des Ausgangs ANWSOD:

Diese Funktion wird nur wirksam, wenn $D_6 = 1$ ist. Dann wird der Zustand des Bit D_7 an den Ausgang ANWSOD ausgegeben und bleibt bis zur nächsten Ausgabe erhalten.

b) Individuelle Interrupt-Sperre:

Diese Funktion wird nur wirksam, wenn $D_3 = 1$ ist. Dann wird mit Bit D_0 , D_1 und D_2 die Auslösung von Interrupt-Sprüngen über die Eingänge RST5.5, RST6.5 und RST7.5 gesperrt oder freigegeben.

c) Rücksetzen des Interrupt-Flip-Flops zu RST7.5:

Das dem Eingang RST7.5 zugeordnete Flip-Flop, das durch die HL-Flanke des Eingangssignals gesetzt wird, kann softwaremäßig mit dem Befehl SIM rückgesetzt werden. Das Rücksetzen erfolgt mit Bit D_4 .

Der Befehl RIM (read interrupt mask) hat den Maschinencode 20 und dient zum Einlesen vom Eingang ANWSID und von der Interrupt-Steuerung in den Akkumulator. Das Datenwort im Akkumulator liefert folgende Aussagen, die auch in Bild 8.4-4b dargestellt sind:

a) Einlesen vom Eingang ANWSID:

D7 enthält den Augenblickswert am Eingang ANWSID.

b) Bestehende Interrupt-Anforderungen:

D4, D5 und D6 geben an, welche Eingänge im Augenblick Interrupt anfordern.

c) Generelle Interrupt-Sperre:

Aus Bit D3 ist zu ersehen, ob generell durch die Zentraleinheit Interrupt gesperrt oder freigegeben ist (Sperrern und Freigeben mit Befehl DI und EI).

d) Individuelle Interrupt-Sperre:

Aus Bit D0 bis D2 ist zu ersehen, ob für die einzelnen Eingänge RST5.5, RST6.5 und RST7.5 Interrupt gesperrt oder freigegeben ist (Sperrern und Freigeben mit Befehl SIM).

Beim Rücksetzen der Zentraleinheit werden die Interrupts RST5.5, RST6.5 und RST7.5 generell und individuell gesperrt. Die Sperrung kann außerdem jederzeit generell mit dem Befehl DI und individuell mit dem Befehl SIM erreicht werden. Auch bei Sperrung läßt sich jedoch das Interrupt-Flip-Flop zum Eingang RST7.5 über den Eingang setzen und mit dem Befehl SIM rücksetzen. Außerdem können auch bei Sperrung mit Hilfe des durch den Befehl RIM eingelesenen Datenworts alle bestehenden Interrupt-Anforderungen erkannt werden. Auf diese Weise läßt sich bei Sperrung von Interrupt immer noch eine Interrupt-Bearbeitung nach dem Abfrage-Verfahren (polling) durchführen.

Beim Arbeiten mit dem Interrupt-Eingang TRAP ergibt sich eine programmtechnische Besonderheit durch die Tatsache, daß die zugehörigen Interrupt-Sprünge nicht gesperrt werden können. Bei einem durch TRAP ausgelösten Interrupt-Sprung wird wie bei jedem anderen Interrupt generell Interrupt gesperrt (Funktion wie bei Befehl DI). Vor Rücksprung aus dem Interrupt-Bedien-Programm möchte man im allgemeinen entsprechend dem Zustand vor dem Interrupt-Sprung

wieder Interrupt generell sperren oder freigeben. Anders als bei allen anderen Interrupts kann man jedoch bei TRAP aus der Tatsache, daß ins Interrupt-Bedien-Programm gesprungen wurde, nicht darauf schließen, daß vorher Interrupt freigegeben war. Durch eine besondere schaltungstechnische Maßnahme im Baustein SAB8085A ist es jedoch möglich, daß man auch bei TRAP über den vorangegangenen Zustand Aufschluß erhalten kann: Mit dem ersten Befehl RIM, der nach dem durch TRAP ausgelösten Interrupt-Sprung ausgeführt wird, erhält man in dem Datenbit für die generelle Interrupt-Sperre den Zustand vor Ausführung des Interrupt-Sprungs. Abhängig davon kann man dann vor Verlassen des Interrupt-Bedien-Programms Interrupt gesperrt lassen oder freigeben (Befehl EI). Es ist zu beachten, daß alle durch folgende Befehle RIM eingelesene Datenwörter nur noch generelle Interrupt-Sperrung anzeigen.

Die Verhältnisse bezüglich des Ausgangs ANWSOD und des Interrupt-Systems sind beim ECB85 etwas verwickelt. Zunächst muß, wie schon früher erwähnt, zwischen dem tatsächlich in der Hardware bestehenden aktuellen Zustand und dem erst beim nächsten Starten eines Anwender-Programms wirksam werdenden Schatten-Zustand unterschieden werden. Außerdem ist zu bedenken, daß sowohl das Monitor- als auch das Anwender-Programm hierauf Einfluß nehmen können. In Tabelle 8.4-5a ist der aktuelle und in Tabelle 8.4-5b der Schatten-Status für verschiedene Bedingungen zusammengestellt. Die Angaben sind teilweise schon an früherer Stelle erwähnt worden.

Auch in diesem Kapitel sollen wieder Demonstrationsbeispiele gebracht werden. Zunächst zu der Ein-/Ausgabe über ANWSID und ANWSOD. Das Beispiel ist bezüglich der Funktion ähnlich wie das von Bild 8.3-11. Die Aufgabe besteht darin, über den Eingang ANWSID ein binäres Signal einzulesen und das gleiche Signal über den Ausgang ANWSOD auszugeben. Bild 8.4-6a zeigt die dafür erforderliche externe Schaltung, die über die Anwender-Schnittstelle angeschlossen wird. In Bild 8.4-6b ist der Ablaufplan des Programms und in Bild 8.4-6c das sehr einfache Programm selbst dargestellt. Das Programm kann mit Kommando 0 eingegeben und

mit Kommando 7 gestartet werden. Nach dem Starten muß der Programm-Wahlschalter in die Stellung ANW gebracht werden, da nur dann die Anschlüsse SID und SOD des Bausteins SAB8085A auf die Anschlüsse ANWSID und ANWSOD der Anwender-Schnittstelle geschaltet sind. Es ist zu beachten, daß in dieser Stellung des Programm-Wahlschalters das Monitor-Programm nicht mit der R-Taste gestartet werden kann. Durch das eingegebene Anwender-Programm muß bei geschlossenem Schalter der Peripherie-Schaltung die Leuchtdiode hell und bei offenem Schalter dunkel sein. Der Zusammenhang ist gegenüber dem Beispiel im Kapitel 8.3 umgekehrt, da wegen der größeren Belastbarkeit des Ausgangs ANWSOD vor der Leuchtdiode kein Treiber-Inverter verwendet wird.

Das Demonstrationsbeispiel für die Interrupt-Steuerung ist ebenfalls sehr einfach. Es wird der Interrupt-Eingang RST5.5 benutzt, an den nach Bild 8.4-7a ein Schalter angeschlossen ist. Bei diesem Interrupt-Eingang besteht die Interrupt-Anforderung so lange, wie L-Pegel anliegt bzw. der periphere Schalter geschlossen ist. Um das Arbeiten der Zentraleinheit im Hauptprogramm bzw. im Interrupt-Programm sichtbar zu machen, soll in diesem Beispiel das Hauptprogramm das Ausgeben des Zeichens H auf der linken Stelle des Anzeigefelds und das Interrupt-Programm das Ausgeben des Zeichens I an derselben Stelle bewirken. Bild 8.4-7b zeigt die Ablaufpläne der beiden Programme und Bild 8.4-7c die Programme selbst. Es ist zu beachten, daß das Interrupt-Programm bei dem Interrupt RST5.5 zugeordneten Adresse 082C beginnen muß, die im Bereich des EPROM-Steckplatzes Nr. 2 liegt. Deswegen wird für dieses Demonstrationsbeispiel ein gelöschter EPROM-Baustein SAB2758 oder SAB2716 benötigt, der in den EPROM-Steckplatz Nr. 2 einzustecken ist. Mit Kommando 0 wird das Hauptprogramm, beginnend mit Adresse 1800, eingegeben. Ebenfalls mit Kommando 0 wird das Interrupt-Programm eingegeben, jedoch vorsichtshalber zunächst beginnend z.B. mit Adresse 1900; anschließend kann man es dann nach Umlegen des Programmierschalters in die Stellung PRGR mit Kommando 2 in den EPROM ab Adresse 082C übertragen. Danach ist der Programmierschalter wieder in die Ruhestellung zu bringen. Das Hauptprogramm wird mit Kommando 7 gestartet. Wenn der externe Interrupt-Schalter offen ist, muß auf dem Anzeigefeld das Zeichen H erscheinen. Solange der Schalter dagegen geschlossen ist, wird das Interrupt-Programm ausgeführt, und es erscheint auf dem Anzeigefeld das

Zeichen I. Bei diesem Beispiel ist das Interrupt-Programm so kurz, daß normalerweise auch noch nach dessen Beendigung der Schalter geschlossen ist und damit die Interrupt-Anforderung fortbesteht. Damit wird bei geschlossenem Interrupt-Schalter ständig zwischen Interrupt- und Hauptprogramm hin und her gesprungen. Das Hauptprogramm kann sich jedoch nicht auswirken, da es sofort nach Einsprung wieder durch Interrupt unterbrochen wird. Es könnte die Befürchtung entstehen, daß bei diesen ständigen Interrupt-Sprüngen der Kellerspeicher fortlaufend mit Rücksprungsadressen vollgeschrieben wird, da der Befehl RET hinter dem Befehl EI steht. Dies ist jedoch nicht der Fall, da der Interrupt erst hinter dem auf EI folgenden Befehl wirksam wird. Es wird also zunächst ins Hauptprogramm zurückgesprungen (wobei die beim vorangegangenen Einsprung in das Interrupt-Programm im Kellerspeicher abgespeicherte Rücksprungsadresse wieder herausgeholt wird), ehe ein neuer Interrupt-Sprung stattfindet.

An dieser Stelle sollen noch einige allgemeine Anmerkungen zu den Interrupt-Eingängen angefügt werden. Normalerweise möchte man bei jeder Anforderung ein und nur genau einmal das Interrupt-Programm ausführen. Sowohl die I-Taste des ECB85 als auch der in dem vorstehenden Beispiel verwendete Schalter erlauben jedoch im Normalfall keinen definierten Interrupt-Betrieb, da sie prellen und in Bezug auf das Interrupt-Programm unvorhersagbar lange geschlossen sind. Die bestehenden Probleme lassen sich folgendermaßen lösen:

- a) I-Taste: Die I-Taste ist nur in Verbindung mit dem Monitor-Programm zu verwenden. Soll ein Anwender-Programm mit der I-Taste unterbrochen werden, ist in diesem Programm Interrupt generell und der Interrupt RST7.5 speziell freizugeben (Bild 8.4-4a), wie es schon im Beispiel des Kapitels 3.5 gezeigt ist. Interrupt ist während der Abarbeitung des Monitor-Programms gesperrt, und das RST7.5-Flip-Flop wird vor dem Starten eines Anwender-Programms mit einem Kommando durch das Monitor-Programm rückgesetzt. Daher spielt das Prellen der I-Taste keine Rolle.
- b) Eingang RST5.5 und RST6.5: Wenn zum Auslösen von Interrupt Schalter verwendet werden, sind diese zunächst zu entprellen, wofür Bild 8.4-8a

eine elektronische Lösung zeigt. Ferner ist zu beachten, daß die Interrupt-Anforderung bei diesen Eingängen so lange besteht, wie das Interrupt-Signal anliegt. Um eine mehrfache Ausführung des Interrupt-Programms bei länger anliegendem Anforderungssignal zu vermeiden, kann man extern ein Flip-Flop vorschalten, das mit der Vorderflanke des Anforderungssignals gesetzt und softwaremäßig z.B. über den Ausgang ANWSOD rückgesetzt werden kann. Bild 8.4-7b zeigt die Schaltung. Zum Rücksetzen muß vor dem Rücksprung aus dem Interrupt-Programm der Ausgang ANWSOD mit dem Befehl SIM kurzzeitig auf 0 und dann wieder auf 1 gesetzt werden.

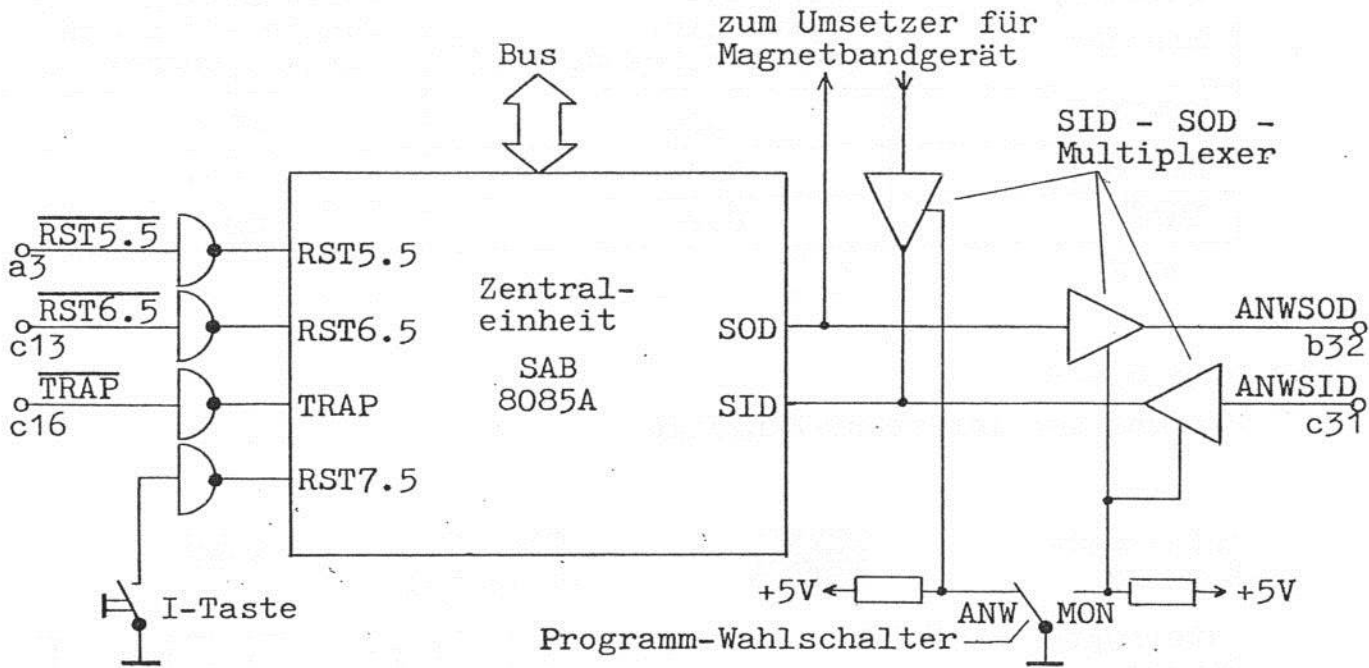
- c) Eingang TRAP: Der Eingang TRAP ist bereits intern mit einem Flip-Flop ausgerüstet, das automatisch bei Sprung in das Interrupt-Programm rückgesetzt wird. Das Interrupt-Signal kann daher beliebig lang (mindestens aber 6 μ s) sein, da nur die Vorderflanke entscheidend ist. Lediglich wenn zum Auslösen von Interrupt ein Schalter verwendet wird, ist Entprellung nach Bild 8.4-8a erforderlich.

Es ist noch darauf hinzuweisen, daß bei den Interrupt-Eingängen RST5.5, RST6.5 und TRAP dem aktiven Zustand (Interrupt-Anforderung) L-Pegel zugeordnet ist. Die Steuer-Ausgänge der seriellen Schnittstelle, die zum Anfordern von Interrupt vorgesehen sind (Kapitel 8.3), haben jedoch als aktiven Zustand H-Pegel. Will man die parallele Ein-/Ausgabe mit Interrupt betreiben, ist daher ein Inverter zwischenzuschalten, wie es Bild 8.4-8c zeigt. In diesem Fall braucht man sich um das Wegnehmen der Interrupt-Anforderung rechtzeitig mit dem Rücksprung aus dem Interrupt-Programm keine Gedanken zu machen, da der Baustein SAB8155 mit dem Einlesen (bei Eingabe-Betrieb) bzw. Ausgeben (bei Ausgabe-Betrieb) des Datenworts durch die Zentraleinheit selbsttätig das Interrupt-Anforderungssignal wieder in den Ruhezustand bringt.

Bild 8.4-1

Anschlüsse des Bausteins SAB8085A an der Anwender-Schnittstelle

a) Schaltung



b) statische Daten

INTR - allg. Interrupt

		L-Pegel *)	H-Pegel *)
Eingänge	$\overline{\text{RST 5.5}}, \overline{\text{RST 6.5}}, \text{TRAP}$	$U \leq 0,8\text{V}$ $-I \leq 0,2\text{mA}$	$U \geq 2\text{V}$ $I \leq 20\mu\text{A}$
	ANWSID	$U \leq 0,8\text{V}$ $-I \leq 0,4\text{mA}$	$U \geq 2\text{V}$ $I \leq 20\mu\text{A}$
Ausgänge	ANWSOD	$U \leq 0,4\text{V}$ bei $I=12\text{mA}$	$U \geq 2,4\text{V}$ bei $-I=2,6\text{mA}$

*) Bei den Stromangaben für Ein- und Ausgänge bedeutet ein positives Vorzeichen, daß der Strom in den ECB85 hineinfließt und ein negatives Vorzeichen, daß der Strom herausfließt. Es wird dabei die technische Stromrichtung (entgegen dem Elektronenstrom) angenommen.

Bild 8.4-2

Sprungziele bei Interrupt

fest

(im Monitor-EPROM fest)

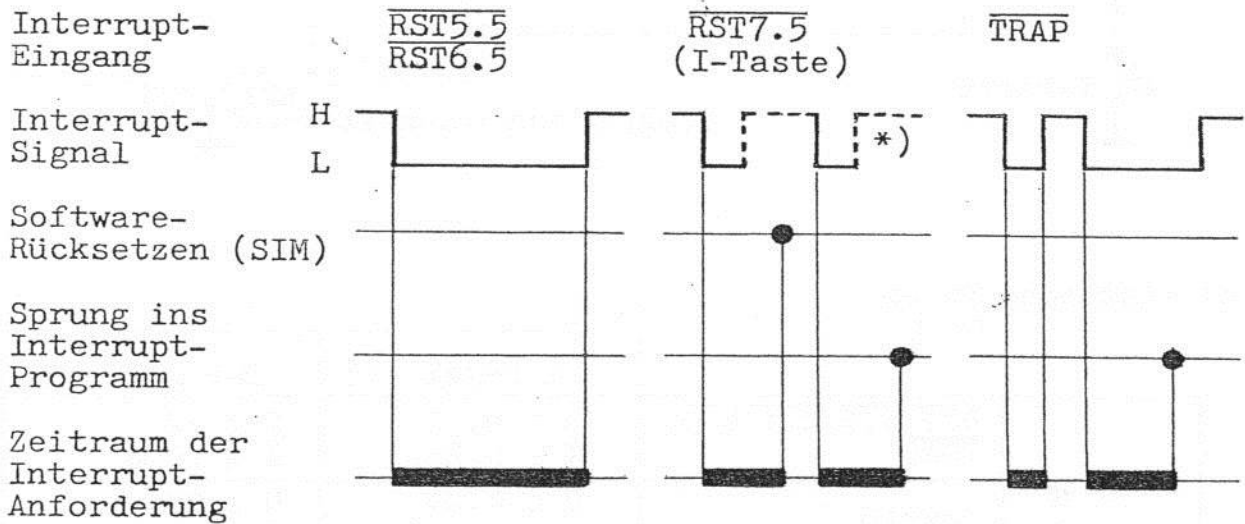
Interrupt-Eingang	direktes Sprungziel (Adresse, hexadezimal)	Weitersprung durch Monitor-Programm (Adresse, hexadezimal)
$\overline{\text{RST 5.5}}$	002C	082C
$\overline{\text{RST 6.5}}$	0034	0834
$\overline{\text{TRAP}}$	0024	0824

INTR

beliebig 00..FF muß auf die Datenleitung kommen

Bild 8.4-3

Wirkung der Interrupt-Eingänge



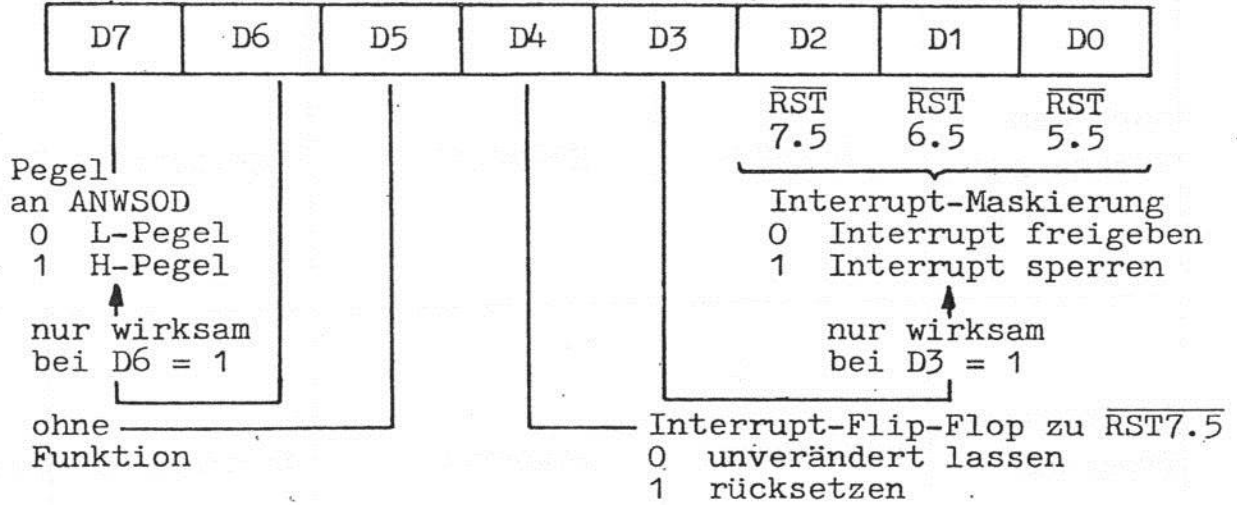
*) für die I-Taste gilt folgende Zuordnung:

- Taste in Ruhe - $\overline{\text{RST 7.5}}$ = H-Pegel
- Taste gedrückt - $\overline{\text{RST 7.5}}$ = L-Pegel

Schreib-Interrupt Mask mit Inhalt des Flip-Flop
Read Interrupt Mask in App.

Bild 8.4-4
 Befehle SIM und RIM

a) Datenwort beim Befehl SIM



b) Datenwort beim Befehl RIM

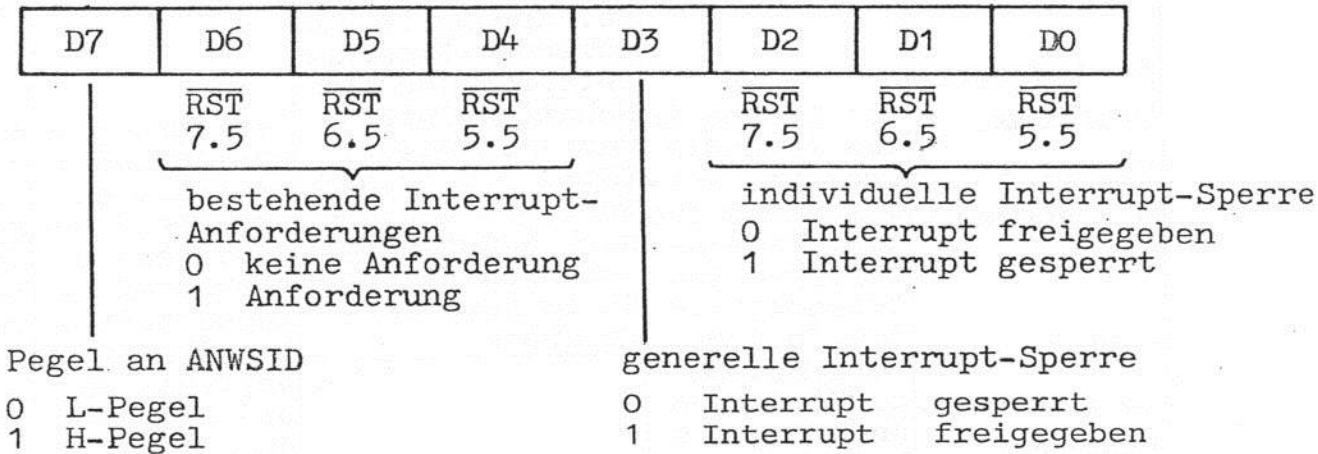


Tabelle 8.4-5a

Aktueller Status des Ausgangs ANWSOD und des Interrupt-Systems

Bedingung	Pegel an Ausgang ANWSOD	Interrupt-System generell	Interrupt-Eingänge individuell	Zustand RST 7.5 - Flip-Flop *
nach dem Rücksetzen	L-Pegel	gesperrt	gesperrt	rückgesetzt
im Monitor-Programm	unverändert	gesperrt	unverändert	unverändert
im Anwender-Programm, gestartet mit Kommando 7	Bei Sprung in das Anwender-Programm entsprechend dem Schatten-Interrupt-Statuswort (Bild 3.8-3b, SIM-Wort, Bit D6 u. D7), danach Änderung mit Befehl SIM im Anwender-Programm	Bei Sprung in das Anwender-Programm entsprechend dem Schatten-Interrupt-Statuswort (Bild 3.8-3b, RIM-Wort, Bit D3), danach Änderung mit Befehlen EI und DI im Anwender-Programm	Bei Sprung in das Anwender-Programm entsprechend dem Schatten-Interrupt-Statuswort (Bild 3.8-3b, SIM-Wort, Bit D0 bis D3), danach Änderung mit Befehl SIM im Anwender-Programm	Bei Sprung in das Anwender-Programm entsprechend dem Schatten-Interrupt-Statuswort (Bild 3.8-3b, SIM-Wort, Bit 4) danach Änderung mit Befehl SIM im Anwender-Programm
im Anwender-Programm, gestartet mit Kommando 8 oder 9		gesperrt		
nach Interrupt-Sprung	unverändert	gesperrt	unverändert	rückgesetzt bei Annahme des Interrupts RST7.5

*) Setzen jederzeit durch Drücken der I-Taste möglich

Tabelle 8.4-5b

Schatten-Status des Ausgangs ANWSOD und des Interrupt-Systems

Bedingung	SIM-Wort 1)	RIM-Wort 1)
nach dem Rücksetzen	neutraler Wert 2)	entsprechend aktuellem Status (Tabelle 8.4-5a)
im Monitor-Programm außer bei Kommando 6	unverändert	unverändert
im Monitor-Programm bei Kommando 6	Änderung durch Kommando 6	
im Anwender-Programm, gestartet mit Kommando 7	Nachdem vor dem Sprung in das Anwender-Programm das Monitor-Programm das SIM-Wort durch Ausführung eines Befehls SIM auf die Hardware übertragen hat, setzt es das SIM-Wort im Schattenbereich auf den neutralen Wert 00	Nach Unterbrechen des Anwender-Programms (Drücken der I-Taste bei Kommando 7, Erreichen eines Unterbrechungspunktes bei Kommando 8, Beendigung der Befehlsausführung bei Kommando 9) führt das Monitor-Programm einen Befehl RIM aus und speichert das Datenwort im Schatten-Interrupt-Statusregister (Bild 3.8-3b)
im Anwender-Programm, gestartet mit Kommando 8 oder 9		

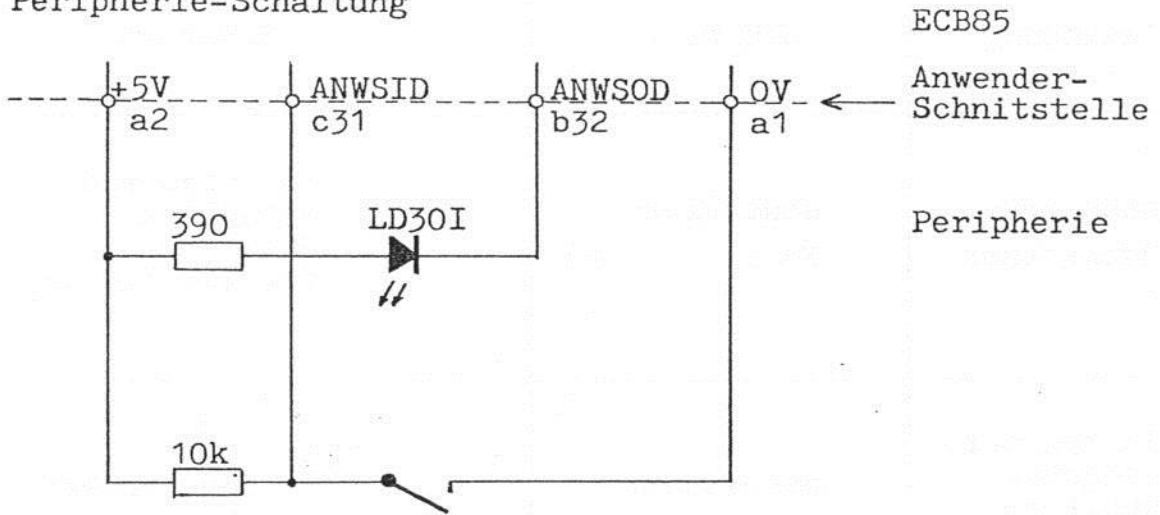
1) Zur Definition siehe Bild 3.8-3b

2) Der neutrale Wert des SIM-Wortes ist 00, da damit der Ausgang ANWSOD, das RST 7.5-Flip-Flop, und die individuellen Interrupt-Eingänge unbeeinflusst bleiben (Bild 3.8-3b)

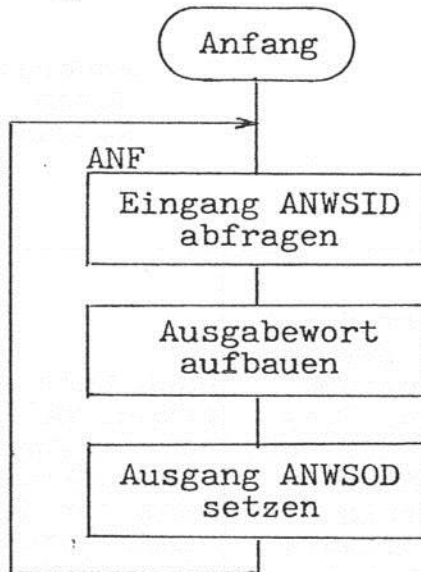
Bild 8.4-6

Demonstrationsbeispiel für ANWSID und ANWSOD

a) Peripherie-Schaltung



b) Ablaufplan des Programms



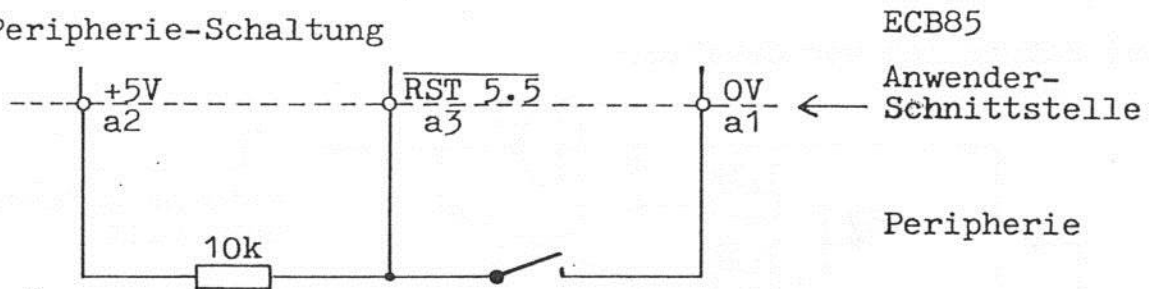
c) Programm

Hexadezi- mal-Adresse	Hexadezi- mal-Befehl	Assembler-Befehl	Kommentar
1800	20	ANF: RIM	;Eingang ANWSID abfragen
1801	E680	ANI 80H	;Ausgabewort
1803	C640	ADI 40H	; aufbauen
1805	30	SIM	;Ausgang ANWSOD setzen
1806	C30018	JMP ANF	;Schleife

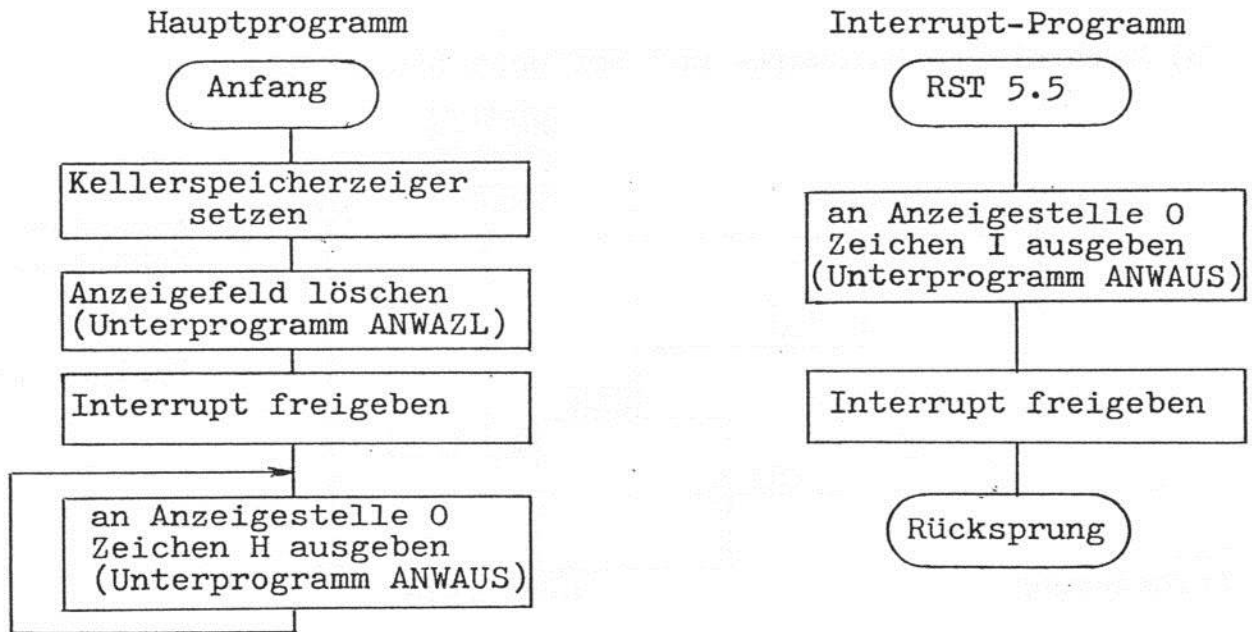
Bild 8.4-7

Demonstrationsbeispiel für Interrupt

a) Peripherie-Schaltung



b) Ablaufpläne der Programme



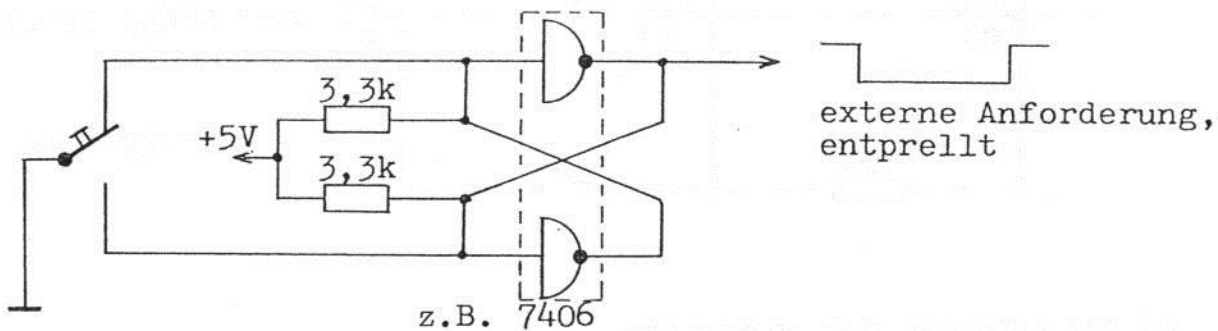
c) Programme

Hexadezi- mal-Adresse	Hexadezi- mal-Befehl	Assembler-Befehl	Kommentar
		;Hauptprogramm	
1800	31FF1B	ANF: LXI SP,1BFFH	;Kellerspeicherzeiger ; setzen
1803	CDBC07	CALL 07BCH	;Anzeigefeld löschen ;(Unterprogramm ANWAZL)
1806	3E0E	MVI A,0EH	;Interrupt RST 5.5 frei
1808	30	SIM	; freigeben
1809	FB	EI	
180A	3E76	SCHL:MVI A,76H	;an Anzeigestelle 0
180C	0600	MVI B,0	; Zeichen H ausgeben
180E	CDC707	CALL 07C7	; (Unterprogramm ANWAUS)
1811	C30A18	JMP SCHL	;Schleife
		;Interrupt-Programm	
082C	3E06	MVI A,6	;an Anzeigestelle 0
082E	0600	MVI B,0	; Zeichen I ausgeben
0830	CDC707	CALL 07C7H	; (Unterprogramm ANWAUS)
0833	FB	EI	;Interrupt freigeben
0834	C9	RET	;Rücksprung

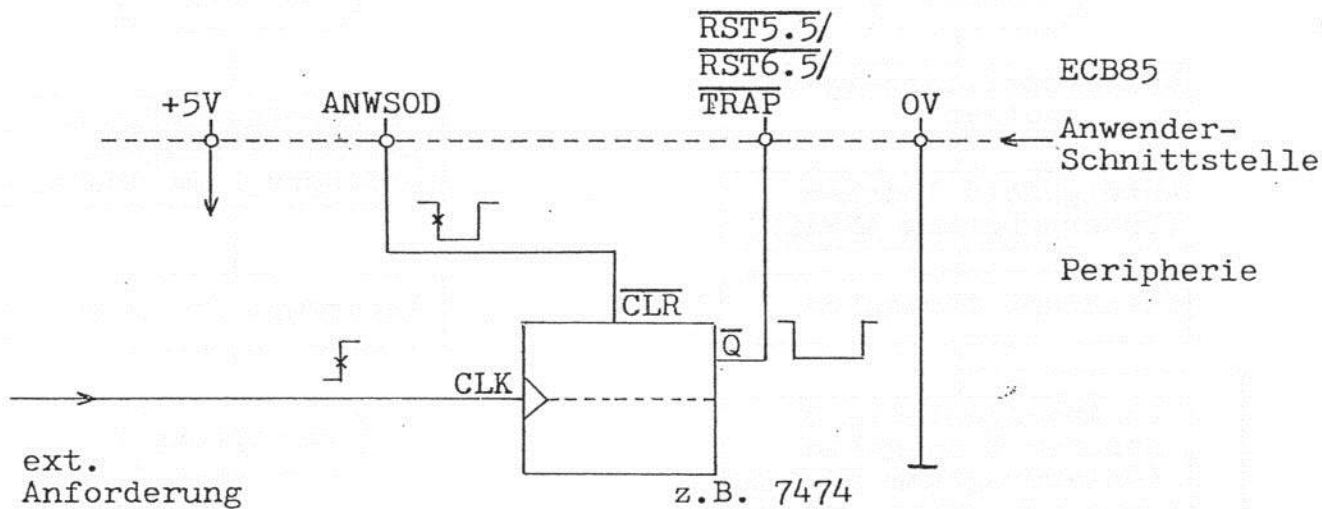
Bild 8.4-8

Besondere Maßnahmen in Zusammenhang mit den Interrupt-Eingängen

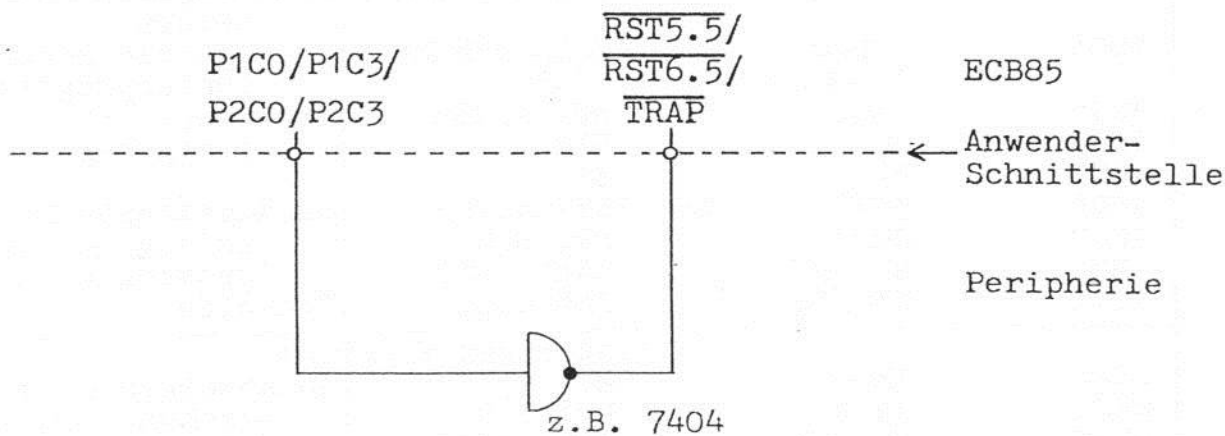
a) Entprellen von Schaltern



b) Kombinierte Hardware- und Software-Steuerung



c) Betrieb der parallelen Ein-/Ausgabe mit Interrupt



9. Technische Unterlagen und Arbeitshilfen

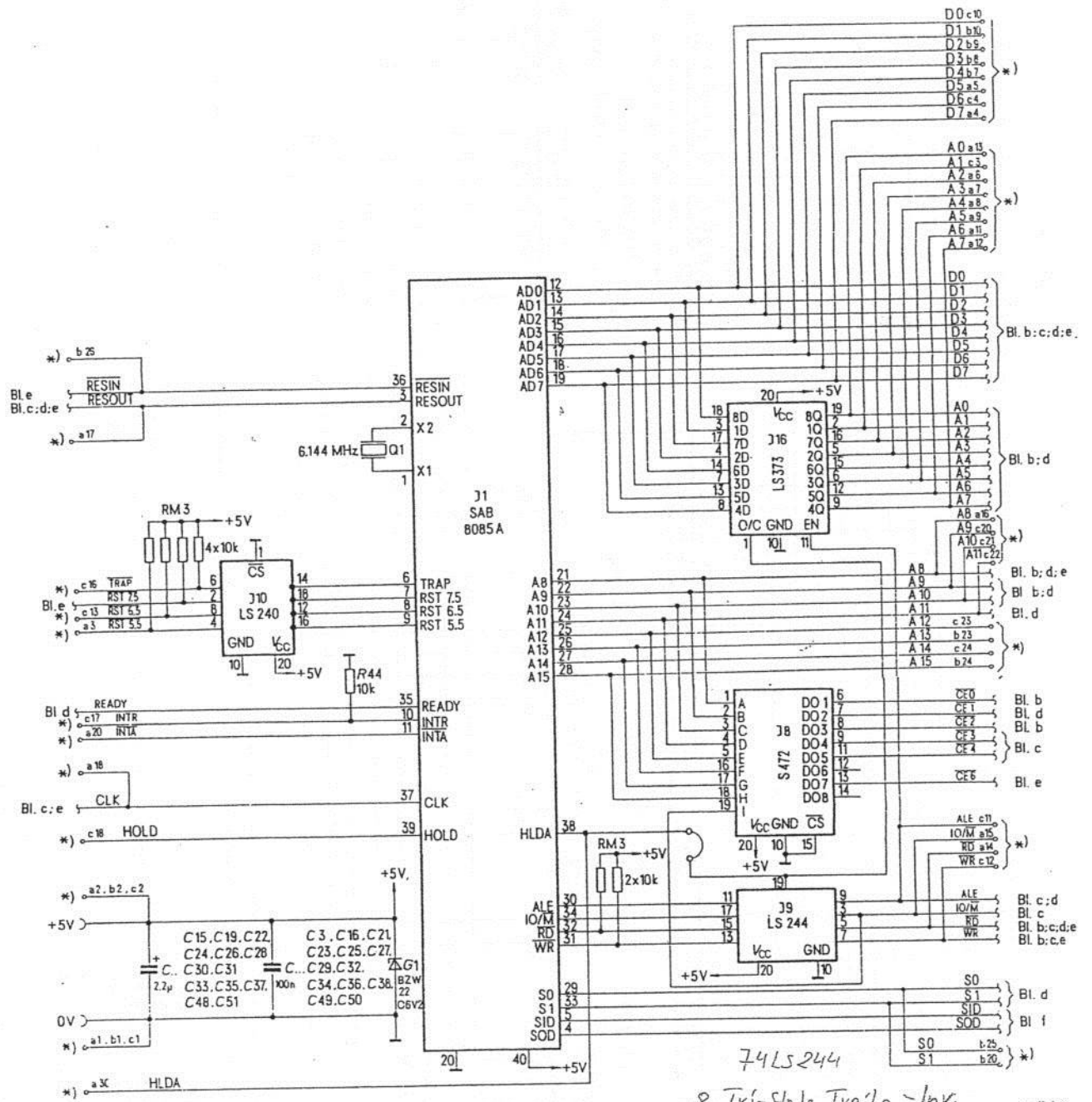
In den folgenden Kapiteln sind die technischen Unterlagen des EC885 und nützliche Arbeitshilfen zusammengefaßt. Zunächst werden zur Hardware die ausführlichen Stromlaufpläne gebracht. Darauf folgt die Software mit dem kompletten Listing des Monitor-Programms. Am Schluß ist noch eine Übersetzungstabelle für Assembler-Befehle angefügt, die bei der manuellen Assemblierung von Anwender-Programmen die Arbeit erleichtert.

9.1 Hardware-Unterlagen

Der ausführliche Stromlaufplan des ECB85 ist in Bild 9.1-1a bis 9.1-1f dargestellt. Der Belegungsplan ist in Bild 9.1-2 angegeben. Mit Hilfe dieser Unterlagen läßt sich für Messungen jeder interne Punkt des Geräts lokalisieren.

Bild 9.1-1a

Stromlaufplan, Teil: Zentraleinheit mit Hilfsschaltungen

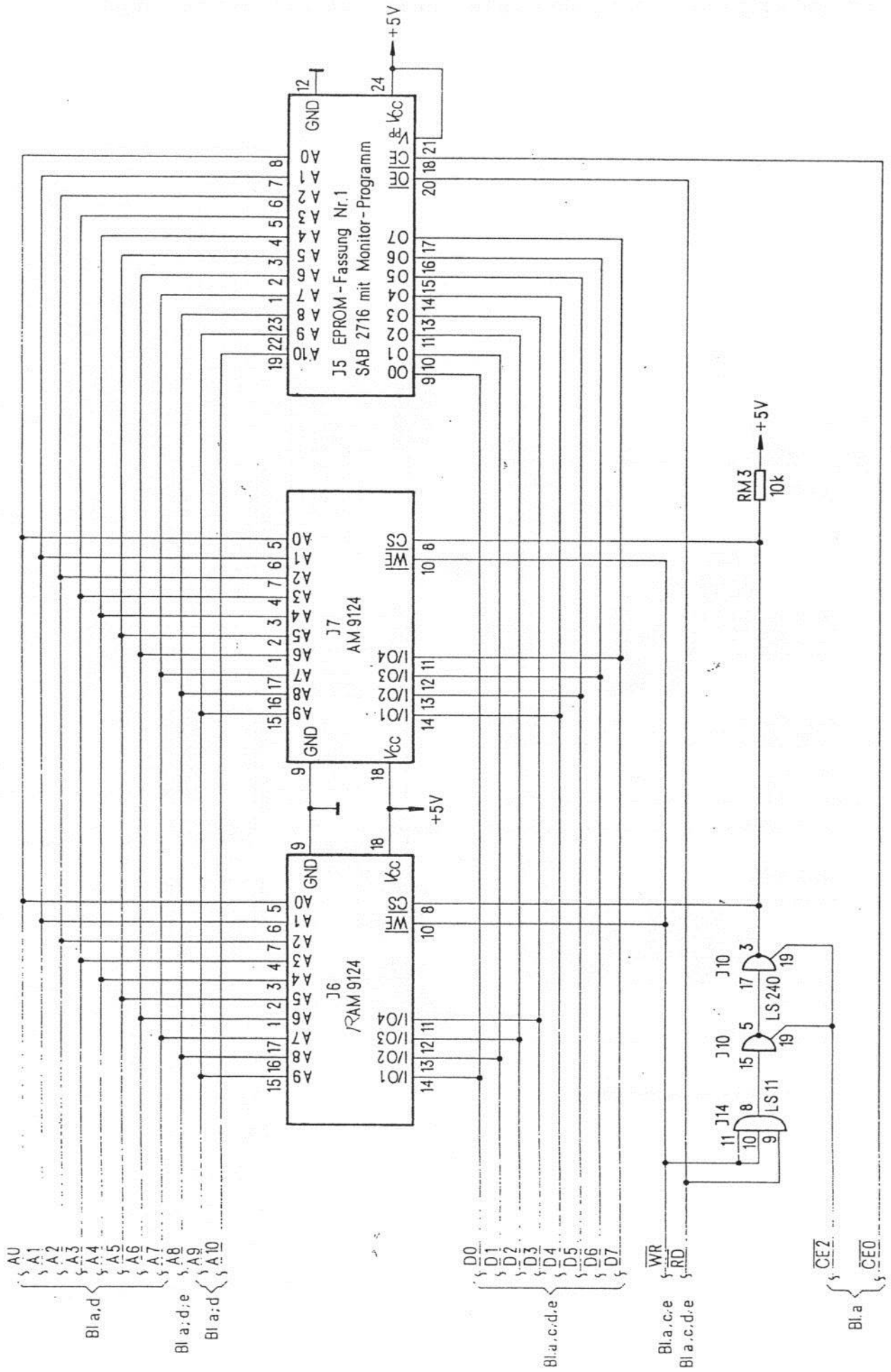


*) Anschlüsse der Anwender-Schnittstelle

- 74LS244
- 8 Tri-State Treiber - Inv.
- 74LS472
- PROM 512x8
- 74LS373
- 8 FF-Speicher mit Treiber (Tri-Stat)
- 74LS240
- 8 Tri-State Treiber

Bild 9.1-1b

Stromlaufplan, Teil: Schreib-/Lese-Speicher (Teilfunktion) und EPROM Nr. 1

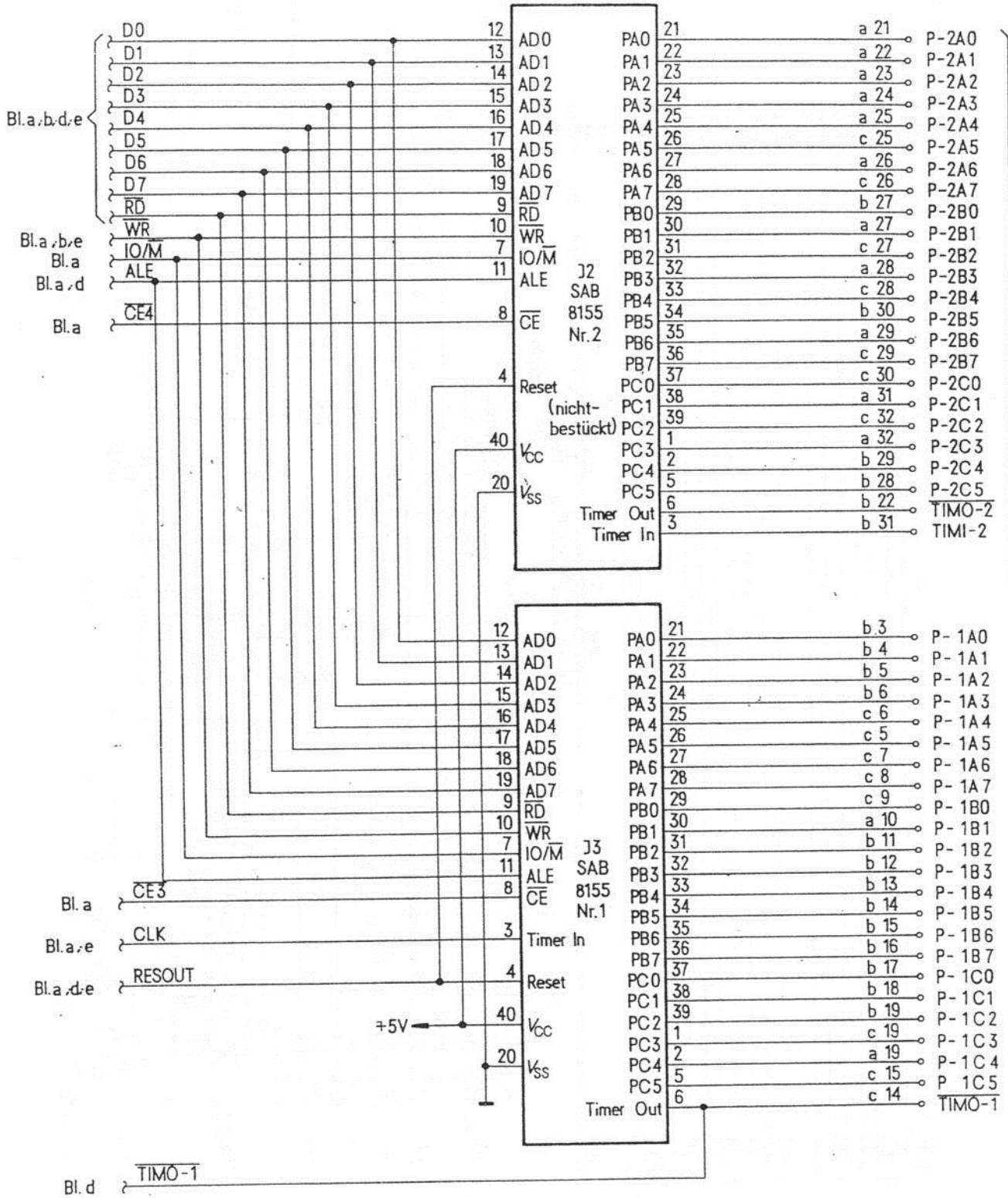


44-5240
& Typ-5-2e - 1981

3 r d 3-Fing-Singe

Bild 9.1-1c

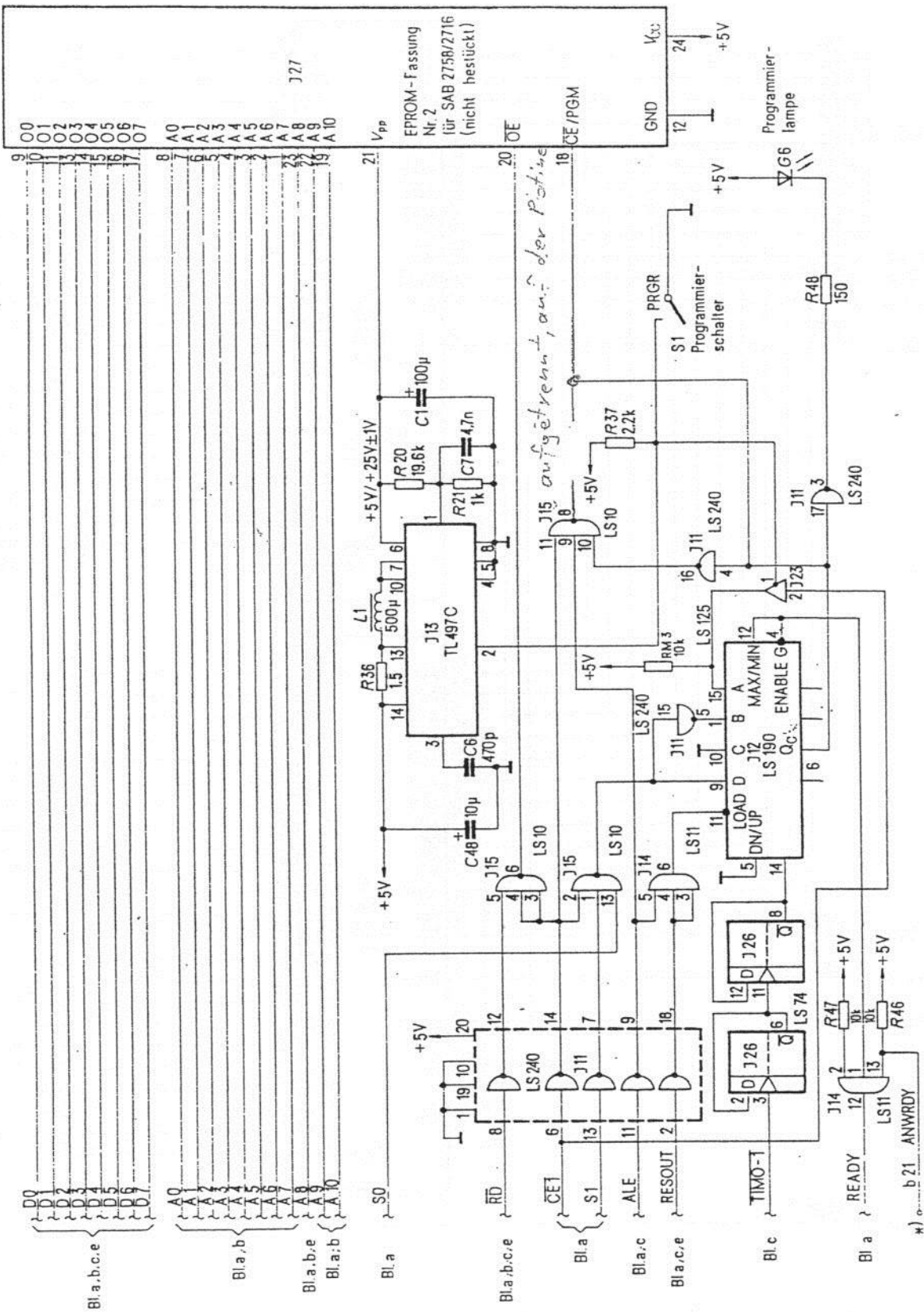
Stromlaufplan, Teil: Schreib-/Lese-Speicher (Teilfunktion),
parallele Ein-/Ausgabe, Zähler



*) Anschlüsse der Anwender-Schnittstelle

Bild 9.1-1d

Stromlaufplan, Teil: EPROM Nr. 2 und EPROM-Programmierer



*) Anschlüsse der Anwender-Schmittstelle

Bild 9.1-1e

Stromlaufplan, Teil: Tasten-/Anzeige-Interface mit Tasten- und Anzeigefeld

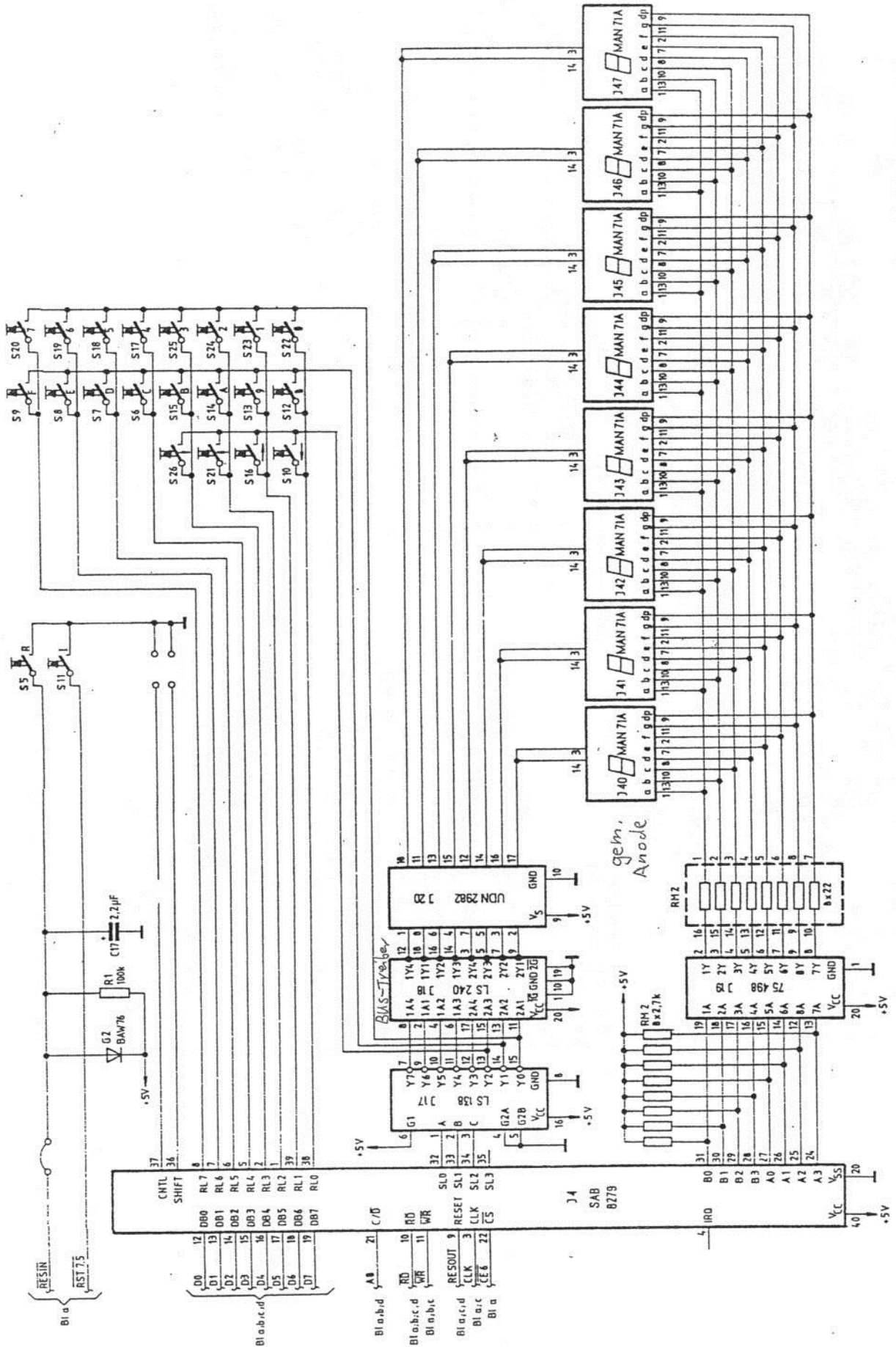
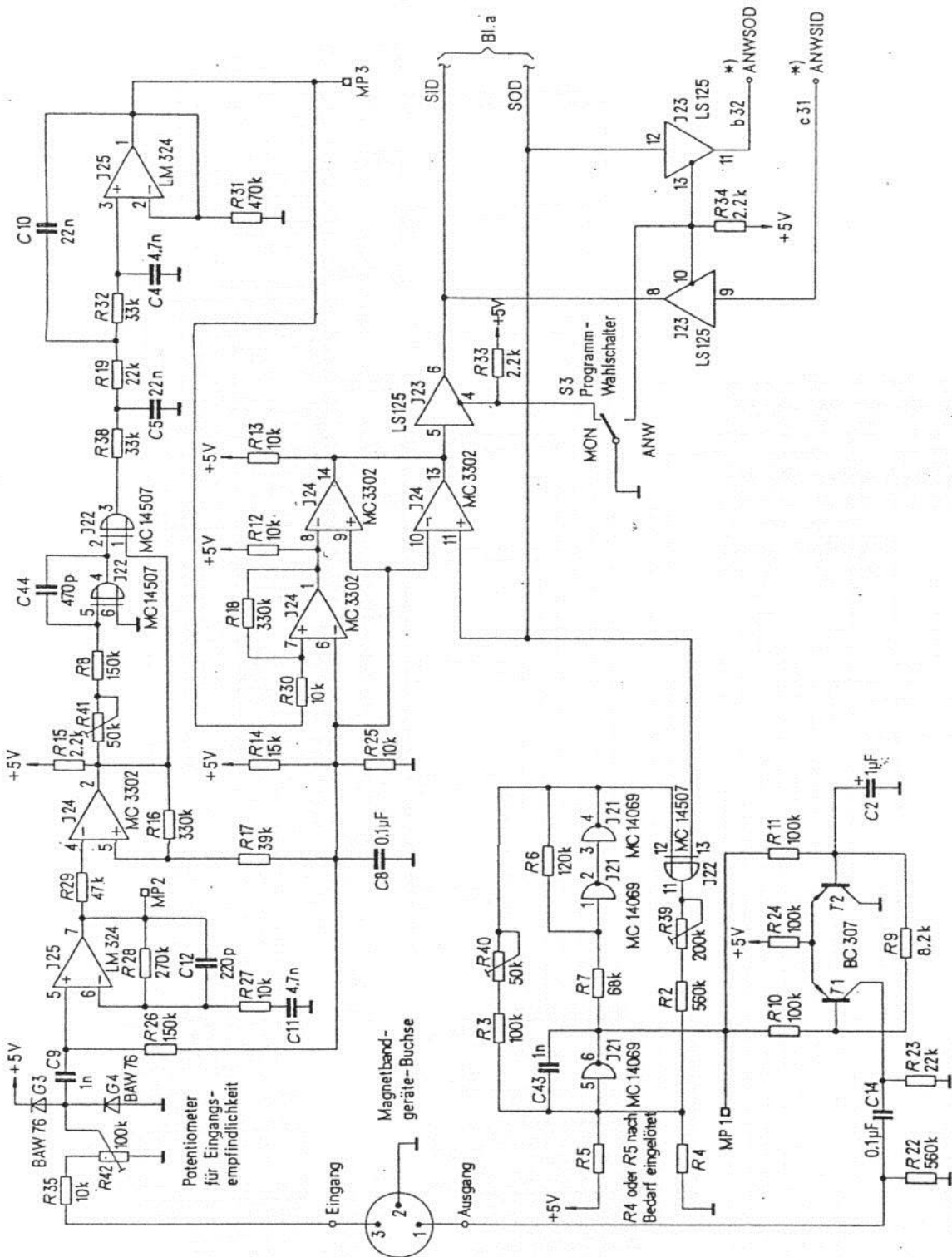


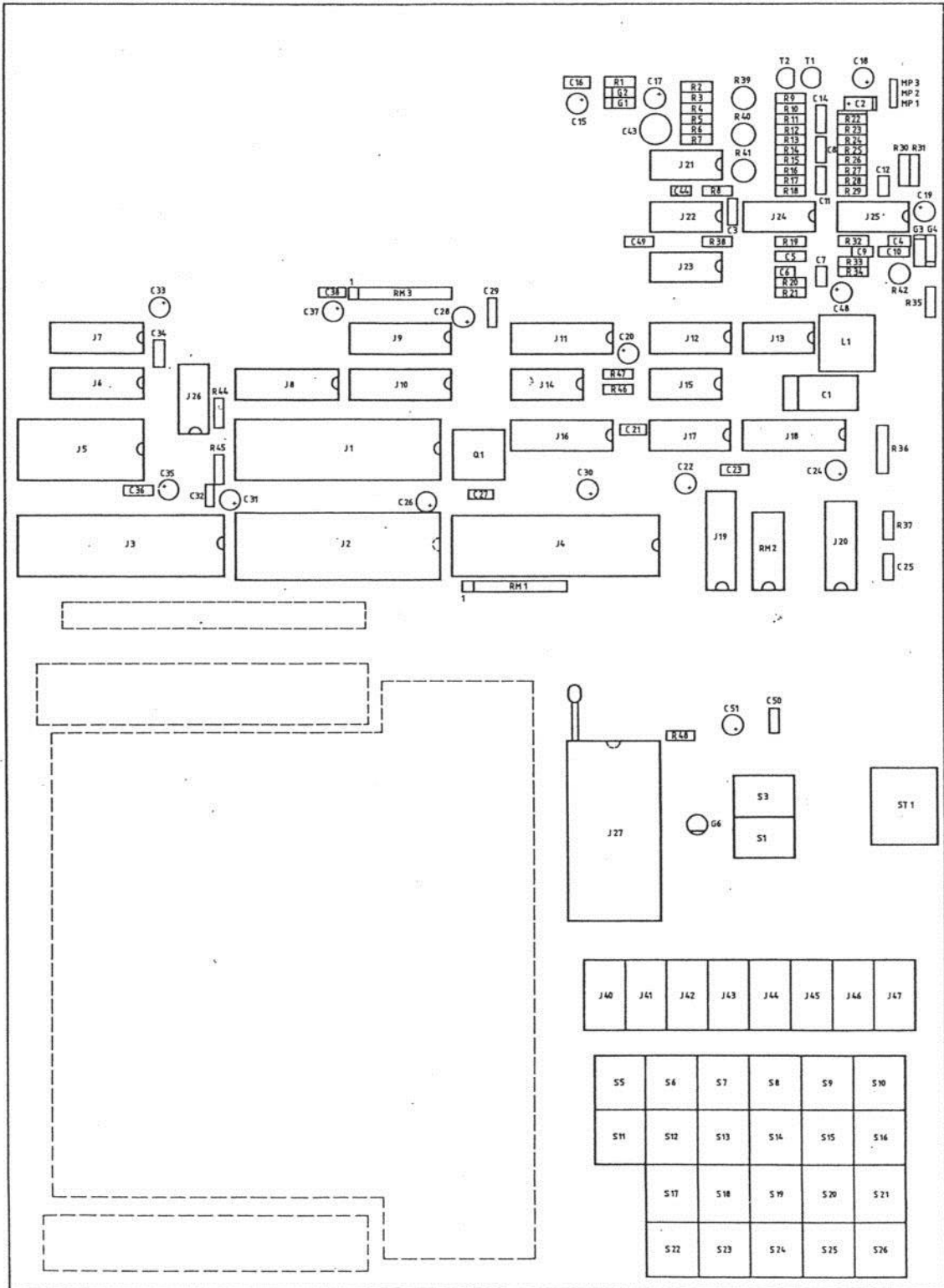
Bild 9.1-1f

Stromlaufplan, Teil: Umsetzer für Magnetbandgerät



**) Anschlüsse der Anwender - Schnittstelle

Bild 9.1-2
Belegungsplan



9.2 Software-Unterlagen

Tabelle 9.2-1 bringt das vollständige Listing des Monitor-Programms. Das Listing enthält detaillierte Kommentare, so daß der etwas erfahrene Mikrocomputer-Anwender daraus die Wirkungsweise entnehmen kann. Das Monitor-Programm befindet sich bei Lieferung in einem Baustein SAB2716 auf dem ECB85.

Tabelle 9.2.1

Vollständiges Listing des Monitor-Programms

ASM80 :F1:M8521.SRC NOOBJECT PRINT(:F1:M8521.LP) MOD85 XREF

ISIS-II 8080/8085 MACRO ASSEMBLER, V4.0 MODULE PAGE 1

```

LOC  OBJ          LINE      SOURCE STATEMENT
      1 $PAGELENGTH(63)
      2 ;#####
      3 ;#
      4 ;#          M O N I T O R - P R O G R A M M   Z U   E C B 8 5          #
      5 ;#
      6 ;#          V E R S I O N :   V 2 . 1       D A T U M :   1 4 . 0 3 . 8 0   #
      7 ;#
      8 ;#####
      9
     10
     11 ;*****
     12 ;
     13 ;          O R G A N I S A T O R I S C H E R   T E I L
     14 ;
     15 ;*****
     16
     17 ;-----
     18 ; ORGANISATION DES ARBEITSSPEICHERS
     19 ; *****
     20 ; -DER MONITOR BENUTZT IM BAUSTEIN SAB8155/0 DIE LETZTEN 76 BYTE DES
     21 ; RAM-SPEICHERS (ADRESSE FFB4H BIS FFFFH). DIESER RAMBEREICH DARF
     22 ; VOM ANWENDER NICHT BENUTZT WERDEN, WENN DAS MONITORPROGRAMM BENUTZT
     23 ; WIRD.
     24 ; -DER RAMBEREICH VON FFOOH BIS FFB3H STEHT DEM ANWENDER ZUR VERFUE-
     25 ; GUNG.
     26 ;-----
     27
FFB4  28          ORG      OFFB4H          ;ANFANGSADRESSE DES VOM MONITOR BE-
      29 MONRAM:          ;      NUTZTEN RAM-SPEICHER
      30
      31          ;RETTZELLEN ZUR STATUS-ABSPEICHERUNG BEI UNTERBRECHUNG DES
      32          ;*****
      33          ;ANWENDERPROGRAMMS.
      34          ;*****
FFB4  35 FRET:  DS      1          ;RETTZELLE FUER FLAGS
FFB5  36 ARET:  DS      1          ;RETTZELLE FUER AKKU
FFB6  37 CRET:  DS      1          ;RETTZELLE FUER REGISTER C
FFB7  38 BRET:  DS      1          ;RETTZELLE FUER REGISTER B
FFB8  39 ERET:  DS      1          ;RETTZELLE FUER REGISTER E
FFB9  40 DRET:  DS      1          ;RETTZELLE FUER REGISTER D
FFBA  41 LRET:  DS      1          ;RETTZELLE FUER REGISTER L
FFBB  42 HRET:  DS      1          ;RETTZELLE FUER REGISTER H
FFBC  43 PRET:  DS      2          ;RETTZELLE FUER PROGRAM COUNTER
FFBE  44 SRET:  DS      2          ;RETTZELLE FUER STACKPOINTER
FFCO  45 IRET:  DS      1          ;RETTZELLE FUER INT.-STATUS (VON RIM)
FFC1  46          DS      1          ;RETTZELLE FUER INT.-STATUS (FUER SIM)
      47
      48          ;ADRESSEN DER MONITOR-PARAMETER
      49          ;*****
FFC2  50 BEFREG: DS      8          ;BEFEHLSREGISTER
FFCA  51 REGAD:  DS      1          ;REGISTER-ZEIGER

```

LOC	OBJ	LINE	SOURCE STATEMENT
FFCB		52	MEMAD: DS 3 ;SPEICHER-ZEIGER
FFCE		53	AA: DS 2 ;ANF-ADRESSE
FFD0		54	EA: DS 2 ;END-ADRESSE
FFD2		55	BA: DS 2 ;BESTIMMUNGS-ADRESSE
FFD4		56	H1: DS 2 ;HALTADRESSE 1
FFD6		57	H2: DS 2 ; " 2
FFD8		58	AU: DS 2 ;ANF.ADR UMZURECHN. BEREICH
FFDA		59	EU: DS 2 ;END " " "
FFDC		60	DAT: DS 1 ;DATENBYTE
FFDD		61	ADR: DS 1 ;ADRESSBYTE
FFDE		62	MODUS: DS 1 ;MODUS KENNZELLE
FFDF		63	KOMZE: DS 1 ;KOMMANDO ZELLE
FFEO		64	RAMHZ: DS 1 ;HILFSZELLE FUER ZEICHEN AN
		65	; SCHREIBMARKENPOSITION
FFE1		66	DEFAUL: DS 1 ;DEFAULTWERT KENNZELLE
FFE2		67	DS 30 ;STACK FUER MAX. SCHACHTELUNGSTIEFE=15
		68	STKINI: ;ADRESSE ZUM SETZEN DES STACKPOINTERS
		69	
		70	
		71	;
		72	; KONSTANTENDEFINITIONEN, PARAMETER UND EIN-/AUSGABE-ADRESSEN
		73	;
		74	
		75	;EIN-/AUSGABEADRESSEN
		76	;*****
00EE		77	DATEA EQU DEEH ;DATENADR. SAB 8279
00EF		78	KOSEA EQU DEFH ;KOMMANDO/STATUS-ADRESSE SAB 8279
00FD		79	TIMFH EQU OFDH ;TIMERADR. SAB 8155 /0 (MSB)
00FC		80	TIMEL EQU OFCH ; " " " (LSB)
00F8		81	KOADR EQU OFBH ;KOMMANDO/STATUSADR. SAB 8155/0
		82	
		83	;PARAMETER FUER MAGNETBAND-EIN/AUSGABE
		84	;*****
0269		85	BITTIME EQU 0269H ;BITZEIT FUER 600 BIT/S
01A5		86	HALFBIT EQU 01A5H ;HALBE BITZEIT FUER 600 BIT/S
000B		87	BITSO EQU 11 ;ANZAHL DER AUSZUGEB.BITS
		88	; INKL. ZWEI STOP- UND EINEM STARTBIT
0009		89	BITSI EQU 9 ;ANZAHL DER ZU LESENDEN BITS
		90	; INKL. EIN STOPBIT
		91	
		92	\$EJECT

```

LOC  OBJ          LINE          SOURCE STATEMENT
-----
          93 ;-----
          94 ; INTERRUPTVEKTORADRESSEN MIT SPRUNGTABELLE FUER RST1 - RST6.5, TRAP
          95 ; *****
          96 ; -BEI DIESEN RST-BEFEHLEN UND DEM TRAP-INTERRUPT ERFOLGT EIN SPRUNG
          97 ; INS ANWENDER-EPROM.
          98 ; -ES GILT FOLGENDE ADRESSENZUORDNUNG:  RST1 --> 808H
          99 ;                                     RST2 --> 810H
         100 ;                                     RST3 --> 818H
         101 ;                                     RST4 --> 820H  TRAP  --> 824H
         102 ;                                     RST5 --> 828H  RST5.5 --> 832H
         103 ;                                     RST6 --> 830H  RST6.5 --> 834H
         104 ;-----
         105
0008          106      ORG      08H      ;ADRESSE FUER RST 1
0008 C30808    107      JMP      808H
0010          108      ORG      10H      ;ADRESSE FUER RST 2
0010 C31008    109      JMP      810H
0018          110      ORG      18H      ;ADRESSE FUER RST 3
0018 C31808    111      JMP      818H
0020          112      ORG      20H      ;ADRESSE FUER RST 4
0020 C32008    113      JMP      820H
0024          114      ORG      24H      ;ADRESSE FUER TRAP
0024 C32408    115      JMP      824H
0028          116      ORG      28H      ;ADRESSE FUER RST 5
0028 C32808    117      JMP      828H
002C          118      ORG      2CH      ;ARDESSE FUER RST 5.5
002C C32C08    119      JMP      82CH
0030          120      ORG      30H      ;ADRESSE FUER RST 6
0030 C33008    121      JMP      830H
0034          122      ORG      34H      ;ADRESSE FUER RST 6.5
0034 C33408    123      JMP      834H
         124
         125
         126 ;-----
         127 ; VEKTORADRESSEN FUER RST7 UND RST7.5 UND VERARBEITUNG DIESER RESTARTS
         128 ; *****
         129 ; - RST7  BEI ANTREFFEN EINES RST7-BEFEHLS IN EINEN ANWENDER-PROGRAMM
         130 ; - RST7.5 BEI DRUECKEN DER I-TASTE
         131 ;-----
         132
0038          133      ORG      38H      ;ADRESSE FUER RST7
         134
0038 E3        135 RST7:  XTHL          ;PROGRAMM-COUNTER IN ANW.-STACK UM EINS
0039 2B        136      DCX      H          ; ERNIEDRIGEN (WEGEN RST 7 BEFEHL)
003A E3        137      XTHL          ;
003B F3        138      DI          ;UNTERBINDE INTERRUPTS
         139
003C          140      ORG      3CH      ;ADRESSE FUER RST7.5
         141
         142      ;STATUS DER CPU RETTEN
         143      ;*****
003C 22BAFF    144 RST75: SHLD  LRET  ;H & L  RETTEN

```


LOC	OBJ	LINE	SOURCE STATEMENT
003F	E1	145	POP H ;HOLE PC VOM ANWENDER STACK
0040	22BCFF	146	SHLD PRET ;PC RETTEN
0043	F5	147	PUSH PSW ;AKKU & FLAGS AUF ANW.STACK
0044	E1	148	POP H ;AKKU & FLAGS VON ANW.STACK IN H&L
0045	22B4FF	149	SHLD FRET ;AKKU & FLAGS RETTEN
0048	210000	150	LXI H,0 ;ANWENDER STACKZEIGER RETTEN UEBER H&L
004B	39	151	DAD SP
004C	22BEFF	152	SHLD SRET
004F	21BAFF	153	LXI H,DRET+1;STACKZEIGER ZUM RETTEN DER UEBRIGEN REGISTER
0052	F9	154	SPHL ; SETZEN
0053	D5	155	PUSH D ;D & E RETTEN
0054	C5	156	PUSH B ;B & C RETTEN
0055	20	157	RIM ;HOLE INTERRUPT STATUS & MASKE
0056	F608	158	ORI 08H ;SETZE EI-BIT
0058	32C0FF	159	STA IRET ;RETTE INTERRUPT STATUS & MASKE
005B	97	160	SUB A ;LOESCHE INT.-STATUS FUER SIM
005C	32C1FF	161	STA IRET+1
005F	C39ADD	162	JMP ANFANG ;SPRUNG INS MONITOR-PROGRAMM
		163	
		164	NEXADR: ;NAECHSTE ADRESSE IM PROGRAMMSPEICHER
		165	
		166	
		167	-----
		168	; RESET- UND RSTO-VERARBEITUNG
		169	;
		170	; - BEIM RUECKSETZEN BZW. RSTO-BEFEHL WIRD UEBER DIE SERIELLE EIN-/
		171	; AUSGABE DES SAB 8085 DER STATUS DES ANWENDER/MONITOR-WAHLSCHALTERS
		172	; ABGEFRAGT.
		173	; - SCHALTERSTELLUNG: MONITOR --> SPRUNG INS MONITORPROGRAMM
		174	; ANWENDER --> SPRUNG INS ANWENDER-EPROM
		175	; (ADRESSE 800H)
		176	; - NACH DER SCHALTERABFRAGE IST SOD=0
		177	-----
		178	
0000		179	ORG 00H ;ADRESSE FUER RESET BZW. RSTO
		180	
0000	3E00	181	START: MVI A,11000000B ;1 --> SOD
0002	C36200	182	JMP START1 ;UEBERSPRINGE INTERRUPTVEKTORADRESSEN UND
		183	;PROGRAMME
		184	ORG NEXADR
0062	30	185	START1: SIM
0063	00	186	NOP ;
0064	00	187	NOP
0065	00	188	NOP
0066	00	189	NOP
0067	00	190	NOP
0068	00	191	NOP
0069	20	192	RIM ;SID LESEN
006A	07	193	RLC
006B	3E40	194	MVI A,01000000B ;0 --> SOD
006D	30	195	SIM
006E	D20008	196	JNC 800H ;WAR SID=1 ?

LOC	OBJ	LINE	SOURCE STATEMENT
		197	;NEIN, SPRUNG INS ANWENDER EPROM
0071	00	198	NOP ;
0072	00	199	NOP
0073	00	200	NOP
0074	00	201	NOP
0075	00	202	NOP
0076	00	203	NOP
0077	20	204	RIM ;JA, LESE SID
0078	07	205	RLC
0079	DA0008	206	JC 800H ;WAR SID=0 ?
		207	;NEIN, SPRUNG INS ANWENDER-EPROM
		208	;JA, WEITER IM MONITOR HAUPTPROGRAMM
		209	
		210	
		211	;*****
		212	;
		213	; H A U P T P R O G R A M M
		214	;
		215	;*****
		216	
		217	;-----
		218	; INITIALISIERUNG FUER MONITORPROGRAMM
		219	;-----
		220	
007C	3E0F	221	MVI A,0FH ;KOMMANDO 'F' LADEN
007E	32DFFF	222	STA KOHZE
0081	20	223	RIM ;INTR.-STATUS-REG. INITIALISIEREN
0082	32C0FF	224	STA IRET
0085	97	225	SUB A
0086	32C1FF	226	STA IRET+1
0089	21CEFF	227	LXI H,AA ;RAM BEREICH VON ADR'AA' - 'ADR' LOESCHEN
008C	0610	228	MVI B,16
008E	77	229	LOES: MOV M,A
008F	23	230	INX H
0090	05	231	DCR B
0091	C28E00	232	JNZ LOES
		233	
		234	;SAB 8279 INITIALISIEREN
		235	;*****
0094	310000	236	CLRALL: LXI SP,STKINI ;STACKPONTER SETZEN FUER UP-AUFRUF
0097	CDBC07	237	CALL ANWAZL
009A	97	238	ANFANG: SUB A ;BETRIEBSART SETZEN
009B	D3EF	239	OUT KOSEA ;/ACHT 8-BIT-ZEICHEN ANZEIGE
		240	;/LINKSEINGABE
		241	;/KODIERTE TASTATUR-ABTASTUNG
		242	;/"2-KEY-LOCKOUT"
009D	3EC2	243	MVI A,DC2H ;TASTEN-FIFO LOESCHEN
009F	D3EF	244	OUT KOSEA
		245	
		246	;ANFANGSZUSTAND HERSTELLEN
		247	;*****
00A1	310000	248	LXI SP,STKINI ;STACKPONTER SETZEN

LOC	OBJ	LINE	SOURCE STATEMENT
00A4	168D	249	MVI D,80H ;SCHREIBMARKE AUF KOMMANDOFELD
00A6	1E7F	250	MVI E,7FH ;SCHREIBMASKE SETZEN
00A8	3ADFFF	251	LDA KOMZE ;ERLAUBTES KOMMANDO ?
00AB	FE0C	252	CPI OCH
00AD	D2CB00	253	JNC ANFA1 ;NEIN, SPRUNG
00B0	4F	254	MOV C,A ;JA, KOMMANDO MIT DEZIMALPUNKT ANZEIGEN
00B1	CDC905	255	CALL UMW
00B4	F680	256	ORI 80H
00B6	42	257	MOV B,D
00B7	CDA805	258	CALL AUS
00BA	79	259	MOV A,C ;KOMMANDO 7 ?
00BB	D607	260	SUI 7
00BD	CAD400	261	JZ ANFA2 ;JA, PROGRAM-COUNTER ANZEIGEN
00C0	3D	262	DCR A ;NEIN, KOMMANDO 8 ?
00C1	CAD400	263	JZ ANFA2 ;JA, PROGRAM-COUNTER ANZEIGEN
00C4	3D	264	DCR A ;NEIN, KOMMANDO 9 ?
00C5	CA36D2	265	JZ CMD6A ;JA, REGISTER NACH WAHL ANZEIGEN
00C8	CD5E04	266	CALL BEDIEN ;NEIN, AUF BEDIENUNG WARTEN
00CB	3E88	267 ANFA1:	MVI A,88H ;IN KOMMANDOFELD BALKEN ANZEIGEN
00CD	42	268	MOV B,D
00CE	CDA805	269	CALL AUS
00D1	CD5E04	270	CALL BEDIEN ;AUF NEUES KOMMANDO WARTEN
00D4	3E05	271 ANFA2:	MVI A,05H ;REGISTER-ZEIGER AUF PC SETZEN
00D6	32CAFF	272	STA REGAD
00D9	C33602	273	JMP CMD6A ;PROGRAM-COUNTER ANZEIGEN
		274	
		275	
		276	-----
		277	; KOMMANDO D: SPEICHER LESEN/BESCHREIBEN
		278	-----
		279	
00DC	1640	280 CMDO:	MVI D,40H ;SCHREIBMARKE AUF ADRESS-ANFANG
00DE	1E04	281 CMDOA:	MVI E,04H ;SCHREIBMASKE SETZEN
00E0	2ACCFE	282	LHLD MEMAD+1 ;SPEICHERZEIGER IN HL
00E3	7E	283	MOV A,M ;DATEN-BYTE LESEN UND IM RAM ABSPEICHERN
00E4	32CBFF	284	STA MEMAD
00E7	CD0606	285	CALL DISPLAY ;ADRESSE, '=', DATEN ANZEIGEN
00EA	21CCFF	286	LXI H, MEMAD+1 ;SPEICHERZEIGER-ADRESSE IN H&L
00ED	3E80	287	MVI A,80H ;MODUS 'SPEICHER LESEN/BESCHREIBEN' SETZEN
00EF	32DEFF	288	STA MODUS
00F2	CD5E04	289	CALL BEDIEN ;AUF BEDIENUNG WARTEN
00F5	1602	290	MVI D,02H ;SCHREIBMARKE AUF DATENFELD ANFANG
00F7	2ACCFE	291	LHLD MEMAD+1 ;SPEICHERZEIGER IN H&L
00FA	23	292	INX H ;SPEICHERZEIGER +1
00FB	FE12	293	CPI 12H ;TASTE ' ' BETAETIGT ?
00FD	C20201	294	JNZ M1 ;NEIN, SPEICHERZEIGER ABSPEICHERN
0100	2B	295	DCX H ;JA, SPEICHERZEIGER -2
0101	2B	296	DCX H
0102	22CCFF	297 M1:	SHLD MEMAD+1 ;SPEICHERZEIGER IM RAM ABSPEICHERN
0105	C3DE00	298	JMP CMDOA
		299	
			300 \$EJECT

LOC	OBJ	LINE	SOURCE STATEMENT
		301	-----
		302	; KOMMANDO 1: SPEICHERBEREICH MIT KONSTANTE FUELLEN
		303	-----
		304	
0108	CD9606	305	CMD1: CALL AANF ;ANFANGSADRESSE ANFORDERN
010B	CD9FD6	306	CALL EANF ;ENDADRESSE ANFORDERN
010E	CDEAD6	307	CALL DATANF ;DATENBYTE ANFORDERN
0111	3ADCFF	308	LDA DAT ;DATENBYTE IN B
0114	47	309	MOV B,A
0115	2AD0FF	310	LHLD EA ;ENDADRESSE IN D&E
0118	EB	311	XCHG
0119	2ACEFF	312	LHLD AA ;ANFANGSADRESSE IN H&L
011C	78	313	FILL1: MOV A,B ;DATENBYTE IN AKKU
011D	CDE805	314	CALL SPEICH ;DATENBYTE ABSPEICHERN
0120	C21C01	315	JNZ FILL1 ;BEI FEHLER NOCHMAL VERSUCHEN
0123	CD4006	316	CALL GLEICH ;ANFANGSADRESSE = ENDADRESSE ?
0126	CA9400	317	JZ CLRALL ;JA, KOMMANDO AUSGEFUEHRT
0129	23	318	INX H ;NEIN, ANFANGSADRESSE INKREMENTIEREN
012A	C31C01	319	JMP FILL1
		320	
		321	
		322	-----
		323	; KOMMANDO 2: SPEICHERBEREICH VERSCHIEBEN
		324	-----
		325	
012D	CD9606	326	CMD2: CALL AANF ;ANFANGSADRESSE (AA) ANFORDERN
0130	CD9FD6	327	CALL EANF ;ENDADRESSE (EA) ANFORDERN
0133	CDA806	328	CALL BANF ;BESTIMMUNGSADRESSE (BA) ANFORDERN
0136	2AD2FF	329	LHLD BA ;BA IN B&C
0139	E5	330	PUSH H
013A	C1	331	POP B
013B	C5	332	PUSH B ;BA IN STACK
013C	2ACEFF	333	LHLD AA ;AA IN H&L
013F	E5	334	PUSH H ;AA IN STACK
0140	CD3306	335	CALL DSUB ;AA > BA ? (FLAGS SETZEN)
0143	C1	336	POP B ;AA IN B&C
0144	D1	337	POP D ;BA IN D&E
0145	2AD0FF	338	LHLD EA ;EA IN H&L → EA in Adv. = ED0 RAM in 8-55
0148	D25501	339	JNC UMS1 ;SPRUNG WENN AA > BA
014B	E5	340	PUSH H ;EA IN STACK
014C	CD3306	341	CALL DSUB ;BYTEZAHL (BZ) := EA-AA
014F	EB	342	XCHG ;BZ IN D&E, BA IN H&L
0150	19	343	DAD D ;EB := BA + BZ IN H&L
0151	C1	344	POP B ;EA IN B&C
0152	C35901	345	JMP UMS3
		346	
0155	CD3306	347	UMS1: CALL DSUB ;BZ := EA-AA IN D&E
0158	EB	348	XCHG ;BA IN H&L, BZ IN D&E
0159	0A	349	UMS3: LDAX B ;BYTE AUS QUELLE IN AKKU
015A	CDE805	350	CALL SPEICH ;BYTE AUS AKKU ANS ZIEL
015D	C25901	351	JNZ UMS3 ;BEI FEHLER NOCHMAL VERSUCHEN
0160	7A	352	MOV A,D ;BYTEZAehler NULL ?

LOC	OBJ	LINE	SOURCE STATEMENT
0161	B3	353	ORA E
0162	CA9400	354	JZ CLRALL ;JA, KOMMANDO AUSGEFUEHRT
0165	1B	355	DCX D ;NEIN, BYTEZAEHLER -1
0166	E5	356	PUSH H
0167	CD3306	357	CALL DSUB ;ZIEL < QUELLE ?
016A	E1	358	POP H
016B	D27201	359	JNC UMS4 ;JA, ZIEL UND QUELLE -1
016E	23	360	INX H ;NEIN, ZIEL UND QUELLE +1
016F	23	361	INX H
0170	03	362	INX B
0171	03	363	INX B
0172	2B	364	UMS4: DCX H
0173	0B	365	DCX B
0174	C35901	366	JMP UMS3 ;NAECHSTES BYTE VERSCHIEBEN
		367	
		368	
		369	-----
		370	; KOMMANDO 3: ADRESSEN UMRECHNEN
		371	-----
		372	
0177	CD5407	373	CMD3: CALL SDEF ;DEFAULT-KENNZELLE SETZEN
017A	CD9606	374	CALL AANF ;ANFANGSADRESSE (AA) ANFORDERN
017D	CD9F06	375	CALL EANF ;END-ADRESSE (EA) ANFORDERN
0180	CDA806	376	CALL BANF ;BESTIMMUNGSADRESSE (BA) ANFORDERN
0183	CD5A07	377	CALL RDEF ;DEFAULT-KENNZELLE RUECKSETZEN
0186	CDCC06	378	CALL AUANF ;ANF. UMZURECHN.BER. (AU) ANFORDERN
0189	CDD506	379	CALL EUANF ;END. UMZURECHN.BER. (EU) ANFORDERN
018C	2ADAFF	380	LHLD EU ;EU IN D&E
018F	EB	381	XCHG
0190	2AD8FF	382	LHLD AU ;AU IN H&L
0193	7E	383	UMR1: MOV A,M ;OP-CODE LESEN
0194	CD5F07	384	CALL K ;OP-CODE KLASSIFIZIEREN
0197	CAA801	385	JZ UMR3 ;SPRUNG WENN BEFEHL ADRESSE ENTHAELT
019A	CD4D06	386	UMR2: CALL GLEICH ;AU = EU ?
019D	CA9400	387	JZ CLRALL ;JA, KOMMANDO AUSGEFUEHRT
01A0	23	388	INX H ;AU +1
01A1	05	389	DCR B ;KLASSE -1
01A2	C29A01	390	JNZ UMR2 ;SPRUNG BEI MEHR-BYTE-BEFEHLEN
01A5	C39301	391	JMP UMR1 ;NAECHST. BEFEHLS-CODE LESEN
		392	
		393	;ADRESSE LESEN UND UMRECHNEN
01A8	D5	394	UMR3: PUSH D ;EU RETTEN
01A9	E5	395	PUSH H ;AU RETTEN
01AA	23	396	INX H ;ADRESSE IN B&C
01AB	4E	397	MOV C,M
01AC	23	398	INX H
01AD	46	399	MOV B,M
		400	;LIEGT ADRESSE IN VERSCHOBENEN BEREICH ?
01AE	2AD0FF	401	LHLD EA ;EA IN D&E
01B1	EB	402	XCHG
01B2	2ACEFF	403	LHLD AA ;AA IN H&L
01B5	79	404	MOV A,C ;ADRESSE > AA ?

LOC	OBJ	LINE	SOURCE STATEMENT
01B6	95	405	SUB L ;(ADRESSE - AA => 0 ?)
01B7	78	406	MOV A,B
01B8	9C	407	SBB H
01B9	DAD4D1	408	JC UMREND ;NEIN, NICHT UMRECHNEN
01BC	7B	409	MOV A,E ;ADRESSE =< EA ?
01BD	91	410	SUB C ;(EA - ALTADR => NULL ?)
01BE	7A	411	MOV A,D
01BF	98	412	SBB B
01CD	DAD4D1	413	JC UMREND ;NEIN, NICHT UMRECHNEN
		414	;JA, NEUE ADRESSE := (ADRESSE-AA)+BA
01C3	79	415	MOV A,C ;ADRESSE - AA IN D&E
01C4	95	416	SUB L
01C5	5F	417	MOV E,A
01C6	78	418	MOV A,B
01C7	9C	419	SBB H
01C8	57	420	MOV D,A
01C9	2AD2FF	421	LHLD BA ;BA IN H&L
01CC	19	422	DAD D ;NEUE ADRESSE := <D&E> + BA
		423	
		424	;ADRESSE DURCH NEUE ERSETZEN
01CD	EB	425	XCHG ;NEUE ADRESSE IN D&E
01CE	E1	426	POP H ;BEFEHLSADR. IN H&L
01CF	E5	427	PUSH H
01D0	23	428	INX H ;ADRESSE DURCH NEUE ERSETZEN
01D1	73	429	MOV M,E
01D2	23	430	INX H
01D3	72	431	MOV M,D
01D4	E1	432	UMREND: POP H ;AU IN H&L
01D5	D1	433	POP D ;EU IN D&E
01D6	D6D3	434	MVI B,3 ;LADE KLASSE 3
01D8	C39A01	435	JMP UMR2 ;RESTLICH. BEREICH UMRECHNEN
		436	
		437	
		438	-----
		439	; KOMMANDO 4: IN AUSGABE-EINHEIT SCHREIBEN
		440	-----
		441	
01DB	CDDE06	442	CMD4: CALL ADRAF ;PERIPHERIEADRESSE ANFORDERN
01DE	1E64	443	CMD4A: MVI E,64H ;SCHREIBMASKE SETZEN
01ED	16D2	444	MVI D,02H ;SCHREIBMARKE SETZEN
01E2	3ED3	445	MVI A,0D3H ;OUT-BEFEHL IN RAM
01E4	32C2FF	446	STA BEFREG
01E7	CD0202	447	CALL IOLBA ;PERIPHERIEADRESSE MIT DATUM ANZEIGEN,
		448	;AUF BEDienung WARTEN, I/O-BEFEHL AUSFUEHREN
01EA	C3DE01	449	JMP CMD4A
		450	
		451	\$EJECT

LOC	OBJ	LINE	SOURCE STATEMENT
		452	-----
		453	; KOMMANDO 5: VON EINGABEEINHEIT LESEN
		454	-----
		455	
01ED	CDDE06	456	CMD5: CALL ADRANF ;PERIPHERIEADRESSE ANFORDERN
01F0	3EDB	457	MVI A,0DBH ;IN-BEFEHL IN RAM
01F2	32C2FF	458	STA BEFREG
01F5	CD1F02	459	CALL IOLBB ;I/O-BEFEHL IN RAM AUSFUEHREN
01F8	1E67	460	CMD5A: MVI E,67H ;SCHREIBMASKE SETZEN
01FA	1610	461	MVI D,10H ;SCHREIBMARKE SETZEN
01FC	CDD202	462	CALL IOLBA ;PERIPHERIEADRESSE MIT DATUM ANZEIGEN,
		463	,AUF BEDIENUNG WARTEN, I/O-BEFEHL AUSFUEHREN
01FF	C3F801	464	JMP CMD5A
		465	
0202	97	466	IOLBA: SUB A ;ANZEIGEN NR.1 U. 2 DUNKEL
0203	0640	467	MVI B,40H
0205	CDA805	468	CALL AUS
0208	CDA805	469	CALL AUS
020B	21DDFF	470	LXI H,ADR ;I/O-ADRESSE ANZEIGEN
020E	CD1F06	471	CALL BYTE
0211	3E48	472	MVI A,48H ; '=' ANZEIGEN
0213	CDA805	473	CALL AUS
0216	21DCFF	474	LXI H,DAT ;DATENBYTE ANZEIGEN
0219	CD1F06	475	CALL BYTE
021C	CD5E04	476	CALL BEDIEN ;AUF BEDIENUNG WARTEN
021F	3ADDFE	477	IOLBB: LDA ADR ;I/O-ADRESSE IN RAM
0222	32C3FF	478	STA BEFREG+1
0225	3EC9	479	MVI A,0C9H ;RETURN-BEFEHL IN RAM
0227	32C4FF	480	STA BEFREG+2
022A	3ADCFE	481	LDA DAT ;DATENBYTE IN A
022D	CDC2FF	482	CALL BEFREG ;IO-BEFEHL IN RAM AUSFUEHREN
0230	32DCFF	483	STA DAT ;DATENBYTE ABSPEICHERN
0233	C9	484	RET
		485	
		486	
		487	-----
		488	; KOMMANDO 6: REGISTER AUSLESEN/BESCHREIBEN
		489	-----
		490	
0234	1608	491	CMD6: MVI D,08H ;SCHREIBMARKE AUF ANFANG DES DATENFELDES
0236	21CAFF	492	CMD6A: LXI H,REGAD ;REGISTERZEIGER-ADRESSE IN H&L
0239	7E	493	MOV A,M ;REGISTERZEIGER IN AKKU
023A	E607	494	ANI 07H ;HW-BITS AUSBLENDEN
023C	77	495	MOV M,A
023D	C24202	496	JNZ REGL1 ;SPRUNG WENN ZEIGER UNGLEICH NULL
0240	3C	497	INR A ;SONST ZEIGER = 1 (0 NICHT ERLAUBT)
0241	77	498	MOV M,A
0242	87	499	REGL1: ADD A ;REGISTERKURZNAMEN AUS TABELLE 'REGP'
0243	4F	500	MOV C,A ; UEBER H&L ADRESSIEREN
0244	0600	501	MVI B,00H
0246	C5	502	PUSH B ;TABELLENZEIGER RETTEN
0247	218B02	503	LXI H,REGP-2

LOC	OBJ	LINE	SOURCE STATEMENT
024A	09	504	DAD B
024B	7E	505	MOV A,M ;REGISTERKURZNAMEN AUSGEBEN
024C	0640	506	MVI B,40H
024E	CDA805	507	CALL AUS
0251	23	508	INX H
0252	7E	509	MOV A,M
0253	CDA805	510	CALL AUS
0256	3E48	511	MVI A,48H ; '=' AUSGEBEN
0258	CDA805	512	CALL AUS
025B	C1	513	POP B ;REGISTERINHALT IN RETTZELLEN UEBER H&L
025C	21B2FF	514	LXI H,FRET-2; ADRESSIEREN
025F	09	515	DAD B
0260	E5	516	PUSH H ;H&L RETTEN
0261	0602	517	MVI B,02H ;NW.BYTE LESEN UND ANZEIGEN
0263	CD1F06	518	CALL BYTE
0266	23	519	INX H
0267	0608	520	MVI B,08H ;HW.BYTE LESEN UND ANZEIGEN
0269	CD1F06	521	CALL BYTE
026C	1E70	522	MVI E,70H ;SCHREIBMASKE SETZEN
026E	97	523	SUB A ;MODUS 'REG. LESEN/BESCHREIBEN' SETZEN
026F	32DEFF	524	STA MODUS
0272	E1	525	POP H ;H&L ZEIGT AUF AKTUELLE REG.ADR.
0273	CD5E04	526	CALL BEDIEN ;AUF BEDIENUNG WARTEN
0276	21CAFF	527	LXI H,REGAD
0279	FE12	528	CPI 12H ;TASTE ' ' BETAETIGT ?
027B	CA8202	529	JZ CMD6B ;JA, REGISTERZEIGER -1
027E	34	530	INR M ;NEIN, REGISTERZEIGER +1
027F	C33402	531	JMP CMD6 ;NEUEN REGISTER MIT INHALT ANZEIGEN
		532	
0282	35	533	CHD6B: DCR M ;REGISTERZEIGER -1
0283	7E	534	MOV A,M
0284	E607	535	ANI 07H ;REGISTERZEIGER = 0 ?
0286	C23402	536	JNZ CMD6 ;NEIN, NEUES REG. MIT INHALT ANZEIGEN
0289	35	537	DCR M ;JA, REGISTERZEIGER NOCHMAL -1
028A	C33402	538	JMP CMD6 ;NEUES REGISTER MIT INHALT ANZEIGEN
		539	
		540	;TABELLE FUER REGISTERKURZNAMEN
		541	;*****
028D	7771	542	REGP: DW 7177H ;7-SEGMENTMUSTER FUER AF
028F	7C39	543	DW 397CH ; " " BC
0291	5E79	544	DW 795EH ; " " DE
0293	7638	545	DW 3876H ; " " HL
0295	7339	546	DW 3973H ; " " PC
0297	6D73	547	DW 736DH ; " " SP
0299	066D	548	DW 6D06H ; " " IS
		549	
		550	\$EJECT

LOC	OBJ	LINE	SOURCE STATEMENT
		551	-----
		552	; KOMMANDO 7: PROGRAMM OHNE HALTEPUNKTE STARTEN (ECHTZEIT)
		553	-----
		554	
029B	CD5407	555	CMD7: CALL SDEF ;DEFAULT-KENNZELLE SETZEN
029E	CDB106	556	CALL SANF ;STARTADRESSE ANFORDERN (DEFAULT IST PC)
02A1	3EC3	557	MVI A,DC3H ;OBJ.CODE FUER JMP-BEFEHL IN RAM ABSPEICHERN
02A3	32C2FF	558	STA BEFREG
02A6	2ABCFF	559	LHLD PRET ;SPRUNGZIEL (STARTADRESSE) IN RAM ABSPEICHERN
02A9	22C3FF	560	SHLD BEFREG+1
02AC	3AC1FF	561	LDA IRET+1 ;ANWENDER INTERRUPTMASKE IN AKKU
02AF	F610	562	ORI 10H ;RST7.5-FLIPFLOP RUECKSETZEN
02B1	3D	563	SIM ;INTR.-MASKE SETZEN
02B2	3AC0FF	564	LDA IRET ;INT STATUS IN AKKU
02B5	E608	565	ANI 08H ;EI-BIT GESETZT ?
02B7	CA7903	566	JZ PRGSTR ;NEIN, PROGRAMM SO STARTEN
02BA	FB	567	EI ;JA, INTERRUPTSYSTEM FREIGEBEN
02BB	C37903	568	JMP PRGSTR ;PROGRAMM STARTEN
		569	
		570	
		571	-----
		572	; KOMMANDO 8: PROGRAMM MIT HALTEPUNKTEN STARTEN
		573	-----
		574	
02BE	CD5407	575	CMD8: CALL SDEF ;DEFAULT-KENNZELLE SETZEN
02C1	CDB106	576	CALL SANF ;STARTADRESSE (=PC) ANFORDERN
02C4	CDBA06	577	CALL H1ANF ;HALTADRESSE 1 ANFORDERN
02C7	CDC306	578	CALL H2ANF ;HALTADRESSE 2 ANFORDERN
02CA	CD0703	579	CMD85: CALL STEP ;EINEN BEFEHL AUSFUEHREN
02CD	0640	580	MVI B,40H ;NEUEN PC ANZEIGEN
02CF	3E73	581	MVI A,73H
02D1	CDA805	582	CALL AUS
02D4	3E39	583	MVI A,39H
02D6	CDA805	584	CALL AUS
02D9	3E48	585	MVI A,48H
02DB	CDA805	586	CALL AUS
02DE	21BDFE	587	LXI H,PRET+1
02E1	CD1F06	588	CALL BYTE
02E4	2B	589	DCX H
02E5	CD1F06	590	CALL BYTE
02E8	2ABCFF	591	LHLD PRET ;HALTEPUNKT ERREICHT ?
02EB	EB	592	XCHG
02EC	2AD4FF	593	LHLD H1
02EF	CD4006	594	CALL GLEICH
02F2	CA9A00	595	JZ ANFANG ;JA, UNTERBRECHEN DER ABARBEITUNG
02F5	2AD6FF	596	LHLD H2
02F8	CD4006	597	CALL GLEICH
02FB	CA9A00	598	JZ ANFANG ;JA, UNTERBRECHEN DER ABARBEITUNG
02FE	C3CA02	599	JMP CMD85 ;NEIN, NAECHSTEN BEFEHL AUSFUEHREN
		600	
		601	\$EJECT

LOC	OBJ	LINE	SOURCE STATEMENT
		602	;
		603	; KOMMANDO 9: PROGRAMM IM EINZELSCHRITT ABARBEITEN
		604	;
		605	
0301	CDD703	606	CMD9: CALL STEP ;EINEN BEFEHL AUSFUEHREN
0304	C39A00	607	JMP ANFANG
		608	
		609	
		610	;
		611	; LOKALES UNTERPROGRAMM 'STEP' ZUR AUSFUEHRUNG EINES EINZEL-
		612	; SCHRITTES, MIT EINSPRUNGPUNKT 'PRGSR' ZUM STARTEN EINES
		613	; PROGRAMMS FUER ECHTZEITABLAUF.
		614	;
		615	
0307	21C1FF	616	STEP: LXI H, IRET+1; INTR.MASKE SETZEN
030A	7E	617	MOV A, M
030B	3D	618	SIM
030C	3600	619	MVI M, D
030E	2B	620	DCX H
030F	7E	621	MOV A, M ;EI-BIT AUS RETTZELLE LESEN
0310	E608	622	ANI DBH
0312	77	623	MOV M, A
0313	20	624	RIM ;INTERRUPT STATUS LESEN
0314	86	625	ORA M ;ALTE EI-BIT EINFUEGEN
0315	77	626	MOV M, A ;INTERRUPT STATUS RETTEN
0316	2ABCFF	627	LHLD PRET ;PROGRAM-COUNTER IN H&L
0319	11C2FF	628	LXI D, BEFREG;SCHATTENPROGRAMM ANFANG IN D&E
031C	0600	629	MVI B, 0 ;KLASSE 0 EINSTELLEN
		630	
		631	;BEFEHL VOLLSTAENDIG EINLESEN
		632	;*****
031E	7E	633	STEPA: MOV A, M ;BEFEHLSBYTE LESEN UND IN SCHATTENPROGRAMM
031F	12	634	STAX D ; LADEN
0320	05	635	DCR B ;KLASSE MINUS EINS
0321	CA2F03	636	JZ STEPC ;SPRUNG WENN ALLE BYTES GELESEN
0324	FC5F07	637	CM K ;BEI 1.BYTE OP-CODE KLASSIFIZIEREN
0327	13	638	INX D ;SCHATTEN-PC ERHOEHEN
0328	23	639	INX H ;PC IN H&L ERHOEHEN
0329	22BCFF	640	SHLD PRET ;NEUE PC ABSPEICHERN
032C	C31E03	641	JMP STEPA ;NAECHSTES BYTE LESEN
		642	
		643	;OPERATIONSCODE PRUEFEN / VERTEILER
		644	;*****
032F	79	645	STEP3: MOV A, C ;OP-CODE IN AKKU
0330	FE76	646	CPI 76H ;HLT-BEFEHL NICHT AUSFUEHREN !
0332	CAB303	647	JZ STEPO ;SPRUNG BEI HLT BEFEHL
0335	FEE9	648	CPI 0E9H
0337	CAB803	649	JZ STEP1 ;SPRUNG BEI PCHL
033A	FEC3	650	CPI 0C3H
033C	CAE703	651	JZ STEP2 ;SPRUNG BEI UNBEDINGTEN JMP
033F	FECD	652	CPI 0CDH
0341	CAD903	653	JZ STEP3 ;SPRUNG BEI UNBEDINGTEN CALL

LOC	OBJ	LINE	SOURCE STATEMENT
0344	FEC9	654	CPI 0C9H
0346	CAF303	655	JZ STEP4 ;SPRUNG BEI UNBEDINGTEN RET
0349	E6F7	656	ANI 0F7H
034B	FEF3	657	CPI 0F3H
034D	CABF03	658	JZ STEP5 ;SPRUNG BEI EI ODER DI
0350	E6C7	659	ANI 0C7H
0352	FEC2	660	CPI 0C2H
0354	CACC03	661	JZ STEP6 ;SPRUNG BEI BEDINGTEN JMP
0357	FEC4	662	CPI 0C4H
0359	CAD403	663	JZ STEP7 ;SPRUNG BEI BEDINGTEN CALL
035C	FEC0	664	CPI 0C0H
035E	CAEE03	665	JZ STEP8 ;SPRUNG BEI BEDINGTEN RET
0361	FEC7	666	CPI 0C7H
0363	CA0204	667	JZ STEP9 ;SPRUNG BEI RST
		668	
		669	;TRANSFER-, LOG.- UND ARITHM.- BEFEHL AUSFUEHREN
		670	;*****
0366	3EC3	671	MVI A,0C3H ;JMP-BEFEHLSCODE IN RAM LADEN
0368	12	672	STAX D
0369	13	673	INX D
036A	2189D3	674	LXI H,STEPE ;RUECKSPRUNGADRESSE IN RAM LADEN
036D	7D	675	MOV A,L
036E	12	676	STAX D
036F	13	677	INX D
0370	7C	678	MOV A,H
0371	12	679	STAX D
0372	21000D	680	LXI H,0 ;MONITOR-STACKPOINTER RETTEN
0375	39	681	DAD SP
0376	22C8FF	682	SHLD BEFREG+6
		683	PRGSTR: ;REGISTER LADEN:
0379	31B4FF	684	LXI SP,FRET ;/SP AUF ANFANG RETTZELLEN
037C	F1	685	POP PSW ;/AKKU & FLAGS LADEN
037D	C1	686	POP B ;/B&C LADEN
037E	D1	687	POP D ;/D&E LADEN
037F	2ABEFF	688	LHLD SRET ;/SP LADEN
0382	F9	689	SPHL
0383	2ABAFF	690	LHLD LRET ;/H&L LADEN
0386	C3C2FF	691	JMP BEFREG ;ANWENDER-PROGRAMM IN RAM STARTEN
		692	; BZW AUSFUEHRUNG DES BEFEHLS MIT RUECKSPRUNG
		693	; NACH AUSFUEHRUNG.
		694	STEPE: ;CPU-STATUS (ANWEND.) RETTEN UND FUER MONITOR
		695	; SETZEN
0389	22BAFF	696	SHLD LRET ;/RETTE H&L
038C	32B5FF	697	STA ARET ;/RETTE AKKU
038F	1F	698	RAR ;/SCHIEBE CY IN AKKU
0390	21000D	699	LXI H,0 ;/RETTE STACKPOINTER
0393	39	700	DAD SP
0394	22BEFF	701	SHLD SRET
0397	17	702	RAL ;/CY WIEDER LADEN
0398	3AB5FF	703	LDA ARET ;/AKKU WIEDER LADEN
039B	31BAFF	704	LXI SP,DRET+1 ;/SP SETZEN ZUM RETTEN D. UEBRIGEN REG.
039E	D5	705	PUSH D ;/REGISTER D&E RETTEN

LOC	OBJ	LINE	SOURCE STATEMENT		
039F	C5	706	PUSH	B	;/REGISTER B&C RETTEN
03A0	F5	707	PUSH	PSW	;/AKKU & FLAGS RETTEN
03A1	2AC8FF	708	LHLD	BEFREG+6	;/MONITOR STACKPOINTER WIEDER LADEN
03A4	F9	709	SPHL		
03A5	21COFF	710	LXI	H, IRET	;/EI-BIT AUS RETTZELLE LESEN
03A8	7E	711	MOV	A, M	
03A9	E608	712	ANI	08H	
03AB	77	713	MOV	M, A	
03AC	20	714	RIM		;/INTERRUPT STATUS LESEN
03AD	B6	715	ORA	M	;/ALTE EI-BIT EINFUEGEN
03AE	77	716	MOV	M, A	;/INTERRUPT-STATUS RETTEN
03AF	23	717	INX	H	
03B0	3600	718	MVI	M, 0	
03B2	C9	719	RET		
		720			
		721			;/VERZWEIGUNGS- UND STEUERBEFEHLE AUSFUEHREN
		722			;
03B3	2B	723	STEPO:	DCX H	;/PC WIEDER AUF HLT-BEFEHL SETZEN
03B4	22BCFF	724	SHLD	PRET	;/ (SPRUNG AUF SICH SELBST UM NICHT
03B7	C9	725	RET		;/ AUSSER MONIT. KONTROLLE ZU GERATEN)
		726			
03B8	2ABAFF	727	STEP1:	LHLD LRET	;/H&L RETTZELLEN IN SCHATTEN PC FUER PCHL
03BB	22BCFF	728	SHLD	PRET	
03BE	C9	729	RET		
		730			
03BF	79	731	STEP5:	MOV A, C	;/OP-CODE ('EI' ODER 'DI') IN AKKU
03C0	E608	732	ANI	08H	;/EI-BIT MASKIEREN
03C2	4F	733	MOV	C, A	
03C3	21COFF	734	LXI	H, IRET	
03C6	7E	735	MOV	A, M	
03C7	E6F7	736	ANI	0F7H	;/ALTES EI-BIT LOESCHEN
03C9	B1	737	ORA	C	;/NEUES EI-BIT EINFUEGEN
03CA	77	738	MOV	M, A	
03CB	C9	739	RET		
		740			
03CC	79	741	STEP6:	MOV A, C	;/OP-CODE (BED. 'JUMP') IN AKKU
03CD	CD9807	742	CALL	JUMP	;/SPRUNGBEDINGUNG ERFUELLT ?
03D0	DAE703	743	JC	STEP2	;/JA, SPRUNG
03D3	C9	744	RET		;/NEIN, RETURN
		745			
03D4	79	746	STEP7:	MOV A, C	;/OP-CODE (BED. 'CALL') IN AKKU
03D5	CD9807	747	CALL	JUMP	;/SPRUNGBEDINGUNG ERFUELLT ?
03D8	D0	748	RNC		;/NEIN, RETURN
03D9	2ABCFF	749	STEP3:	LHLD PRET	;/JA, ANWENDER PROGRAMM-COUNTER AUF
03DC	EB	750	XCHG		;/ ANWENDER STACK
03DD	2ABEFF	751	LHLD	SRET	
03E0	2B	752	DCX	H	
03E1	72	753	MOV	M, D	
03E2	2B	754	DCX	H	
03E3	73	755	MOV	M, E	
03E4	22BEFF	756	SHLD	SRET	
03E7	2AC3FF	757	STEP2:	LHLD BEFREG+1	;/SPRUNGADRESSE IN SCHATTEN PC

LOC	OBJ	LINE	SOURCE STATEMENT		
03EA	22BCFF	758	SHLD	PRET	
03ED	C9	759	RET		
		760			
03EE	79	761	STEP8: MOV	A,C	;OP-CODE (BED.'RET') IN AKKU
03EF	CD9807	762	CALL	JUMP	;SPRUNGBEDINGUNG ERFUELLT ?
03F2	D0	763	RNC		;NEIN, BEFEHL AUSGEFUEHRT
03F3	2ABEFF	764	STEP4: LHLD	SRET	;ANWENDER SP IN H&L
03F6	5E	765	MOV	E,M	;NW PC-BYTE VON ANW.STACK
03F7	23	766	INX	H	;STACKPOINTER +1
03F8	56	767	MOV	D,M	;HW PC-BYTE VON ANW.STACK
03F9	23	768	INX	H	;STACKPOINTER +1
03FA	22BEFF	769	SHLD	SRET	;STACKPOINTER WIEDER IN RETTZELLE
03FD	EB	770	XCHG		;PC VON D&E IN RETTZELLE
03FE	22BCFF	771	SHLD	PRET	
0401	C9	772	RET		
		773			
0402	2ABCFF	774	STEP9: LHLD	PRET	;ANWENDER PROGRAMCOUNTER AUF
0405	EB	775	XCHG		; ANWENDERSTACK
0406	2ABEFF	776	LHLD	SRET	
0409	2B	777	DCX	H	
040A	72	778	MOV	M,D	
040B	2B	779	DCX	H	
040C	73	780	MOV	M,E	
040D	22BEFF	781	SHLD	SRET	
0410	79	782	MOV	A,C	;OP-CODE ('RST'-BEFEHL) IN AKKU
0411	E638	783	ANI	38H	;RST-ADRESSE MASKIEREN
0413	32BCFF	784	STA	PRET	;PC MIT RST-ADRESSE LADEN
0416	97	785	SUB	A	; (HW. BYTE = 0)
0417	32BDFF	786	STA	PRET+1	
041A	C9	787	RET		
		788			
		789	\$EJECT		

LOC	OBJ	LINE	SOURCE STATEMENT
		790	-----
		791	; KOMMANDO A: INFORMATION AUF MAGNETBAND SCHREIBEN
		792	; KOMMANDO B: INFORMATION VON MAGNETBAND LESEN
		793	-----
		794	
		795	CHDA:
041B	3ECD	796	CHDB: MVI A,COH ;SOD AUF HIGH SETZEN
041D	30	797	SIM ;/STOPBITS AUSGEBEN
041E	CD9606	798	CALL AANF ;ANFANGSADRESSE (AA) ANFORDERN
0421	CD9F06	799	CALL EANF ;ENDADRESSE (EA) ANFORDERN
0424	2AD0FF	800	LHLD EA ;EA IN D&E
0427	EB	801	XCHG
0428	2ACEFF	802	LHLD AA ;AA IN H&L
042B	4E	803	ABSL: MOV C,M ;DATENBYTE FUER AUSGABE LADEN
042C	E5	804	PUSH H ;AA NACH STACK
042D	D5	805	PUSH D ;EA NACH STACK
042E	116902	806	LXI D,BITTIME ;BITZEITEN LADEN
0431	21A501	807	LXI H,HALFBIT
0434	3ADFFF	808	LDA KOMZE ;KOMMANDO = 'B' ?
0437	FE0B	809	CPI OBH
0439	CA4804	810	JZ MCMDB ;JA, VON MAGNETBAND LESEN
		811	;NEIN, AUF MAGNETBAND SCHREIBEN ('A')
043C	CD4606	812	CALL COUT ;BYTE AUSGEBEN
043F	D1	813	POP D ;EA IN D&E
0440	E1	814	POP H ;AA IN H&L
0441	79	815	MOV A,C ;AKTUELLE AKTION ANZEIGEN
0442	CD0606	816	CALL DISPLAY
0445	C35404	817	JMP ABTEST ;SPRUNG
		818	;VON MAGNETBAND LESEN
0448	CD6406	819	MCMDB: CALL CIN ;BYTE EINLESEN
044B	D1	820	POP D ;EA IN D&E
044C	E1	821	POP H ;AA IN H&L
044D	79	822	MCMDBF: MOV A,C ;DATENBYTE ABSPEICHERN UND ANZEIGEN
044E	CDE805	823	CALL SPEICH
0451	C24D04	824	JNZ MCMDBF ;WIEDERHOLUNG BEI FEHLER
		825	;TEST UND NAECHSTES ZEICHEN
0454	CD4006	826	ABTEST: CALL GLEICH ;AA = EA ?
0457	CA9400	827	JZ CLRALL ;JA, KOMMANDO AUSGEFUEHRT
045A	23	828	INX H ;NEIN, NAECHSTES BYTE
045B	C32B04	829	JMP ABSL
		830	
		831	\$EJECT

```
LOC OBJ      LINE      SOURCE STATEMENT
832 ;*****
833 ;
834 ;      U N T E R P R O G A M M E
835 ;
836 ;*****
837
838 ;-----
839 ; PROGRAMM      : BEDIEN
840 ;
841 ; FUNKTION      : BEDIEN WARTET AUF EINE TASTENBETAETIGUNG. WAEHREND
842 ;                DER WARTEZEIT BLINKT DIE SCHREIBMARKE (CURSOR).
843 ;                WIRD EINE TASTE GEDRUECKT, SO WIRD DER TASTENCODE
844 ;                EINGELESEN UND WIE FOLGT AUSGEWERTET:
845 ;                -STEUERZEICHEN
846 ;                <-- = ROTIERE SCHREIBMARKE LINKS
847 ;                --> = ROTIERE SCHREIBMARKE RECHTS
848 ;                "" = 1)BEI ABSCHLUSS EINER ADRESS- ODER
849 ;                "V"  DATENEINGABE ERFOLGT EIN RUECKSPRUNG
850 ;                MIT DER TASTENCODE IM AKKUMULATOR
851 ;                2)BEI KOMMANDOQUITTUNG EINES ZUGELASSENEN
852 ;                KOMMANDOS ERFOLGT EIN SPRUNG ZUM KOMMAN-
853 ;                DOPROGRAMM UEBER SPRUNGTABELLE
854 ;                (AUSNAHME: BEI KOMMANDO 9 UND ANZ. DUNKEL)
855 ;                3)BEI KOMMANDOQUITTUNG EINES NICHT
856 ;                ZUGELASSENEN KOMMANDOS ERFOLGT KEINE
857 ;                AUSWERTUNG
858 ;                -KOMMANDOZEICHEN
859 ;                1)ERLAUBTE KOMMANDOS WERDEN ANGEZEIGT UND
860 ;                ABGESPEICHERT
861 ;                2)NICHT ERLAUBTE KOMMANDOS WERDEN IGNORIERT
862 ;                -ANDERE HEX-ZEICHEN
863 ;                ZEICHEN WERDEN ANGEZEIGT UND AN ZEIGERADRESSE
864 ;                (H&L) ABGESPEICHERT.
865 ;                BEI MODUS 80 (KOMMANDO 0) UND SCHREIBMARKE
866 ;                AUF DATENFELD, DATENBYTE ABSPEICHERN;
867 ;                BEI SCHREIBMARKE AUF ADRESSFELD
868 ;                SPEICHERZELLE LESEN UND ANZEIGEN.
869 ;                DIE SCHREIBMARKE WIRD VERSCHOBEN.
870 ;
871 ; EING.PARAM. : H&L, ZEIGERADRESSE
872 ;            : D&E, SCHREIBMARKE UND SCHREIBMASKE
873 ;            : MODUS
874 ; EINGABEN VON: SAB 8279 (TASTATUR)
875 ; AUSG.PARAM. : A, TASTENCODE 12H (TASTE ' ')
876 ;            : TASTENCODE 13H (TASTE 'V')
877 ;            : D, SCHREIBMARKE
878 ;            : KOMZE, (KOMMANDO ZEICHEN)
879 ;            : SPEICHERZELLEN (ZEIGERADRESSE IST H&L)
880 ;            : " (BEI MODUS 80H)
881 ; AUSGABEN AN : SAB 8279 (ANZEIGEN)
882 ; AUFGER.PROG.: KOGEN, AUS, RDEF, UMW, SPEICH, DISPLAY
883 ;
```

LOC	OBJ	LINE	SOURCE STATEMENT
		884	; BEMERKUNG : DAS UNTERPROGRAMM WIRD NUR UEBER DIE ABSCHLUSS-
		885	TASTEN ' ' UND 'V' VERLASSEN.
		886	WIRD EIN ERLAUBTES KOMMANDO ABGESCHLOSSEN,
		887	ERFOLGT EIN SPRUNG INS ENTSPRECH. KOMMANDOPROGRAMM,
		888	ANSONSTEN ERFOLGT EIN RUECKSPRUNG INS AUFRUFENDE
		889	PROGRAMM.
		890	-----
		891	
		892	;WARTEN AUF TASTENBETAETIGUNG / BLINKZEIT
		893	;*****
045E	010040	894	BEDIEN: LXI B,4000H ;BLINKZEIT IN B&C
0461	DBEF	895	TAST: IN KOSEA ;TASTE BETAETIGT ?
0463	E607	896	ANI 07H
0465	C27304	897	JNZ CLOE ;JA, AUSWERTUNG
0468	OD	898	DCR C ;NEIN, BLINKZEITZAEHLER ERNIEDRIGEN
0469	C26104	899	JNZ TAST ;ZEIT ABGELAUFEN ?
046C	05	900	DCR B
046D	C26104	901	JNZ TAST ;NEIN !
047D	C38905	902	JMP BLIN ;JA, ZUSTANDSWECHSEL
		903	
		904	;AUSWERTUNG
		905	;*****
0473	42	906	CLOE: MOV B,D ;WIRD BLANK ANGEZEIGT ?
0474	C0BD05	907	CALL KOGEN
0477	DBEE	908	IN DATEA
0479	E67F	909	ANI 7FH
047B	C28504	910	JNZ LES ;NEIN, SPRUNG
047E	3AEOFF	911	LDA RAMHZ ;JA, ZWISCHENGESPEICHERTES ZEICHEN ANZEIGEN
0481	42	912	MOV B,D
0482	CDA805	913	CALL AUS
		914	
0485	3E40	915	LES: MVI A,40H ;TASTENCODE EINLESEN
0487	D3EF	916	OUT KOSEA
0489	DBEE	917	IN DATEA
048B	E63F	918	ANI 3FH ;SHIFT- UND CTRL-BIT AUSBLENDEN
048D	FE10	919	CPI 10H ;HEXZEICHEN EINGEGEBEN ?
048F	DADF04	920	JC HEXZEI ;JA, AUSWERTUNG
		921	;STEUERTASTE '<--' BETAETIGT ?
0492	CA7805	922	JZ LEFT ;JA, AUSFUEHRUNG
0495	FE11	923	CPI 11H ;STEUERTASTE '-->' BETAETIGT ?
0497	CA8205	924	JZ RIGHT ;JA, AUSFUEHRUNG
049A	4F	925	MOV C,A ;KOMMANDO ABSCHLUSS ?
049B	7A	926	MOV A,D
049C	07	927	RLC
049D	79	928	MOV A,C
049E	D0	929	RNC ;NEIN, RUECKSPRUNG
		930	
		931	;KOMMANDO ABSCHLUSS
		932	;*****
049F	CD5A07	933	CALL RDEF ;DEFAULT-KZ RUECKSETZEN
04A2	3ADFFF	934	LDA KOMZE ;KOMMANDO ERLAUBT ?
04A5	FE0C	935	CPI 0CH

LOC	OBJ	LINE	SOURCE STATEMENT
04A7	D25E04	936	JNC BEDIEN ;NEIN, WARTEN AUF NAECHSTE EINGABE
04AA	FE09	937	CPI 09H ;KOMMANDO 9 UND ANZEIGE DUNKEL ?
04AC	C2B704	938	JNZ TABZ ;NEIN, SPRUNG
04AF	7B	939	MOV A,E
04BD	FE7F	940	CPI 7FH
04B2	CA9400	941	JZ CLRALL ;JA, REGISTER ANZEIGEN
04B5	3E09	942	MVI A,09H ;NEIN, KORREKTUR
04B7	D5	943	TABZ: PUSH D ;ADRESSE FUER KOMMANDOPROGRAMM
04B8	11C704	944	LXI D,BTBL ; AUS ADRESSTABELLE LADEN
04BB	6F	945	MOV L,A
04BC	2600	946	MVI H,0
04BE	29	947	DAD H
04BF	19	948	DAD D
04C0	5E	949	MOV E,M
04C1	23	950	INX H
04C2	56	951	MOV D,M
04C3	EB	952	XCHG
04C4	D1	953	POP D
04C5	E3	954	XTHL ;RUECKSPRUNGADRESSE ERSETZEN
04C6	C9	955	RET ;SPRUNG INS KOMMANDO-PROGRAMM
		956	
		957	;ADRESS-SPRUNGTABELLE FUER KOMMANDO-PROGRAMM:
		958	;*****
04C7	DC00	959	BTBL: DW CMD0
04C9	0801	960	DW CMD1
04CB	2D01	961	DW CMD2
04CD	7701	962	DW CMD3
04CF	DB01	963	DW CMD4
04D1	ED01	964	DW CMD5
04D3	3402	965	DW CMD6
04D5	9B02	966	DW CMD7
04D7	BE02	967	DW CMD8
04D9	0103	968	DW CMD9
04DB	1B04	969	DW CMDA
04DD	1B04	970	DW CMDB
		971	
		972	;HEX-ZEICHEN:
		973	;*****
04DF	4F	974	HEXZEI: MOV C,A ;ZEICHEN = KOMMANDOZEICHEN ?
04E0	7A	975	MOV A,D
04E1	07	976	RLC
04E2	79	977	MOV A,C
04E3	D20705	978	JNC ECHO ;NEIN, SPRUNG
		979	
		980	;KOMMANDO-ZEICHEN:
		981	;*****
04E6	FE0C	982	CPI 0CH ;KOMMANDO ERLAUBT ?
04E8	D25E04	983	JNC BEDIEN ;NEIN, WARTEN AUF NAECHSTE EINGABE
04EB	32DFFF	984	STA KOMZE ;JA, KOMMANDOZEICHEN ABSPEICHERN
04EE	1E7F	985	MVI E,7FH ;RESTL. FELD LOESCHEN UND SPERREN
04F0	97	986	SUB A
04F1	0640	987	MVI B,40H

LOC	OBJ	LINE	SOURCE STATEMENT
04F3	CDA805	988	CALL AUS
04F6	3ED2	989	MVI A,DD2H
04F8	D3EF	990	OUT KOSEA
04FA	79	991	MOV A,C
04FB	CDC9D5	992	CALL UMW ;KOMMANDO ANZEIGEN
04FE	F68D	993	ORI 80H
0500	42	994	MOV B,D
0501	CDA805	995	CALL AUS
0504	C35EQ4	996	JMP BEDIEN ;WARTE AUF QUITTUNG
		997	
		998	;ANDERES HEX-ZEICHEN:
		999	;*****
0507	CDC9D5	1000	ECHO: CALL UMW ;ZEICHEN ANZEIGEN
050A	42	1001	MOV B,D
050B	CDA805	1002	CALL AUS
		1003	
		1004	;ZEICHEN ABSPEICHERN:
050E	E5	1005	ABSP: PUSH H ;ADRESSZEIGER RETTEN
050F	23	1006	INX H ;ADRESSZEIGER +2
0510	23	1007	INX H
0511	06C0	1008	MVI B,0C0H
0513	78	1009	PACK: MOV A,B ;ADRESS-OFFSET ANHAND SCHREIBMARKE ERRECHNEN
0514	A3	1010	ANA E ; UND ADRESSZEIGER MIT OFFSET ERHOEHEN
0515	C22505	1011	JNZ PACK2
0518	78	1012	MOV A,B
0519	17	1013	RAL
051A	3F	1014	CMC
051B	DA1F05	1015	JC PACK1
051E	2B	1016	DCX H
051F	1F	1017	PACK1: RAR
0520	47	1018	MOV B,A
0521	A2	1019	ANA D
0522	C23305	1020	JNZ PACK4
0525	78	1021	PACK2: MOV A,B
0526	17	1022	RAL
0527	78	1023	MOV A,B
0528	D22E05	1024	JNC PACK3
052B	E67F	1025	ANI 7FH
052D	37	1026	STC
052E	1F	1027	PACK3: RAR
052F	47	1028	MOV B,A
0530	C31305	1029	JMP PACK
0533	78	1030	PACK4: MOV A,B ;OFFSET GERADE ?
0534	17	1031	RAL
0535	3EFO	1032	MVI A,0FOH
0537	DA4205	1033	JC PACK5 ;JA, NIEDERW. HALBBYTE ERSETZEN
053A	79	1034	MOV A,C ;NEIN, HOEHERW. HALBBYTE ERSETZEN
053B	87	1035	ADD A
053C	87	1036	ADD A
053D	87	1037	ADD A
053E	87	1038	ADD A
053F	4F	1039	MOV C,A

LOC	OBJ	LINE	SOURCE STATEMENT
0540	3E0F	1040	MVI A,0FH
0542	A6	1041	PACK5: ANA M ;HALBBYTE LESEN
0543	B1	1042	ORA C ;HEX-ZEICHEN EINFUEGEN
0544	77	1043	MOV M,A ;BYTE ZURUECKSCHREIBEN
0545	E1	1044	POP H ;ADRESSZEIGER WIEDER MIT GRUNDWERT LADEN
0546	3ADEFF	1045	LDA MODUS ;MODUS ABFRAGEN:
0549	FE80	1046	CPI 80H ;SPEICHER LESEN/BESCHREIBEN ?
054B	C26A05	1047	JNZ CURE ;NEIN, SPRUNG
054E	7A	1048	MOV A,D ;SCHREIBMARKE AUF DATENFELD ?
054F	E603	1049	ANI 03H
0551	CA5D05	1050	JZ LESBE ;NEIN, SPRUNG
0554	2ACFF	1051	LHLD MEMAD+1 ;JA, DATEN AN ADRESSE ABSPEICHERN
0557	3ACBFF	1052	LDA MEMAD
055A	CDE805	1053	CALL SPEICH
055D	2ACFF	1054	LESBE: LHLD MEMAD+1 ;DATEN VON ADRESSE LESEN
0560	7E	1055	MOV A,M
0561	32CBFF	1056	STA MEMAD
0564	C0606	1057	CALL DISPLAY ;ADRESSE UND DATENBYTE ANZEIGEN
0567	21CCFF	1058	LXI H,MEMAD+1
056A	7A	1059	CURE: MOV A,D ;SCHREIBMARKE NACH RECHTS, BIS ZUR NAECHSTEN
056B	0F	1060	RECHTS: RRC ; NICHTMASKIERTEN POSITION SCHIEBEN
056C	47	1061	MOV B,A
056D	DA5E04	1062	JC BEDIEN ;SPRUNG WENN SCHREIBMARKE AM BILDRAND
0570	A3	1063	ANA E
0571	C26805	1064	JNZ RECHTS
0574	50	1065	MOV D,B
0575	C38905	1066	JMP BLIN ;AUF NAECHSTE EINGABE WARTEN
		1067	
		1068	;STEUERTASTE '<--'
		1069	;*****
0578	7A	1070	LEFT: MOV A,D ;SCHREIBMARKE NACH LINKS ROTIEREN
0579	07	1071	LEFT1: RLC
057A	57	1072	MOV D,A
057B	A3	1073	ANA E
057C	C27905	1074	JNZ LEFT1
057F	C38905	1075	JMP BLIN
		1076	
		1077	;STEUERTASTE '-->'
		1078	;*****
0582	7A	1079	RIGHT: MOV A,D ;SCHREIBMARKE NACH RECHTS ROTIEREN
0583	0F	1080	RIGH1: RRC
0584	57	1081	MOV D,A
0585	A3	1082	ANA E
0586	C28305	1083	JNZ RIGH1
		1084	
		1085	;ZUSTANDSWECHSEL SCHREIBMARKE
		1086	;*****
0589	42	1087	BLIN: MOV B,D ;ZEICHEN VON SCHREIBMARKENPOSITION LESEN
058A	CDBD05	1088	CALL KOGEN
058D	DBEE	1089	IN DATEA
058F	47	1090	MOV B,A
0590	E67F	1091	ANI 7FH ;BLANK ?

LOC	OBJ	LINE	SOURCE STATEMENT
0592	CA9E05	1092	JZ BL1 ;JA, BLANK DURCH ZEICHEN ERSETZEN
0595	78	1093	MOV A,B ;NEIN, ZEICHEN DURCH BLANK ERSETZEN
0596	32E0FF	1094	STA RAMHZ
0599	E680	1095	ANI 80H
059B	C3A105	1096	JMP BL2
		1097	
059E	3AE0FF	1098	BL1: LDA RAMHZ
05A1	42	1099	BL2: MOV B,D ;ZEICHEN ANZEIGEN
05A2	CDA805	1100	CALL AUS
05A5	C35E04	1101	JMP BEDIEN ;WARTEN AUF NAECHSTE EINGABE
		1102	
		1103	
		1104	;
		1105	; PROGRAMM : AUS
		1106	;
		1107	; FUNKTION : 7-SEGMENTMUSTER ANZEIGEN
		1108	; EING.PARAM. : <A>= AUSZUGEBENDEN ZEICHEN ALS 7-SEGMENTMUSTER
		1109	; = SCHREIBMARKENPOSITION
		1110	; AUSG.PARAM. : = NEUE SCHREIBMARKENPOSITION
		1111	; <A>= 0 (DER AKKU WIRD GELOESCHT)
		1112	; VERAEND.REG.: PSW, B
		1113	; AUFGER.PROG.: KOGEN
		1114	;
		1115	
05A8	C5	1116	AUS: PUSH B
05A9	4F	1117	MOV C,A ;7-SEGMENTMUSTER IN C
05AA	DBEF	1118	AUS1: IN KOSEA ;ANZEIGE VERFUEGBAR ?
05AC	C680	1119	ADI 80H
05AE	DAAA05	1120	JC AUS1 ;NEIN, WARTEN
05B1	CDBD05	1121	CALL KOGEN ;JA, 7-SEGMENTMUSTER ANZEIGEN
05B4	79	1122	MOV A,C
05B5	D3EE	1123	OUT DATEA
05B7	C1	1124	POP B
05B8	78	1125	MOV A,B ;SCHREIBMARKE NACH RECHTS
05B9	0F	1126	RRC
05BA	47	1127	MOV B,A
05BB	97	1128	SUB A ;AKKU LOESCHEN
05BC	C9	1129	RET
		1130	
		1131	\$EJECT

```

LOC OBJ      LINE      SOURCE STATEMENT
-----
1132 ;-----
1133 ; PROGRAMM      : KOGEN
1134 ;
1135 ; FUNKTION      : KOGEN GENERIERT, ABHAENGIG VON DER SCHREIBMARKEN-
1136 ;                  POSITION DAS KOMMANDO "READ DISPLAYRAM" FUER DEN
1137 ;                  SAB 8279 UND GIBT DAS KOMMANDO AUS.
1138 ; EING.PARAM.   : <B> = SCHREIBMARKE
1139 ; AUSG.PARAM.   : <A> = KOMMANDO 'READ DISPLAYRAM'
1140 ; AUSGABEN AN   : KOMMANDO "READ DISPLAYRAM" AN SAB 8279
1141 ; VERAEND.REG.: PSW,B
1142 ; AUFGER.PROG.: -
1143 ;-----
1144
05BD 78      1145 KOGEN:  MOV    A,B      ;SCHREIBMARKE IN A
05BE 065F    1146          MVI    B,5FH    ;KOMMANDO RAHMEN IN B
05C0 07      1147 KOG2:  RLC          ;KOMMANDO GENERIEREN
05C1 04      1148          INR    B
05C2 D2C005  1149          JNC    KOG2
05C5 78      1150          MOV    A,B      ;KOMMANDO AN SAB8279 AUSGEBEN
05C6 D3EF    1151          OUT   KOSEA
05C8 C9      1152          RET
1153
1154
1155 ;-----
1156 ; PROGRAMM      : UMW
1157 ;
1158 ; FUNKTION      : UMW WANDELT DAS NIEDERWERTIGE HALBBYTE DES AKKUS
1159 ;                  IN EIN 7-SEGMENTMUSTER UM.
1160 ; EING.PARAM.   : <A> = HEX-ZEICHEN IM NIEDERW. HALBBYTE
1161 ; AUSG.PARAM.   : <A> = 7-SEGMENTMUSTER
1162 ; VERAEND.REG.: PSW
1163 ; AUFGER.PROG.: -
1164 ;-----
1165
05C9 E5      1166 UMW:   PUSH   H
05CA E60F    1167          ANI    0FH      ;NIEDERW.HALBBYTE MASKIEREN
05CC 21D805  1168          LXI    H,HEXD  ;LADE H&L MIT TABELLENADR. ('HEXD')
05CF 85      1169          ADD    L      ;ADIERE HEX-CODE HINZU
05D0 6F      1170          MOV    L,A
05D1 3E00    1171          MVI    A,00H
05D3 8C      1172          ADC    H
05D4 67      1173          MOV    H,A
05D5 7E      1174          MOV    A,M      ;7-SEGMENTMUSTER IN AKKU
05D6 E1      1175          POP    H
05D7 C9      1176          RET
1177
1178          ;TABELLE FUER HEX-CODE-ANZEIGE
1179          ;*****
05D8 3F      1180 HEXD:  DB     3FH      ;7-SEGMENTMUSTER FUER 0
05D9 06      1181          DB     06H      ;          "          " 1
05DA 5B      1182          DB     5BH      ;          "          " 2
05DB 4F      1183          DB     4FH      ;          "          " 3

```

LOC	OBJ	LINE	SOURCE STATEMENT
05DC	66	1184	DB 66H ; " " 4
05DD	6D	1185	DB 6DH ; " " 5
05DE	7D	1186	DB 7DH ; " " 6
05DF	07	1187	DB 07H ; " " 7
05E0	7F	1188	DB 7FH ; " " 8
05E1	6F	1189	DB 6FH ; " " 9
05E2	77	1190	DB 77H ; " " A
05E3	7C	1191	DB 7CH ; " " B
05E4	39	1192	DB 39H ; " " C
05E5	5E	1193	DB 5EH ; " " D
05E6	79	1194	DB 79H ; " " E
05E7	71	1195	DB 71H ; " " F
		1196	
		1197	
		1198	-----
		1199	; PROGRAMM : SPEICH
		1200	;
		1201	; FUNKTION : SPEICH SPEICHERT DAS DATENBYTE AUS A UNTER DIE
		1202	IN H&L STEHENDE ADRESSE AB. ANSCHLIESSEND WIRD
		1203	EIN KONTROLL-LESEN DURCHGEFUEHRT.
		1204	ADRESSE UND GELESENE DATEN WERDEN AM 7-SEGMENT-
		1205	DISPLAY ANGEZEIGT. ENTSPRICHT DAS GELESENE NICHT
		1206	DEM GESCHRIEBENEN SO WIRD DAS ZERO-BIT RUECKGESETZT
		1207	WENN DIE ADRESSE KLEINER 1000H IST, WIRD DER TIMER
		1208	FUER DIE EPROM-PROGRAMMIERUNG INITIALISIERT.
		1209	; EING.PARAM. : <A> = DATENBYTE
		1210	<H&L> = SPEICHERADRESSE
		1211	; AUSG.PARAM. : ZEROFLAG=0 BEI FEHLER, SONST 1
		1212	; AUSGABEN AN : ANZEIGE DER ADRESSE UND DES GELESENEN DATENBYTES
		1213	; VERAEND.REG.: PSW
		1214	; AUFGER.PROG.: DISPLAY
		1215	-----
		1216	
05E8	F5	1217	SPEICH: PUSH PSW ;ADR < 1000H ?
05E9	7C	1218	MOV A,H
05EA	FE10	1219	CPI 10H
05EC	D2FB05	1220	JNC SPEI1 ;NEIN, DATENBYTE ABSPEICHERN
		1221	;JA, ZAEHLER IM SAB 8155 SETZEN
05EF	3E65	1222	MVI A,65H ;/RECHTECKGENERATOR
05F1	D3FD	1223	OUT TIMEH ;/3,125 MS TAKT
05F3	3E80	1224	MVI A,80H
05F5	D3FC	1225	OUT TIMEL
05F7	3ECO	1226	MVI A,0COH ;START ZAEHLER
05F9	D3F8	1227	OUT KOADR
05FB	F1	1228	SPEI1: POP PSW ;DATENBYTE ABSPEICHERN
05FC	F5	1229	PUSH PSW
05FD	77	1230	MOV M,A
05FE	7E	1231	MOV A,M ;BYTE LESEN UND MIT
05FF	CD0606	1232	CALL DISPLA ; DER ADRESSE ANZEIGEN
0602	E3	1233	XTHL ;IST-SOLL-WERT VERGLEICH
0603	BC	1234	CMP H
0604	E1	1235	POP H

LOC	OBJ	LINE	SOURCE STATEMENT
0605	C9	1236	RET
		1237	
		1238	
		1239	-----
		1240	; PROGRAMM : DISPLA
		1241	;
		1242	; FUNKTION : DISPLA ZEIGT DIE ADRESSE AUS H&L UND DAS DATENBYTE
		1243	; AUS A AN DEN 7-SEGMENTANZEIGEN AN
		1244	; EING.PARAM. : <A> = DATENBYTE
		1245	; <H&L> = SPEICHERADRESSE
		1246	; AUSGABEN AN : ANZEIGENFELD: XXXX=YY
		1247	; VERAEND.REG.: -
		1248	; AUFGER.PROG.: AUS, BYAUS
		1249	-----
		1250	
0606	F5	1251	DISPLA: PUSH PSW
0607	C5	1252	PUSH B
0608	4F	1253	MOV C,A ;DATENBYTE IN C
0609	0640	1254	MVI B,40H ;SCHREIBMARKE AUF ANFANG
060B	7C	1255	MOV A,H
060C	CD2006	1256	CALL BYAUS ;HW-ADRESSTEIL ANZEIGEN
060F	7D	1257	MOV A,L
0610	CD2006	1258	CALL BYAUS ;NW-ADRESSTEIL ANZEIGEN
0613	3E48	1259	MVI A,48H
0615	CDA805	1260	CALL AUS ; '=' ANZEIGEN
0618	79	1261	MOV A,C
0619	CD2006	1262	CALL BYAUS ;DATENBYTE ANZEIGEN
061C	C1	1263	POP B
061D	F1	1264	POP PSW
061E	C9	1265	RET
		1266	
		1267	\$EJECT

LOC	OBJ	LINE	SOURCE STATEMENT
		1268 ;	-----
		1269 ;	PROGRAMME : BYTE, BYAUS
		1270 ;	
		1271 ;	FUNKTION : DAS DURCH H&L ADRESSIERTE DATENBYTE WIRD AUS DEM
		1272 ;	SPEICHER GELESEN (UP-BYTE) ODER STEHT IM AKKUMULATOR
		1273 ;	(UP-BY AUS).
		1274 ;	DAS DATENBYTE WIRD IN ZWEI HAEFTEN ZERLEGT, DIE
		1275 ;	HALBBYTES IN 7-SEGMENTMUSTER GEWANDELT UND AM
		1276 ;	ANZEIGENFELD ANGEZEIGT. DIE SCHREIBMARKE WIRD AUTO-
		1277 ;	MATISCH NACH RECHTS VERSCHOBEN.
		1278 ;	EING.PARAM. : BYTE - <HL> = ADRESSE DES AUSZUGEB. BYTE
		1279 ;	 = SCHREIBMARKE
		1280 ;	BYAUS- <A> = AUSZUGEB. BYTE
		1281 ;	 = SCHREIBMARKE
		1282 ;	AUSG.PARAM. : = SCHREIBMARKE UM 2 ST. NACH RECHTS VERSCHOBEN
		1283 ;	AUFGER.PROG.: UMW, AUS
		1284 ;	VERAEND.REG.: PSW, B
		1285 ;	-----
		1286	
061F	7E	1287	BYTE: MOV A, M ; BYTE AUS SPEICHER IN A LESEN
0620	F5	1288	BYAUS: PUSH PSW ; HOEHERW. HALBBYTE ANZEIGEN <i>vom Akku</i>
0621	0F	1289	RRC
0622	0F	1290	RRC
0623	0F	1291	RRC
0624	0F	1292	RRC
0625	CDC905	1293	CALL UMW
0628	CDA805	1294	CALL AUS
062B	F1	1295	POP PSW ; NIEDERW. HALBBYTE ANZEIGEN
062C	CDC905	1296	CALL UMW
062F	CDA805	1297	CALL AUS
0632	C9	1298	RET
		1299	
		1300	
		1301 ;	-----
		1302 ;	PROGRAMM : DSUB
		1303 ;	
		1304 ;	FUNKTION : DER INHALT DES B&C-REG. WIRD VON DEM DES H&L REG.
		1305 ;	SUBTRAHIERT
		1306 ;	EING.PARAM. : <H&L> UND <B&C> = 16 BIT WOERTER
		1307 ;	AUSG.PARAM. : <H&L> = RESULTAT
		1308 ;	CY=1 WENN RESULTAT NEGATIV
		1309 ;	CY=0 WENN RESULTAT POSITIV
		1310 ;	VERAEND.REG.: PSW, H, L
		1311 ;	AUFGER.PROG.: -
		1312 ;	-----
		1313	
0633	C5	1314	DSUB: PUSH B ; ADDIERE ZUM INHALT VON REGISTER H&L DAS
0634	37	1315	STC ; ZWEIER-KOMPLEMENT VON REGISTER B&C
0635	79	1316	MOV A, C
0636	2F	1317	CMA
0637	8D	1318	ADC L
0638	6F	1319	MOV L, A

LOC	OBJ	LINE	SOURCE STATEMENT
0639	78	1320	MOV A,B
063A	2F	1321	CHA
063B	8C	1322	ADC H
063C	67	1323	MOV H,A
063D	3F	1324	CHC
063E	C1	1325	POP B
063F	C9	1326	RET
		1327	
		1328	
		1329	-----
		1330	; PROGRAMM : GLEICH
		1331	;
		1332	; FUNKTION : GLEICH VERGLEICHT D&E MIT H&L AUF ><=
		1333	; EING.PARAM. : <D&E> = VERGLEICHSWORT 1
		1334	; <H&L> = VERGLEICHSWORT 2
		1335	; AUSG.PARAM. : ZERO=1 WENN <H&L> = <D&E>
		1336	; ZERO=0 WENN <H&L> >< <D&E>
		1337	; CY =1 WENN <H&L> > <D&E>
		1338	; CY =0 WENN <H&L> =< <D&E>
		1339	; VERAEND.REG.: PSW
		1340	; AUFGER.PROG.: -
		1341	-----
		1342	
0640	7A	1343	GLEICH: MOV A,D ;D UND H VERGLEICHEN
0641	BC	1344	CHP H
0642	CO	1345	RNZ ;FERTIG WENN NICHT GLEICH
0643	7B	1346	MOV A,E ;E UND L VERGLEICHEN
0644	BD	1347	CHP L
0645	C9	1348	RET
		1349	
		1350	\$EJECT

```

LOC OBJ      LINE      SOURCE STATEMENT
-----
1351 ; -----
1352 ; PROGRAMM      : COUT
1353 ;
1354 ; FUNKTION      : INHALT VON REG. C SERIELL UEBER SOD AUSGEBEN
1355 ;                  FORMAT: SERIELLE, ASYNCHRONE UEBERTRAGUNG MIT
1356 ;                  1 STARTBIT, 8 DATENBITS UND 2 STOPPBITS.
1357 ;                  DIE BAUDRATE WIRD UEBER <D&E> BESTIMMT.
1358 ;
1359 ; EING.PARAM.    : <C> = AUSZUGEBENDES DATANWORT (8 BIT)
1360 ;                  <D&E> = 'BITTIME'-WERT FUER BAUDRATE ENTSPRECHEND
1361 ;                  TABELLE.
1362 ;
1363 ;                  +-----+
1364 ;                  ! BAUDRATE ! BITTIME ! HALFBIT !
1365 ;                  !           ! (D&E)  ! (H&L)  !
1366 ;                  +-----+
1367 ;                  !   110   ! 08C6  ! 04E3  !
1368 ;                  !   150   ! 06B2  ! 03D9  !
1369 ;                  !   300   ! 03D7  ! 026C  !
1370 ;                  !   600   ! 0269  ! 01A5  !
1371 ;                  !  1200  ! 01B2  ! 0159  !
1372 ;                  !  2400  ! 0157  ! 012C  !
1373 ;                  !  4800  ! 0129  ! 0115  !
1374 ;                  !  9600  ! 0112  ! 0109  !
1375 ;                  +-----+
1376 ; AUSGABEN AN : SERIELLE AUSGABE UEBER SOD
1377 ; VERAEND.REG.: PSW,H,L
1378 ; AUFGER.PROG.: -
1379 ; -----
0646 C5      1380 COUT:  PUSH  B
0647 060B    1381      MVI  B,BITSO ;SETZE ANZAHL DER ZU UEBERTRAGENEN BITS
0649 AF      1382      XRA  A      ;SETZE CARRY ZURUECK
064A 3E80    1383 C01:  MVI  A,80H  ;BEREITE SOD ENABLE BIT VOR
064C 1F      1384      RAR          ;SCHIEBE CY IN SOD POSITION
064D 30      1385      SIM          ;GEBE CY-BIT AUF SOD
064E 210000  1386      LXI  H,D    ;SETZE BITZEIT
0651 19      1387      DAD  D
0652 2D      1388 C02:  DCR  L      ;WARTE EIN BITZEIT
0653 C25206  1389      JNZ  C02
0656 25      1390      DCR  H
0657 C25206  1391      JNZ  C02
065A 37      1392      STC          ;SETZE CY HIGH FUER STOP-BITS
065B 79      1393      MOV  A,C    ;SCHIEBE NAECHSTE BIT IN CY
065C 1F      1394      RAR
065D 4F      1395      MOV  C,A
065E 05      1396      DCR  B      ;ALLE BITS UEBERTRAGEN ?
065F C24A06  1397      JNZ  C01    ;NEIN, DAS NAECHST. BIT AUSGEBEN
0662 C1      1398      POP  B      ;JA, UNTERPROGRAMM ENDE
0663 C9      1399      RET
1400
1401 $EJECT

```

LOC	OBJ	LINE	SOURCE STATEMENT
		1402	-----
		1403	; PROGRAMM : CIN
		1404	;
		1405	; FUNKTION : DATENBYTE UEBER SID SERIELL EINLESEN
		1406	; FORMAT: SERIELLE ASYNCHRONE UEBERTRAGUNG MIT
		1407	; 1 STARTBIT, 8 DATENBITS UND MIND. 1 STOPPBIT.
		1408	; DIE BAUDRATE WIRD UEBER <D&E> UND <H&L> BESTIMMT
		1409	; EING.PARAM. : <D&E> = 'BITTIME'-WERT FUER BAUDRATE ENTSPR. TABELLE
		1410	; <H&L> = 'HALFBIT'-WERT FUER BAUDRATE ENTSPR. TABELLE
		1411	; EINGABEN VON: SERIELLEN DATEN VON SID
		1412	; AUSG.PARAM. : <C> = DATENBYTE VON SID
		1413	; VERAUND.REG.: PSW,B,C,H,L
		1414	; AUFGER.PROG.: -
		1415	-----
		1416	
0664	E5	1417	CIN: PUSH H ;H&L AUF STACK
0665	0609	1418	MVI B,BITSI ;ANZAHL DER ZU LESENDEN BITS LADEN
0667	E1	1419	CI1: POP H ;'HALFBIT'-WERT LADEN UND IM STACK ABLEGEN
0668	E5	1420	PUSH H
0669	2D	1421	RIM ;LESE SID
066A	B7	1422	ORA A
066B	FA6706	1423	JM CI1 ;WARTE BIS STARTBIT ERKANNT
066E	2D	1424	CI2: DCR L ;WARTE BIS STARTBITMITTE
066F	C26E06	1425	JNZ CI2
0672	25	1426	DCR H
0673	C26E06	1427	JNZ CI2
0676	2D	1428	RIM ;STARTBIT SICHER ERKANNT ?
0677	B7	1429	ORA A
0678	FA6706	1430	JM CI1 ;NEIN, WARTE AUF STARTBIT
		1431	;JA, DATENBITS EINLESEN
067B	210000	1432	CI3: LXI H,0 ;'BITTIME'-WERT LADEN
067E	19	1433	DAD D
067F	2D	1434	CI4: DCR L ;EINE BITZEIT WARTEN
0680	C27F06	1435	JNZ CI4
0683	25	1436	DCR H
0684	C27F06	1437	JNZ CI4
0687	2D	1438	RIM ;LESE SID
0688	17	1439	RAL ;SID IN CY
0689	05	1440	DCR B ;IST DIES DAS ERSTE STOPBIT ?
068A	CA9406	1441	JZ CI5 ;JA, SPRUNG
068D	79	1442	MOV A,C ;SCHIEBE DATENBIT IN REGISTER C
068E	1F	1443	RAR
068F	4F	1444	MOV C,A
0690	00	1445	NOP ;FUER GLEICHE LAUFZEIT IN CIN UND COUT
0691	C37B06	1446	JMP CI3 ;DAS NAECHSTE BIT LESEN
0694	E1	1447	CI5: POP H
0695	C9	1448	RET
		1449	
		1450	\$EJECT

LOC	OBJ	LINE	SOURCE STATEMENT
		1451 ;	-----
		1452 ;	PROGRAMME : AANF, EANF, BANF, SANF, H1ANF, H2ANF, AUANF, EUANF
		1453 ;	DATANF, ADRANF
		1454 ;	
		1455 ;	FUNKTION : PARAMETEREINGABE ANFORDERN. DIE KURZBEZEICHNUNG DES
		1456 ;	EINZUGEBENEN PARAMETERS WIRD ZUSAMMEN MIT EINEM DE-
		1457 ;	FAULTWERT ANGEZEIGT.
		1458 ;	AANF = ANFANGSADRESSE (AA) ANFORDERN
		1459 ;	EANF = ENDADRESSE (EA) ANFORDERN
		1460 ;	BANF = BESTIMMUNGSADRESSE ANFORDERN
		1461 ;	SANF = STARTADRESSE ANFORDERN
		1462 ;	H1ANF = HALTADRESSE NR. 1 ANFORDERN
		1463 ;	H2ANF = " NR. 2 "
		1464 ;	AUANF = ANFANGSADRESSE DES UMGEB. BEREICHS ANFORDERN
		1465 ;	EUANF = ENDADRESSE " " " "
		1466 ;	ADRANF = PERIPHERIEADRESSE ANFORDERN
		1467 ;	DATANF = DATENBYTE ANFORDERN
		1468 ;	DIE PARAMETER WERDEN IN ZUGEORDNETE SPEICHERZELLEN
		1469 ;	ABGESPEICHERT
		1470 ;	EING.PARAM. : -
		1471 ;	AUSG.PARAM. : EINGEGEBENE PARAMETER IN ZUGEORDNETER SPEICHERZELLE
		1472 ;	AUSGABEN AN : KURZBEZEICHNUNG UND DEFAULTWERT ZU DER ANZEIGE
		1473 ;	VERAEND.REG. : ALLE
		1474 ;	AUFGER.PROG. : AUS, BYTE, BEDIEN, SDEF
		1475 ;	-----
		1476	
0696	21CEFF	1477	AANF: LXI H,AA ;ADRESSZEIGER LADEN
0699	017777	1478	LXI B,7777H ;KURZBEZEICHNUNG 'AA' LADEN
069C	C31F07	1479	JMP PARAM
069F	21D0FF	1480	EANF: LXI H,EA ;ADRESSZEIGER LADEN
06A2	017779	1481	LXI B,7977H ;KURZBEZEICHNUNG 'EA' LADEN
06A5	C31F07	1482	JMP PARAM
06A8	21D2FF	1483	BANF: LXI H,BA ;ADRESSZEIGER LADEN
06AB	01777C	1484	LXI B,7C77H ;KURZBEZEICHNUNG 'BA' LADEN
06AE	C31F07	1485	JMP PARAM
06B1	21B0FF	1486	SANF: LXI H,PRET ;ADRESSZEIGER LADEN
06B4	01776D	1487	LXI B,6D77H ;KURZBEZEICHNUNG 'SA' LADEN
06B7	C31F07	1488	JMP PARAM
06BA	21D4FF	1489	H1ANF: LXI H,H1 ;ADRESSZEIGER LADEN
06BD	010676	1490	LXI B,7606H ;KURZBEZEICHNUNG 'H1' LADEN
06C0	C31F07	1491	JMP PARAM
06C3	21D6FF	1492	H2ANF: LXI H,H2 ;ADRESSZEIGER LADEN
06C6	015B76	1493	LXI B,765BH ;KURZBEZEICHNUNG 'H2' LADEN
06C9	C31F07	1494	JMP PARAM
06CC	21D8FF	1495	AUANF: LXI H,AU ;ADRESSZEIGER LADEN
06CF	013E77	1496	LXI B,773EH ;KURZBEZEICHNUNG 'AU' LADEN
06D2	C31F07	1497	JMP PARAM
06D5	21DAFF	1498	EUANF: LXI H,EU ;ADRESSZEIGER LADEN
06D8	013E79	1499	LXI B,793EH ;KURZBEZEICHNUNG 'EU' LADEN
06DB	C31F07	1500	JMP PARAM
		1501	
		1502	

LOC	OBJ.	LINE	SOURCE STATEMENT
06DE	21DCFF	1503	ADRANF: LXI H,ADR-1 ;ADRESSZEIGER LADEN
06E1	010000	1504	LXI B,0000H ;KURZBEZEICHNUNG 'PA' LADEN
06E4	117773	1505	LXI D,7377H
06E7	C3F306	1506	JMP PAAN
		1507	
06EA	21DBFF	1508	DATANF: LXI H,DAT-1 ;ADRESSZEIGER LADEN
06ED	010000	1509	LXI B,0000H ;KURZBEZEICHNUNG 'DB' LADEN
06FD	117C5E	1510	LXI D,5E7CH
		1511	
		1512	;EIN-BYTE PARAMETER ANFORDERN
		1513	;*****
06F3	78	1514	PAAN: MOV A,B ;KURZBEZEICHNUNG ANZEIGEN
06F4	0640	1515	MVI B,40H
06F6	CDA805	1516	CALL AUS
06F9	79	1517	MOV A,C
06FA	CDA805	1518	CALL AUS
06FD	7A	1519	MOV A,D
06FE	CDA805	1520	CALL AUS
0701	7B	1521	MOV A,E
0702	CDA805	1522	CALL AUS
0705	3E48	1523	MVI A,48H ; '=' ANZEIGEN
0707	CDA805	1524	CALL AUS
070A	3AE1FF	1525	LDA DEFAULT ;DEFAULTWERT LOESCHEN ?
070D	07	1526	RLC
070E	DA1407	1527	JC PAANF1 ;NEIN, SPRUNG
0711	23	1528	INX H ;JA, PARAMETER LOESCHEN
0712	3600	1529	MVI M,D
0714	CD1F06	1530	PAANF1: CALL BYTE ;DEFAULTWERT ANZEIGEN
0717	2B	1531	DCX H
0718	1E7C	1532	MVI E,7CH ;SCHREIBMASKE SETZEN
071A	1602	1533	MVI D,D2H ;SCHREIBMARKE AUF ANFANG SETZEN
071C	C34807	1534	JMP PARAM3 ;SPRUNG
		1535	
		1536	;ZWEI-BYTE PARAMETER ANFORDERN
		1537	;*****
071F	78	1538	PARAM: MOV A,B ;1.ZEICHEN DER ADRESS-KURZBEZEICHNUNG
0720	0640	1539	MVI B,40H ; ANZEIGEN
0722	CDA805	1540	CALL AUS
0725	79	1541	MOV A,C ;2.ZEICHEN DER ADRESS-KURZBEZEICHNUNG
0726	CDA805	1542	CALL AUS ; ANZEIGEN
0729	3E48	1543	MVI A,48H ; '=' ZEICHEN ANZEIGEN
072B	CDA805	1544	CALL AUS
072E	3AE1FF	1545	LDA DEFAULT ;DEFAULTWERT LOESCHEN ?
0731	07	1546	RLC
0732	DA3A07	1547	JC PARAM2 ;NEIN, SPRUNG
0735	97	1548	SUB A ;JA, PARAMETER LOESCHEN
0736	77	1549	MOV M,A
0737	23	1550	INX H
0738	77	1551	MOV M,A
0739	2B	1552	DCX H
073A	23	1553	PARAM2: INX H ;DEFAULTWERT ANZEIGEN
073B	0608	1554	MVI B,08H

LOC	OBJ	LINE	SOURCE STATEMENT
073D	CD1F06	1555	CALL BYTE
0740	2B	1556	DCX H
0741	CD1F06	1557	CALL BYTE
0744	1E70	1558	MVI E,70H ;SCHREIBMASKE SETZEN
0746	1608	1559	MVI D,08H ;SCHREIBMARKE AUF ANFANG SETZEN
0748	3ADEFF	1560	PARAM3: LDA MODUS ;PARAM EING. KENNZEICHNEN
0748	F610	1561	ORI 10H
074D	32DEFF	1562	STA MODUS
0750	CD5E04	1563	CALL BEDIEN ;AUF EINGABE DER PARAMETER WARTEN
0753	C9	1564	RET
		1565	
		1566	
		1567	-----
		1568	; PROGRAMME : SDEF, RDEF
		1569	;
		1570	; FUNKTION : DEFAULTKENNZELLE SETZEN/RUECKSETZEN
		1571	; AUSG.PARAM. : <DEFAULT> = FFH (KENNZELLE GESETZT)
		1572	; <DEFAULT> = 00H (" RUECKGESETZT)
		1573	; VERAEND.REG.: A
		1574	-----
		1575	
0754	3EFF	1576	SDEF: MVI A,OFFH
0756	32E1FF	1577	STA DEFAULT
0759	C9	1578	RET
		1579	
075A	97	1580	RDEF: SUB A
075B	32E1FF	1581	STA DEFAULT
075E	C9	1582	RET
		1583	
		1584	\$EJECT

LOC	OBJ	LINE	SOURCE STATEMENT
		1585	-----
		1586	; PROGRAMM : K
		1587	;
		1588	; FUNKTION : K KLASSIFIZIERT DEN OPERATIONS-CODE EINES 8085-BEFEHLS
		1589	; 1-BYTE-BEFEHL KLASSE=1 ZERO=1
		1590	; 2-BYTE-BEFEHL KLASSE=2 ZERO=1
		1591	; 3-BYTE-BEFEHL KLASSE=3 ZERO=1
		1592	; 3-BYTE-BEFEHL MIT ADRESSE KLASSE=3 ZERO=0
		1593	; EING.PARAM. : <A> = OPERATIONS-CODE
		1594	; AUSG.PARAM. : = KLASSE 01H = 1-BYTE-BEFEHL
		1595	; 02H = 2-BYTE-BEFEHL
		1596	; 03H = 3-BYTE-BEFEHL
		1597	; <ZERO-FLAG> = 1 BEI BEFEHL MIT DATEN
		1598	; <ZERO-FLAG> = 0 BEI BEFEHL MIT ADRESSE
		1599	; <C> = OP-CODE
		1600	; VERAEND.REG.: PSW,B,C
		1601	; AUFGER.PROG.: --
		1602	-----
		1603	
075F	4F	1604	K: MOV C,A ;OPERATIONS-CODE IN C ABSPEICHERN
0760	0603	1605	MVI B,3 ;3-BYTE-BEFEHL MIT ADRESSE ?
0762	FEC3	1606	CPI 0C3H ;OP=11000011 ?
0764	C8	1607	RZ ;JA, RETURN
0765	FEC0	1608	CPI 0CDH ;OP=11001101 ?
0767	C8	1609	RZ ;JA, RETURN
0768	E6E7	1610	ANI 0E7H ;OP=001XX010 ?
076A	FE22	1611	CPI 22H
076C	C8	1612	RZ ;JA, RETURN
076D	E6C7	1613	ANI 0C7H ;OP=11XXX010 ?
076F	FEC2	1614	CPI 0C2H
0771	C8	1615	RZ ;JA, RETURN
0772	FEC4	1616	CPI 0C4H ;OP=11XXX100 ?
0774	C8	1617	RZ ;JA, RETURN
0775	79	1618	MOV A,C ;3-BYTE-BEFEHL OHNE ADRESSE ???
0776	E6CF	1619	ANI 0CFH ;OP=00XX0001 ???
0778	FE01	1620	CPI 01H
077A	CA9607	1621	JZ KEND ;JA, SPRUNG
077D	0602	1622	MVI B,2 ;2-BYTE-BEFEHL ?
077F	79	1623	MOV A,C
0780	E6F7	1624	ANI 0F7H ;OP=1101X011 ?
0782	FED3	1625	CPI 0D3H
0784	CA9607	1626	JZ KEND ;JA, SPRUNG
0787	E6C7	1627	ANI 0C7H ;OP=00XXX110 ?
0789	FE06	1628	CPI 06H
078B	CA9607	1629	JZ KEND ;JA, SPRUNG
078E	FEC6	1630	CPI 0C6H ;OP=11XXX110 ?
0790	CA9607	1631	JZ KEND ;JA, SPRUNG
		1632	
0793	0601	1633	K1: MVI B,1 ;ALSO 1-BYTE-BEFEHL
0795	97	1634	SUB A
0796	BB	1635	KEND: CMP B ;ZERO-FLAG SETZEN
0797	C9	1636	RET

LOC	OBJ	LINE	SOURCE STATEMENT
		1637	
		1638	
		1639	-----
		1640	; PROGRAMM : JUMP
		1641	;
		1642	; FUNKTION : JUMP PRUEFFT OB BEI DEN BEDINGTEN JMP, CALL
		1643	: UND RET BEFEHLEN DIE SPRUNGBEDINGUNG ERFUELLT
		1644	: IST. WENN JA WIRD DAS CARRY-FLAG GESETZT, WENN
		1645	: NICHT WIRD DAS CARRY-FLAG RUECKGESETZT
		1646	; EING.PARAM. : <A> = OPERATIONS-CODE
		1647	: <FRET> = FLAGS
		1648	; AUSG.PARAM. : <CARRY> = 1 WENN SPRUNGBEDINGUNG ERFUELLT
		1649	: <CARRY> = 0 WENN SPRUNGBEDINGUNG NICHT ERFUELLT
		1650	; VERAEND.REG.: PSW,B,C
		1651	; AUFGER.PROG.: --
		1652	-----
		1653	
0798	4F	1654	JUMP: MOV C,A ;OPERATIONS-CODE IN C
0799	E630	1655	ANI 30H ;BIT A5 UND A4 AUSBLENDEN
079B	0640	1656	MVI B,40H ;FLAG-MASKE 40H IN B
079D	CAAE07	1657	JZ JUMP1 ;SPRUNG WENN A5=0 UND A4=0
07A0	0680	1658	MVI B,80H ;FLAG-MASKE 80H IN B
07A2	EAAE07	1659	JPE JUMP1 ;SPRUNG WENN A5=1 UND A4=1
07A5	FE20	1660	CPI 20H
07A7	0604	1661	MVI B,04H ;FLAG-MASKE 04H IN B
07A9	CAAE07	1662	JZ JUMP1 ;SPRUNG WENN A5=1 UND A4=0
		1663	;A5 = 0 UND A4 = 1
07AC	0601	1664	MVI B,01H ;FLAG-MASKE 01H IN B
07AE	3AB4FF	1665	JUMP1: LDA FRET ;FLAGS IN AKKU
07B1	AD	1666	ANA B ;AKTUELLE FLAGBIT AUSBLENDEN
07B2	47	1667	MOV B,A ;UND IN B ABSPEICHERN
07B3	79	1668	MOV A,C ;OP-CODE IN AKKU
07B4	E608	1669	ANI 08H ;BIT A3 (SOLL-FLAG) AUSBLENDEN
07B6	80	1670	ADD B ;ADIERE SOLL-FLAG UND IST-FLAG
07B7	37	1671	STC ;CY=1
07B8	E8	1672	RPE ;RUECKSPRUNG WENN SPRUNGBED. ERFUELLT
07B9	3F	1673	CMC ;CY=0 WENN SPRUNGBED NICHT ERFUELLT
07BA	C9	1674	RET
		1675	\$EJECT

LOC	OBJ	LINE	SOURCE STATEMENT
		1676	*****
		1677	;
		1678	ANWENDERPROGRAMME
		1679	;
		1680	*****
		1681	
		1682	
07BC		1683	ORG 7BCH
		1684	
		1685	
		1686	-----
		1687	PROGRAMM : ANWAZL (ANZEIGE LOESCHEN)
		1688	;
		1689	FUNKTION : DIE UEBER DEN TASTATUR-/ANZEIGEBAUSTEIN SAB8279 GESTEU-
		1690	ERTE ANZEIGE WIRD GELOESCHT (ALLE ANZEIGEN "NULLEN"
		1691	D.H. DUNKEL).
		1692	;
		1693	EING.PARAM. : -
		1694	EINGABEN VON: 8279 (STATUS)
		1695	AUSG.PARAM. : -
		1696	AUSGABEN AN : 8279 (LOESCH-KOMMANDO)
		1697	VERAEND.REG.: PSW
		1698	AUFGER.PROG.: ANWAUS
		1699	BEMERKUNG : BEI VERWENDUNG DES KOMMANDOS "ANZEIGE LOESCHEN" KANN
		1700	ES VORKOMMEN, DASS NICHT ALLE STELLEN GELOESCHT WERDEN
		1701	(EIGENSCHAFT DES 8279). AUS DIESEM GRUND WERDEN ALLE
		1702	STELLEN EINZELN GELOESCHT.
		1703	-----
		1704	
07BC 97		1705	ANWAZL: SUB A ;CODE FUER ZEICHEN "NULL" LADEN
07BD 0608		1706	MVI B,8 ;B-REG. FUER ANZEIGESTELLE SETZEN
07BF 05		1707	SLAZL: DCR B ;ANZEIGESTELLE DEKREMENTIEREN
07CD CDC707		1708	CALL ANWAUS ;ZEICHEN "NULL" AUSGEBEN
		1709	;ANZEIGESTELLE = 0?
07C3 C2BFD7		1710	JNZ SLAZL ;NEIN! NAECHSTE STELLE
07C6 C9		1711	RET ;JA! RUECKSPRUNG
		1712	
		1713	
		1714	\$EJECT

LOC	OBJ	LINE	SOURCE STATEMENT
		1715 ;	-----
		1716 ;	PROGRAMM : ANWAUS (AUSGABE ZUR ANZEIGE)
		1717 ;	
		1718 ;	FUNKTION : DER IM A-REG. UEBERGEBENE 7-SEGMENT-CODE WIRD UEBER
		1719 ;	DEN TASTATUR-/ANZEIGEBAUSTEIN SAB8279 ZUR ANZEIGE AUS-
		1720 ;	GEGEBEN. DIE ANZEIGE-POSITION (STELLE) IST IM B-REG.
		1721 ;	ZU UEBERGEBEN.
		1722 ;	
		1723 ;	EING.PARAM. : 7-SEGMENT-CODE IM A-REG.
		1724 ;	ANZEIGEPPOSITION IM B-REG. (0-7, WOBEI DIE ZAEHLUNG VON
		1725 ;	LINKS NACH RECHTS EFOLGST)
		1726 ;	EINGABEN VON: -
		1727 ;	AUSG.PARAM. : -
		1728 ;	AUSGABEN AN : 8279 (ANZEIGE-RAM)
		1729 ;	VERAEND.REG.: KEINE
		1730 ;	-----
		1731	
07C7	F5	1732	ANWAUS: PUSH PSW ;/7-SEGMENT-CODE IM KELLERSPEICHER ABLEGEN
07C8	3E80	1733	MVI A,80H ;MASKE FUER "SCHREIBEN ANZEIGE-RAM 0" LADEN
07CA	80	1734	ADD B ;NR. DER ANZEIGEPPOSITION ZUR MASKE ADDIEREN
07CB	D3EF	1735	OUT DEFH ;ERZEUGTES KOMMANDOWORT ZUM 8279 AUSGEBEN
07CD	F1	1736	POP PSW ;/7-SEGMENT-CODE LADEN
07CE	D3EE	1737	OUT DEEH ;7-SEGMENT-CODE ZUR ANZEIGE UEBER 8279 AUSGEBEN
		1738	;/DEEH IST DIE ADRESSE DES ANZEIGE- UND DES
		1739	; FIFO-RAM IM 8279
07DD	C9	1740	RET ;RUECKSPRUNG
		1741	
		1742 ;	-----
		1743 ;	PROGRAMM : ANWEIN (EINGABE VON TASTATUR)
		1744 ;	
		1745 ;	FUNKTION : DER PUFFERSPEICHER (FIFO-RAM) FUER DIE TASTATUREINGABE
		1746 ;	WIRD GELOESCHT UND GEWARTET BIS EINE TASTATUREINGABE
		1747 ;	ERFOLGT. WIRD EIN ZEICHEN EINGEGEBEN, ERFOLGT DER
		1748 ;	RUECKSPRUNG MIT UEBERGABE DES ZEICHENCODES IM A-REG.
		1749 ;	
		1750 ;	EING.PARAM. : -
		1751 ;	EINGABEN VON: 8279 (STATUS,FIFO-RAM)
		1752 ;	AUSG.PARAM. : ZEICHEN IM A-REG.
		1753 ;	AUSGABEN AN : 8279 (FIFO-RAM)
		1754 ;	VERAEND.REG.: PSW
		1755 ;	-----
		1756	
07D1	3EC2	1757	ANWEIN: MVI A,0C2H ;"FIFO-STATUS" DES 8279 LOESCHEN
07D3	D3EF	1758	OUT DEFH
07D5	3E40	1759	MVI A,40H ;KOMMANDOWORT "FIFO-RAM LESEN" ZUM 8279
07D7	D3EF	1760	OUT DEFH ; AUSGEBEN
07D9	DBEF	1761	WAREIN: IN DEFH ;"FIFO-STATUS" DES 8279 LESEN
07DB	E60F	1762	ANI DFH ;TASTATUR-EINGABE ?
		1763	; /BIT3-0 = ANZAHL DER ZEICHEN IM FIFO >< 0 ?
07DD	CAD907	1764	JZ WAREIN ;NEIN! /WARTEN
07ED	DBEE	1765	IN DEEH ;JA! EINGEGEBENS ZEICHEN EINLESEN
07E2	C9	1766	RET ;RUECKSPRUNG

LOC	OBJ	LINE	SOURCE STATEMENT
		1767	
		1768	
		1769	-----
		1770	PROGRAMM : ANWRUF (AUFRUFPROGRAMM)
		1771	;
		1772	FUNKTION : 'ANWRUF' IST EIN HAUPTPROGRAMM. ES IST ABLAUFFAEHIG
		1773	UND KANN SOMIT MIT DEN MONITORKOMMANDOS "7", "8" ODER
		1774	"9" GESTARTET WERDEN. ES DIENST ZUM AUFRUF DES UNTER-
		1775	PROGRAMMS 'ANWMUL' (MULTIPLIKATION). DABEI WERDEN DIE
		1776	INHALTE DER REGISTER D UND E MULTIPLIZIERT UND DAS ER-
		1777	GEBNIS IM H&L-REGISTER ABGELEGT. DIESE REGISTER KOEN-
		1778	NEN MIT DEM MONITORKOMMANDO "6" GELESEN UND BESCHRIE-
		1779	BEN WERDEN. AM PROGRAMM ENDE WIRD DER PROZESSOR DURCH
		1780	DEN HALT-BEFEHL ANGEHALTEN.
		1781	-----
		1782	
07E3	31B2FF	1783	ANWRUF: LXI SP,MONRAM-2;KELLERSPEICHERZEIGER SETZEN
		1784	;/KELLERSPEICHER IM FREIEN RAMTEIL DES
		1785	; SAB8255/0
07E6	3E0B	1786	MVI A,OBH ;INTERRUPT-EINGANG RST7.5 FREIGEBEN
07E8	30	1787	SIH
07E9	CDEE07	1788	CALL ANWMUL ;AUFRUF ANWMUL
07EC	FB	1789	EI ;INTERRUPT FREIGEBEN
		1790	;/FUER I-TASTE BEI REALZEITTEST
07ED	76	1791	HLT ;PROZESSOR ANHALTEN
		1792	
		1793	SEJECT

LOC	OBJ	LINE	SOURCE STATEMENT
		1794	-----
		1795	; PROGRAMM : ANWMUL (MULTIPLIKATION)
		1796	; -----
		1797	; FUNKTION : MULTIPLIKATION ZWEIER GANZER, VORZEICHENLOSER 8-BIT
		1798	; BINAER-ZAHLEN.
		1799	; PRODUKT := MULTIPLIKATOR * MULTIPLIKAND
		1800	; -----
		1801	; EING.PARAM. : MULTIPLIKATOR (8 BIT) IM E-REG.
		1802	; MULTIPLIKAND (8 BIT) IM D-REG.
		1803	; EINGABEN VON: -
		1804	; AUSG.PARAM. : PRODUKT (16 BIT) IM H&L-REG.
		1805	; AUSGABEN AN : -
		1806	; VERAEND.REG.: PSW, D, H, L
		1807	-----
		1808	; VERWENDUNG DER ZENTRALEINHEIT-REGISTER
		1809	;
		1810	; <D&E> = MULTIPLIKATOR (ERWEITERT AUF 16 BIT
		1811	; MIT <D>:=0)
		1812	; A = SCHIEBEREGISTER FUER MULTIPLIKAND
		1813	; <C> = STELLENZAehler
		1814	; <H&L> = ZWISCHENSUMME UND PRODUKT
		1815	-----
07EE	DE08	1816	ANWMUL: MVI C,8 ;STELLENZAehler AUF 8 SETZEN
07F0	7A	1817	MOV A,D ;/MULTIPLIKAND IN A-REG. LADEN
07F1	210000	1818	LXI H,0 ;ZWISCHENSUMME AUF 0 SETZEN
07F4	54	1819	MOV D,H ;/MULTIPLIKATOR AUF 16 BIT ERWEITERN
07F5	29	1820	MUL1: DAD H ;ZWISCHENSUMME UM EINE STELLE NACH LINKS
		1821	; SCHIEBEN
07F6	17	1822	RAL ;MULTIPLIKAND UM EINE STELLE NACH LINKS
		1823	; SCHIEBEN
		1824	;HERAUSGESCHOBENES BIT = 1 ?
07F7	D2FB07	1825	JNC MUL2 ;NEIN! /SPRUNG
07FA	19	1826	DAD D ;JA! MULTIPLIKATOR ZUR ZWISCHENSUMME ADD.
07FB	0D	1827	MUL2: DCR C ;STELLENZAehler DEKREMENTIEREN
		1828	;STELLENZAehler = 0 ?
07FC	C2F507	1829	JNZ MUL1 ;NEIN! /NAECHSTE STELLE
07FF	C9	1830	RET ;JA! RUECKSPRUNG
		1831	-----
		1832	\$EJECT

LOC OBJ LINE SOURCE STATEMENT

1833 END

PUBLIC SYMBOLS

EXTERNAL SYMBOLS

USER SYMBOLS

AA	A FFCE	AANF	A 0696	ABSL	A 042B	ABSP	A 050E	ABTEST	A 0454	ADR	A FFDD	ADRANF	A 06DE
ANFA1	A 00CB	ANFA2	A 00D4	ANFANG	A 009A	ANWAUS	A 07C7	ANWAZL	A 07BC	ANWEIN	A 07D1	ANWHUL	A 07EE
ANWRUF	A 07E3	ARET	A FF85	AU	A FFDB	AUANF	A 06CC	AUS	A 05A8	AUS1	A 05AA	BA	A FF02
BANF	A 06A8	BEDIEN	A 045E	BEFREG	A FFC2	BITSI	A 0009	BITSO	A 000B	BITTIM	A 0269	BL1	A 059E
BL2	A 05A1	BLIN	A 0589	BRET	A FF87	BTBL	A 04C7	BYAUS	A 0620	BYTE	A 061F	CI1	A 0667
CI2	A 066E	CI3	A 067B	CI4	A 067F	CIS	A 0694	CIN	A 0664	CLOE	A 0473	CLRALL	A 0094
CHDD	A 00DC	CHDDA	A 00DE	CHD1	A 0108	CHD2	A 012D	CHD3	A 0177	CHD4	A 01DB	CHD4A	A 01DE
CHD5	A 01ED	CMD5A	A 01F8	CMD6	A 0234	CMD6A	A 0236	CHD6B	A 0282	CMD7	A 029B	CHD8	A 02BE
CHD85	A 02CA	CMD9	A 03D1	CMDA	A 041B	CMDB	A 041B	C01	A 064A	C02	A 0652	COUT	A 0646
CRET	A FF86	CURE	A 056A	DAT	A FFDC	DATANF	A 06EA	DATEA	A 00EE	DEFAULT	A FFE1	DISPLA	A 0606
DRET	A FF89	DSUB	A 0633	EA	A FFDD	EANF	A 069F	ECHO	A 0507	ERET	A FF88	EU	A FFDA
EUANF	A 06D5	FILL1	A 011C	FRET	A FF84	GLEICH	A 0640	H1	A FF04	H1ANF	A 06BA	H2	A FF06
H2ANF	A 06C3	HALFBI	A 01A5	HEXD	A 05D8	HEXZEI	A 04DF	HRET	A FF8B	IOLBA	A 0202	IOLBB	A 021F
IRET	A FFC0	JUMP	A 0798	JUMP1	A 07AE	K	A 075F	K1	A 0793	KEND	A 0796	KOADR	A 00F8
KOG2	A 05C0	KOGEN	A 05BD	KOMZE	A FFDF	KOSEA	A 00EF	LEFT	A 0578	LEFT1	A 0579	LES	A 0485
LESBE	A 055D	LOES	A 008E	LRET	A FF8A	M1	A 0102	HCMDB	A 0448	MCHDBF	A 044D	MEHAD	A FFCB
MODUS	A FFDE	HONRAM	A FF84	MUL1	A 07F5	MUL2	A 07FB	NEXADR	A 0062	PAAN	A 06F3	PAANF1	A 0714
PACK	A 0513	PACK1	A 051F	PACK2	A 0525	PACK3	A 052E	PACK4	A 0533	PACK5	A 0542	PARAM	A 071F
PARAM2	A 073A	PARAM3	A 0748	PRET	A FF8C	PRGSTR	A 0379	RAMHZ	A FFE0	RDEF	A 075A	RECHTS	A 056B
REGAD	A FFCA	REGL1	A 0242	REGP	A 028D	RIGH1	A 0583	RIGHT	A 0582	RST7	A 0038	RST75	A 003C
SANF	A 06B1	SDEF	A 0754	SLAZL	A 078F	SPEI1	A 05FB	SPEICH	A 05E8	SRET	A FF8E	START	A 0000
START1	A 0062	STEP	A 0307	STEPD	A 03B3	STEP1	A 0388	STEP2	A 03E7	STEP3	A 03D9	STEP4	A 03F3
STEP5	A 038F	STEP6	A 03CC	STEP7	A 03D4	STEP8	A 03EE	STEP9	A 0402	STEPA	A 031E	STEPC	A 032F
STEPE	A 0389	STKINI	A 0000	TABZ	A 0487	TAST	A 0461	TIMEH	A 00FD	TIMEL	A 00FC	UMR1	A 0193
UMR2	A 019A	UMR3	A 01A8	UHREND	A 01D4	UHS1	A 0155	UHS3	A 0159	UHS4	A 0172	UHW	A 05C9
WAREIN	A 07D9												

ASSEMBLY COMPLETE, NO ERRORS

STEPE	674	694#					
STKINI	68#	236	248				
TABZ	938	943#					
TAST	895#	899	901				
TIMEH	79#	1223					
TIMEL	80#	1225					
UMR1	383#	391					
UHR2	386#	390	435				
UMR3	385	394#					
UMREND	408	413	432#				
UMS1	339	347#					
UMS3	345	349#	351	366			
UMS4	359	364#					
UMW	255	992	1000	1166#	1293	1296	
WAREIN	1761#	1764					

CROSS REFERENCE COMPLETE

9.3 Übersetzungstabelle für die Assembler-Befehle

Tabelle 9.3-1 bringt eine vollständige Übersetzungstabelle für 8080/8085-Befehle, die für die manuelle Assemblierung gedacht ist. Die Assembler-Befehle sind alphabetisch geordnet. Es sind alle Befehle aufgeführt, die einen eigenen Operationscode haben. Die Tabelle umfaßt vier Spalten mit folgenden Bedeutungen:

- A = Assembler-Befehl
- B = Zahl der Bytes, die der Befehl umfaßt
- H = Operationscode (erstes Byte) des Befehls
in Hexadezimal-Darstellung
- T = Zahl der Taktzyklen, die für die Ausführung
eines Befehls erforderlich ist. Ein Taktzyklus
umfaßt beim ECB85 0,326 μ s (Taktfrequenz = 3,072MHz).

In den Assembler-Befehlen sind Daten- und Adreßteil nur symbolisch angedeutet. Es werden dabei folgende Abkürzungen benutzt:

- d8 == 8-Bit-Datenteil
- d16 = 16-Bit-Datenteil
- a8 = 8-Bit-Adreßteil
(Ein-/Ausgabe-Adresse)
- a16 = 16-Bit-Adreßteil
(Speicher-Adresse)

Als Zahl der Taktzyklen sind bei bedingten Sprungbefehlen zwei Zahlen angegeben. Die erste Zahl gilt für den Fall, daß der Sprung nicht ausgeführt wird (Sprungbedingung nicht erfüllt), die zweite dann, wenn der Sprung erfolgt (Sprungbedingung erfüllt). Eine Taktperiode dauert 326ns (3.072MHz).

Die Befehle RIM und SIM gibt es nur beim System 8085, nicht jedoch beim 8080.

Tabelle 9.3-1

Übersetzungstabelle von Assembler- in Maschinenbefehle

A		B	H	T
ACI	d8	2	CE	7
ADC	A	1	8F	4
ADC	B	1	88	
ADC	C	1	89	
ADC	D	1	8A	
ADC	E	1	8B	
ADC	H	1	8C	
ADC	L	1	8D	
ADC	M	1	8E	7
ADD	A	1	87	4
ADD	B	1	80	
ADD	C	1	81	
ADD	D	1	82	
ADD	E	1	83	
ADD	H	1	84	
ADD	L	1	85	
ADD	M	1	86	7
ADI	d8	2	C6	4
ANA	A	1	A7	
ANA	B	1	A0	
ANA	C	1	A1	
ANA	D	1	A2	
ANA	E	1	A3	
ANA	H	1	A4	
ANA	L	1	A5	7
ANA	M	1	A6	
ANI	d8	2	E6	18
CALL	a16	3	CD	
CC	a16	3	DC	9/18
CM	a16	3	FC	4
CMA		1	2F	
CMC		1	3F	4
CMP	A	1	BF	

A		B	H	T
CMP	B	1	B	4
CMP	C	1	B9	
CMP	D	1	BA	
CMP	E	1	BB	
CMP	H	1	BC	
CMP	L	1	BD	
CMP	M	1	BE	
CNC	a16	3	D4	9/18
CNZ	a16	3	C4	
CP	a16	3	F4	
CPE	a16	3	EC	7
CPI	d8	2	FE	
CPO	a16	3	E4	9/18
CZ	a16	3	CC	10
DAA		1	27	
DAD	B	1	09	
DAD	D	1	19	
DAD	H	1	29	
DAD	SP	1	39	
DCR	A	1	3D	
DCR	B	1	05	
DCR	C	1	0D	
DCR	D	1	15	
DCR	E	1	1D	
DCR	H	1	25	
DCR	L	1	2D	
DCR	M	1	35	10
DCX	B	1	0B	6
DCX	D	1	1B	
DCX	H	1	2B	
DCX	SP	1	3B	4
DI		1	F3	
EI		1	FB	

(Fortsetzung)

Tabelle 9.3-1
(Fortsetzung)

A	B	H	T
HLT	1	76	5
IN a8	2	DB	10
INR A	1	3C	4
INR B	1	04	
INR C	1	0C	
INR D	1	14	
INR E	1	1C	
INR H	1	24	
INR L	1	2C	
INR M	1	34	
INX B	1	03	6
INX D	1	13	
INX H	1	23	
INX SP	1	33	7/10
JC a16	3	DA	
JM a16	3	FA	
JMP a16	3	C3	
JNC a16	3	D2	
JNZ a16	3	C2	
JP a16	3	F2	
JPE a16	3	EA	
JPO a16	3	E2	
JZ a16	3	CA	
LDA a16	3	3A	13
LDAX B	1	0A	7
LDAX D	1	1A	
LHLD a16	3	2A	10
LXI B,d16	3	01	
LXI D,d16	3	11	
LXI H,d16	3	21	4
LXI SP,d16	3	31	
MOV A,A	1	7F	4
MOV A,B	1	78	

A	B	H	T
MOV A,C	1	79	4
MOV A,D	1	7A	
MOV A,E	1	7B	
MOV A,H	1	7C	
MOV A,L	1	7D	
MOV A,M	1	7E	
MOV B,A	1	47	
MOV B,B	1	40	
MOV B,C	1	41	
MOV B,D	1	42	
MOV B,E	1	43	7
MOV B,H	1	44	
MOV B,L	1	45	
MOV B,M	1	46	
MOV C,A	1	4F	
MOV C,B	1	48	
MOV C,C	1	49	
MOV C,D	1	4A	
MOV C,E	1	4B	
MOV C,H	1	4C	
MOV C,L	1	4D	4
MOV C,M	1	4E	
MOV D,A	1	57	
MOV D,B	1	50	
MOV D,C	1	51	
MOV D,D	1	52	
MOV D,E	1	53	
MOV D,H	1	54	
MOV D,L	1	55	
MOV D,M	1	56	
MOV E,A	1	5F	7
MOV E,B	1	58	
MOV E,C	1	59	

(Fortsetzung)

Tabelle 9.3-1
(Fortsetzung)

A	B	H	T
MOV E,D	1	5A	4
MOV E,E	1	5B	
MOV E,H	1	5C	
MOV E,L	1	5D	
MOV E,M	1	5E	7
MOV H,A	1	67	
MOV H,B	1	60	
MOV H,C	1	61	
MOV H,D	1	62	4
MOV H,E	1	63	
MOV H,H	1	64	
MOV H,L	1	65	
MOV H,M	1	66	7
MOV L,A	1	6F	
MOV L,B	1	68	
MOV L,C	1	69	
MOV L,D	1	6A	4
MOV L,E	1	6B	
MOV L,H	1	6C	
MOV L,L	1	6D	
MOV L,M	1	6E	7
MOV M,A	1	77	
MOV M,B	1	70	
MOV M,C	1	71	
MOV M,D	1	72	
MOV M,E	1	73	4
MOV M,H	1	74	
MOV M,L	1	75	
MVI A,d8	2	3E	
MVI B,d8	2	06	
MVI C,d8	2	0E	
MVI D,d8	2	16	7
MVI E,d8	2	1E	

A	B	H	T
MVI H,d8	2	26	7
MVI L,d8	2	2E	
MVI M,d8	2	36	10
NOP	1	00	
ORA A	1	B7	
ORA B	1	B0	
ORA C	1	B1	
ORA D	1	B2	4
ORA E	1	B3	
ORA H	1	B4	
ORA L	1	B5	
ORA M	1	B6	7
ORI d8	2	F6	
OUT a8	2	D3	10
PCHL	1	E9	6
POP B	1	C1	
POP D	1	D1	10
POP H	1	E1	
POP PSW	1	F1	
PUSH B	1	C5	
PUSH D	1	D5	12
PUSH H	1	E5	
PUSH PSW	1	F5	
RAL	1	17	
RAR	1	1F	4
RC	1	D8	6/12
RET	1	C9	10
RIM	1	20	
RLC	1	07	4
RM	1	F8	
RNC	1	D0	
RNZ	1	C0	6/12
RP	1	F0	

(Fortsetzung)

Tabelle 9.3-1
(Fortsetzung)

A	B	H	T
RPE	1	E8	6/12
RPO	1	E0	
RRC	1	OF	4
RST 0	1	C7	12
RST 1	1	CF	
RST 2	1	D7	
RST 3	1	DF	
RST 4	1	E7	
RST 5	1	EF	
RST 6	1	F7	
RST 7	1	FF	
RZ	1	C8	6/12
SBB A	1	9F	4
SBB B	1	98	
SBB C	1	99	
SBB D	1	9A	
SBB E	1	9B	
SBB H	1	9C	
SBB L	1	9D	
SBB M	1	9E	
SBI d8	2	DE	7
SHLD a16	3	22	16
SIM	1	30	4
SPHL	1	F9	6
STA a16	3	32	13
STAX B	1	02	7
STAX D	1	12	
STC	1	37	4
SUB A	1	97	
SUB B	1	90	
SUB C	1	91	
SUB D	1	92	

A	B	H	T
SUB E	1	93	4
SUB H	1	94	
SUB L	1	95	
SUB M	1	96	7
SUI d8	2	D6	4
XCHG	1	EB	
XRA A	1	AF	
XRA B	1	A8	
XRA C	1	A9	
XRA D	1	AA	
XRA E	1	AB	
XRA H	1	AC	
XRA L	1	AD	7
XRA M	1	AE	
XRI d8	2	EE	16
XTHL	1	E3	

Tabelle 9.4-1

a) Zusammenstellung der Kommandos und ihre Funktion

Kommando-Symbol	Funktion	Funktionsgruppe
0	Speicher auslesen und beschreiben	Kommandos mit Bezug auf den Speicher
1	Speicherbereich mit Konstante füllen	
2	Speicherinhalt übertragen	
3	Adressen umrechnen	
4	über Ausgabe-Kanal ausgeben	Kommandos zum Testen der Hardware
5	über Eingabe-Kanal eingeben	
6	Register auslesen und beschreiben	Kommandos zum Testen der Software
7	Programm ohne Unterbrechungspunkte starten	
8	Programm mit Unterbrechungspunkten starten	
9	Programm im Einzelschritt abarbeiten	
A	Information auf Magnetband schreiben	Kommandos mit Bezug auf das Magnetbandgerät
b	Information von Magnetband lesen	

b) Symbole im Adreßfeld und ihre Funktion

Symbole für Parameternamen

AA Anfangsadresse
 EA Endadresse
 bA Bestimmungsadresse
 SA Startadresse
 H1 Unterbrechungsadresse 1
 H2 Unterbrechungsadresse 2
 AU Anfangsadresse des um-zurechnenden Bereichs
 EU Endadresse des um-zurechnenden Bereichs
 PA Peripherie-Adresse
 db Datenbyte

Symbole für Registerpaar-Namen

AF Akkumulator & Zustandswort(Flag)
 bC Registerpaar B & C
 dE Registerpaar D & E
 HL Registerpaar H & L
 PC Programmzähler(program counter)
 SP Kellerspeicher-Zeiger (stack pointer)
 IS Interrupt-Status

Questo documento e' stato scaricato dal sito del Museo del computer.

Tutti i marchi appartengono ai legittimi proprietari. Non siamo responsabili di eventuali errori o mancanze presenti in questo documento.

Se questo documento vi e' stato utile, valutate la possibilita' di fare una piccola donazione alla nostra fondazione, che da anni lavora per preservare la storia dell'informatica all'indirizzo donazioni.museo.computer

In caso di pubblicazione o diffusione, siete pregati di citare la fonte.

This document was downloaded from the Computer Museum website.

All trademarks belong to their respective owners. We are not responsible for any errors or omissions in this document.

If this document has been useful to you, consider making a small donation to our foundation, which has been working for years to preserve the history of information technology at donazioni.museo.computer

In case of publication or diffusion, please quote the source.

FONDAZIONE MUSEO DEL COMPUTER

Via per Occhieppo 29
13891 CAMBURZANO (BIELLA) - ITALY
Tel +39 015 8853201
Fax +39 015 8853201

Web : www.museodelcomputer.org
Mail : info@museodelcomputer.org
PEC : museodelcomputer@pec.it

C.F. 94064520037
P.I. 02314610037
Registro P.G. 237

Member of ICOM n. 70884